LNCS 6484

ARCoSS

Amin Saberi (Ed.)

Internet and Network Economics

6th International Workshop, WINE 2010
Stanford, CA, USA, December 2010
Proceedings

Springer

Amin Saberi (Ed.)

Internet and Network Economics

6th International Workshop, WINE 2010
Stanford, CA, USA, December 13-17, 2010
Proceedings

Springer

Volume Editor

Amin Saberi
Stanford University
Department of Management Science and Engineering
Stanford, CA 94305, USA
E-mail: saberi@stanford.edu

Library of Congress Control Number: 2010939947

CR Subject Classification (1998): C.2, F.2, D.2, H.4, F.1, H.3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-642-17571-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-17571-8 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180 5 4 3 2 1 0

Preface

The present volume contains the papers accepted for presentation at the 6th International Workshop on Internet and Network Economics (WINE), an interdisciplinary forum devoted to the analysis of algorithmic and economic problems arising in the context of the Internet and the World Wide Web.

WINE 2010 was held December 13–17 in Stanford University. It was colocated with the 7th Workshop on Algorithms and Models for the Web Graph (WAW 2010).

In response to call for papers, the Program Committee received 95 submissions. The committee conducted a thorough evaluation and electronic discussion and eventually selected 52 papers (33 regular and 19 short papers) for inclusion in the proceedings.

This volume contains all the accepted papers. The workshop program also includes six distinguished lectures of Daron Acemoglu (Massachusetts Institute of Technology), Jennifer Chayes (Microsoft Research), Michael Kearns (University of Pennsylvania), Jon Kleinberg (Cornell University), Nimrod Megiddo (IBM Research), and Rakesh Vohra (Northwestern University).

We wish to thank the creators of EasyChair, a free conference management system, which significantly assisted the work of the Program Committee. We would also like to thank Google, Microsoft Research, and Yahoo! Research for supporting WINE. Finally, we wish to thank Stanford University, especially the staff of Stanford Computer Forum and the department of Management Science and Engineering.

December 2010

Amin Saberi
Yinyu Ye

Conference Organization

Program Chairs

Amin Saberi
Yinyu Ye

Program Committee

Daron Acemoglu, Christina Aperjis, Moshe Babaioff, Kostas Bimpikis, Jennifer Chayes, Ning Chen, Costis Daskalakis, Xiaotie Deng, Nikhil Devanur, Edith Elkind, Jason Hartline, Nicole Immorlica, Ramesh Johari, Jon Kleinberg, Robert Kleinberg, Stefano Leonardi, Mohammad Mahdian, Vangelis Markakis, Aranyak Mehta, S. Muthukrishnan, Hamid Nazerzadeh, David Pennock, Maria Polukarov, Tim Roughgarden, Amin Saberi, Adam Szeidl, Eva Tardos, Rakesh Vohra, and Yinyu Ye

Local Organization

Mohammad Mahdian, Aranyak Mehta, Amin Saberi, Yoav Shoham, and Yinyu Ye

External Reviewers

Saeed Alaei, Adam Kalai, Ioannis Caragiannis, Aleksandrs Slivkins, Angelo Fanelli, Ariel Procaccia, Azarakhsh Malekian, Balasubramanian Sivan, Brendan Lucier, Bumin Yenmez, Chinmay Karande, Chris Wilkens, David Thompson, Felix Fischer, Georgios Chalkiadakis, Georgios Piliouras, Guido Schaefer, Heiko Roglin, Hu Fu, Ilan Lobel, Itai Ashlagi, Jiajin Yu, Jie Zhang, John Augustine, Josh Letchford, Kejun Wang, Mahyar Salek, Michael Schapira, Mohamed Mostagir, Nikolai Gravin, Nima Haghpanah, Omer Tamuz, Orestis Telelis, Paul Harrenstein, Paul Spirakis, Pinyan Lu, Piotr Faliszewski, Qiqi Yan, Radhika Arava, Renato Paes Leme, Samuel Ieong, Shaddin Dughmi, Shahar Dobzinski, Shayan Oveis Gharan, Sven Seuken, Tian-Ming Bu, Utku Ozan Candogan, Vahab Mirrokni, Victor Naroditskiy, Viet Le Truc, Vincent Conitzer, Xia Hua, Xiaorui Sun, Yajun Wang, Yashodhan Kanoria, Yoram Bachrach, and Yossi Azar

Table of Contents

Regular Papers

Short Papers

Efficient Computation of the Shapley Value for Centrality in Networks

Karthik V. Aadithya[1], Balaraman Ravindran[2], Tomasz P. Michalak[3], and Nicholas R. Jennings[3]

[1] The University of California, Berkeley, USA
aadithya@eecs.berkeley.edu
[2] Indian Institute of Technology Madras, India
ravi@cse.iitm.ac.in
[3] School of Electronics and Computer Science, University of Southampton, UK
{tpm,nrj}@ecs.soton.ac.uk

Abstract. The Shapley Value is arguably the most important normative solution concept in coalitional games. One of its applications is in the domain of networks, where the Shapley Value is used to measure the relative importance of individual nodes. This measure, which is called node centrality, is of paramount significance in many real-world application domains including social and organisational networks, biological networks, communication networks and the internet. Whereas computational aspects of the Shapley Value have been analyzed in the context of conventional coalitional games, this paper presents the first such study of the Shapley Value for network centrality. Our results demonstrate that this particular application of the Shapley Value presents unique opportunities for efficiency gains, which we exploit to develop exact analytical formulas for Shapley Value based centrality computation in both weighted and unweighted networks. These formulas not only yield efficient (polynomial time) and error-free algorithms for computing node centralities, but their surprisingly simple closed form expressions also offer intuition into why certain nodes are relatively more important to a network.

1 Introduction

The Shapley Value (SV) is a fundamental normative solution concept in coalitional games. Given a scenario where agents are allowed to realize collective payoffs through mutual co-operation, the SV postulates a fair method to evaluate each agent's individual contribution. One of the many applications of the SV is in the domain of networks, where it is used to measure the importance of individual nodes, which is known as *game theoretic network centrality* [1,2]. Although centrality plays a key role in many real-life network applications, efficient algorithms for its measurement via the SV remain unknown.

We now introduce the concept of "centrality". In the networks context, it is often paramount to determine which nodes and edges are more critical than others. Classic examples include identifying the most important highways in a road network, the most influential people in a social network or the most critical functional entities in a protein network. For such applications, the concept of *centrality* aims to quantify the importance of individual nodes/edges in a network.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 1–13, 2010.

Conventional centrality measures[1] usually work by assigning a score to each node in the network, which in some way corresponds to the importance of that node for the application at hand. For instance, if the application is to design an infrastructure network (such as a power transmission network or communication network) with minimum vulnerability to random node failures, a conventional centrality measure might work by analysing the consequences of failure at each individual node. The more adverse the consequences of failure, the higher the node centrality.

Such a conventional centrality metric, however, suffers from the following drawbacks:

1. By considering only the failure of individual nodes, it completely ignores real-world situations where multiple nodes can fail simultaneously. For example, if the network is so designed that no single node's failure carries any consequence, but the failure of certain specific *pairs of nodes* can bring down the entire network, the above centrality metric would fail to give a higher centrality score to the nodes belonging to these critical pairs.
2. Because each node is treated separately, the hidden assumption is that node failures occur independently of each other. As a result, real-world phenomena such as cascading node failures, that have been known to precipitate widespread disruption in a very short time [4], are outside the scope of this centrality analysis.

In short, conventional centrality measures fail to recognize that in many network applications, it is not sufficient to merely understand the relative importance of nodes as stand-alone entities. Rather, the key requirement is to understand the importance of each node *in terms of its utility when combined with other nodes* [5]. For instance, in the above infrastructure network, an ideal centrality measure would assign a score to a node v based on the failure probabilities (and consequences thereof) of *every subset of nodes containing* v, rather than just failure of the single node v. This approach would automatically allow the ideal centrality measure to give due consideration to real-world failure patterns such as cascading failures and simultaneous multiple node failures. On the other hand, this flexibility, which comes from the ability to take into account the contributions of all possible combinations of nodes (rather than just one node at a time), is absent in conventional centrality measures, which is a crucial limitation in many applications.

Game theoretic network centrality [1, 2] has been proposed as a framework that would address this limitation. Given the network to be analysed, the idea is to define a co-operative game where the agents (players) are the nodes of the network. Then the SV of each agent (node) in this game is interpreted as a *centrality measure* because it represents the average marginal contribution made by each node to every possible combination of the other nodes. This paradigm of SV-based network centrality thus confers a high degree of flexibility (which was completely lacking in traditional centrality metrics) to model real-world network phenomena. Indeed, this new paradigm has already been proved to be more useful than traditional centrality measures for certain real-life network applications [1, 6].

From a computational perspective, however, evaluating game theoretic network centrality using the original SV formula involves an analysis of the marginal contribution

[1] An overview of conventional centrality measures (such as degree centrality, betweenness centrality, closeness centrality and eigenvalue centrality) can be found in [3].

of every node (i.e. player) to every coalition. Thus, given a network $G(V, E)$, a direct application of the SV formula involves considering $O(2^{|V(G)|})$ coalitions. Such an exponential computation is clearly prohibitive for bigger networks (of, e.g., 100 or 1000 nodes). For such networks, the only feasible approach currently outlined in the literature is Monte-Carlo sampling, which is not only inexact, but also very time-consuming.

The above problem of *exponential complexity in the number of agents* is a fundamental challenge associated with computing the SV. As a result, for conventional coalitional games, this issue has received considerable attention in the literature. As an alternative to the straightforward (but exponential) listing of all possible coalitions, some authors [7,8] have proposed more efficient representations for coalitional games. In addition to being concise for many games, these representations are expressly designed to possess desirable computational properties, including efficient SV computation. Thus, the *choice of representation* has been the foremost consideration for efficient SV computation in the context of conventional coalitional games.

The networks domain, by contrast, poses a very different set of challenges:

1. Unlike conventional coalitional games, conciseness is usually not an issue in the networks context. This is because the games that aim to capture network centrality notions are completely specified by (a) the underlying network compactly represented as a graph, and (b) a *concise closed-form characteristic function expression* for evaluating coalition values (please see next section for an example).
2. Rather, the fundamental issue in the networks context is that: because the games are designed to reflect network centrality, the characteristic function definition often depends highly non-trivially on the underlying graph structure. Specifically, the value assigned by the characteristic function to each subset of nodes depends not just on the subgraph induced by those nodes, but also on the relationship between that subgraph and the rest of the network. For example, the value assigned to a coalition of nodes may be based on shortest path lengths to nodes outside the coalition, or it may depend non-trivially on the relationship between the coalition and its neighbors.

Therefore, the challenge we face in this paper is to *efficiently compute the SV, given a network and a game defined over it, where coalition values for this game are given by a closed-form expression that depends non-trivially on the network*. The key question here is how to take advantage of (a) the network structure, and (b) the functional form for the coalition values, so as to compute SVs efficiently, i.e., without the need to enumerate all possible coalitions.

Against this background:

[1] Our key contribution in this paper is to demonstrate that it is possible to exactly and efficiently compute SV-based network centralities of practical interest defined on large networks which exceed thousands of nodes! By contrast, the only previously known method that scaled to such large networks was Monte-Carlo simulation, which was neither exact nor particularly efficient.

[2] For four different measures of network centrality, we develop exact closed-form formulas for the SVs. We present pseudo-codes of linear and polynomial time algorithms to implement these formulas.

The remainder of the paper is organized as follows. Section 2 presents an example of how a coalitional game may be used to capture the notion of network centrality.

Section 3 analyses four types of centrality-related coalitional games and presents polynomial time SV algorithms for all of them. Conclusions follow.

2 SV as a Centrality Measure

As mentioned in the introduction, the paradigm of game theoretic network centrality based on the SV has been proposed in [1, 2] and further explored in [6]. This section presents an example to illustrate the advantages of this paradigm over conventional centrality measures.

Consider the notion of "closeness centrality" of a node in a graph $G(V, E)$, which is traditionally defined as the reciprocal of the average distance of that node from other (reachable) nodes in the graph [3]. This definition captures the intuitive idea that a node "in close proximity to many other nodes" is more valuable by virtue of its central location, and hence should be assigned a higher centrality score.

The above measure, however, fails to recognize the importance of combinations of nodes. For example, consider a typical real-world application of closeness centrality: that of disseminating a piece of information to all nodes in the network. At any time point t in the dissemination process, define the random variable C_t to be the subset of nodes most actively involved in propagating the information. In this situation, a new node added to C_t would make maximum contribution to the diffusion of information only if it is "in close proximity to nodes that are not currently in close proximity to any node in C_t". Thus, while conventional closeness centrality only takes into account *average proximity to all other nodes*, the actual importance of a node in the real-world application is based on a very different measure: *proximity to nodes that are not in close proximity to the random variable* C_t.

We now show how coalitional game theory can be used to construct a centrality measure that faithfully models the above *real-world importance* of a node. Let C be any subset of nodes from the given network $G(V, E)$. Then, for every such C, assign a value $\nu(C)$ given by

$$\nu(C) = \sum_{v \in V(G)} \frac{1}{1 + \min\{d(u, v) | u \in C\}}$$

where $d(u, v)$ is the distance between nodes u and v (measured as the shortest path length between u and v in graph G).

The map ν defined above captures a fundamental centrality notion: that the *intrinsic value* of a subset of nodes C in the context of a real-world application (such as information dissemination) is proportional to the overall proximity of the nodes in C to the other nodes in the network. In effect, the map ν carries the original definition of closeness centrality to a global level, where a measure of importance is assigned to every possible combination of nodes.

The map ν above is therefore a *characteristic function* for a coalitional game, where each vertex of the network is viewed as an agent playing the game. It follows that if a node v has a high SV in this game, it is likely that v would "contribute more" to an arbitrary randomly chosen coalition of nodes C in terms of increasing the proximity of C to other nodes on the network. Thus, computing the SVs of this game yields a centrality score for each vertex that is a much-improved characterization of closeness centrality.

The only difficulty in adopting such a game-theoretically inspired centrality measure is the previously mentioned problem of *exponential complexity in the number of agents*. In the next section, we show how to overcome this difficulty and compute the SV for many centrality applications (including the above formulation) in time polynomial in the size of the network.

3 Algorithms for SV-Based Network Centrality

In this section, we present 4 characteristic function formulations $\nu(C)$, each designed for a different real-world application. While each formulation captures a different *flavor of centrality*, they all embrace one fundamental centrality idea: that the definition for $\nu(C)$ must somehow quantify the *sphere of influence* of the coalition C over the other nodes. For instance, in our first game formulation, we start with the simplest possible idea that the sphere of influence of a coalition of nodes C is the set of all nodes immediately reachable (within one hop) from C. Subsequent games further generalize this notion of sphere of influence. For example, the second formulation specifies a more sophisticated sphere of influence: one that includes only those nodes which are immediately reachable in at least k different ways from C. The other two formulations extend the notion of sphere of influence to weighted graphs. The third game, for instance, defines sphere of influence as the set of all nodes within a cutoff distance of C (as measured by shortest path lengths on the weighted graph). The fourth and final formulation is an extreme generalization: it allows the sphere of influence of C to be specified by an arbitrary function $f(.)$ of the distance between C and the other nodes.

Throughout this section, we assume the reader is familiar with concepts of graph theory, including weighted and unweighted graphs, vertex degrees, neighboring vertices and shortest paths. We do not define these concepts here but suggest the reference [9]. The terms "network" and "graph" are used interchangeably in this paper, as are the terms "node" and "vertex". All the weighted graphs considered in this paper are positive weighted. We do not use digraphs in this paper, so all graphs are assumed to be undirected.

We also assume familiarity with the concepts of co-operative game theory, including the definition of coalitional games in characteristic function form and the Shapley Value. We do not define these concepts here but suggest the reference [10].

We now set the notation for a general coalitional game played on a network. Given a graph $G(V,E)$ with vertex set V and edge set E, we use G to define a coalitional game $g(V(G),\nu)$ with set of agents $V(G)$ and characteristic function ν. Here the agents of the coalitional game are the vertices of the graph G. Thus a coalition of agents C is simply any subset of $V(G)$. The characteristic function $\nu : 2^{V(G)} \rightarrow \mathbb{R}$ can be any function that depends on the graph G as long as it satisfies the condition $\nu(\emptyset) = 0$. We use the phrase "value of coalition C" to informally refer to $\nu(C)$.

3.1 Game 1: $\nu_1(C)$ = #Agents At-Most 1 Degree Away

Given an unweighted, undirected network $G(V,E)$. We first define "fringe" of a subset $C \subseteq V(G)$ as the set $\{v \in V(G) : v \in C \text{ (or) } \exists u \in C \text{ such that } (u,v) \in E(G)\}$, i.e., the fringe of a coalition includes all nodes reachable from the coalition in at most one hop.

Based on the fringe, we define the coalitional game $g_1(V(G), \nu_1)$ with respect to the network $G(V, E)$ by the characteristic function $\nu_1 : 2^{V(G)} \rightarrow \mathbb{R}$ given by

$$\nu_1(C) = \begin{cases} 0 & \text{if } C = \emptyset \\ \text{size}(\text{fringe}(C)) & \text{else} \end{cases}$$

This coalitional game has been extensively discussed in [1], where the authors motivate the game by arguing that the SVs of nodes in this game constitute a centrality metric that is superior to degree centrality for some applications. It is therefore desired to compute the SVs of all nodes for this game. We shall now present an exact formula for this computation rather than obtaining it through Monte-Carlo simulation as was done in [1].

To evaluate the SV of node v_i, consider all possible permutations of the nodes in which v_i would make a positive marginal contribution to the coalition of nodes occurring before itself. Let the set of nodes occurring before node v_i in a random permutation of nodes be denoted C_i. Let the neighbors of node v_i in the graph $G(V, E)$ be denoted $N_G(v_i)$ and the degree of node v_i be denoted $deg_G(v_i)$.

The key question to ask is: what is the necessary and sufficient condition for node v_i to marginally contribute node $v_j \in N_G(v_i) \cup \{v_i\}$ to fringe(C_i)? Clearly this happens if and only if neither v_j nor any of *its* neighbors are present in C_i. Formally $(N_G(v_j) \cup \{v_j\}) \cap C_i = \emptyset$.

Given that permutations are chosen uniformly at random for computing the SV, combinatorial arguments can be used to show that the above condition is satisfied with probability $\frac{1}{1+deg_G(v_j)}$. Denote by B_{v_i,v_j} the Bernoulli random variable that v_i marginally contributes v_j to fringe(C_i). Thus:

$$E[B_{v_i,v_j}] = \Pr[(N_G(v_j) \cup \{v_j\}) \cap C_i = \emptyset] = \frac{1}{1 + deg_G(v_j)}$$

Therefore, the Shapley Value $SV_{g_1}(v_i)$, which is the expected marginal contribution of v_i, is given by:

$$SV_{g_1}(v_i) \;=\; \sum_{v_j \in \{v_i\} \cup N_G(v_i)} E[B_{v_i,v_j}] \;=\; \sum_{v_j \in \{v_i\} \cup N_G(v_i)} \frac{1}{1 + deg_G(v_j)}$$

which is an *exact closed-form expression* for computing the SV of each node on the network.

Algorithm 1 describes an $O(V+E)$ procedure that directly implements the above equation to compute the exact SVs of all nodes in the network. By contrast, Monte-Carlo simulation requires $O(V+E)$ operations for *every* iteration. Moreover, the results obtained using Monte-Carlo are statistical in nature and may not be sufficiently accurate unless a large number of iterations are carried out.

Algorithm 1. Computing SVs for Game 1

Input: Unweighted graph $G(V, E)$
Output: SVs of all nodes in $V(G)$ for game g_1

```
foreach v ∈ V(G) do
    ShapleyValue[v] = 1/(1+deg_G(v));
    foreach u ∈ N_G(v) do
        ShapleyValue[v] += 1/(1+deg_G(u));
    end
end
return ShapleyValue;
```

It is possible to derive some intuition from the above formula. If a node has a high degree, the number of terms in its SV summation above will also be high. But the terms themselves would be inversely related to the degree of neighboring nodes. This gives the intuition that a node will have high centrality not only when its degree is high, but also whenever its degree tends to be higher in comparison to the degree of its neighboring nodes. In other words, *power comes from being connected to those who are powerless*, a fact that is well-recognized [11] by the centrality literature.

3.2 Game 2: $\nu_2(C)$ = #Agents with At-Least k Neighbors in C

We now consider a more general game formulation for an unweighted graph $G(V, E)$, where the value of a coalition includes the number of agents who are either in the coalition or are adjacent to at least k agents who are in the coalition. Formally, we consider game g_2 characterised by $\nu_2 : 2^{V(G)} \to \mathbb{R}$, where

$$\nu_2(C) = \begin{cases} 0 & \text{if } C = \emptyset \\ |\{v : v \in C \text{ (or) } |N_G(v) \cap C| \geq k\}| & \text{else} \end{cases}$$

Note that this game reduces to game g_1 for $k = 1$.

The motivation for this generalization is that in many real-life networks, the value of a coalition is interpreted as the number of agents who can be "influenced" by the coalition. For instance, in a viral marketing or innovation diffusion analysis [12], it is usually assumed that an agent will "be influenced" only if atleast k of his neighbors have already been convinced, which suggests such a game formulation.

Adopting notation from the previous subsection, we again ask: what is the necessary and sufficient condition for node v_i to marginally contribute node $v_j \in N_G(v_i) \cup \{v_i\}$ to the value of the coalition C_i?

Clearly, if $deg_G(v_j) < k$, we have $E[B_{v_i,v_j}] = \delta(v_i, v_j)$, i.e, $E[B_{v_i,v_j}] = 1$ for $v_i = v_j$ and 0 otherwise.

For $deg_G(n_j) \geq k$, we split the argument into two cases. If $v_j \neq v_i$, the condition for marginal contribution is that exactly $(k-1)$ neighbors of v_j already belong to C_i and $v_j \notin C_i$. On the other hand, if $v_j = v_i$, marginal contribution happens if and only if C_i originally consisted of at most $(k-1)$ neighbors of v_j.

So for $deg_G(v_j) \geq k$ and $v_j \neq v_i$, we have

$$E[B_{n_i,n_j}] = \binom{deg_G(v_j) - 1}{k-1} \frac{(k-1)!(1 + deg_G(v_j) - k)!}{(1 + deg_G(v_j))!} = \frac{1 + deg_G(v_j) - k}{deg_G(v_j)(1 + deg_G(v_j))}$$

And for $deg_G(v_i) \geq k$ and $v_j = v_i$, we have

$$E[B_{v_i,v_i}] = \frac{k}{1 + deg_G(v_i)}$$

As before, the SVs are given by substituting the above formulas into:

$$SV_{g_2}(v_i) = \sum_{v_j \in N_G(v_i) \cup \{v_i\}} E[B_{v_i,v_j}]$$

Although this game is a generalization of game g_1, it can still be solved to obtain the SVs of all nodes in $O(V + E)$ time, as formalised by Algorithm 2.

Algorithm 2. Computing SVs for Game 2

Input: Unweighted graph $G(V, E)$, positive integer k
Output: SVs of all nodes in $V(G)$ for game g_2

```
foreach v ∈ V(G) do
    ShapleyValue[v] = min(1, k/(1+deg_G(v)));
    foreach u ∈ N_G(v) do
        ShapleyValue[v] +=
        max(0, (deg_G(u)−k+1)/(deg_G(u)(1+deg_G(u))));
    end
end
return ShapleyValue;
```

An even more general formulation of the game is possible by allowing k to be a function of the agent, i.e, each node $v_i \in V(G)$ is assigned its own unique attribute $k(v_i)$. This translates to an application of the form: agent i is convinced if and only if atleast k_i of his neighbors are convinced, which is a frequently used model in the literature [12].

The above proof does not use the fact that k is constant across all nodes. So this generalized formulation can be solved by a simple modification to the original SV expression:

$$SV(v_i) = \frac{k(v_i)}{1 + deg_G(v_i)} + \sum_{v_j \in N_G(v_i)} \frac{1 + deg_G(v_j) - k(v_j)}{deg_G(v_j)(1 + deg_G(v_j))}$$

The above equation (which is also implementable in $O(V + E)$ time) assumes that $k(v_i) \leq 1 + deg_G(v_i)$ for all nodes v_i. This condition can be assumed without loss of generality because all cases can still be modeled (we set $k(v_i) = 1 + deg_G(v_i)$ for the extreme case where node v_i is never convinced no matter how many of its neighbors are already convinced).

3.3 Game 3: $\nu_3(C)$ = #Agents At-Most d_{Cutoff} Away

Hitherto, our games have been confined to unweighted networks. But in many applications, it is necessary to model real-world networks as weighted graphs. For example, in a co-authorship network, each edge is often assigned a weight proportional to the number of joint publications the corresponding authors have produced [13].

This subsection extends the game g_1 to the case of weighted networks. Whereas game g_1 equates $\nu(C)$ to the number of nodes located within one hop of some node in C, our new formulation in this subsection equates $\nu(C)$ to the number of nodes located within a distance d_{cutoff} of some node in C. Here, distance between two nodes is measured as the length of the shortest path between the nodes in the given weighted graph $G(V, E, W)$, where $W : E \to \mathbb{R}^+$ is the weight function.

Formally, we define the game g_3, where for each coalition $C \subseteq V(G)$,

$$\nu_3(C) = \begin{cases} 0 & \text{if } C = \emptyset \\ \text{size}(\{v_i : \exists v_j \in C \mid \text{distance}(v_i, v_j) \leq d_{\text{cutoff}}\}) & \text{else} \end{cases}$$

We shall now show that even this highly general centrality game g_3 is amenable to analysis which yields an exact formula for SVs. However, in this case the algorithm for implementing the formula is not linear in the size of the network, but has $O(VE + V^2\log(V))$ complexity.

Let us introduce some extra notation. Define the *extended neighborhood* $N_G(v_j, d_{\text{cutoff}}) = \{v_k \neq v_j : \text{distance}(v_k, v_j) \leq d_{\text{cutoff}}\}$, i.e, the set of all nodes whose distance from v_j is at most d_{cutoff}. Denote the size of $N_G(v_j, d_{\text{cutoff}})$ by $deg_G(v_j, d_{\text{cutoff}})$.

With this notation, the necessary and sufficient condition for node v_i to marginally contribute node v_j to the value of coalition C_i is: $\text{distance}(v_i, v_j) \leq d_{\text{cutoff}}$ and distance $(v_j, v_k) > d_{\text{cutoff}}\ \forall v_k \in C_i$. That is, neither v_j nor any node in *its* extended neighborhood should be present in C_i. But from the discussion of previous subsections, we know that the probability of this event is exactly $\frac{1}{1+deg_G(v_j,d_{\text{cutoff}})}$. Therefore, the exact formula for SV of node v_i in game g_3 is:

$$SV_{g_3}(v_i) = \sum_{v_j \in \{v_i\} \cup N_G(v_i, d_{\text{cutoff}})} \frac{1}{1 + deg_G(v_j, d_{\text{cutoff}})}$$

Algorithm 3 works as follows: for each node v in the network $G(V,E)$, the extended neighborgood $N_G(v, d_{\text{cutoff}})$ and its size $deg_G(v, d_{\text{cutoff}})$ are first computed using Dijkstra's algorithm in $O(E+V\log(V))$ time [14]. The results are then used to directly implement the above equation, which takes maximum time $O(V^2)$. In practice this step runs much faster because the worst case situation only occurs when every node is reachable from every other node within d_{cutoff}. Overall the complexity of the algorithm is $O(VE + V^2\log(V))$.

Algorithm 3. Computing SVs for Game 3

Input: Weighted graph $G(V, E, W)$, $d_{\text{cutoff}} > 0$
Output: SVs of all nodes in G for game g_3

```
foreach v ∈ V(G) do
    DistanceVector D = Dijkstra(v,G);
    extNeighbors(v) = ∅; extDegree(v) = 0;
    foreach u ∈ V(G) such that u ≠ v do
        if D(u) ≤ d_cutoff then
            extNeighbors(v).push(u);
            extDegree(v)++;
        end
    end
end
foreach v ∈ V(G) do
    ShapleyValue[v] = 1/(1+extDegree(v));
    foreach u ∈ extNeighbors(v) do
        ShapleyValue[v] += 1/(1+extDegree(u));
    end
end
return ShapleyValue;
```

We make one final observation: that the above proof does not depend on d_{cutoff} being constant across all nodes. Indeed, each node $v_i \in V(G)$ may be assigned its own unique value $d_{\text{cutoff}}(v_i)$, where $\nu(C)$ would be the number of agents v_i who are within a distance $d_{\text{cutoff}}(v_i)$ from C. For this case, the above proof gives:

$$SV(v_i) = \sum_{\substack{v_j : \text{distance}(v_i, v_j) \\ \leq d_{\text{cutoff}}(v_j)}} \frac{1}{1 + deg_G(v_j, d_{\text{cutoff}}(v_j))}$$

3.4 Game 4: $\nu_4(C) = \sum_{v_i \in V(G)} f(\text{Distance}(v_i, C))$

This subsection further generalizes game g_3, again taking motivation from real-life network problems. In game g_3, all agents at distances $d_{\text{agent}} \leq d_{\text{cutoff}}$ contributed equally to the value of a coalition. However, this assumption may not always hold true because in some applications, we intuitively expect agents closer to a coalition to contribute more to its value. For instance, we expect a Facebook user to exert more influence over his immediate circle of friends than over "friends of friends", even though both may satisfy the d_{cutoff} criterion. Similarly, we expect a virus-affected computer to infect a neighboring computer more quickly than a computer two hops away.

In general, we expect that an agent at distance d from a coalition would contribute $f(d)$ to its value, where $f(.)$ is a positive valued decreasing function of its argument. More formally, we define the game g_4 where the value of a coalition C is given by:

$$\nu_4(C) = \begin{cases} 0 & \text{if } C = \emptyset \\ \sum_{v_i \in V(G)} f(d(v_i, C)) & \text{else} \end{cases}$$

where $d(v_i, C)$ is the minimum distance $\min\{\text{distance}(v_i, v_j) | v_j \in C\}$.

It is possible to solve for SVs in the above formulation by constructing a marginal contribution network (MC-Net) [8]. However, the MC-Net so constructed would have $O(V^3)$ rules. In the discussion below, we give a more efficient algorithm that runs in $O(VE + V^2\log(V))$. This is a considerable improvement because most real-world networks for which this formulation computes centralities are sparse, i.e, $E \sim O(V)$.

The key question to ask is: what is the expected value of the marginal contribution of v_i through node $v_j \neq v_i$ to the value of coalition C_i? Let this marginal contribution be denoted $MC(v_i, v_j)$. Clearly:

$$MC(v_i, v_j) = \begin{cases} 0 & \text{if distance}(v_i, v_j) \geq d(v_j, C_i) \\ f(\text{distance}(v_i, v_j)) - f(d(v_j, C_i)) & \text{else} \end{cases}$$

Let $D_{v_j} = \{d_1, d_2 ... d_{|V|-1}\}$ be the distances of node v_j from all other nodes in the network, sorted in increasing order. Let the nodes corresponding to these distances be $\{w_1, w_2 ... w_{|V|-1}\}$ respectively. Let $k_{ij} + 1$ be the number of nodes (out of these $|V| - 1$) whose distances to v_j are $\leq$ distance(v_i, v_j). Let $w_{k_{ij}+1} = v_i$ (i.e, among all nodes that have the same distance from v_j as v_i, v_i is placed last in the increasing order).

We use *literal* w_i to mean $w_i \in C_i$ and the literal $\overline{w_i}$ to mean $w_i \notin C_i$. Define a sequence of boolean variables $p_k = \overline{v_j} \wedge \overline{w_1} \wedge \overline{w_2} \wedge ... \wedge \overline{w_k}$ for each $0 \leq k \leq |V| - 1$. Finally denote expressions of the form $MC(v_i, v_j | F)$ to mean the marginal contribution of v_i to C_i through v_j given that the coalition C_i satisfies the boolean expression F.

$$\begin{aligned} MC(v_i, v_j | p_{k_{ij}+1} \wedge w_{k_{ij}+2}) &= f(d_{k_{ij}+1}) - f(d_{k_{ij}+2}) \\ MC(v_i, v_j | p_{k_{ij}+2} \wedge w_{k_{ij}+3}) &= f(d_{k_{ij}+1}) - f(d_{k_{ij}+3}) \\ \vdots \qquad & \qquad \vdots \qquad \qquad \vdots \\ MC(v_i, v_j | p_{|V|-2} \wedge w_{|V|-1}) &= f(d_{k_{ij}+1}) - f(d_{|V|-1}) \\ MC(v_i, v_j | p_{|V|-1}) &= f(d_{k_{ij}+1}) \end{aligned}$$

With this notation, we obtain expressions for $MC(v_i, v_j)$ by splitting over the above *mutually exclusive* and *exhaustive* (i.e, covering all possible non-zero marginal contributions) cases.

The probabilities $\Pr(p_k \wedge w_{k+1})$ are found by elementary combinatorics which gives:

$$\Pr(p_k \wedge w_{k+1}) = \frac{k!}{(k+2)!} = \frac{1}{(k+1)(k+2)} \; \forall\, 1 + k_{ij} \leq k \leq |V| - 2$$

Using the $MC(v_i, v_j)$ equations and the probabilities $\Pr(p_k \wedge w_{k+1})$:

$$\begin{aligned} E[MC(v_i, v_j)] &= \left[\sum_{k=1+k_{ij}}^{|V|-2} \frac{f(\text{distance}(v_i, v_j)) - f(d_{k+1})}{(k+1)(k+2)} \right] + \frac{f(\text{distance}(v_i, v_j))}{|V|} \\ &= \frac{f(\text{distance}(v_i, v_j))}{k_{ij} + 2} - \sum_{k=k_{ij}+1}^{|V|-2} \frac{f(d_{k+1})}{(k+1)(k+2)} \end{aligned}$$

For $v_i = v_j$, a similar analysis produces:

$$E[MC(v_i, v_i)] = f(0) - \sum_{k=0}^{|V|-2} \frac{f(d_{k+1})}{(k+1)(k+2)}$$

Finally the exact SVs are given by:

$$SV_{g_4}(v_i) = \sum_{v_j \in V(G)} E[MC(v_i, v_j)]$$

Algorithm 4 implements the above formulas. For each vertex v, a vector of distances to every other vertex is first computed using Dijkstra's algorithm [14]. This yields a vector D_v that is already sorted in increasing order. This vector is then traversed in reverse, to compute the backwards cumulative sum $\sum \frac{f(d_{k+1})}{(k+1)(k+2)}$. At each step of the backward traversal, the SV of the appropriate node w is updated according to the $E[MC(w, v)]$ equation. After the traversal, the SV of v itself is updated according to the $E[MC(v, v)]$ equation. This process is repeated for all nodes v so that at the end of the algorithm, all SVs have been computed exactly in $O(VE + V^2 \log(V))$ time.

Algorithm 4. Computing SVs for Game 4

Input: Weighted graph $G(V, E, W)$, function $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$
Output: SVs of all nodes in G for game g_4

```
Initialise: ∀v ∈ V(G) set ShapleyValue[v]=0;
foreach v ∈ V(G) do
    [Distances D, Nodes w] = Dijkstra(v,G);
    sum = 0; index = |V|-1; prevDistance = -1,
    prevSV = -1;
    while index > 0 do
        if D(index) == prevDistance then
            currSV = prevSV;
        else
            currSV = f(D(index))/(1+index) − sum;
        end
        ShapleyValue[w(index)] += currSV;
        sum += f(D(index))/(index(1+index));
        prevDistance = D(index), prevSV =
        currSV;
        index--;
    end
    ShapleyValue[v] += f(0) − sum;
end
return ShapleyValue;
```

4 Summary and Conclusions

Game	Graph	$\nu(C)$	Complexity
g_1	UW	≤ 1 degree away	$V + E$
g_2	UW	$\geq k$ neighbors $\in C$	$V + E$
g_3	W	$\leq d_{\text{cutoff}}$ away	$VE + V^2 \text{log} V$
g_4	W	$\sum_{v_i} f(d(v_i, C))$	$VE + V^2 \text{log} V$

$\{W = \text{weighted},\ UW = \text{unweighted}\}$

The table to the left presents a brief summary of the SV algorithms discussed in this paper. These algorithms enable efficient centrality computation for many real-world applications including the analysis of social networks, information diffusion, spread of epidemics, biological and biochemical networks, viral marketing and internet/web phenomena. The conclusion is that many centrality-related co-operative games of interest played on real-life networks can in fact be solved for SVs analytically. The resulting algorithms are not only error-free but also run in polynomial time and in practice, much faster than Monte-Carlo methods.[2]

References

1. Suri, N.R., Narahari, Y.: Determining the top-k nodes in social networks using the Shapley Value. In: AAMAS 2008: Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems, pp. 1509–1512 (2008)
2. Gómez, D., González-Arangüena, E., Manuel, C., Owen, G., Del Pozo, M., Tejada, J.: Centrality and power in social networks: A game theoretic approach. Mathematical Social Sciences 46(1), 27–54 (2003)
3. Koschützki, D., Lehmann, K.A., Peeters, L., Richter, S., Tenfelde-Podehl, D., Zlotowski, O.: Centrality indices. Network analysis. In: Brandes, U., Erlebach, T. (eds.) Network Analysis. LNCS, vol. 3418, pp. 16–61. Springer, Heidelberg (2005)
4. Watts, D.J.: Six degrees: The science of a connected age. W.W. Norton & Company, New York (2004)
5. Kempe, D., Kleinberg, J., Tardos, É.: Influential nodes in a diffusion model for social networks. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 1127–1138. Springer, Heidelberg (2005)
6. Suri, N.R., Narahari, Y.: A Shapley Value based approach to discover influential nodes in social networks. IEEE Transactions on Automation Science and Engineering 99, 1–18 (2010)
7. Deng, X., Papadimitriou, C.: On the complexity of cooperative solution concepts. Mathematics of Operations Research 19(2), 257–266 (1994)
8. Ieong, S., Shoham, Y.: Marginal contribution nets: a compact representation scheme for coalitional games. In: EC 2005: Proceedings of the Sixth ACM Conference on Electronic Commerce, pp. 193–202 (2005)
9. Bondy, J.A., Murty, U.S.R.: Graph theory and its applications. Macmillan, Basingstoke (1976)
10. Gibbons, R.: A primer in game theory. Harvester Wheatsheaf (1994)

[2] We have performed simulations on public domain real-life networks; the results show that the algorithms proposed in this paper are not only accurate but also deliver significant speedups over Monte-Carlo simulation. However, due to space constraints, we are unable to present these results here.

11. Bonacich, P.: Power and centrality: A family of measures. American Journal of Sociology 92(5), 1170–1182 (1987)
12. Valente, T.W.: Social network thresholds in the diffusion of innovations. Social Networks 18(1), 69–89 (1996)
13. Newman, M.E.J.: Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. Physical Review E, 64(1), 0161321–0161327 (2001)
14. Cormen, T.H.: Introduction to Algorithms. MIT Press, Cambridge (2001)

On Approximate Nash Equilibria in Network Design*

Susanne Albers and Pascal Lenzner

Department of Computer Science, Humboldt-Universität zu Berlin,
Unter den Linden 6, 10099 Berlin, Germany
{albers,lenzner}@informatik.hu-berlin.de

Abstract. We study a basic network design game where n self-interested agents, each having individual connectivity requirements, wish to build a network by purchasing links from a given set of edges. A fundamental cost sharing mechanism is Shapley cost sharing that splits the cost of an edge in a fair manner among the agents using the edge. In this paper we investigate if an optimal minimum-cost network represents an attractive, relatively stable state that agents might want to purchase. We resort to the concept of α-approximate Nash equilibria. We prove that for single source games in undirected graphs, any optimal network represents an $H(n)$-approximate Nash equilibrium, where $H(n)$ is the n-th Harmonic number. We show that this bound is tight. We extend the results to cooperative games, where agents may form coalitions, and to weighted games. In both cases we give tight or nearly tight lower and upper bounds on the stability of optimal solutions. Finally we show that in general source-sink games and in directed graphs, minimum-cost networks do not represent good states.

1 Introduction

Today many networks are not built and maintained by a central authority but rather by a large number of economic agents that usually have selfish interests. As a result, game-theoretic approaches for modeling network formation and agent behavior have received considerable research interest over the past years, see e.g. [2,4,5,6,7,8,9,10,11,13,16,19,21].

We study a very basic network design game that has received a lot of attention [1,4,5,6,7,12,14,17]. Let $G = (V, E, c)$ be a graph with a non-negative cost function $c : E \mapsto \mathbb{R}_+^0$. The graph may be directed or undirected. There are n selfish agents, each having to connect a set of terminals in G. A strategy $S_i \subseteq E$ of an agent i is an edge set connecting the desired terminals. Considering all agents, we obtain a combination $\mathcal{S} = (S_1, \ldots, S_n)$ of strategies. Edges used by the agents have to be paid for. A fundamental cost sharing mechanism is Shapley cost sharing, proposed by Anshelevich et al. [4] for network design games. In Shapley cost

* Work supported by a Gottfried Wilhelm Leibniz Award of the German Research Foundation.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 14–25, 2010.

sharing the cost of an edge is split in a fair manner among the agents using the edge. More specifically, in an *unweighted game*, if k agents use an edge e, then each of them pays a share of $c(e)/k$. Thus, given a combination $\mathcal{S}$ of strategies, the cost of an agent i, $1 \leq i \leq n$, is $cost_i(\mathcal{S}) = \sum_{e \in S_i} c(e)/|\{j : e \in S_j\}|$. In a *weighted game*, each agent i has a positive weight w_i and pays a share proportional to its weight. For any edge $e \in S_i$, agent i pays a share of $c(e)w_i/W_e$, where $W_e = \sum_{j:e \in S_j} w_j$ is the total weight of the agents j using e in their strategies. The cost of agent i is $cost_i(\mathcal{S}) = \sum_{e \in S_i} c(e)w_i/W_e$.

Previous work has analyzed stable states in which agents have no incentive to deviate from their strategies. In a standard non-cooperative game a combination $\mathcal{S}$ of strategies forms a *Nash equilibrium* if no agent has a better strategy with a strictly smaller cost if all other agents adhere to their strategies. In cooperative games, where coordination among agents is allowed, one is interested in *strong Nash equilibria* that are resilient to deviations of coalitions of agents [3]. A combination $\mathcal{S}$ of strategies forms a strong Nash equilibrium if there exists no coalition of agents that can jointly change strategy such that every agent of the coalition achieves a strictly smaller cost. There exist two performance measures evaluating Nash equilibria relative to globally optimal solutions. The *price of anarchy* is the maximum ratio of the total cost incurred by any Nash equilibrium to the cost paid by an optimal solution [18]. The *price of stability* is the minimum ratio, i.e. the cost ratio of the best Nash equilibrium relative to the optimum [4]. Anshelevich et al. [4] showed that, for unweighted non-cooperative games, the price of anarchy is n while the price of stability is $H(n)$. Here $H(n) = \sum_{i=1}^{n} 1/i$ is the n-th Harmonic number, which is closely approximated by the natural logarithm, i.e. $\ln(n+1) \leq H(n) \leq \ln n + 1$. For unweighted cooperative games the price of anarchy is $H(n)$ [1,12].

In this paper we study if an optimal solution – which is a minimum-cost network establishing the required connections – represents an attractive, relatively stable state that agents might want to purchase. If the n agents buy an optimal solution, which extra cost does any agent incur compared to a strategy deviation? The motivation for our study is twofold. (1) In Nash equilibria there exist agents that pay a high cost compared to the average agent cost in an optimal solution. In a worst-case equilibrium this cost factor can be as high as n; even in a best-case equilibrium the factor can be $H(n)$. With this information in mind the agents might be interested in purchasing an optimal solution provided that the incentive of a strategy deviation is not too high. (2) The only known protocol to reach a good equilibrium, attaining a price of stability of $H(n)$, relies on optimal solutions. Anshelevich et al. [4] showed that if the agents start in an optimal solution, then a sequence of improving moves converges to a Nash equilibrium whose cost is at most $H(n)$ times the optimum cost. Hence, if agents start in an optimal solution, they might as well consider remaining in this solution provided that the state has favorable properties.

We address the above issues by studying *approximate Nash equilibria* in which the equilibrium constraint is relaxed [5,7]. In a non-cooperative game a combination $\mathcal{S}$ of strategies forms an α-approximate Nash equilibrium, for some

$\alpha \geq 1$, if no agent can improve its cost by a factor of more than α assuming that all the other agents adhere to their strategies. Formally, for no agent i exists a strategy change S_i' such that $cost_i(S_1, \ldots, S_i', \ldots, S_n) < cost_i(\mathcal{S})/\alpha$. In cooperative games a combination $\mathcal{S}$ of strategies is an α-approximate strong Nash equilibrium if no coalition of agents can change strategy such all agents of the coalition improve their cost by a factor of more than α. More specifically, for no non-empty coalition I of agents exists a strategy change S_I' such that $cost_i(\mathcal{S}_I', \mathcal{S}_{-I}) < cost_i(\mathcal{S})/\alpha$ holds for all $i \in I$. Here $\mathcal{S}_{-I}$ is the vector of the original strategies of agents $i \notin I$.

We evaluate the quality of optimal solutions for a variety of settings. The main conclusion is that optimal solutions represent good states for single source games in undirected graphs. This holds true for unweighted games, considering both non-cooperative and cooperative agent behavior, as well as for weighted games. On the other hand, in general source-sink games and in directed graphs, optimal solutions do not represent satisfying states.

Previous work. Research on the network design game defined above was initiated by Anshelevich et al. [5]. In this first paper the authors considered general cost sharing schemes that are not restricted to Shapley cost sharing. The cost of an edge may be split in an arbitrary way among agents. Anshelevich et al. [5] considered undirected graphs. First they studied single source games in which each agent i has to connect one terminal t_i to a common source s, $1 \leq i \leq n$. They showed that the cost of an optimal solution can be shared among the agents such that the resulting strategies form a Nash equilibrium. Anshelevich et al. [5] also studied general source-sink games where each agent has to connect an arbitrary set of terminals. Here the cost on an optimal solution can shared such that the agents' strategies form a 3-approximate Nash equilibrium.

In a second paper Anshelevich et al. [4] investigated network design with Shapley cost sharing. They first focused on unweighted games and showed that in directed and undirected graphs the price of anarchy is n while the price of stability is upper bounded by $H(n)$. This upper bound of $H(n)$ is tight for directed graphs. Additionally Anshelevich et al. [4] studied weighted games and showed a lower bound of $\Omega(\max\{n, \log W\})$ on the price of stability, where W is the total weight of all the agents.

Further work on unweighted games was presented by Chekuri et al. [6] and Fiat et al. [14]. Both papers address single source games in undirected graphs. Chekuri et al. [6] showed that the price of anarchy is $O(\sqrt{n}\log^2 n)$ if agents join the game sequentially and perform best-response moves. Fiat et al. [14] proved that the price of stability is $O(\log\log n)$ if each vertex of the graph is the terminal of some agent. Chen and Roughgarden [7] further investigated weighted games in directed graphs. They assume that each agent has to connect a terminal pair (s_i, t_i) and proved that, for any $\alpha = \Omega(\log w_{\max})$, the price of stability of $O(\alpha)$-approximate Nash equilibria is $O((\log W)/\alpha)$. Here $w_{\max}$ is the maximum weight of any agent. In particular, there exists an $O(\log W)$-approximate Nash equilibrium whose cost is within a constant factor of optimal. Cooperative network design games were studied in [3,12]. It shows that the price of anarchy drops to $H(n)$. Finally, approximate pure Nash equilibria for a different class of graphical games were recently studied by Nguyen and Tardos [20].

Our contribution. We evaluate the stability of optimal solutions in network design games with Shapley cost sharing, complementing the existing results for this classical cost sharing mechanism. In Section 2 we present a comprehensive study of unweighted games. We focus mostly on single source games in undirected graphs. First, for non-cooperative games, we prove that any optimal solution represents an $H(n)$-approximate Nash equilibrium. We show that this bound is tight. There exist games in which an optimal solution does not form an α-approximate Nash equilibrium for any $\alpha < H(n)$.

Then, in Section 2, we investigate cooperative games where agents may coordinate their actions. We consider a general scenario where coalitions of up to c agents may be formed, for any $1 \leq c \leq n$. We prove that any optimal solution is a $2c(\ln(n/c)+2)$-approximate strong Nash equilibrium. The analysis is considerably more involved than that for non-cooperative games. More specifically we show that, given any tree establishing the required connections and any coalition I of agents, there exists one agent $i \in I$ whose cost shares along the path from t_i to the source do not grow too much. For cooperative games, allowing coalitions of up to c agents, we give a nearly matching lower bound: There exist games in which an optimal solution does not represent an α-approximate strong Nash equilibrium for $\alpha < c' \ln(n/c')$, where $c' = \min\{c, \lfloor n/e \rfloor\}$. Hence, for $c < \lfloor n/e \rfloor$ the bound is $\alpha < c \ln(n/c)$; for large coalitions of size $c \geq \lfloor n/e \rfloor$, the bound is $\lfloor n/e \rfloor$ and hence linear in n. This behavior is consistent with our upper bound. Moreover, in Section 2, we consider general source-sink games, in which each agent has to connect an individual set of terminals, as well as directed graphs. In both cases we show negative results, even for non-cooperative games. There are general source-sink games for which an optimal solution is an $\Omega(n)$-approximate Nash equilibrium. In directed graphs the approximation guarantee α can even be unbounded.

In Section 3 we study weighted games. We consider single source games in undirected graphs. We show that in non-cooperative games, any optimal solution is an α-approximate Nash equilibrium, where $\alpha = w_{\max} \sum_{k=0}^{n-1} 1/(w_{\max}+k)$. This bound is again tight. Optimal solutions generally do not form α-approximate Nash equilibria, for $\alpha < w_{\max} \sum_{k=0}^{n-1} 1/(w_{\max}+k)$. The latter expression is upper bounded by $w_{\max}(\ln(W/w_{\max})+1)$. Here $w_{\max}$ and W denote again the maximum and total weight of the agents.

Here we finally relate our results to those of Chen and Roughgarden [7] mentioned above for non-cooperative games. In this paper we evaluate the quality of optimal solutions, which are solutions of specific interest, and develop explicit bounds not resorting to O-notation. On the other hand, Chen and Roughgarden develop asymptotic trade-offs. For unweighted games these trade-offs imply the existence of an $O(\log n)$-approximate Nash equilibrium whose cost is within a constant factor of the optimum cost. The protocol starts in an optimal solution and then performs a sequence of improving deviations. Our results show that the protocol can, and indeed will, remain in the optimal solution.

2 Unweighted Games

In this section we study games with classical Shapley cost sharing, i.e. agents have uniform weights. If an edge e is used by k agents, then each agent has to pay a share of $c(e)/k$. We first consider the standard setting where agents are non-cooperating entities. Then we consider the setting where agents cooperate and may form coalitions. For both scenarios we focus on single source games in undirected graphs. More specifically given an undirected graph $G = (V, E, c)$, each agent i, $1 \leq i \leq n$, has to connect a terminal t_i to a common source s, where $t_i, s \in V$. Finally we address general source-sink games and games in directed graphs.

2.1 Non-cooperative Games

We first prove an upper bound on the quality of optimal solutions and then give a matching lower bound.

Theorem 1. *In single source games, any optimal solution represents an $H(n)$-approximate Nash equilibrium.*

Proof. Let E_{OPT} be the edge set used by an optimal solution to establish the required connections. As we study single source games, E_{OPT} forms a tree. Consider the combination $\mathcal{S}$ of strategies in which every agent i connects its terminal t_i to the common source s using only edges of E_{OPT}. Let P_i be the simple path used by agent i and let $cost_i(P_i)$ denote the corresponding cost paid by i within $\mathcal{S}_{OPT}$. We observe that path P_i is unique in E_{OPT}.

Now suppose that an agent i changes strategy and selects a different path Q_i, $Q_i \neq P_i$, in order to connect t_i to s. Let $cost_i(Q_i)$ be the associated cost incurred by agent i when performing this strategy change. We will show

$$cost_i(P_i) \leq H(n) cost_i(Q_i), \tag{1}$$

which establishes the theorem.

Let $v_1, \ldots, v_\ell$, $\ell \geq 2$, be the vertices where P_i and Q_i separate and merge again; Figure 1 shows an example. More specifically, starting at t_i, paths P_i and Q_i first traverse a common subpath $P_i(1) = Q_i(1)$ until reaching vertex v_1 where the two paths separate. Vertex v_1 may be equal to t_i, in which case paths $P_i(1) = Q_i(1)$ are empty. After v_1 path P_i traverses a subpath $P_i(2)$ while Q_i uses a subpath $Q_i(2)$. These subpaths use disjoint edge sets and meet again only at vertex v_2. In general, suppose that P_i and Q_i merge at a vertex v_j, with j being even. Then P_i and Q_i traverse a common subpath $P_i(j+1) = Q_i(j+1)$ until reaching v_{j+1}, where P_i and Q_i separate into disjoint subpaths $P_i(j+2)$ and $Q_i(j+2)$, meeting again at v_{j+2}. Finally, let $P_i(\ell+1) = Q_i(\ell+1)$ be the subpath between v_ℓ and s. For any odd number j, the subpath $P_i(j) = Q_i(j)$ may be empty. For any even j, the subpath $Q_i(j)$ contains at least one edge that does not belong to E_{OPT} because the optimal solution does not contain cycles. Let $Q'_i(j)$, with $Q'_i(j) \subseteq Q_i(j)$, be the set of edges not contained in E_{OPT}.

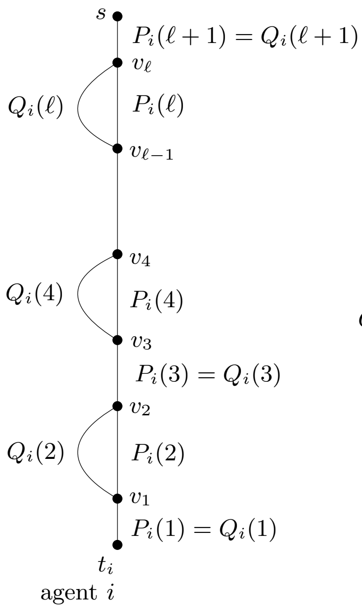


Fig. 1. Paths P_i and Q_i

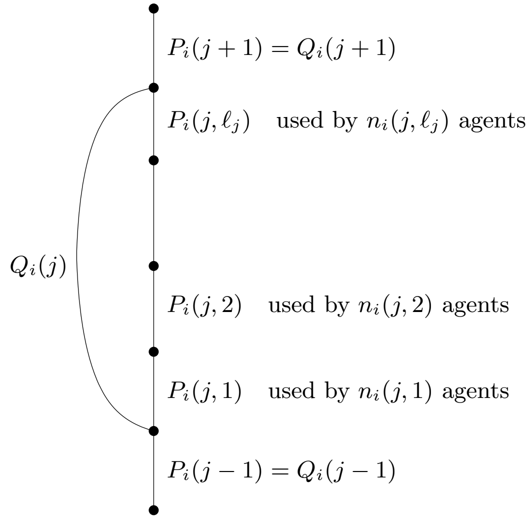


Fig. 2. Subpaths $P_i(j)$ and $Q_i(j)$

Let $cost_i(P_i(j))$ and $cost_i(Q_i(j))$ denote the costs paid by agent i on $P_i(j)$ and $Q_i(j)$, respectively, $1 \leq j \leq \ell+1$. We have $cost_i(P_i) = \sum_{j=1}^{l+1} cost_i(P_i(j))$ and $cost_i(Q_i) = \sum_{j=1}^{l+1} cost_i(Q_i(j))$, where $cost_i(P_i(j)) = cost_i(Q_i(j))$ for any odd index j. We will prove $cost_i(P_i(j)) \leq H(n) cost_i(Q_i(j))$, for any even j, which implies inequality (1).

Consider a fixed even j and partition $P_i(j)$ into a sequence of maximal subpaths $P_i(j,1), \ldots, P_i(j,\ell_j)$ such that, for any $1 \leq k \leq \ell_j$, the number of agents using a given edge e of $P_i(j,k)$ in E_{OPT} is the same for all the edges of this subpath, cf. Figure 2. Let $n_i(j,k)$ be the number of agents using the edges of $P_i(j,k)$ within E_{OPT}, for any $1 \leq k \leq \ell_j$. Since the subpaths are maximal we have $n_i(j,1) < \ldots < n_i(j,\ell_j)$. As E_{OPT} is a minimum cost tree we have $cost(P_i(j,k)) \leq cost(Q_i'(j))$, where $cost(P_i(j,k))$ and $cost(Q_i')$ denote the total edge costs of subpath $P_i(j,k)$ and edge set $Q_i'(j)$, respectively, $1 \leq k \leq \ell_j$. If we had $cost(P_i(j,k)) > cost(Q_i'(j))$, then in E_{OPT} we could replace $P_i(j,k)$ by $Q_i'(j)$ obtaining a solution with a strictly smaller cost. The connectivity requirements would still be maintained as agents using $P_i(j,k)$ in E_{OPT} could traverse subpaths $P_i(j,k-1), \ldots, P_i(j,1)$ and $Q_i(j)$ to reach v_j, from where they could again follow their original path to source s. In E_{OPT} agent i pays a share of $cost(P_i(j,k))/n_i(j,k)$ for $P_i(j,k)$, where $cost(P_i(j,k))$ is the total cost of edges on $P_i(j,k)$. Summing over all k and making use of the fact that the sequence $n_i(j,k)$ is strictly increasing with $n_i(j,1) \geq 1$, we obtain that the total cost paid by agent i on $P_i(j)$ is $cost_i(P_i(j)) = \sum_{k=1}^{\ell_j} cost(P_i(j,k))/n_i(j,k) \leq$

$\sum_{k=1}^{\ell_j} cost(P_i(j,k))/k \leq H(n) cost(Q'_i(j))$. As $Q'_i(j)$ is not part of E_{OPT}, agent i has to fully cover its edge cost when traversing $Q_i(j)$ and hence $cost(Q'_i(j)) \leq cost_i(Q_i(j))$. We conclude $cost_i(P_i(j)) \leq H(n) cost_i(Q_i(j))$. □

Theorem 2. *There exists a single source game in which the unique optimal solution does not represent an α-approximate Nash equilibrium, for any $\alpha < H(n)$.*

Proof. Consider a graph consisting of $n+1$ vertices $v_1, \ldots, v_{n+1}$ and n edges $e_i = \{v_i, v_{i+1}\}$, $1 \leq i \leq n$, cf. Figure 3. Associated with each v_i, $1 \leq i \leq n$, is one agent that wishes to connect this vertex to the source $s = v_{n+1}$. Each edge e_i, $1 \leq i \leq n$, has cost 1. Additionally there is an edge $e_0 = \{v_1, v_{n+1}\}$ of cost $1+\epsilon$, where $\epsilon > 0$ is an arbitrarily small constant. The unique optimal solution consists of the set of edges e_i, $1 \leq i \leq n$. In this solution, agent 1 pays a cost of $H(n)$. On the other hand choosing edge e_0, agent 1 incurs a cost of only $1+\epsilon$. □

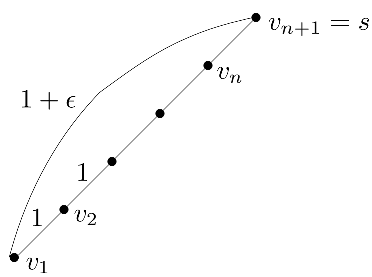


Fig. 3. A single source game without cooperation

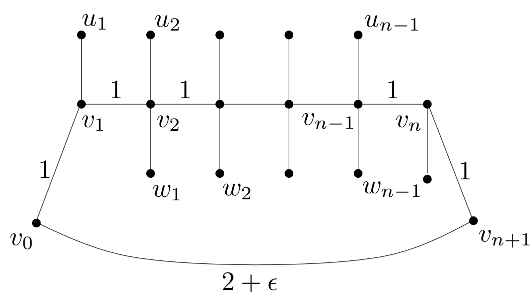


Fig. 4. A source-sink game

2.2 Cooperative Games

We study general cooperative games in which coalitions of up to c agents may be formed, for any $1 \leq c \leq n$.

Theorem 3. *In single source games, any optimal solution represents an α-approximate strong Nash equilibrium, where $\alpha = 2c(\ln(n/c) + 2)$, if coalitions up to size c are allowed.*

In order to establish the theorem, we first prove a property of trees T in which agents connect terminals to the root of T using the edges of the tree. The property holds for any tree T but when using the property in the proof of Theorem 3, T will be an optimal solution of a single source game. So let T be an arbitrary tree with root s. There are n agents, each of which has to connect a terminal of T to s using the edges of T. Let A denote the set of agents i whose terminal t_i is different from s. For any agent $i \in A$, let P_i be the path from t_i to s in T. We partition P_i into maximal subpaths $P_i(1), \ldots, P_i(l_i)$ such that, for any subpath, the number of agents using the edges of the subpath does not vary. Let $n_i(j)$ be the number of agents using the edges of $P_i(j)$, $1 \leq j \leq l_i$. Define

$$N_i(T) = \sum_{j=1}^{l_i} \frac{1}{n_i(j)},$$

which intuitively is the sum of the fractions paid by agent i on $P_i(1), \ldots, P_i(l_i)$, ignoring edge costs. The following lemma states that in any non-empty coalition $I \subseteq A$ there exists an agent i whose value $N_i(T)$ is logarithmic in $|A|/|I|$.

Lemma 1. *Let T be an arbitrary tree and A be the set of agents whose terminal is not equal to the root of T. For any $I \subseteq A$, $I \neq \emptyset$, there exists an $i \in I$ satisfying $N_i(T) \leq 2\ln(\frac{2|A|}{|I|}) + 1$.*

Proof. Due to space limitations we present the main ideas of the proof. A complete proof is given in the full version of the paper.

We prove a slightly stronger bound on $N_i(T)$. Given T and A, a vertex $v \neq s$ in T is called a *branching vertex* if v has at least two children rooting subtrees both of which contain terminals. Let B be the set of branching vertices. We will prove

$$N_i(T) \leq 2\ln\left(\frac{|A|+|B|}{|I|}\right) + 1. \tag{2}$$

The lemma then follows because $|B| < |A|$.

We prove (2) by induction on the number m of edges of T. In the base case we have $m = 1$. The tree consists of a single edge $\{v, s\}$ and A is the set of agents that have to connect v to s. For any $I \subseteq A$, $I \neq \emptyset$, and any $i \in I$ there holds $N_i(T) = 1/|A| \leq 1 \leq 2\ln(\frac{|A|+|B|}{|I|}) + 1$.

Next consider a tree T with $m > 1$ edges. If there is an agent $i \in I$ whose terminal t_i is equal to a child of s, then the analysis is simple. For this agent we have $N_i(T) \leq 1$ and as above we conclude $N_i(T) \leq 2\ln((|A| + |B|)/|I|) + 1$ because $|A| \geq |I|$ and $|B| \geq 0$. In the following we assume that, for no agent $i \in I$, the terminal t_i is equal to a child of s. We distinguish two cases depending on whether s has a degree of 1 or a degree larger than 1.

Suppose that s has degree 1. Let $\{s', s\}$ be the edge adjacent to s in T, and let T' be the tree rooted at s'. Let $A' \subseteq A$, be the set of agents i whose terminal t_i is a vertex of T' but not equal to the root s'. There holds $I \subseteq A'$ because we assume that, for no agent of I, the terminal is equal to a child of s. For any $i \in I$, consider the path P_i from t_i to s and the path P'_i from t_i to s'. Obviously P_i consists of P'_i followed by edge $\{s', s\}$. Partition both P_i and P'_i into maximal subpaths $P_i(1), \ldots, P_i(l_i)$ and $P'_i(1), \ldots, P'_i(l'_i)$, respectively, such that the edges of a subpath are used by a non-varying number of agents. Let $n_i(j)$ and $n'_i(j)$ be the number of agents using $P_i(j)$ and $P'_i(j)$, respectively. We have $P_i(j) = P'_i(j)$ and hence $n_i(j) = n'_i(j)$, for $j = 1, \ldots, l'_i - 1$. If the number $n'_i(l'_i)$ of agents using $P'_i(l'_i)$ is equal to the number of agents using edge $\{s', s\}$, then $l_i = l'_i$ and $P_i(l_i)$ consists of $P'_i(l_i)$ followed by $\{s', s\}$. Otherwise $l_i = l'_i + 1$ as well as $P_i(l'_i) = P'_i(l'_i)$ and $P_i(l_i) = \{s', s\}$.

By induction hypothesis, there exists an agent $i \in I$ satisfying $N_i(T') \leq 2\ln((|A'|+|B'|)/|I|)+1$, where B' is the set of branching vertices in T'. In the following we consider this fixed agent i. If $n'_i(l'_i)$ is equal to the number of agents using $\{s', s\}$, then we are done: As argued in the last paragraph $n_i(j) = n'_i(j)$, for $j = 1, \ldots, l'_i - 1$, and $l_i = l'_i$ which implies $n_i(l_i) = n'_i(l'_j)$. Hence $N_i(T) = N_i(T') \leq 2\ln((|A'|+|B'|)/|I|)+1 \leq 2\ln((|A|+|B|)/|I|)+1$ because $|A'| \leq |A|$ and $|B'| \leq |B|$.

If on the other hand $n'_i(l'_i)$ is not equal to the number of agents using $\{s', s\}$, then (a) there exists an agent in A whose terminal is equal to s' or (b) s' is a branching vertex. In case (a) we have $|A| > |A'|$ and in case (b) we have $|B| > |B'|$. Hence in both cases $|A|+|B| > |A'|+|B'|$. Again $n_i(j) = n'_i(j)$, for $j = 1, \ldots, l'_i - 1$, Since $l_i = l'_i + 1$ and $P_i(l'_i) = P'_i(l'_i)$, there holds $n_i(l'_i) = n'_i(l'_i)$ and $n_i(l_i) = 1/|A|$ because edge $\{s', s\}$ is used by all the agents of A. We obtain

$$\begin{aligned} N_i(T) = N_i(T') + \tfrac{1}{|A|} \quad &\leq \quad 2\ln\left(\tfrac{|A'|+|B'|}{|I|}\right) + 1 + \tfrac{2}{2|A|} \\ &\leq 2\left(\ln\left(\tfrac{|A'|+|B'|}{|I|}\right) + \tfrac{1}{|A|+|B|}\right) + 1 \\ &\leq 2\Big(\ln(|A'|+|B'|) + \tfrac{1}{|A'|+|B'|+1} - \ln(|I|)\Big) + 1. \end{aligned}$$

The second inequality holds because $|A| > |B|$ and hence $2|A| > |A|+|B|$. The third inequality follows since $|A|+|B| \geq |A'|+|B'|+1$. For any positive integer K there holds $\ln K + 1/(K+1) \leq \ln(K+1)$. Setting $K = |A'|+|B'|$ we obtain as desired $N_i(T) \leq 2(\ln(|A'|+|B'|+1) - \ln(|I|)) + 1 \leq 2\ln((|A|+|B|)/|I|)+1$.

The analysis of the case that the root s of T has a degree larger than 1 is omitted here. The main idea is to partition T into two trees T_1 and T_2 such that for any agent i whose terminal is in T_j, $j \in \{1,2\}$, there holds $N_i(T) = N_i(T_j)$. Using induction hypothesis one can then show that there exists an agent i with $N_i(T) \leq 2\ln((|A|+|B|)/|I|)+1$. □

Proof (of Theorem 3). Consider any optimal solution and let E_{OPT} be the corresponding edge set. Moreover, let $\mathcal{S}$ be the combination of strategies in which every agent i connects its terminal t_i to the common source s using only edges of E_{OPT}. In order to prove the theorem we show that if any non-empty coalition I of at most c agents changes strategy, then there exist an agent $i \in I$ whose cost before and after strategy change satisfies $\frac{1}{\alpha} cost_i(\mathcal{S}) \leq cost_i(\mathcal{S}_I, \mathcal{S}_{-I})$, where $\alpha = 2c(\ln(n/c)+2)$.

If a coalition I contains an agent i whose terminal t_i is equal to the source s, then there is nothing to show because for this agent $cost_i(\mathcal{S}) = 0$ and the desired inequality trivially holds. Hence in the following we consider non-empty coalitions I not containing an agent i whose terminal is equal to s.

Let A be the set of agents whose terminal is not equal to s. Consider any non-empty coalition $I \subseteq A$ of size at most c. The optimal solution E_{OPT} forms a tree and hence by Lemma 1 there exists an agent $i \in I$ with $N_i(E_{OPT}) \leq 2\ln(2|A|/|I|)+1$. Fix this agent i. We will prove that if I performs any strategy change, for this agent i the desired inequality holds.

For agent i let P_i be the path connecting t_i to s in E_{OPT}. Let Q_i be the path used by the agent when I changes strategy. As in the proof of Theorem 1 we partition P_i and Q_i into subpaths $P_i(1), \ldots, P_i(l+1)$ and $Q_i(1), \ldots, Q_i(l+1)$ along the vertices $v_1 \ldots, v_l$ where P_i and Q_i separate and merge. Let $cost_i(P(j))$ be the cost incurred by agent i for $P_i(j)$ before strategy change, $1 \leq j \leq l+1$. Similarly, let $cost_i(Q(j))$ be the cost paid by the agent for $Q_i(j)$ after strategy change, $1 \leq j \leq l+1$. We have $P_i(j) = Q_i(j)$, for any odd number j, and hence $\frac{1}{|I|} cost_i(P(j)) \leq cost_i(Q(j))$ because at most $|I|-1$ additional agents of I can join edges of $P_i(j)$ after strategy change. Since $|I| \leq c$ this implies $\frac{1}{\alpha} cost_i(P(j)) \leq cost_i(Q(j))$ for any odd number j. In the following we show that the last inequality also holds for any even j.

For any even j we partition $P_i(j)$ into maximal subpaths $P_i(j,1), \ldots, P_i(j,l_j)$ such that all the edges of a subpath $P_i(j,k)$ are used by the same number $n_i(j,k)$ of agents, $1 \leq k \leq l_j$, considering the time before strategy change. Let $Q_i'(j) \subseteq Q_i(j)$ be the non-empty set of edges not contained in E_{OPT}. For any path π let $cost(\pi)$ be the total cost of the edges of π. There holds $cost(P_i(j,k)) \leq cost(Q_i'(j))$, for any $1 \leq k \leq l_j$ and $cost(Q_i'(j)) \leq cost(Q_i(j))$. Hence

$$cost_i(P_i(j)) = \sum_{k=1}^{l_j} cost(P_i(j,k))/n_i(j,k) \leq cost(Q_i'(j)) \sum_{k=1}^{l_j} 1/n_i(j,k).$$

Consider the partitioning of P_i into maximal subpaths such that the edges of a subpath are used by the same number of agents. Paths $P_i(j,1), \ldots, P_i(j,l_j)$ are a subsequence of this partition and hence $\sum_{k=1}^{l_j} 1/n_i(j,k) \leq N_i(E_{OPT})$. Moreover $cost_i(Q_i'(j)) \geq cost(Q_i'(j))/|I|$ because the cost of the edges of $Q_i'(j)$, which are not part of E_{OPT}, must be fully covered by the coalition I and agent i pays a share of at least $1/|I|$. Thus $cost(P_i(j)) \leq |I| cost_i(Q_i'(j)) N_i(E_{OPT}) \leq |I| \cdot N_i(E_{OPT}) cost_i(Q_i(j))$. By our choice of agent i and Lemma 1, $N_i(E_{OPT}) \leq 2\ln(2|A|/|I|) + 1 \leq 2\ln(2n/|I|) + 1$. We obtain $cost(P_i(j)) \leq |I|(2\ln(2n/|I|) + 1) cost_i(Q_i(j)) \leq c(2\ln(2n/c)+1) cost_i(Q_i(j))$. The last inequality holds because $|I|(2\ln(2n/|I|) + 1)$ is increasing in $|I|$. We conclude $cost(P_i(j)) \leq 2c(\ln(n/c) + 2) cost_i(Q_i(j))$ and, as desired, $\frac{1}{\alpha} cost(P_i(j)) \leq cost(Q_i(j))$. □

Theorem 4. *There exists a single source game, allowing coalitions of size up to c, in which the unique optimal solution does not represent an α-approximate strong Nash equilibrium, for any $\alpha < c' \ln(n/c')$, where $c' = \min\{c, \lfloor n/e \rfloor\}$,*

The proof is given in the full version of the paper.

2.3 Source-Sink Games and Directed Graphs

A natural question is if the results of the previous sections can be extended to (a) general source-sink games in which each agent has to connect an individual set of terminals or (b) to directed graphs. Unfortunately, this is not the case. Even for non-cooperative games we can show high lower bounds on the approximation factor α.

Theorem 5. *There exists a general source-sink game in which the unique optimal solution represents an α-approximate Nash equilibrium with $\alpha = \Omega(n)$.*

Proof. Consider the graph depticted in Figure 4, shown at the end of Section 2.1. There are n vertices $v_1, \ldots, v_n$ which are connected by edges $e_i = \{v_i, v_{i+1}\}$, $1 \leq i \leq n-1$. Furthermore, there are vertices $u_1, \ldots, u_{n-1}$, and $w_1, \ldots, w_{n-1}$ with corresponding edges $\{u_i, v_i\}$ and $\{v_{i+1}, w_i\}$, $1 \leq i \leq n-1$. Agent i, $1 \leq i \leq n-1$, has to connect u_i and w_i. There are two additional vertices v_0 and v_{n+1} with associated edges $e_0 = \{v_0, v_1\}$ and $e_n = \{v_n, v_{n+1}\}$. Agent n has to connect terminals v_0 and v_{n+1}. All edges mentioned so far have a cost of 1. Finally, there is an edge $e' = \{v_0, v_{n+1}\}$ of cost $2 + \epsilon$. The unique optimal solution purchases all the edges e_i, $0 \leq i \leq n$, in addition to $\{u_i, v_i\}$ and $\{v_{i+1}, w_i\}$, $1 \leq i \leq n-1$. In this solution agent n pays a cost of $2 + (n-1)/2 \geq n/2$, whereas purchasing edge e' incurs a cost of $2 + \epsilon$. □

Theorem 6. *For any C, there exist single source games in directed graphs in which an optimal solution does not form a C-approximate Nash equilibrium.*

The proof is given in the full version of the paper.

3 Weighted Games

We scale the agents' weights such that the minimum weight is equal to 1 and hence $w_i \geq 1$, for all the agents. Let $w_{\max} = \max_{1 \leq i \leq n} w_i$ be the maximum weight of any agent. We consider single source games in undirected graphs and extend Theorems 1 and 2. Again we give tight upper and lower bounds on the value of α such that any optimal solution represents an α-approximate Nash equilibrium. The proofs of the following Theorems 7 and 8 are presented in the full version of this paper. The expression $\alpha = w_{\max} \sum_{k=0}^{n-1} 1/(w_{\max} + k)$ is upper bounded by $w_{\max}(\ln(W/w_{\max}) + 1)$.

Theorem 7. *In single source games, any optimal solution represents an α-approximate Nash equilibrium, where $\alpha = w_{\max} \sum_{k=0}^{n-1} 1/(w_{\max} + k)$.*

Theorem 8. *There exists a single source game in which the unique optimal solution does not represent an α-approximate Nash equilibrium, for any $\alpha < w_{\max} \sum_{k=0}^{n-1} 1/(w_{\max} + k)$.*

References

1. Albers, S.: On the value of coordination in network design. SIAM Journal on Computing 38, 2273–2302 (2009)
2. Alon, N., Demaine, E.D., Hajiaghayi, M., Leighton, T.: Basic network creation games. In: Proc. 22nd Annual ACM Symposium on Parallel Algorithms and Architectures, pp. 106–113 (2010)
3. Aumann, R.J.: Acceptable points in general cooperative n-person games. In: Tucker, A.W., Luce, R.D. (eds.) Contributions to the Theory of Games, vol. IV. Annals of Mathematics Studies, vol. 40, pp. 287–324 (1959)

4. Anshelevich, E., Dasgupta, A., Kleinberg, J.M., Tardos, E., Wexler, T., Roughgarden, T.: The price of stability for network design with fair cost allocation. SIAM Journal on Computing 38, 1602–1623 (2008)
5. Anshelevich, E., Dasgupta, A., Tardos, E., Wexler, T.: Near optimal network design with selfish agents. Theory of Computing 4, 77–109 (2008)
6. Chekuri, C., Chuzhoy, J., Lewin-Eytan, L., Naor, J., Orda, A.: Non-cooperative multicast and facility location games. In: Proc. 7th ACM Conference on Electronic Commerce (EC), pp. 72–81 (2006)
7. Chen, H.-L., Roughgarden, T.: Network design with weighted players. Theory of Computing Systems 45, 302–324 (2009)
8. Chen, H.-L., Roughgarden, T., Valiant, G.: Designing network protocols for good equilibria. SIAM Journal on Computing 39, 1799–1832 (2010)
9. Corbo, J., Parkes, D.: The price of selfish behavior in bilateral network formation. In: Proc. 24th Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 99–107 (2005)
10. Demaine, E.D., Hajiaghayi, M., Mahini, H., Zadimoghaddam, M.: The price of anarchy in cooperative network creation games. In: Proc. 26th International Symposium on Theoretical Aspects of Computer Science (STACS), pp. 301–312 (2009)
11. Devanur, N.R., Mihail, M., Vazirani, V.V.: Strategyproof cost-sharing mechanisms for set cover and facility location games. Decision Support Systems 39, 11–22 (2005)
12. Epstein, A., Feldman, M., Mansour, Y.: Strong equilibrium in cost sharing connection games. In: Proc. 8th ACM Conference on Electronic Commerce, pp. 84–92 (2007)
13. Fabrikant, A., Luthra, A., Maneva, E., Papadimitriou, C.H., Shenker, S.: On a network creation game. In: Proc. 22nd Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 347–351 (2003)
14. Fiat, A., Kaplan, H., Levy, M., Olonetsky, S., Shabo, R.: On the price of stability for designing undirected networks with fair cost allocations. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 608–618. Springer, Heidelberg (2006)
15. Gupta, A., Srinivasan, A., Tardos, E.: Cost-sharing mechanisms for network design. Algorithmica 50, 98–119 (2008)
16. Halevi, Y., Mansour, Y.: A network creation game with nonuniform interests. In: Deng, X., Graham, F.C. (eds.) WINE 2007. LNCS, vol. 4858, pp. 287–292. Springer, Heidelberg (2007)
17. Hoefer, M.: Non-cooperative tree creation. Algorithmica 53, 104–131 (2009)
18. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. In: Meinel, C., Tison, S. (eds.) STACS 1999. LNCS, vol. 1563, pp. 404–413. Springer, Heidelberg (1999)
19. Laoutaris, N., Poplawski, L.J., Rajaraman, R., Sundaram, R., Teng, S.-H.: Bounded budget connection (BBC) games or how to make friends and influence people, on a budget. In: Proc. 27th Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 165–174 (2008)
20. Nguyen, T., Tardos, E.: Approximate pure Nash equilibria via Lovász local lemma. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 160–171. Springer, Heidelberg (2009)
21. Pal, M., Tardos, E.: Group strategyproof mechanisms via primal-dual algorithms. In: Proc. 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 584–593 (2003)

The Efficiency of Fair Division with Connected Pieces

Yonatan Aumann and Yair Dombb

Department of Computer Science, Bar-Ilan University, Ramat Gan 52900, Israel
aumann@cs.biu.ac.il, yair.biu@gmail.com

Abstract. We consider the issue of fair division of goods, using the cake cutting abstraction, and aim to bound the possible degradation in social welfare due to the fairness requirements. Previous work has considered this problem for the setting where the division may allocate each player any number of unconnected pieces. Here, we consider the setting where each player must receive a single connected piece. For this setting, we provide tight bounds on the maximum possible degradation to both utilitarian and egalitarian welfare due to three fairness criteria — proportionality, envy-freeness and equitability.

1 Introduction

Cake Cutting. The problem of fair division of goods is the subject of extensive literature in the social sciences, law, economics, game theory and more. The famous "cake cutting" problem abstracts the fair division problem in the following way. There are n players wishing to divide between themselves a single "cake". The different players may value differently the various sections of the cake, e.g. one player may prefer the marzipan, another the cherries, and a third player may be indifferent between the two. The goal is to obtain a "fair" division of the cake amongst the players. There are several possible definitions to what constitutes a "fair" division, with *proportionality*, *envy-freeness* and *equitability* being the major fairness criteria considered (these notions will be defined in detail later). Much previous work considered the problem of obtaining a fair division under these (and other) criteria.

Social Welfare. While fairness is clearly a major consideration in the division of goods, another important consideration is the social welfare resulting from the division. Clearly, a division may be envy-free but very inefficient, e.g. in the total welfare it provides to the players. Accordingly, the question arises what, if any, is the tradeoff between these two desiderata? How much social welfare does one have to sacrifice in order to achieve fairness? The answer to this question may, of course, depend both on the exact definition of fairness, and on the social welfare of interest.

The first analysis of such questions was provided in [CKKK09], where Caragiannis et al. consider the three leading fairness criteria — proportionality, envy-freeness and equitability — and quantify the possible loss in utilitarian social welfare due to such fairness requirements. Here we continue this line of research,

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 26–37, 2010.

extending the results in two ways. Firstly, the [CKKK09] analysis allows dividing the cake into any number of pieces, possibly even infinite. Thus, each player may get a collection of pieces, rather than a single one. While this may be acceptable in some cases, it may not be so in others, or at least highly undesirable, e.g. in the division of real estate, where players naturally prefer getting a connected plot. Similarly, in the cake scenario itself, allowing unconnected pieces may lead to a situation where, in Stromquist's words [Str80], "a player who hopes only for a modest interval of the cake may be presented instead with a countable union of crumbs". Accordingly, in this work, we focus on divisions in which each player gets a single connected piece of the cake. In addition, we consider both the utilitarian and the egalitarian social welfare functions, whereas Caragiannis et al. considered only utilitarian welfare. For each of these welfare functions, we give tight bounds on the possible loss in welfare due to the three fairness criteria.

1.1 Definitions and Notations

We consider a 1-dimensional cake, represented by the interval $[0, 1]$, and so each cut is some point $p \in [0, 1]$. The cake has to be shared between n players (we denote $[n] = \{1, \ldots, n\}$), each having a valuation function $v_i(\cdot)$ assigning a non-negative value to every possible interval of the cake. As customary, we require that for all i, $v_i(\cdot)$ is a nonatomic measure on $[0, 1]$ having $v_i(0, 1) = 1$.[1] A set of valuation functions $\{v_i(\cdot)\}_{i=1}^n$ defines an *instance* of the cake cutting problem.

Since we consider only divisions in which every player gets a single connected interval, a division of the cake to n players can be represented by a vector

$$x = (x_1, \ldots, x_{n-1}, \pi) \in [0, 1]^{n-1} \times S_n$$

with $0 \le x_1 \le \cdots \le x_{n-1} \le 1$. Here, x_i determines the position of the i-th cut, and π is a permutation that determines which piece is given to which player. For convenience, we denote $x_0 = 0$ and $x_n = 1$, so we can write that player $i \in [n]$ receives the interval $(x_{\pi(i)-1}, x_{\pi(i)})$. We use the notation $u_i(x)$ for the utility that player i gets in the division x, i.e. $u_i(x) = v_i(x_{\pi(i)-1}, x_{\pi(i)})$. We denote by X the set of all possible division vectors, and note that X is a compact set.

Fairness Criteria. We say that a division $x \in X$ is:

- **Proportional.** If every player gets at least $\frac{1}{n}$ of the cake (by her own valuation). Formally, x is a proportional division if for all $i \in [n]$, $u_i(x) \ge \frac{1}{n}$.
- **Envy-Free.** If no player prefers getting a piece alloted to another player. Formally, x is an envy-free division if for all $i, j \in [n]$, i values j's piece as most as her own, i.e. $u_i(x) = v_i(x_{\pi(i)-1}, x_{\pi(i)}) \ge v_i(x_{\pi(j)-1}, x_{\pi(j)})$.
- **Equitable.** If all the players get the exact same utility in x (by their own valuations). Formally, x is equitable if for all $i, j \in [n]$, $u_i(x) = u_j(x)$.

[1] The assumption of $v_i(0, 1) = 1$ for all i normalizes the utilities of the players, implying, e.g. that 50% for one player is worth to society exactly as 50% for any other player. When this assumption is removed, some of the bounds we provide here for the Price of Fairness change; we discuss this further in a the full version of the paper.

It is well known that proportional divisions with connected pieces exist for every cake-cutting instance; Stromquist [Str80] showed (by a non-constructive argument) that connected envy-free divisions also always exist. In this paper we show (Theorem 3) that connected equitable divisions are also guaranteed to exist (for non-connected pieces this is already known from [DS61]).

Social Welfare Functions. For a division $x \in X$, we denote by $u(x)$ the utilitarian social welfare of x, i.e. $u(x) = \sum_{i \in [n]} u_i(x)$. Likewise, we denote by $eg(x)$ the egalitarian social welfare of x, which is $eg(x) = \min_{i \in [n]} u_i(x)$. Note that both these social welfare functions are continuous and thus have maxima in X.

The Price of Fairness. As described above, we aim to quantify the degradation in social welfare due to the different fairness requirements. This is captured by the notion of *Price of Fairness*, in its three forms — *Price of Proportionality*, *Price of Envy-freeness* and *Price of Equitability*, defined as follows. The *Price of Proportionality* (resp. Envy-Freeness, Equitability) of a cake-cutting instance I, with respect to some predefined social welfare function, is defined as the ratio between the maximum possible social welfare for the instance, taken over all possible divisions, and the maximum social welfare attainable when divisions must be proportional (resp. envy-free, resp. equitable). When considering divisions with connected pieces, this restriction is applied to both maximizations. For example, if $X_{EF} \subseteq X$ is the set of all (connected) envy-free divisions of an instance, the egalitarian Price of Envy-Freeness for this instance is

$$\frac{\max_{x \in X} eg(x)}{\max_{y \in X_{EF}} eg(y)} .$$

In this work we show bounds on the maximum utilitarian and egalitarian Price of Proportionality, Envy-Freeness and Equitability of *any* instance.

1.2 Results

We analyze the utilitarian and egalitarian Price of Proportionality, Price of Envy-Freeness and Price of Equitability for divisions with connected pieces. We provide tight bounds (in some cases, up to an additive constant factor) for all six resulting cases. The results are summarized in Table 1; the last row presents the relevant previous results of [CKKK09] for comparison. The upper bounds mean that the respective price of fairness of *any* instance in the class is never greater than the bound; the lower bounds mean that there *exists* an example of an instance with at least this price of fairness.

Utilitarian Welfare. For the utilitarian social welfare, we show an upper bound of $\frac{\sqrt{n}}{2} + 1 - o(1)$ on the price of envy-freeness. This, we believe, is the first non-trivial upper bound on the Price of Envy-Freeness. It seems that such bounds are hard to obtain since we need to consider the “best” possible envy-free division, while no efficient method for explicitly constructing any envy-free divisions is known. We show that the same bound also applies to the Price of Proportionality. This

Table 1. Summary of results

Price of:	Proportionality	Envy-Freeness	Equitability	
Utilitarian	UB: $\frac{\sqrt{n}}{2}+1-o(1)$ LB: $\frac{\sqrt{n}}{2}$		UB: n LB: $n-1+\frac{1}{n}$	connected pieces (this work)
Egalitarian (tight)	1	$\frac{n}{2}$	1	
Utilitarian	UB: $2\sqrt{n}-1$ LB: $\frac{\sqrt{n}}{2}$	UB: $n-\frac{1}{2}$ LB: $\frac{\sqrt{n}}{2}$	UB: n LB: $\frac{(n+1)^2}{4n}$	non-connected pieces [CKKK09]

upper bound is (nearly) matched by a lower bound presented in [CKKK09] that also applies to the connected case.

For the Price of Equitability, we show that it is always bounded by n (though simple, this does require a proof since an equitable division need not even give each player $1/n$). We also provide an almost matching lower bound, showing that for any n there exists an instance with utilitarian Price of Equitability arbitrarily close to $n-1+\frac{1}{n}$.

Egalitarian Welfare. When considering the egalitarian social welfare, we show that there is no price for either proportionality or equitability. That is, for any instance there exist both proportional and equitable divisions for which the minimum amount any player gets is no less than if there were no fairness requirements. While perhaps not surprising, the proof for the Price of Equitability is somewhat involved; we note that we are not aware of any previous proof that altogether establishes the existence of an equitable division with connected pieces. For the Price of Envy-Freeness, we show that it is bounded by $n/2$, and provide a matching family of instances that exhibit this price, for any n.

Trading Fairness for Efficiency. The price of fairness is a measure that characterizes the trade-off between fairness and social welfare, and does so by quantifying the amount of welfare we need to give up to achieve fairness. However, we may also be interested in the "reverse" question: measuring the amount of fairness that may have to be sacrificed to achieve social optimality. To answer this question, we define natural measures that *quantify* the "unfairness" of different divisions (one for each fairness criterion). We then prove that in order to achieve utilitarian optimum, we may have to give up infinite fairness (by all three measures), and that to achieve egalitarian optimum, we may have to accept divisions in which a player may think that some other player received a piece worth $(n-1)$-times more than his own. However, as the results for the price of fairness indicate, there is no conflict between egalitarian optimality and proportionality or equitability.

1.3 Related Work

The problem of fair division dates back to the ancient times, and takes many forms. The property to be divided may be divisible or indivisible: Divisible goods can be "cut" into pieces of any size without destroying their value (like a cake, a piece of land, or an investment account), while indivisible goods must be given in whole to one person (e.g. a car, a house, or an antique vase). Since such items cannot be divided, the problem is usually to divide a *set* of such goods between a number of players. Fair division may also relate to the allocation of chores (of which every party likes to get as little as possible); this problem is of a somewhat different flavor from the allocation of goods, and also has the divisible and indivisible variants.

Modern mathematical treatment of fair division started at the 1940s [Ste49], and was initially concerned mainly with finding methods for allocation of divisible goods. Different algorithms — both discrete and continuous ("moving knife algorithms") — were presented (e.g. [Str80, EP84] and [BT95], which also surveys older algorithms), as well as non-constructive existence theorems [DS61, Str80]. Several books have also appeared on the subject [BT96, RW98, Mou04]. Following the evaluation and cut queries model of Robertson and Webb [RW98], much attention was given to finding lower bounds on the number of steps required for fair divisions [MIBK03, SW03, EP06, Str08, Pro09]. In particular, Stromquist [Str08] proves that no finite protocol (even unbounded) can be devised for an envy-free division of a cake among three or more people in which each player receives a connected piece. However, we note that this result applies only to the model presented in that work (which resembles that of Robertson and Webb), and not for cases where, for example, a mediator has full information of the players' valuation functions and proposes a division based on this information.

Unlike most of the work on cake cutting, the different notions of the price of fairness are not concerned with *procedures* for obtaining divisions, but rather with the *existence* of divisions with different properties (relating to social optimality and fairness). These notions, namely the Price of Proportionality, Envy-Freeness, and Equitability, were first presented in a recent paper by Caragiannis et al. [CKKK09]. This line of work somewhat resembles the line of work on the Price of Stability [ADK$^+$04], which attracted much attention in the past decade. The work in [CKKK09] analyzes the price of fairness (via the above three measures) with the utilitarian welfare function for divisible and indivisible goods and chores, giving tight bounds (up to a constant multiplicative factor) in most cases. However, unlike in this work, no special attention was given to the case of connected pieces in divisible goods, and egalitarian welfare was not considered.

Due to the strict page limit in these proceedings, some of the proofs are omitted from this extneded abstract; these proofs can be found in the full version of the paper.

2 The Price of Envy-Freeness and Proportionality

2.1 Utilitarian Welfare

Theorem 1. *The utilitarian Price of Envy-Freeness for cake-cutting instances with n players and connected pieces is bounded from above by* $\frac{\sqrt{n}}{2} + 1 - o(1)$.

In fact, we prove an even stronger claim: The distance of *any* envy-free division (and not just the "best" one) from utilitarian optimality is never greater than the above bound.

Proof. Let x be an envy-free division of the cake, and $u(x) = \sum_{i \in [n]} u_i(x)$ its utilitarian welfare. We show that any other division to connected pieces y has $u(y) \leq \left(\frac{\sqrt{n}}{2} + 1 - \frac{n}{4n^2 - 4n + 2\sqrt{n}}\right) \cdot u(x)$. Our proof is based on the following key observation:

> *Assume that for some $i \in [n]$, $u_i(y) \geq \alpha \cdot u_i(x)$. Since i values any other piece in the division x at most as much as her own, it has to be that in y, i gets an interval that intersects pieces that belonged to at least $\lceil \alpha \rceil$ different players (possibly including i herself).*

We will say that in the division y, player i gets the j-th cut of x if in y, i is given a piece starting at a point $p < x_j$ and ending at the point $p' > x_j$. A more formal statement of our observation is therefore that if in y, i gets at most α cuts of x, it holds that $u_i(y) \leq (\alpha + 1) \cdot u_i(x)$. We can thus bound the ratio $\frac{u(y)}{u(x)}$ by the solution to the following optimization problem, which aims to find values $\{u_i(x)\}_{i=1}^n$ and $\{\alpha_i\}_{i=1}^n$ (the number of cuts of x each player gets) that maximize this ratio.

$$\text{maximize} \quad \frac{\sum_{i=1}^n (\alpha_i + 1) u_i(x)}{\sum_{i=1}^n u_i(x)} \tag{1}$$

$$\text{subject to} \quad \sum_{i=1}^n \alpha_i = n - 1 \tag{2}$$

$$u_i(x) \geq \frac{1}{n} \qquad \forall 1 \leq i \leq n \tag{3}$$

$$(\alpha_i + 1) u_i(x) \leq 1 \qquad \forall 1 \leq i \leq n \tag{4}$$

$$\alpha_i \in \{0, \ldots, n-1\} \qquad \forall 1 \leq i \leq n \tag{5}$$

(3) is a necessary condition for the envy-freeness of x that provides a lower bound for the denominator, and (4) is equivalent to $u_i(y) \leq 1$.

We therefore concentrate on bounding the solution to the above optimization problem. To this end, the following observations are useful:

1. For any choice of values $\{u_i(x)\}_{i=1}^n$, the optimal assignment for the α_i variables is greedy, i.e. giving each player i, in non-increasing order of $u_i(x)$, the maximum possible value for α_i that does not violate any constraints. (Otherwise, there are players i, j with $u_i(x) > u_j(x)$ and $\alpha_j \geq 1$ such that increasing α_i by one at the expense of α_j is feasible and yields an increase of $u_i(x) - u_j(x) > 0$ in the numerator of (1), without affecting the denominator.) We thus can divide the players into two sets: Those with "high" $u_i(x)$ values, who receive strictly positive α_i values, and those with "low" $u_i(x)$ values, for which $\alpha_i = 0$.

2. Since the players with low $u_i(x)$ values add the same amount to both the numerator and the denominator in the objective function, maximum is obtained when these values are minimized; i.e. in the optimal solution $u_i(x) = \frac{1}{n}$ for all these players.
3. The solution to the problem above is clearly bounded from above by the solution to the same problem where the α_i variables need not have integral values. Clearly, in the optimal solution to such a problem, all the players with $\alpha_i > 0$ have $(\alpha_i + 1)u_i(x) = 1$.

We can thus bound the solution to our optimization problem by the solution to the following problem. Let K be a variable that denotes the number of players that will have $\alpha_i > 0$; by observation (3) above, for every such player, $(\alpha_i + 1)u_i(x) = 1$, and thus their total contribution to the numerator is K. We therefore seek a bound for:

$$\text{maximize} \qquad \frac{K + (n-K)\cdot\frac{1}{n}}{\sum_{i=1}^{K} u_i(x) + (n-K)\cdot\frac{1}{n}} \tag{6}$$

$$\text{subject to} \qquad \sum_{i=1}^{K}\left(\frac{1}{u_i(x)} - 1\right) = n-1 \tag{7}$$

$$K \le n \tag{8}$$

where (7) follows from (2) in the original program, combined with $\alpha_i = \frac{1}{u_i(x)} - 1$ for all i, which follows from our observation that $(\alpha_i + 1)u_i(x) = 1$.

It can be verified (e.g. using Lagrange multipliers) that for any value of $K \le n$ this is maximized when $u_i(x) = u_j(x)$ for all $i, j \in [K]$, i.e. when $u_i(x) = \frac{K}{n-K+1}$ for all $i \in [K]$. We thus conclude that the maximum solution to the above problem maximizes the ratio

$$\frac{K + (n-K)\cdot\frac{1}{n}}{K\cdot\frac{K}{n+K-1} + (n-K)\cdot\frac{1}{n}}\ ;$$

by elementary calculus this is maximized at $K = \sqrt{n}$, where the value is

$$\frac{(n\sqrt{n} + n - \sqrt{n})(n + \sqrt{n} - 1)}{n^2 + (n-\sqrt{n})(n+\sqrt{n}-1)} = \frac{\sqrt{n}}{2} + 1 - \frac{n}{4n^2 - 4n + 2\sqrt{n}}$$

as stated.

Since every envy-free division is in particular proportional, we immediately get that the bound on the utilitarian Price of Envy-Freeness also applies to the Price of Proportionality:

Corollary 1. *The utilitarian Price of Proportionality for cake-cutting instances with n players and connected pieces is bounded from above by $\frac{\sqrt{n}}{2} + 1 - o(1)$.*

We conclude by showing that these bounds are essentially tight (up to a small additive factor). The construction we use is identical to the one in [CKKK09].

Proposition 1. *The utilitarian Price of Proportionality (and thus also the utilitarian Price of Envy-Freeness) with n players and connected pieces is bounded from below by $\frac{\sqrt{n}}{2}$.*

2.2 Egalitarian Welfare

The following proposition is straightforward.

Proposition 2. *The egalitarian Price of Proportionality is* 1*.*

Theorem 2. *The egalitarian Price of Envy-Freeness for cake-cutting instances with n players and connected pieces is upper bounded by $\frac{n}{2}$, and this bound is tight. Furthermore, the upper bound holds also for the case of non-connected pieces.*

Proof. Note that if an instance admits a division y with $eg(y) \geq \frac{1}{2}$, this y is envy-free, and thus the Price of Envy-Freeness for this instance is 1. If this is not the case, then for every division y, $eg(y) < \frac{1}{2}$. However, the instance at hand has an envy-free division x and this division is in particular proportional and thus has $eg(x) \geq \frac{1}{n}$; the upper bound follows immediately.

It remains to show that this bound is tight for the connected case. Let $\epsilon > 0$ be an arbitrarily small constant, and consider n players with the following valuation functions. For $i = 1, \ldots, (n-1)$, player i assigns a value of $\frac{1}{2} + \epsilon$ to the piece $(\frac{i}{n} - \epsilon, \frac{i}{n} + \epsilon)$ (her "favorite piece"), a value of $\frac{1}{2} - \epsilon$ to the piece $(1 - \frac{2i+1}{2n} - \epsilon, 1 - \frac{2i+1}{2n} + \epsilon)$ (her "second-favorite piece"), and value of 0 to the rest of the cake. Finally, player n assigns a uniform value to the entire cake.

Any piece worth at least $\frac{1}{n} + 2\epsilon$ to player n has a physical size at least $\frac{1}{n} + 2\epsilon$, and thus contains the "favorite piece" of some other player, making this player envy player n. Thus, in any envy-free division of the cake, player n has utility of less than $\frac{1}{n} + 2\epsilon$. On the other hand, a division in which every player has utility of at least $\frac{1}{2} - \epsilon$ can be achieved by giving players $i = 1 \ldots \lfloor \frac{n-1}{2} \rfloor$ their favorite pieces, players $i = (\lfloor \frac{n-1}{2} \rfloor + 1) \ldots (n-1)$ their second-favorite pieces, and player n the interval $(\frac{1}{2} + \epsilon, 1)$ (the remaining parts of the cake can be given to any of the players closest to them). Tightness follows as ϵ approaches zero.

3 The Price of Equitability

To talk about the Price of Equitability, we first have to make sure that the concept is well-defined. When non-connected pieces are allowed, it is known that every cake cutting instance has an equitable division [DS61]. However, the proof of Dubins and Spanier allows a "piece" of the cake to be any member of the σ-algebra of subsets, which is quite far from our restricted case of single intervals. Another result by Alon [Alo87] establishes the existence of an equitable division giving every player exactly $\frac{1}{n}$ by each measure; however, such a division may require up to $n^2 - 1$ cuts. The question thus arises whether equitable divisions with connected pieces always exist; to the best of our knowledge, this question has not been addressed before, and we answer it here to the affirmative. Furthermore, we show that such a division requires no sacrifice of egalitarian welfare.

Theorem 3. *For every cake-cutting instance there exists an equitable division of the cake with connected pieces. Furthermore, there always exists such a division in which the egalitarian social welfare is as high as possible in any division with connected pieces.*

This holds even for cake cutting instances for which players' valuation of the entire cake need not be 1*.*

Proof. Recall that $eg(\cdot)$ has a maximum in X; we denote $OPT = \max_{x \in X} eg(x)$. Let $Y \subset X$ be the set of divisions with egalitarian value OPT, i.e. $Y = \{y = (y_1, \ldots, y_{n-1}, \pi) \in X \mid eg(y) = OPT\}$.

We note that Y is a compact set; this follows from the fact that it is a closed subset of X (which is compact itself). To show that Y is closed, we show that $\overline{Y} = X \setminus Y$ is open. Let $z \in \overline{Y}$ be some division having egalitarian welfare smaller than OPT. In particular, there must exist a player i and $\epsilon > 0$ such that $u_i(z) \leq OPT - \epsilon$. Since player i's valuation of the cake is a nonatomic measure, there must exist $\delta > 0$ such that extending i's piece to the interval $(z_{\pi(i)-1} - \delta, z_{\pi(i)} + \delta)$ increases i's utility (compared to the original division z) by less than ϵ. Therefore, in the ball of radius δ around z (e.g. in L_∞), every division still gives i utility smaller than OPT, and thus this ball does not intersect Y. It thus follows that $\overline{Y}$ is an open set, and so Y is closed and compact.

Recall that our aim is to show that Y has an equitable division. We first define a function $\Delta : Y \to \mathbb{R}$, measuring the distance of a division from equitability:

$$\Delta(y) = \max_{i,j \in [n]} \{u_i(y) - u_j(y)\} = \max_{i \in [n]} \{u_i(y) - OPT\} .$$

We complete the proof by showing that for any ϵ, there exists a devision $y^{(\epsilon)} \in Y$, such that $\Delta(y^{(\epsilon)}) \leq \epsilon$. Since Y is a compact set and $\Delta(\cdot)$ is continuous, the image of Y is also compact and thus necessarily contains zero (as it is a closed subset of $\mathbb{R}$ containing a point $p < \epsilon$ for every $\epsilon > 0$). We therefore conclude that there exists some $y^* \in Y$ with $\Delta(y^*) = 0$; such y^* is clearly equitable.

It remains to prove that for any ϵ, $y^{(\epsilon)}$ exists; we prove this by induction on the number of players n. For $n = 1$ there is only one possible division, which obtains exactly OPT for the single player. Assume for $n-1$, we prove for n. Let y be any division in Y, and assume w.l.o.g. that y uses the identity permutation. We first construct a division y' such that for $i = 1, \ldots, n-1$, $u_i(y') = OPT$, by sequentially moving the border y'_i (between players i and $i+1$) to the left as far as possible while keeping that $u_i(y') \geq OPT$. This is possible since in y, $u_i(y) \geq OPT$ and the borders only need to move to the left. Consider the resulting y'. If $u_n(y') \leq OPT + \epsilon$ we are finished; otherwise, let y'' be the division obtained from y' by moving the border y''_{n-1} (between players $n-1$ and n) as far right as necessary so that $u_n(y'') = OPT + \epsilon$. Now, omit the rightmost piece (that of player n), and consider the $(n-1)$-player cake cutting problem, on the remaining cake. (Note that the players' valuation of the entire new cake need not be identical to their valuation of the original cake, and that the new cake has a different set Y' of egalitarian-optimal divisions.)

Now, in this new problem the egalitarian maximum cannot exceed OPT, as that would induce an egalitarian maximum greater than OPT for the entire problem. On the other hand, an egalitarian value of OPT is clearly attainable, as it is obtained by y'' (reduced to the first $n-1$ players). Hence, OPT is also the egalitarian maximum for the new $(n-1)$-player problem. Thus, by the inductive hypothesis, there exists a division for this problem that obtains egalitarian welfare OPT and such that no player gets more than $OPT+\epsilon$. Combining this solution with the piece $(y''_{n-1}, 1)$ given to player n, we obtain $y^{(\epsilon)} \in Y$, such that no player gets more than $OPT+\epsilon$.

The proof for the bound on the utilitarian Price of Equitability can be found in the full version of the paper.

Theorem 4. *The utilitarian Price of Equitability for cake-cutting instances with n players and connected pieces is bounded from above by n, and for any n there is an example in which it is arbitrarily close to $n-1+\frac{1}{n}$.*

4 Trading Fairness for Efficiency

The work on the Price of Fairness is concerned with the trade-off between two goals of cake division: Fairness, and efficiency (in terms of social welfare). However, the results we presented so far, as well as the results in [CKKK09], concentrate on one direction of this trade-off, namely *how much efficiency may have to be sacrificed to achieve fairness.* We now turn to look at the analogue question of *how much fairness may have to be given up to achieve social optimality.* Sadly, it seems that at least for the connected-pieces case, most of the results are somewhat pessimistic.

To answer such questions, one must first provide a way to quantify unfairness. We suggest the following definitions. For $\alpha \geq 1$, we say that a division x:

- is *α-unproportional* if some player $i \in [n]$ has $u_i(x) \leq \frac{1}{\alpha \cdot n}$.
- *has envy of α* if there exist players $i, j \in [n]$ for which

$$v_i(x_{\pi(j)-1}, x_{\pi(j)}) \geq \alpha \cdot v_i(x_{\pi(i)-1}, x_{\pi(i)}) = \alpha \cdot u_i(x) ,$$

  i.e. if some i feels that $j \neq i$ received a piece worth α-times more than the one she got.
- is *α-inequitable* if there are players $i, j \in [n]$ with $u_i(x) \geq \alpha \cdot u_j(x)$.

Using these "unfairness" notions, we can obtain the following simple results.

Proposition 3. *There are cake-cutting instances where a utilitarian-optimal division is necessarily infinitely unfair, by all three measures above.*

We already know (Proposition 2 and Theorem 3) that egalitarian optimality is not in conflict with neither proportionality nor equitability. However, this is not the case for envy:

Proposition 4. *There are cake-cutting instances where an egalitarian-optimal division necessarily has envy arbitrarily close to $n-1$, and this is the maximum possible envy for such divisions.*

5 Conclusions and Open Problems

In this work we analyzed the possible degradation in social welfare due to fairness requirements, when requiring that each player obtains a single connected piece. We obtain that the results vary considerably, depending on the fairness criteria used, and the social welfare function in consideration. The bounds range from provably no degradation for proportionality and equitability under the egalitarian welfare, through an $O(\sqrt{n})$ degradation for envy-freeness and proportionality under the utilitarian welfare, to an $O(n)$ degradation for equitability under the utilitarian welfare and for envy-freeness under the egalitarian welfare. We have also seen that if we seek to trade fairness to achieve social optimality, the "exchange rate" may (at the worst case) be infinite for utilitarian welfare (for all three fairness criteria), or linear for egalitarian welfare and envy-freeness.

Many open questions await further research, including:

- *Small number of connected pieces.* Most works on cake cutting either require that each player gets a single connected piece (as we do in this work), or allow giving a player any union of intervals. A natural middle ground is to require that each player receives only a small number of pieces, e.g. a constant number. The question thus arises to bound the degradation to the social welfare under such requirements. In such an analysis it would be particularly interesting to see how the bounds on degradation behave as a function of the number of permissible pieces.
- *The Egalitarian Price of Fairness with non-connected pieces.* [CKKK09] provide bounds on the Price of Fairness using the utilitarian welfare function, for the setting in which non-connected pieces are permissible. Bounding the egalitarian Price of Fairness in this setting remains open. A trivial upper bound on the Price of Envy-freeness is $\frac{n}{2}$, and we have examples of instances where this price is strictly larger than 1, but obtaining tight bounds seems to require additional work and techniques.
- *The egalitarian Price of Proportionality and Equitability for indivisible goods.* [CKKK09] provide analysis for the utilitarian Price of Fairness for such goods. A simple tight bound of $\frac{n}{2}$ can be shown for the egalitarian Price of Envy-Freeness for this case; it thus remains open to determine the egalitarian Price of Proportionality and Equitability for such goods.
- *The Price of Fairness for connected chores.* As we already mentioned, fair division of chores has a somewhat different flavor from division of goods, and may require different techniques. One possible motivation for requiring connected division of chores may be, for example, a case in which a group of workers have to maintain the cleanliness of a (heterogeneous) beach strip, and so would like to give each worker a connected area to be responsible for.

Acknowledgement. We thank Ariel Procaccia for providing helpful comments on an earlier draft of this work.

References

[ADK$^+$04] Anshelevich, E., Dasgupta, A., Kleinberg, J.M., Tardos, É., Wexler, T., Roughgarden, T.: The price of stability for network design with fair cost allocation. In: FOCS, pp. 295–304 (2004)

[Alo87] Alon, N.: Splitting necklaces. Advances in Mathematics 63(3), 247–253 (1987)

[BT95] Brams, S.J., Taylor, A.D.: An envy-free cake division protocol. The American Mathematical Monthly 102(1), 9–18 (1995)

[BT96] Brams, S.J., Taylor, A.D.: Fair Division: From cake cutting to dispute resolution. Cambridge University Press, New York (1996)

[CKKK09] Caragiannis, I., Kaklamanis, C., Kanellopoulos, P., Kyropoulou, M.: The efficiency of fair division. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 475–482. Springer, Heidelberg (2009)

[DS61] Dubins, L.E., Spanier, E.H.: How to cut a cake fairly. The American Mathematical Monthly 68(1), 1–17 (1961)

[EP84] Even, S., Paz, A.: A note on cake cutting. Discrete Applied Mathematics 7(3), 285–296 (1984)

[EP06] Edmonds, J., Pruhs, K.: Cake cutting really is not a piece of cake. In: SODA 2006: Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm, pp. 271–278. ACM, New York (2006)

[MIBK03] Magdon-Ismail, M., Busch, C., Krishnamoorthy, M.S.: Cake-cutting is not a piece of cake. In: Alt, H., Habib, M. (eds.) STACS 2003. LNCS, vol. 2607, pp. 596–607. Springer, Heidelberg (2003)

[Mou04] Moulin, H.J.: Fair Division and Collective Welfare. MIT Press Books, vol. 0262633116. MIT Press, Cambridge (2004)

[Pro09] Procaccia, A.D.: Thou shalt covet thy neighbor's cake. In: IJCAI, pp. 239–244 (2009)

[RW98] Robertson, J., Webb, W.: Cake-cutting algorithms: Be fair if you can. A K Peters, Ltd., Natick (1998)

[Ste49] Steinhaus, H.: Sur la division pragmatique. Econometrica 17(Supplement: Report of the Washington Meeting), 315–319 (1949)

[Str80] Stromquist, W.: How to cut a cake fairly. The American Mathematical Monthly 87(8), 640–644 (1980)

[Str08] Stromquist, W.: Envy-free cake divisions cannot be found by finite protocols. Electronic Journal of Combinatorics 15(1) (January 2008)

[SW03] Sgall, J., Woeginger, G.J.: A lower bound for cake cutting. In: Di Battista, G., Zwick, U. (eds.) ESA 2003. LNCS, vol. 2832, pp. 459–469. Springer, Heidelberg (2003)

Collusion in VCG Path Procurement Auctions

Yoram Bachrach[1], Peter Key[1], and Morteza Zadimoghaddam[2]

[1] Microsoft Research, Cambridge, UK
{yobach,peter.key}@microsoft.com
[2] MIT, Cambridge, MA
zadimoghaddam@csail.mit.edu

Abstract. We consider collusion in path procurement auctions, where payments are determined using the VCG mechanism. We show that collusion can increase the utility of the agents, and in some cases they can extract any amount the procurer is willing to offer. We show that computing how much a coalition can gain by colluding is NP-complete in general, but that in certain interesting restricted cases, the optimal collusion scheme can be computed in polynomial time. We examine the ways in which the colluders might share their payments, using the core and Shapley value from cooperative game theory. We show that in some cases the collusion game has an empty core, so although beneficial manipulations exist, the colluders would find it hard to form a stable coalition due to inability to decide how to split the rewards. On the other hand, we show that in several common restricted cases the collusion game is convex, so it has a non-empty core, which contains the Shapley value. We also show that in these cases colluders can compute core imputations and the Shapley value in polynomial time.

1 Introduction

Collusion is an agreement between agents to defraud in order to obtain an unfair advantage [22]. We examine collusion in path procurement auctions (PPAs), where a buyer procures a path from a source s to a target t in a graph $G = \langle V, E \rangle$. Each edge $e_i \in E$ is owned by a_i, who incurs a cost c_i when her edge is used. The cost c_i is known only to a_i. The buyer must compensate edges on the chosen path for their costs. Given the private costs, a mechanism can find the minimal cost $s - t$-path. The mechanism can ask each a_i for the minimal amount it would be willing to receive to allow using e_i. If a_i answers (bids) truthfully, this is her cost c_i. However, the costs are the agents' private information and they may bid strategically to increase their payment. VCG mechanisms [23,10,13] are used to incentivise agents to reveal their true costs. VCG has desirable properties, but is susceptible to collusion. Though any single agent is incentivised to bid truthfully, *several* agents may *coordinate* bids and split the gains from manipulating. We show how agents might collude and share the gains in VCG PPAs. Our model follows the *collusion game* of [4], but applied to PPAs.

1.1 Preliminaries

In VCG mechanisms we have an agent set $N = \{1, \ldots, n\}$. The mechanism chooses an alternative from the set K. Agents report a type $\theta_i \in \Theta_i$, representing her preferences

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 38–49, 2010.

over K, and each agent i has a valuation $w_i(k, \theta_i)$ depending on the chosen $k \in K$. The mechanism uses the choice rule $k : \Theta_1 \times ... \times \Theta_n \rightarrow K$, and agent i must also make a payment r_i to the mechanism, according to a payment rule $t_i : \Theta_1 \times ... \times \Theta_n \rightarrow \mathbb{R}$. We assume quasi-linear utility $u_i(k, p_i, \theta_i) = w_i(k, \theta_i) - r_i$. An agent i may manipulate and report type $\theta'_i = s_i(\theta_i)$, according to its strategy s_i. Groves mechanisms use $k^*(\theta') = arg \max_{k \in K} \sum_i w_i(k, \theta'_i)$ and payment rule: $r_i(\theta') = h_i(\theta'_{-i}) - \sum_{j \neq i} w_j(k^*, \theta'_j)$, where $h_i : \Theta_{-i} \rightarrow \mathbb{R}$ only depends on the reported types of agents other than i. We consider the case of VCG, where: $h_i(\theta'_{-i}) = \sum_{j \neq i} w_j(k^*_{-i}(\theta'_{-i}), \theta'_j)$.

Our collusion analysis uses coalitional game theory. A transferable utility coalitional game is composed of a set N of n agents, and a characteristic function mapping any agent subset (coalition) to a value $v : 2^N \rightarrow \mathbb{R}$, indicating the total utility these agents achieve together. The function only defines the gains a coalition achieves, not how to distribute them. An *imputation* $(p_1, \ldots, p_n)$ divides the the gains among the agents, where $p_i \in \mathbb{R}$, such that $\sum_{i=1}^{n} p_i = v(N)$. We call p_i the payoff of agent i, and denote $p(C) = \sum_{i \in C} p_i$. A key issue is choosing the appropriate imputation. A basic imputation requirement is *individual rationality*: for any $i \in N$, $p_i \geq v(\{i\})$. Otherwise, agent i is incentivized to work alone. Similarly, coalition B *blocks* imputation p if $p(B) < v(B)$, since B's members are better off working on their own. A solution concept focusing on this is the *core* [12]: the set of all imputations p not blocked by any coalition, so for any $C \subseteq N$ we have: $p(C) \geq v(C)$.

Another solution concept is the Shapley value [20] which defines a *single* value division. It focuses on *fairness*, rather than stability. The Shapley value fulfills important fairness axioms [20,25] and has been used to fairly share gains or costs. The Shapley value of an agent depends on its marginal contribution to possible coalition permutations. We denote by π a permutation (ordering) of the agents, and by Π the set of all possible such permutations. Given permutation $\pi \in \Pi = (i_1, \ldots, i_n)$, the marginal worth vector $m^\pi[v] \in \mathbb{R}^n$ is defined as $m^\pi_{i_1} = v(\{i_1\})$ and for $k > 1$ as $m^\pi_{i_k}[v] = v(\{i_1, i_2, \ldots, i_k\}) - v(\{i_1, i_2, \ldots, i_{k-1}\})$. The convex hull of all the marginal vectors is called the *Weber Set*. Weber showed [24] that the Weber set of any game contains its core. The Shapley value is the centroid of the marginal vectors.

Definition 1. *The Shapley value is the payoff vector:* $\phi[v] = \frac{1}{n!} \sum_{\pi \in \Pi} m^\pi[v]$.

Our analysis is based on the notion of convex games. For convex games it is known [21] that the core is non-empty, and that the Weber Set is identical to the core. The Shapley value is a convex combination of the marginal vectors and lies in the Weber Set, so in convex games, the Shapley value lies in the core.

Definition 2. *A game is convex if:* $\forall A, B \subseteq I$, $v(A \cup B) \geq v(A) + v(B) - v(A \cap B)$.

2 Collusion in VCG Path Procurement Auctions

Consider a PPA in a graph $G = \langle V, E \rangle$, where the buyer procures edges $P \subseteq E$ forming an $s-t$-path from a set of agents, each owning an edge in the graph. We identify an agent a_i with her edge $e_i \in E$. Each agent has a cost c_i associated with her edge and the mechanism asks each a_i to provide a bid b_i for using the edge. If the agent is truthful, she would report c_i. Given the edges' true costs, one can find the minimal cost $s-t$-path, but the costs are private information. The canonical solution to induce truthfulness is the VCG mechanism. As discussed in Section 1.1, using VCG prices makes truthful cost revelation the dominant strategy, and results in procuring the cheapest path. Given the edge costs, this path can easily be computed in polynomial time.

Observation 1 (Computing VCG Prices). *Let $G = \langle V, E\rangle$ be a path procurement domain, with cost c_i for edge $e_i \in E$, and let b_i be the bid of e_i. Denote the minimal cost path (according to the declared b_i's) as $(e_{i_1}, e_{i_2}, \ldots, e_{i_x})$ (of x edges), and let the optimal path not including e_i be $e_{j_1}, e_{j_2}, \ldots, e_{j_y}$ (of y edges). If e_i is on the chosen path, the payment to e_i's owner is $p_i = \sum_{s=1}^{y} b_{j_s} - \sum_{s=1}^{x} b_{i_s} + b_i$, otherwise $p_i = 0$.*

2.1 Colluding in VCG Path Procurement Auctions

We begin with collusion examples. Denote the payment to agent a_i when all the agents bid truthfully (i.e. a_i bids her true cost so $b_i = c_i$) as p_i. Given a set of edges $C \subseteq E$, we denote the VCG payments of all of them under truthful revelation as $p(C) = \sum_{e_i \in C} p_i$.

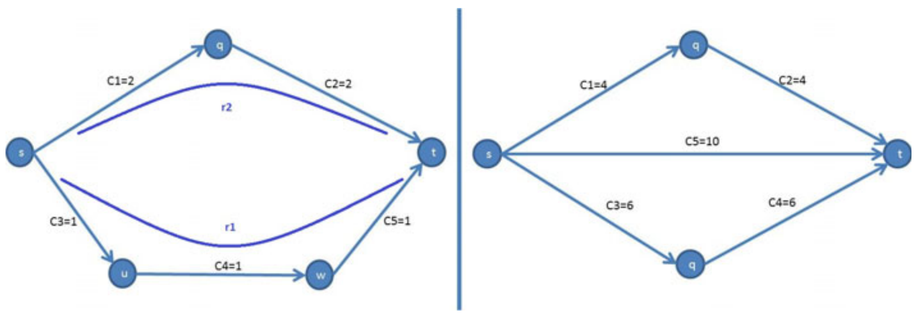


Fig. 1. Left: domain for Examples 1, 2, 3. Right: domain for Example 4.

Example 1 (Collusion on the cheapest path). Consider the graph on the left of Figure 1, with two $s-t$-paths: $r_1 = \langle s, u, w, t\rangle$ with costs $c_3 = 1, c_4 = 1, c_5 = 1$ and $r_2 = \langle s, q, t\rangle$ with edge costs $c_1 = 2, c_2 = 2$. The cheapest path is r_1 with cost $c_{r_1} = 1+1+1 = 3$, and the second cheapest path is r_2 with cost $c_{r_2} = 2+2 = 4$. Consider the agents on r_1: $C = \langle e_3, e_4, e_5\rangle$. If all the edges truthfully declare their costs (so a_i bids b_i where $b_i = c_i$), applying Observation 1 we obtain payments: $p_3 = 2, p_4 = 2, p_5 = 2$. Thus, we have $p(C) = 2 + 2 + 2 = 6$. Suppose each of the agents in C reports having no cost, bidding $b'_3 = b'_4 = b'_5 = 0$. This manipulation does not change the chosen path, as the cheapest path remains r_1. However, the payments do change. Denote the payments when the agents in C bid untruthfully (so $b'_3 = b'_4 = b'_5 = 0$) and the agents in $I \setminus C$ bid truthfully (so $b'_1 = c_1 = 2, b'_2 = c_2 = 2$) as $p' = \langle p'_1, p'_2, \ldots, p'_6\rangle$. Recomputing VCG payments for b' we obtain $p'_3 = p'_4 = p'_5 = 4$. Thus each member of C benefits from this manipulation, and the total payments for the C become $p'(C) =$

$\sum_{e_i \in C} p'_i = 12$. Note the actual costs of the agents in C have not changed, but total payments increased by $12 - 6 = 6$. The cost of the coalition C when r_1 is chosen is $c_1 + c_2 + c_3 = 1 + 1 + 1 = 3$, so through this manipulation, the coalition moves from a utility of $p(C) - \sum_{i \in C} c_i = 6 - 3 = 3$ to $p'(C) - \sum_{i \in C} c_i = 12 - 3 = 9$.

Example 2 (Collusion on a s-t cut). Examine the left of Figure 1 again, but consider the case where $C = \langle e_1, e_3 \rangle$ collude, and e_2, e_4, e_5 bid truthfully. Under truthfully declarations, the chosen path is r_1 with payments: $p_1 = p_2 = 0, p_3 = p_4 = p_5 = 2$. We have $p(C) = p_1 + p_3 = 2$, and since r_1 is chosen, e_3 incurs a cost $c_3 = 1$ so the utility of the coalition C is $p(C) - \sum_{i \in C \cap r_1} c_i = 2 - c_3 = 2 - 1 = 1$. Now suppose the colluders in C manipulate and bid $b'_1 = h$ (for a high number $h > 2$, say $h = 100$), and $b'_3 = 0$[1], while e_2, e_4, e_5 bid truthfully. Again, the manipulation does not change the chosen path which is still r_1, but the payments do change. Again, we denote the payments when the agents in C bid untruthfully ($b'_1 = h, b'_3 = 0$) and the agents in $I \setminus C$ bid truthfully as $p' = \langle p'_1, p'_2, \ldots, p'_5 \rangle$. Recomputing the VCG payments under p' we get $p'_1 = 0, p'_3 = h + 2 - 2 = h$. Thus, $p(C) = h$. Since r_1 is still the chosen path, e_3 still incurs the cost c_3. Thus the new utility of the coalition C is $p(C) - \sum_{i \in C \cap r_1} c_i = h - 1$. Since the payment of the coalition depends on its chosen value for h, its utility is unbounded. One might claim that since a_1 did not increase her utility, she might not be willing to collude (lie for a_3). To get a_1 to cooperate, a_3 can easily *compensate* a_1 via a monetary transfer. Without such a monetary transfers, all the payment goes to e_3. However, using such a transfer, the utility of the coalition of colluders, $p(C) - \sum_{i \in C \cap r_1} c_i = h - 1$, can be shared between e_1 and e_3 in any.

Example 3 (Collusion on the non-optimal path). Consider the left of Figure 1, with the optimal path r_1 and the second cheapest path r_2. Suppose $C = \langle e_1, e_2 \rangle$ collude (edges of a non-optimal path), and e_3, e_4, e_5 bid truthfully. Under truthful declarations the chosen path is r_1, and $p_1 = p_2 = 0$ (as $r_1 = \langle e_3, e_4, e_5 \rangle$ is chosen and not $r_2 = \langle e_1, e_2 \rangle$), so we have $p(C) = 0$, and the utility of C is 0. If C manipulates by bidding $b'_1 = b'_2 = 0$, the chosen path is r_2 rather than r_1, and the payments are $p'_1 = p'_2 = 3$, so we have $p'(C) = 6$. However, since r_2 is chosen, edges e_1, e_2 incur the costs of $c_1 = c_2 = 2$, so the coalition's utility is $p(C) - \sum_{i \in C} c_i = 6 - 4 = 2$. Thus, this manipulation gives C a utility of 2, rather than 0. Without transfers, this utility is shared equally between e_1 and e_2, but it can be shared in any way using transfers.

Example 2 is troublesome, as the colluders achieve unbounded payment from the mechanism[2]. Example 3 shows that even agents on a non-optimal path can manipulate. We now show an example where beneficial manipulations exist, but due to the network structure, the colluders cannot find a stable way to share the gains from manipulating.

Example 4 (Empty Core). Consider Figure 1 on the right. The cheapest path is $r_1 = \langle e_1, e_2 \rangle$ with cost 8, the second cheapest path is $r_2 = \langle e_5 \rangle$ with cost 10, and the third cheapest is $r_3 = \langle e_3, e_4 \rangle$ with cost 12. Under truthful declarations r_1 is chosen, and

[1] For this case, e_3 may as well report its true cost. However, if the coalition has other edges on the cheapest path (e.g. e_4 or e_5), this increases their payment as well.

[2] Colluders who can disconnect s and t get any amount the procurer has. This is not surprising as the good sold is $s - t$ connectivity, and the colluders' cartel controls all the supply.

the payments are $p_1 = p_2 = 6$ (other payments are 0). Coalition $C_t = \langle e_1, e_2 \rangle$ can manipulate similarly to Example 1 by bidding $b'_1 = b'_2 = 0$ to achieve $p'(C_t) = 10 + 10 = 20$. This raises the utility of C_t from $12 - 8 = 4$ to $20 - 8 = 12$. However, Coalition $C_b = \langle e_3, e_4 \rangle$ can manipulate similarly to Example 3 by bidding $b'_3 = b'_4 = 0$ to achieve a $p'(C_b) = 8 + 8 = 16$[3]. This raises the utility of C_b from 0 to $16 - 12 = 4$.

Consider the case where $C = C_t \cup C_b = \{e_1, e_2, e_3, e_4\}$ collude. C doesn't control e_5 so its payment cannot exceed 10 per edge. Either $\langle e_1, e_2 \rangle$ or $\langle e_3, e_4 \rangle$ or $\langle e_5 \rangle$ is chosen, so the total payment for C cannot exceed 20. The minimal cost C incurs to get any payment is $4+4$ (routing through $\langle e_1, e_2 \rangle$). Thus C's utility is bounded by $20-8 = 12$, similarly to C_t, and achievable the same way. Thus, C_b adds no value to coalition C_t. Consider what happens when $C = \{e_1, e_2, e_3, e_4\}$ try to agree on what to bid and how to share the gains. The optimal collusion bids for them get them a utility of 12. Edges e_1, e_2 (of C_t) might claim they deserve all this utility, as they can achieve this utility on their own. However, e_3, e_4 (of C_b) would claim they deserve at least 4, as they achieve 4 on their own. This results in an unstable coalition and in threats between the coalition members[4]. Section 3 characterizes this as a collusion game with an empty core.

In Example 4 , though the colluders have a beneficial manipulation, they find it hard to form a coalition due to inability to decide how to share the reward. We characterize such situations using the collusion game. Despite hopes of having such instability mitigate collusion, we show that for natural coalitions the colluders can always share the gains in a stable way. We focus on coalitions where all colluders are on the cheapest path (as in Example 1) or a non-optimal path (as in Example 3).

2.2 Collusion Schemes

We consider optimal manipulations in VCG PPAs. Such collusion requires trust among the colluders, as they must coordinate and since in many domains collusive behavior is forbidden (the colluders face dire consequences if caught). We first show that in general,

[3] These are the payments where only e_3, e_4 collude, so e_1, e_2 truthfully declares their cost, so under the collusion, the VCG mechanism chooses $\langle e_3, e_4 \rangle$ as the "cheapest" path, and computes the payments using the alternative path $\langle e_1, e_3 \rangle$ of cost 8.

[4] Agents e_3, e_4 might threaten to bid $b'_3 = b'_4 = 0$ creating two zero cost paths, so the result would depend on how the mechanism breaks ties. In this case, the agents on the winning path would get a zero payment. If coalition $\{e_1, e_2, e_3, e_4\}$ breaks down into *two* coalitions $\{e_1, e_2\}$ and $\{e_3, e_4\}$ (each pair bidding in a coordinated manner), we have a normal form game. Each pair chooses the total cost of the path, the pair with lower cost winning and obtaining a total reward of the difference between the paths' costs plus its declared cost. A pure strategy Nash equilibrium is where the truly cheap path bids zero, and the truly expensive path bids highly enough to guarantee the cheap path a positive utility: the total payment to the cheap path is $k(h - l) + l$ where k is the number of edges on it and h and l are the declared path prices, so when h is high enough this exceeds the cheap path's true cost. If these are the *only* two paths, there is another Nash equilibrium: the cheap path bids highly, H, and the expensive path bids zero: the expensive path has a positive utility when winning and the cheap path can only win by bidding zero, in which case it would have a negative utility. When analyzing the core of the collusion game, we assume members dropping out do *not* form a new cartel and bid truthfully. Even under this easier assumption, some collusion games have empty cores.

given a colluder coalition C, finding the optimal collusion or the utility of a colluder coalition when it optimally manipulates for a coalition is NP-complete.

Theorem 1. *Computing the optimal coalition manipulation in a VCG PPA is NP-Complete.*

Proof. Computing the optimal manipulation value is *in* NP (up to any desired degree of numerical accuracy), since we can non-deterministically choose bids and check if we have a manipulation achieving the target utility. To show NP-hardness, we reduce from LONGEST-PATH (LP), where we are given a graph $G = \langle V, E \rangle$ and are asked to return the length of the longest simple path in it, known to be NP-Complete. Given the LP instance $G = \langle V, E \rangle$, we create a graph $G' = \langle V \cup \{s, t\}, E' \rangle$, which contains a copy of G and two other vertices: s which serves as the source and t which serves as the target of the PPA. All of G's edges are also replicated. Also, the source s is connected to the all the vertices in G, and any vertex is G is connected to the target t. We denote all edges (s, v) where $v \in V$ as S, and all edges (u, t) where $u \in V$ as T. We create an edge e_H, connecting s and t. All edges have a cost of $c_e = 1$ except edges in $S \cup T \cup \{e_H\}$. Edges in $S \cup T$ have zero cost, and e_H has a cost H where H is a very high number (for example $H > |E|^2$). The target coalition for which we find an optimal manipulation is $C = E' \setminus e_H = S \cup T \cup E$, all edges except e_H.

Denote by $L = (l_1, \ldots, l_q)$ the longest simple path in G, and its length by q. Coalition C contains L, and so it can have all the edges in $L \cup S \cup T$ bid zero, and all the other edges in C bid $H + 1$. Then, the cheapest path is $(s, l_1, \ldots, l_q, t)$ with a declared cost of zero, so this path is chosen. Under this manipulation, the second cheapest path is (s, t) with cost H, so each edge is paid H, and the coalition is paid $p(C) = (q+2)H$ (there are q edges on the longest path in G, and the edges (s, l_1) and (l_q, t)). The coalition incurs the true cost of 1 on its q edges in L, so C has a total cost of q. Thus, this manipulation obtains C a utility of $u^*(C) = (q + 2)H - q$. It is easy to see that $u^*(C)$ is the maximal utility C can obtain: the cheapest path must have a total cost of at most H or e_H would be the chosen path, so any edge can be paid at most H, and since L is the longest simple path in G it is impossible to have more than q edges of G on the path the mechanism chooses. Since $u^*(C) = (q + 2)H - q$ and since we choose the value of H in the reduction, given $u^*(C)$ we can extract q, the length of the longest simple path in G. This proves we cannot compute the optimal manipulation bids, since given this manipulation we can compute the chosen path and VCG prices and since we know the true edge costs this allows computing $u^*(C)$.

The hardness result of Theorem 1 forces us to examine restricted cases of the manipulation problem. In the extreme case where *all* the edges collude, they can guarantee any payment the procurer can pay[5]. In typical domains, the set of colluders is unlikely to be all the edges or an arbitrary edge subset. A more reasonable colluder set can be a set of neighboring or close edges, or several edges that are all on a single $s - t$ path. We examine cases where we can tractably compute the optimal manipulation. Example 1 is an example of a simple case, where all colluders are on the cheapest $s - t$ path, and the second cheapest path runs in parallel to the cheapest path. Example 3 shows the second

[5] We later show that it suffices for the colluders to be able to disconnect s and t.

simple case, where all colluders are on a non-optimal $s-t$ path, which runs in parallel to the cheapest path, and can underbid the truly optimal path.

Consider the case where all colluders are on the cheapest path. C is *a simple coalition on the cheapest path* if these hold for all $e_i \in C$: edge e_i is on the cheapest $s-t$ path; when removing e_i, the cheapest $s-t$-path contains no edge $e_j \in C$. Similarly, C is *a simple coalition on a non-optimal path* if the following hold: all edges $e_i \in C$ are a non-optimal $s-t$ path, r; the cheapest path r^* does not intersect r, so $r^* \cap r = \emptyset$; r becomes cheapest when $b'_i = 0$ for all $e_i \in C$: $\sum_{e_i \in r \setminus C} c_i < \sum_{e_i \in r^*} c_i$.

The following theorems are regarding a VCG PPA, where edge e_i bids b_i and has cost c_i, and where C is a simple coalition on the cheapest path r_1. We assume that all non-coalition members bid truthfully, so for $e_i \in I \setminus C$ we have $b_i = c_i$.

Theorem 2 (Simple Cheapest Path Collusion). *Let C be a simple coalition of colluders on the cheapest path r_1. The optimal collusion, maximizing payments p_i of any $e_i \in C$ (and C's payment $p(C) = \sum_{e_i \in C} p_i$) is zero bids: $b_i = 0$ for all $e_i \in C$.*

Proof. Denote the cheapest path under truthful declarations as r_1, and the cheapest path under truthful declarations that does not contain any edge in C as r_2. Consider an edge $e_i \in C$ that increases its bid beyond c_i. This increases the cost of r_1 under declared bids. If several agents in C declare such increased costs so that the cost of r_1 under these modified costs is more than the cost of r_2, the path r_2 will be chosen, resulting in a payment of 0 to all agents in C. Since VCG is individually-rational, this manipulation is not beneficial to the colluders. Thus, it suffices to focus on manipulations where the bids of edges in C are such that the cost of r_1 is at most the cost of r_2, so the procured path is r_1. Eliminating any edge $e_i \in C$ disallows the use of r_1, and for any $e_i \in C$ we denote by r_{-i} the cheapest path when eliminating e_i. Since C is a simple coalition on the cheapest path we have $r_{-i} \cap C = \phi$. Thus for $e_i, e_j \in C$ we have $r_{-i} = r_{-j}$. Since e_i, e_j are arbitrary edges in C, this means that the cheapest path after eliminating any edge in C is the same path r. This path r cannot contain any edge $e_i \in C$, so it is simply the cheapest path that does not contain any edge in C, r_2. Denote the edges in $r = r_2$ as $r_2 = \langle e_{j_1}, e_{j_2}, \ldots, e_{j_y} \rangle$ (y edges). Denote the edges of r_1 as $r_1 = \langle e_{i_1}, e_{i_2}, \ldots, e_{i_x} \rangle$ (x edges, containing the edges of C). We assume all agents in r_2 bid truthfully, and denote the total cost of r_2 as $c(r_2)$. Thus, the formula of Observation 1 can be written as: $p_i = \sum_{s=1}^{y} b_{j_s} - \sum_{s=1}^{x} b_{i_s} + b_i = c(r_2) - \sum_{s=1, e_{i_s} \neq i}^{x} b_{i_s}$. Note that the agents in C control the bids $\{b_i | e_i \in C\}$, and since each b_i must be non-negative (the cost of using edge e_i), each p_i is maximized when the bids are minimal. Thus, the optimal manipulation is bidding $b_i = 0$ for all $e_i \in C$.

Theorem 3 (Simple Non-Optimal Path Collusion). *Let C be a simple coalition of colluders on the non optimal path r. The optimal collusion, which maximizes all the payments p_i of any coalition member (and C total payment $p(C) = \sum_{e_i \in C} p_i$) is zero bids: $b_i = 0$ for all $e_i \in C$.*

Proof. The proof is almost identical to Theorem 3. We denote the non optimal path r which contains all the colluders as $\langle e_{i_1}, e_{i_2}, \ldots, e_{i_x} \rangle$, denote the cheapest path (under true costs) as r^*, and obtain: $p_i = c(r^*) - \sum_{s=1, e_{i_s} \neq i}^{x} b_{i_s}$.

Theorem 4 (Cut Collusion). *Let C be a coalition whose removal disconnects s and t, and $h > 0$ be some value. The colluders can bid so that $\sum_{e_i \in C} p_i > h$.*

Proof. At least one $e_x \in C$ must be used in the chosen $s - t$ path, as C is an $s - t$ cut. VCG is individually rational so if all $e \in C$ bid $b'_i = h$, for e_x we have $p_i > h$.

3 The Collusion Game

Consider a PPA over $G = \langle V, E \rangle$ with source s and target t. We examine a subset $C \subseteq N$, who may decide to collude. Under truthful bidding, VCG chooses path $r_1 = \langle e_{i_1}, e_{i_2}, \ldots, e_{i_x} \rangle$ and payments $p^t_1, \ldots, p^t_n$[6]. If the agents in C decide to collude, they can form a coalition and use a collusion scheme, such as those of Section 2.1. Denote the chosen path under the optimal manipulation as $r^* = \langle e^*_1, \ldots, e^*_z \rangle$ and the payments under the manipulation $p^*_1, \ldots, p^*_n$. Some manipulations, such as the optimal manipulation for simple collusion on the cheapest path, do not change the chosen path, so $r^* = r_1$, but increase the payments to coalition members so $p^*_i \leq p^t_i$ for any $i \in C$. Other schemes, such as collusion on a non-optimal path, change the selected path, so $r^* \neq r_1$. The coalition members gain payments, but the members on the chosen path, $C \cap r^*$, also incur the cost of their edges. Thus, the utility of the colluder coalition C is: $u^*(C) = \sum_{i \in C} p^*_i - \sum_{i \in C \cap r^*} c_i$. Using monetary transfers, the coalition's utility can be distributed among its members in any way they choose. We define a coalitional game, based on the total utility a coalition of colluders generates its members.

Definition 3 (Path Procurement Collusion Game). *Given a VCG PPA, the value $v(C)$ of a coalition $C \subseteq N$ is: $v(C) = u^*(C)$. In order to manipulate the colluders must trust each other, or sign a certain enforceable contract, so the coalition C is typically be restricted to only a certain subset of the agents.*

Given the above definition, Theorem 1 simply says that in general it is hard to even compute the value of a coalition in the collusion game. However, Theorem 2, Theorem 3 and Theorem 4 all show that for important restricted cases, finding the optimal manipulation is trivial. The above definition of the game also allows us to apply solution concepts to decide how the colluders might share their rewards. The core characterizes *stability*, where no subset of the coalition is incetivised to operate on its own. The Shapley value characterizes a *fair* allocation of the reward, reflecting each member's contribution. Having defined the collusion game, the theme of Example 4 is simple — this network structure results in the collusion game having an *empty core*[7].

One might hope that most network structures result in empty cores, so the colluders would not have a stable way of sharing the reward. If this were the case, the problem of collusion would be mitigated since despite the existence of profitable manipulations, the colluders would fight amongst themselves regarding the monetary transfers, and never form a lasting coalition. Unfortunately, we show that for the common cases of

[6] The subscript t stands for truthful.

[7] Example 4 has disjoint C_t and C_b where $v(C_t \cup C_b) = v(C_t)$ but $v(C_b) > 0$ so $p(C_t \cup C_b) = p(C_t) + p(C_b) \leq v(C_t)$. One core constraint is $p(C_t) \geq v(C_t)$ so $p(C_t) = v(C_t) = v(C_t \cup C_b)$ and $p(C_b) = 0$. Another is $p(C_b) \geq v(C_b) > 0$, so some core constraints fail.

simple collusion (along the cheapest path or along a non-optimal path), the game always has a nonempty core, and there is a polynomially computable core imputation. Also, regarding *fairness*, we show that the Shapley value, considered "fair", is also in the core and easy to compute. Thus, the colluders can share the gains in a stable and fair manner[8], making collusion a significant problem in such auctions. Our results are based on showing the game is convex. We show convexity by examining the payment of a simple coalition C (on the cheapest path or on a non-optimal path). Denote the cheapest path as r_1 and the cheapest path that contains no edges in C as r_2. Denote the non colluders on the cheapest path r_1 as $T_{r_1} = r_1 \setminus C$. Denote the cost of a path r as $c(r) = \sum_{i \in r} c_i$, and the cost of the edges in T_{r_1} as $c(T_{r_1}) = \sum_{i \in T_{r_1}} c_i$.

Lemma 1 (Shortest Path Collusion Payments). *Let C be a simple coalition of colluders on the cheapest path. The total payment to the colluders under the optimal manipulation is $P^*(C) = |C|(c(r_2) - c(T_{r_1}))$.*

Proof. From Theorem 2, any colluder $i \in C$ would bid $b_i = 0$. Thus, the formula of Observation 1 is simplified to $p_i = c(r_2) - c(T_{r_1})$ (independent of the colluder's identity). Since there are $|C|$ colluders we obtain $P^*(C) = |C|(c(r_2) - c(T_{r_1}))$.

For collusion on a non-optimal path, we denote the optimal (cheapest) path as r_1 and non optimal path that contains C as r. We denote the non colluders on r as $T_r = r \setminus C$. The total cost of the edges in T_r is $c(T_r) = \sum_{i \in T_r} c_i$.

Lemma 2 (Non-Optimal Path Collusion Payments). *Let C be a simple coalition of colluders on a non-optimal path. The total payment to the colluders under the optimal manipulation is $P^*(C) = |C|(c(r_1) - c(T_r)$.*

Proof. The proof is similar to Lemma 1.

Theorem 5 (Convexity of the Collusion Game). *The collusion game is convex for simple coalitions (along the cheapest path or a non-optimal path).*

Proof. We give the proof for a simple coalition along the cheapest path (the other case is almost identical). An alternative definition of convex games is: $\forall S' \subseteq S \subseteq I, \forall i \notin S$: $v(S' \cup \{i\}) - v(S') \leq v(S \cup \{i\}) - v(S)$. We show this for simple coalition on the cheapest path, S. Consider any $S' \subset S$, denote $S \setminus S' = B$, and let a be any agent in $r_1 \setminus S$. Denote $T = r_1 \setminus S \setminus \{a\}$. Denote $|S| = h$ and $|S'| = l$ (where $l \leq h$), and denote $c(r_2) = x$. Using Lemma 1 we can write $v(S), v(S \cup \{a\}), v(S'), v(S' \cup \{a\})$. We have: $v(S \cup \{a\}) = v(S' \cup B \cup \{a\}) = (h+1)(x - c(T)) - c(S') - c(B) - c_a$; $v(S) = v(S' \cup B) = h(x - c_a - c(T)) - c(S') - c(B)$; $v(S' \cup \{a\}) = (l+1)(x - c(B) - c(T)) - C(S') - C_a$; $v(S') = l(x - c(B) - c_a - c(T)) - c(S')$. Opening parentheses and canceling terms we get: $v(S \cup \{a\}) - v(S) = x + h \cdot C_a - c(T) - C_a$; $v(S' \cup \{a\}) - v(S') = x + l \cdot C_a - c(T) - C_a - c(B)$. However, $l \cdot C_a \leq h \cdot C_a$ and $c(B)$ is non-negative, so we have $v(S' \cup \{i\}) - v(S') \leq v(S \cup \{i\}) - v(S)$.

Convexity of the collusion game has implications regarding how the colluders can share the gains. Collusion causes the prices paid to the agents to rise, and monetary transfers

[8] "Fairness" here is for the colluders — the manipulations are devastating for the auctioneer.

allow the colluders to share the utility in any way they desire. Under unstable utility distributions the colluders' coalition is likely to disintegrate, but convexity guarantees a *stable* distribution, so the colluders can distribute the gains so no subset of the colluders would benefit from leaving the coalition. The colluders may also want to share the utility in a *fair* manner, using the Shapley value. In general, even if there are *stable* allocations, the Shapley value may be unstable. Unfortunately, for simple coalitions, a stable allocation always exists, and the Shapley value is also stable.

Corollary 1. *For simple coalitions (on the cheapest path or on a non-optimal path), the collusion game has a non-empty core, containing the Shapley value.*

Proof. The collusion game is convex (Theorem 5), so it has a non-empty core coinciding with the Weber set. The Shapley value is in the Weber set so it is in the core.

A final barrier against collusion is computational complexity. Theorem 1 shows that finding the optimal collusion is hard, but it is trivial for simple coalitions. Corollary 1 guarantees the colluders a fair and stable allocation but it might be hard to compute, even for simple coalitions. We show that for simple coalitions, the colluders can tractably compute a simple core imputation or the Shapley value. Since the game is convex, the Weber set is identical to the core. Given a permutation $\pi = \langle \pi_1, \ldots, \pi_n \rangle$ of the agents and an agent e_i, denote the predecessors of i in π as F^i_π. Denote $m^\pi_i = v(F^i_\pi \cup \{e_i\}) - v(F_i)$, and note this can be computed in polynomial time using Lemma 1 (or Lemma 2). The imputation $\langle m^\pi_1, m^\pi_2, \ldots, m^\pi_n \rangle$ is in the Weber set (the Weber set is the convex hull of all these vectors for different permutations π), and so is a core imputation. A naive way of computing the Shapley value, the centroid of such vectors, requires performing this process for all agent permutations π, requiring exponential time. We show a polynomial algorithm to compute Shapley value.

Theorem 6. *For simple coalitions (on the cheapest path or on a non-optimal path), the Shapley value can be computed in polynomial time.*

Proof. The contribution of edge e_i to coalition C (where $e_i \notin C$), $v(C \cup \{e_i\}) - v(C)$ only depends on $|C|$, not on who the specific members of C are. Due to Lemma 1, we have $p^*(C) = |C|(c(r_2) - c(T^C_{r_1}))$ where $T^C_{r_1} = r_1 \setminus C$ are the non-colluders on the cheapest path. Denote $c(r_1) = x$, $c(r_2) = y$ and $\sum_{i \in C} c_i = z$. We have $p^*(C \cup \{e_i\}) - p^*(C) = (|C|+1)(y - c(r_1 \setminus C \setminus \{e_i\})) - |C|(y - c(r_1 \setminus C)) = (|C|+1)(y - x + z + c_i) - |C|(y - x + z) = |C|(y - x + z + c_i - y + x - z) + (y - x + z + c_i) = |C| \cdot c_i + y - x + z + c_i)$. Collusion on the cheapest path does not change the chosen path, so we have: $v(C \cup \{e_i\}) - v(C) = -\sum_{j \in C \cup \{e_i\}} c_j + p^*(C \cup \{e_i\}) + \sum_{j \in C} c_j - p^*(C) = -c_i + |C| \cdot c_i + y - x + z + c_i = |C| \cdot c_i + y - x + \sum_{i \in C} c_i$.

Consider computing the Shapley value for e_i, $\phi_i(v) = \frac{1}{n!} \sum_{\pi \in \Pi} v(F^i_\pi \cup \{e_i\}) - v(F^i_\pi)$ (where F^i_π are the predecessors of i in π). Denote $\psi_i(v) = \sum_{\pi \in \Pi} v(F^i_\pi \cup \{e_i\}) - v(F^i_\pi)$. We can compute ψ_i by iterating over the possible *numbers* of predecessors i has in π, $|F^i_\pi|$. Let Π_j be all permutations $\pi \in \Pi$ such that $|F^i_\pi| = j$ (i.e. permutations where i has exactly j predecessors. We can denote the total contribution that i has for coalitions of size j as $M_j = \sum_{\pi \in \Pi_j} v(F^i_\pi \cup \{e_i\}) - v(F^i_\pi)$. Thus we have $\psi_i(v) = \sum_{j=0}^{n-1} M_j$. Thus we only need to compute M_j in polynomial time (for $0 \le j \le n-1$).

To compute $M_j = \sum_{\pi \in \Pi_j} v(F^i_\pi \cup \{e_i\}) - v(F^i_\pi)$ we can sum over all possible predecessor sets for i where i has exactly j predecessors, $F = \{F^i_\pi \subseteq N | \pi \in \Pi_j\}$, where $|F| = \binom{n-1}{j}$. Under this notation $M_j = \sum_{C \in F} v(C \cup \{e_i\}) - v(C)$. We've shown that $v(C \cup \{e_i\}) - v(C) = |C| \cdot c_i + y - x + \sum_{i \in C} c_i$, so we have: $M_j = \sum_{C \in F} |C| \cdot c_i + y - x + \sum_{i \in C} c_i$. Since any $C \in F$ have the same size $|C| = j$, we get: $M_j = |F| \cdot (j \cdot c_i + y - x) + \sum_{C \in F} \sum_{i \in C} c_i$. We denote $q = \sum_{C \in F} \sum_{i \in C} c_i$. Thus, $M_j = \binom{n-1}{j} \cdot (j \cdot c_i + y - x) + q$. Consider computing q. Given the coalition C of size $|C| = m$, denote the weights c_i for all $i \in C$ as $W = \langle c_1, \ldots, c_{i-1}, c_{i+1}, \ldots, c_m \rangle$ (W is the set of the costs of all colluders except e_i). Thus, q is simply the sum of weights in all subsets of W of size j, i.e. $q = \sum_{S \subset W || S| = j} \sum_{s \in S} s$. Any weight $w_i \in W$ appears in q exactly $\binom{n-1}{j-1}$ times, so $q = \binom{n-1}{j-1} \sum_{c_x \in W} c_x = \binom{n-1}{j-1} \sum_{e_x \in C \setminus \{e_i\}} c_x$. Given a colluder e_i, we can easily compute $\sum_{e_x \in C \setminus \{e_i\}} c_x$ in polynomial time, and thus compute q in polynomial time. This allows us to compute $M_j = \binom{n-1}{j} \cdot (j \cdot c_i + y - x) + q$ in polynomial time, for any j. We can thus compute $\psi_i(v) = \sum_{j=0}^{n-1} M_j$ in polynomial time, and thus compute the Shapley value of any agent in polynomial time.

4 Related Work

Auctions face untruthful selfish agents, so due to strategic behavior, a mechanism trying to maximize welfare may reach a sub-optimal decision. Proper payment rules incetivize agents to bid truthfully. A prominent scheme for doing so is the VCG mechanism [23,10,13]. Despite its advantages, VCG has many shortcomings [3], including vulnerability to collusion [17]. Collusion can occur in many domains and many of its forms are illegal [17]. Our model follows an analysis of multi-unit auctions [4], but for the PPA domain [1]. We examine coalitional deviations, but as opposed to strong Nash equilibrium [2], we convert the normal-form game to a cooperative game. We provide an *internal* model of collusion, as opposed to *external interventions* models [18,6,19]. We focus on the core [12] and Shapley value [20]. The Shapley value and other power indices are typically hard to compute [16,8], so our result for computing it in some collusion games is interesting. For general collusion games, the colluders can *approximate* [7,16] the Shapley value in order to share their gains. Collusion is also related to shills and false-identity attacks [26,9,5,27], where a single agent pretends to be several agents. A single edge in a VCG PPA can pretend to be several edges to increase payments. Our work is also related to of bidding rings and clubs [15,14], but we assume the colluders have full information on each other's costs. Core selecting auctions [11] are also related to our work, but we do not consider the auctioneer a participating agent, and the core in our collusion game can be empty.

5 Conclusion

We analyzed collusion in VCG PPAs, and showed that such a domain is vulnerable to collusion. Some questions remain open for future research. First, similar analysis can be done for other auctions, such as combinatorial auctions or sponsored search auctions. Second, due to the drawbacks of VCG for PPAs, what alternative mechanisms should be used? Finally, what bids are likely to occur for domains with an empty core?

References

1. Archer, A., Tardos, É.: Frugal path mechanisms. ACM Transactions on Algorithms, TALG (2007)
2. Aumann, R.: Acceptable points in general cooperative n-person games. In: Contributions to the Theory of Games, vol. IV, pp. 287–324 (1959)
3. Ausubel, L.M., Milgrom, P.: The lovely but lonely vickrey auction. In: Combinatorial Auctions, ch. 1, pp. 17–40 (2006)
4. Bachrach, Y.: Honor among thieves: Collusion in multi-unit auctions. In: AAMAS (2010)
5. Bachrach, Y., Elkind, E.: Divide and conquer: False-name manipulations in weighted voting games. In: AAMAS, pp. 975–982 (2008)
6. Bachrach, Y., Elkind, E., Meir, R., Pasechnik, D., Zuckerman, M., Rothe, J., Rosenschein, J.: The cost of stability in coalitional games. In: Mavronicolas, M., Papadopoulou, V.G. (eds.) Algorithmic Game Theory. LNCS, vol. 5814, pp. 122–134. Springer, Heidelberg (2009)
7. Bachrach, Y., Markakis, E., Resnick, E., Procaccia, A., Rosenschein, J., Saberi, A.: Approximating power indices: theoretical and empirical analysis. Autonomous Agents and Multi-Agent Systems 20(2), 105–122 (2010)
8. Bachrach, Y., Rosenschein, J.S.: Computing the Banzhaf power index in network flow games. In: AAMAS, Honolulu, Hawaii, pp. 323–329 (May 2007)
9. Chen, C., Tauman, Y.: Collusion in one-shot second-price auctions. Economic Theory 28, 145–172 (2006)
10. Clarke, E.H.: Multipart pricing of public goods. Public Choice 11(1), 17–33 (1971)
11. Day, R., Milgrom, P.: Core-selecting package auctions. International Journal of Game Theory 36(3), 393–407 (2008)
12. Gillies, D.B.: Some theorems on n-person games (1953)
13. Groves, T.: Incentives in teams. Econometrica 41, 617–663 (1973)
14. Leyton-Brown, K., Shoham, Y., Tennenholtz, M.: Bidding clubs: institutionalized collusion in auctions. In: ACM EC (2000)
15. Leyton-Brown, K., Shoham, Y., Tennenholtz, M.: Bidding clubs in first-price auctions. In: AAAI (2002)
16. Matsui, Y., Matsui, T.: A survey of algorithms for calculating power indices of weighted majority games. Journal of the Operations Research Society of Japan 43 (2000)
17. Milgrom, P.: Putting Auction Theory To Work. Cambridge University Press, Cambridge (2004)
18. Monderer, D., Tennenholtz, M.: Strong mediated equilibrium. Artificial Intelligence 173(1), 180–195 (2009)
19. Resnick, E., Bachrach, Y., Meir, R., Rosenschein, J.: The cost of stability in network flow games. In: Královič, R., Niwiński, D. (eds.) MFCS 2009. LNCS, vol. 5734, pp. 636–650. Springer, Heidelberg (2009)
20. Shapley, L.S.: A value for n-person games. Cont. Theory of Games (1953)
21. Shapley, L.S.: Cores of convex games. Internat. J. Game Theory 1, 12–26 (1971)
22. Sullivan, A., Sheffrin, S.: Economics: principles in action. Prentice-Hall, Englewood Cliffs (2003)
23. Vickrey, W.: Counterspeculation, auctions, and competitive sealed tenders. The Journal of Finance 16(1), 8–37 (1961)
24. Weber, R.J.: Probabilistic values for games. Technical Report, Cowles (1977)
25. Winter, E.: The Shapley Value. Game theory with economic applications (2002)
26. Yokoo, M., Conitzer, V., Sandholm, T., Ohta, N., Iwasaki, A.: Coalitional games in open anonymous environments. In: AAAI (2005)
27. Zuckerman, M., Faliszewski, P., Bachrach, Y., Elkind, E.: Manipulating the quota in weighted voting games. In: AAAI (2008)

Sequential Item Pricing for Unlimited Supply

Maria-Florina Balcan and Florin Constantin

College of Computing, Georgia Institute of Technology
{ninamf,florin}@cc.gatech.edu

Abstract. We study prior-free revenue maximization for a seller with unlimited supply of n item types facing m myopic buyers present for $k < \log n$ days. We show that a certain randomized schedule of posted prices has an approximation factor of $O(\frac{\log m + \log n}{k})$. This algorithm relies on buyer valuations having hereditary maximizers, a novel natural property satisfied for example by gross substitutes valuations. We obtain a matching lower bound with multi-unit valuations. In light of existing results [2], k days can thus improve the approximation by a $\Theta(k)$ factor. We also provide a posted price schedule with the same factor for positive affine allocative externalities, despite an increase in the optimal revenue.

1 Introduction

In most transactions, prices are set on items (not on bundles of items) to simplify buyers' and sellers' decisions. Arguably, a seller's main objective is to maximize profit−this basic problem has received tremendous attention in the optimization literature, often assuming poor information about uncertain demand.

Given a set of buyer valuations for bundles, the optimum revenue is the optimal allocation's total value. Posted prices yielding such a high revenue do not usually exist, because buyers' valuations are private and may be quite complex. A standard compromise [2,3] is then to aim for revenue that is at least (possibly in expectation) a fraction $1/c$ of the optimum for *any* set of buyers, i.e., more formally, to design algorithms with a low revenue approximation factor $c > 1$.

We focus on *unlimited supply* [2,3], a setting relevant to digital media (e.g. DVDs or software programs). A seller has unlimited supply of n item types if the marginal cost of producing an additional copy of any item is negligible. For unlimited supply, the highest possible revenue equals the sum of the maxima of buyers' valuations. Assuming only an upper bound on the m buyers' arbitrary valuations, Balcan et al. [2] provide a *one-shot* randomized price (the same for each item) that yields revenue a $\Theta(1/(\log m + \log n))$ fraction of the optimum. Like [2], we price all items equally, i.e. we use *linear uniform* prices. This involves the least price discrimination possible under static pricing: no buyer or item is favored over another. In fact, some online movie retailers have very limited variability in prices – e.g. iTunes offers only two prices for movies, older movies having a discount.

In practice buyers purchase more than once from the same seller. It is then natural to investigate improved approximation factors if all buyers are present

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 50–62, 2010.

for $k < n$ time periods, that we call *days*. In general, the seller may update prices "on the fly" based on realized demand. For simplicity we only use, like [2,3], price sequences decided ahead of time, but only revealed gradually to buyers. A buyer starts with no items and accumulates them over time. We assume that any buyer is *forward-myopic*, i.e. greedily purchases a preferred set in each day.[1]

We are now ready to state the central question in this paper

Question: *What revenue approximation factor $C_{m,n,k}$ is achievable with m forward-myopic buyers, n items in unlimited supply and k equal item prices?*

Balcan et al. [2]'s results yield $C_{m,n,1} = O(\log m + \log n)$ and $C_{1,n,1} = \Theta(\log n)$.

We provide a general lower bound and an upper bound on the revenue factor achievable (we introduce in detail these results and define these classes shortly).

$$\textbf{\textit{Answer}}: \begin{cases} \textit{Even for concave multi-unit valuations, } C_{m,n,k} = \Theta(\frac{\log m + \log n}{k}) \\ \textit{For valuations with hereditary maximizers, } C_{m,n,k} = O(\frac{\log m + \log n}{k}) \end{cases}$$

We show that no scheme with k successive prices can approximate revenue to a factor lower than $\Theta(\frac{\log m + \log n}{k})$, even for concave multi-unit (i.e. that do not differentiate items) valuations, some of the most basic combinatorial valuations.

Our main result however is a matching upper bound. We show that generating k independent random prices and offering them in decreasing order approximates revenue to no worse than a $\Theta(\frac{\log m + \log n}{k})$ fraction, thus improving [2]'s approximation by a $\Theta(k)$ factor. Our technical contribution is to generalize a guarantee on the expected profit from one random price to k such prices. While the bound for one price uses a standard technique for worst-case bounds, the only improvement for general k that we are aware of (by Akhlaghpour et al. [1]) is exponentially worse than ours as their recursive construction only yields a $\Theta(\log k)$ factor improvement. We connect revenue from a valuation v with the joint area of k rectangles, determined by prices, under v's demand curve F. While each such rectangle covers in expectation a logarithmic fraction of the area under F, we are able to limit the overlaps of rectangles by carefully analyzing the k prices as order statistics. If all valuations have the same maximum (or obviously if $m = n^{O(1)}$), then we can improve our two bounds to $\Theta(\frac{\log n}{k})$.

Our upper bound (in particular our connection between revenue and area under demand curve covered by price-based rectangles) relies on the natural sufficient condition (that we identify) of buyer valuations having *hereditary maximizers* (HM). The HM property essentially states that an algorithm greedily selecting items by their marginal value has at each step a set of maximum value among sets with the same size. In particular, multi-unit valuations and gross substitutes valuations (a classical model in economics, see e.g. [5,7]) have HM.

[1] If buyers chose the lowest price p in a sequence $\mathcal{P}$, then p would be equivalent to $\mathcal{P}$. Steve Jobs, CEO of Apple, Inc., writes about the cut of iPhone prices: "If you always wait for the next price cut [...], you'll never buy any technology product because there is always something better and less expensive on the horizon". `http://www.apple.com/hotnews/openiphoneletter`, September 2007.

Submodular valuations may not have HM, leading to a counter-intuitive phenomenon: the revenue from offering a high price followed by a low one may be less than the revenue from the low one only. Consider three movies: a very good science-fiction (S) one and an animation (A) movie and a drama (D), both of slightly inferior quality. A typical family prefers S to A or D, but A and D (for variety) to any other pair; the family does not strategize about price schedules. If a greedy movie retailer starts with high prices and reduces them afterwards (on all movies) then, despite good revenue on S, it loses the opportunity of more revenue by selling A and D instead. This (submodular) valuation, formalized in Section 2.1, does not have HM. S is preferred to each of the other two, whose pair is preferred to the other pairs. For HM valuations however, this counter-intuitive reversal does not occur, which is critical, as we show, for good sequential revenue.

Finally, we initiate the study of revenue maximization given allocative *externalities* (i.e. influences) between buyers with combinatorial valuations. We provide a rather general model of positive influence of others' ownership of items on a buyer's valuation. For affine, submodular externalities and base valuations with HM we present an influence-and-exploit [8] marketing strategy based on our algorithm for private valuations. This strategy preserves our approximation factor, despite an affine increase (due to externalities) in the optimum revenue.

Related work. The prior-free, unlimited supply domains studied for revenue maximization have been less general than the one in this paper. Balcan et al. [2] present structural results for one-shot pricing and achieve a tight $\Theta(\log m + \log n)$-factor revenue approximation via a single random price. Bansal et al. [3] study buyers with values in $[1, H]$ for one item type and arrival–departure intervals. They obtain almost matching upper and lower bounds on the approximation factor: $O(\log H)$ and $O(\log \log H)$ for deterministic and randomized schemes.

While externalities are natural and well-studied in social networks [9], the corresponding revenue maximization problem has been recently introduced by Hartline et al. [8], who investigate approximation via single-item distribution-based influence-and-exploit marketing strategies. Akhlaghpour et al. [1] study this problem for a seller that cannot use price discrimination amongst buyers.

Paper structure. After introducing notation in Section 2, we review known bounds on $C_{m,n,k}$ and provide a new lower bound in Section 3. Section 4 analyzes hereditary maximizers, a property of valuations leading to a $C_{m,n,k}$ upper bound established in Section 5. Finally, in Section 6, we model externalities, where a buyer's valuation depends on others' items, and extend Section 5's approximation. Due to space constraints, we defer some proofs and examples to the full version[2].

2 Preliminaries

We consider a seller with n item types in unlimited supply. The seller can thus profit from selling copies of an item at any price but aims to maximize its revenue. The seller has $k < n$ sale opportunities called *days*. There are m customers with

[2] Available at `http://arxiv.org/abs/1009.4606` and from the authors' webpages.

quasilinear utilities present in all k days. Customers have valuations over bundles of items (not more than one per type); we denote a generic such valuation[3] by $v_i : 2^{1..n} \to \mathbb{R}$ and its maximum by H_i. We assume that the seller knows only the highest maximum across customers $H = \max_i H_i = \max_{i\in 1..m, S\subseteq 1..n} v_i(S)$.

We treat static pricing first and then dynamic pricing in Section 2.1. We only use the simplest form of pricing, with no item or buyer discrimination. A price vector $\mathbf{p} \in \mathbb{R}^n$ is *linear uniform*[4] if $p_j = p$, $\forall\, j = 1..n$.
Given a price vector, a customer buys a preferred (utility-maximizing) bundle.

Definition 1. *For price vector* $\mathbf{p} \in \mathbb{R}^n$*, the* demand correspondence *[7]* $\mathcal{D}_v(\mathbf{p})$ *of valuation* v *is the set of utility-maximizing bundles at prices* $\mathbf{p}$*:*

$$\mathcal{D}_v(\mathbf{p}) = \mathrm{argmax}_{S\subseteq 1..n}\{v(S) - \textstyle\sum_{j\in S} p_j\} \tag{1}$$

For linear uniform price $\mathbf{p} = p \cdot \mathbf{1}$*, let* $\mathcal{D}_v(p) = \mathcal{D}_v(\mathbf{p})$ *and* $F_v(p) = \min_{S\in\mathcal{D}(p\cdot\mathbf{1})} |S|$ *be the least number of items in a bundle demanded (by valuation*[5] v*) at prices* p*.*

As one would expect, a higher price cannot increase the least quantity bought.

Lemma 1. *[2] For an arbitrary valuation* v *and* $p > p'$*,* $F(p) \le F(p')$*.*

2.1 Sequential Pricing

Assume the seller offers equal item prices $r^d \in \mathbb{R}_+$ in day $d = 1..k$, with $r^1 > \ldots > r^k$. We now define buyers' behavior over time, starting with no items before day 1.

We model any buyer as *forward-myopic*: assume that before day d he buys sets $S_1, \ldots, S_{d-1}$. His utility for items $S \subseteq 1..n \setminus (S_1 \cup \ldots \cup S_{d-1})$ he does not own is

$$u_{d,\ldots,1}(S_1, \ldots, S_{d-1}, S, r^1 \ldots r^d) = v(S_1 \cup \ldots \cup S_{d-1} \cup S) - (\textstyle\sum_{l=1}^{d-1} r^l |S_l|) - r^d |S| \tag{2}$$

i.e. a customer does not anticipate price drops but does take into account past purchases (accumulating items) and payments to decide a utility-maximizing set S to buy today. In this model, a customer buys nothing in a day where the price increases[6], hence our focus on decreasing price sequences: the seller starts with a high price and then gradually reveals discounts, a common retail practice.

Following Def. 1, we denote preferred bundles outside $S_1 \cup \ldots \cup S_{d-1}$ at r^d by

$$\mathcal{D}_v^{S_1,\ldots,S_{d-1}}(r^1 \ldots r^d) = \underset{S\subseteq 1..n\setminus(S_1\cup\cdots\cup S_{d-1})}{\mathrm{argmax}} u_{d,\ldots,1}(S_1, \ldots, S_{d-1}, S, r^1 \ldots r^d)$$

We briefly consider incentive properties before focusing on revenue only.

---

[3] Only in Section 6 does a valuation vary over time (with others' items).
[4] Different (non-uniform) item prices are also (e.g. [5]) called linear prices.
[5] Except for Section 6, v will be clear from context and omitted from $\mathcal{D}$ and F.
[6] We sketch a proof for $d = 2$: let $r^1 < r^2$ and S_i bought in day $i = 1, 2$ with $S_1 \cap S_2 = \emptyset$. Suppose $S_2 \neq \emptyset$; then $v(S_1 \cup \emptyset) - r^1|S_1| - r^2|\emptyset| \le v(S_1 \cup S_2) - r^1|S_1| - r^2|S_2| < v(S_1 \cup S_2) - r^1|S_1| - r^2|S_2|$, i.e. $S_1 \cup S_2$ is preferred to S_1 at price r^1, contradiction.

Incentive considerations. A buyer's utility cannot decrease in any day: there is always the option of not buying anything. Thus, any sequence of prices defines an individually rational mechanism. Furthermore, within a day, as each buyer faces the same prices, buyers have no envy and no profitable item swaps.

Our goal is revenue maximization via (possibly randomized) price sequences decided ahead of time (but only revealed gradually to buyers).

Definition 2. *A pricing scheme $\mathcal{P}$ is a sequence*[7] *of k (possibly random) decreasing prices.* $\text{Rev}_{\mathcal{P}}(v_1,\ldots,v_m)$ *denotes $\mathcal{P}$'s revenue (in expectation for randomized $\mathcal{P}$), for valuations $v_1,\ldots,v_m$ and least favorable tie-breaking decisions by buyers.*

A standard [2,8] revenue benchmark is customers' total willingness to pay. We study worst-case guarantees, that hold regardless of buyer valuations.

Definition 3. *A (possibly randomized) pricing scheme $\mathcal{P}$ is a c-*revenue approximation *(where $c \geq 1$) if* $\sum_{i\in 1..m} \max_{S\subseteq 1..n} v_i(S) \leq$[8] $c \cdot \mathbb{E}[\text{Rev}_{\mathcal{P}}(v_1,\ldots,v_m)]$ *for all valuations $v_1 \ldots v_m$, where the expectation is taken over $\mathcal{P}$'s random choices.*

Recall our main question: assessing what revenue approximation factors $C_{m,n,k}$ are achievable. Clearly, $C_{m,n,k+1} \leq C_{m,n,k}$ and $C_{m,n,k} \leq C_{m+1,n,k}$. Next section formally states known values of $C_{m,n,k}$ for particular (m,n,k) triples, provides some intuition for sequential pricing and presents our lower bound $C_{m,n,k} = \Omega(\frac{\log m+\log n}{k})$. Section 5 presents the upper bound $C_{m,n,k} = O(\frac{\log m+\log n}{k})$.

3 Existing Bounds for $C_{m,n,k}$ and a New Lower Bound

Motivated by worst-case pricing bounds [2,3], we use prices $q_l = H/2^l$ for $l \geq 0$. Algorithm Random_D^H outputs one-shot price q_l where the scaling exponent l is chosen uniformly at random in $0..D-1$. Despite its simplicity, Random_D^H is quite effective in general and as effective as any other algorithm for one buyer.

Lemma 2. *[2] For*[9] $t = 1+\log m+\log n$*,* Random_t^H *is a $4t$-revenue approximation. For one buyer, i.e. $m=1$, this factor is tight (modulo a constant factor). Thus, $C_{m,n,1} = O(\log m+\log n)$ and $C_{1,n,1} = \Theta(\log n)$.*

For $n=1$, $\text{Random}_{1+\log H}^H$ is [3] a 2-approximation, i.e. $C_{m,1,1+\log H} = 2$.

For the rest of this section we assume that the seller fully knows buyers' valuations. This strong assumption will allow us to understand two special settings.

The first setting is concerned with one buyer $m=1$ with a known monotone $(H = v(1..n))$ valuation v and many days $k=n$. In this setting, full revenue can be obtained from v i.e. $C_{1,n,n} = 1$. Assume that items are numbered in the order a greedy algorithm on marginal values would choose them, i.e. for all $i = 1..n$,

---

[7] We can expand, without changing revenue, any shorter sequence $\mathcal{P}$ to k prices by appending to $\mathcal{P}$ copies of its last price.

[8] This benchmark is at least $\max_{S\subseteq 1..n} \sum_{i\in 1..m} v_i(S)$, i.e. the highest joint value of a set.

[9] This paper only uses base 2 logarithms.

$v(\{1..i\}) \geq v(\{1..(i-1), x\}), \forall x \in i..n$. Then, at price $r^i = v(\{1..i\}) - v(\{1..(i-1)\})$ in day $i \in 1..n$, exactly[10] item i is bought (ignoring ties). The sum of all days' revenues telescopes to $v(1..n) = H$.

The second setting yields the first half of the answer to our main question: a lower bound on achievable $C_{m,n,k}$, showing the effect of limited (k) price updates and buyer differences, even with known valuations and identical items.

Theorem 1 (lower bound). *Define* $v_s(x) = \begin{cases} x/2^{s-1}, & \text{if } x \leq 2^{s-1} \\ 1, & \text{if } x > 2^{s-1} \end{cases} \forall 1 \leq s \leq N+1$ *(for* $N = \log n \in \mathbb{Z}$*) to be* $1 + \log n$ *concave multi-unit valuations, each with maximum* 1*:* $v_1(x) = 1, \forall x = 1..n$ *and* $v_{N+1}(x) = x/2^N = x/n, \forall x = 1..n$*. Then the revenue of any sequence of* $k < \log n$ *prices is at most* $2k$*. Thus, even if valuations have the same maximum, any* k*-day pricing algorithm must have a higher revenue approximation factor than* $\frac{1+\log n}{2k}$*, even for* $1 + \log n$ *buyers:* $\frac{1+\log n}{2k} \leq C^{\text{equal maxima}}_{1+\log n, n, k}$*. In general,* $C_{m,n,k} = \Omega(\frac{\log m + \log n}{k})$*.*

Informally, each v_s has a constant non-zero marginal value (MV) for one item in $[1/n, 1]$. A low price is effective for low MV buyers but could profit more from high MV buyers. A high price fails to sell any item to low MV buyers despite getting good revenue from high MV buyers. This reasoning extends to short ($k < \log n$) sequences of prices. For $k = 1$, we get $C_{m,n,1} = \Theta(\log m + \log n)$, showing that [2]'s bound $C_{m,n,1} = O(\log m + \log n)$ is tight for all m.

After this section we will provide positive results only. In preparation we require another piece of bad news highlighting the importance of sequential consistency. Even the seemingly innocuous assumption of decreasing prices can hurt revenue. We now provide a submodular valuation consistent with the movie example in the introduction exhibiting a counterintuitive revenue non-monotonicity.

Example 1. *Let* a *be the science-fiction movie, and* b, c *be the animation and drama. Define a valuation* v *by* $v(a) = 3, v(b) = v(c) = 2.1, v(a,b) = v(a,c) = 3.8, v(b,c) = v(a,b,c) = 4.2$*. For* $r^1 = 1.5$*,* $\mathcal{D}(r^1) = \{\{a\}\}$ *and for* $r^2 = 1, \mathcal{D}(r^2) = \{\{b,c\}\}$*. Neither* b *or* c *is worth \$*1 *given* a*:* $\mathcal{D}^{\{a\}}(r^1, r^2) = \{\emptyset\}$*. Less revenue (\$*1.5*) is obtained from offering* r^1 *followed by* r^2 *than from* r^2 *alone (\$*2*).*

4 Hereditary Maximizers

We now define hereditary maximizers, a new property of valuations and establish it for multi-unit and gross substitutes valuations. In Section 5 we will show that it is sufficient for good sequential revenue: in particular, unlike in Example 1, the revenue from any price sequence is at least that from its lowest price.

Definition 4. *Valuation* v *has* hereditary maximizers *(HM) if given any size* j *value-maximizing bundle* S_j*, one item can be added to it to obtain a size* $j+1$ *such bundle. Letting* $\mathcal{M}^v_j = \text{argmax}_{|S|=j}\, v(S)$*,* v *has HM if*

[10] Assuming v submodular; if not, we can set r^i to v's steepest slope given items owned.

$$\forall\, n \geq j \geq 1,\ \forall\, S_j \in \mathcal{M}_j^v, \exists S_{j+1} \in \mathcal{M}_{j+1}^v \text{ with } S_j \subset S_{j+1} \qquad \text{(HM)}$$
$$\textit{implying } \forall\, n \geq j' > j \geq 1,\ \forall\, S_j \in \mathcal{M}_j^v, \exists S_{j'} \in \mathcal{M}_{j'}^v \text{ with } S_j \subset S_{j'} \qquad \text{(HM*)}$$

Thus, a valuation has hereditary maximizers if a greedy algorithm selecting the highest marginal value item at each step always maintains, regardless of its tie-breaking decisions, a set of maximum value among sets of the same size. Example 1's valuation v *does not* have HM: $\mathcal{M}_1^v = \{\{a\}\}$ but $\mathcal{M}_2^v = \{\{b, c\}\}$.

Several well-studied valuation classes are HM as we show shortly. A multi-unit valuation v, a basic combinatorial valuation, treats all items identically. Hence, for any j, $\mathcal{M}_j^v$ is the collection of all sets of size j and v trivially has HM.

Lemma 3. *A multi-unit valuation has hereditary maximizers.*

A valuation is *gross substitutes*, a well-studied condition in assignment problems [5,7], if raising prices on some items preserves the demand on other items.

Definition 5. *A valuation v is* gross substitutes *(GS) if for any price vectors*[11] $\mathbf{p}' \geq \mathbf{p}$*, and any $A \in \mathcal{D}(\mathbf{p})$ there exists $A' \in \mathcal{D}(\mathbf{p}')$ with $A' \supseteq \{i \in A : p_i = p'_i\}$.*

Remarkably [7], for any set of GS buyers with public valuations, there exists a Walrasian (or competitive) equilibrium with one-shot item (possibly non-uniform) prices, i.e. at which buyers' preferred bundles form a partition of all items. Among GS valuation classes (see [10] for more examples) are unit demand valuations (that define the value of a set as the highest value of an item within the set) and concave multi-unit valuations. We know from Lemma 3 that the latter valuations have HM – this is not a coincidence.

Theorem 2. *[4] A gross substitutes valuation has hereditary maximizers.*

Bertelsen [4] implicitly proves Theorem 2, without defining HM. The full version provides a simpler proof for it via a basic graph-theoretic fact starting, like [4], from Lien and Yan's [10] GS characterization.

Lemma 4. *[10] v is gross substitutes if and only if v is submodular and*

$$\forall \textit{ items } a, b, c, \textit{set } S, v^S(ab) + v^S(c) \leq \max\{v^S(ac) + v^S(b), v^S(bc) + v^S(a)\} \quad (3)$$

i.e. no unique maximizer among $v^S(ab) + v^S(c), v^S(ac) + v^S(b), v^S(bc) + v^S(a)$ where $v^S(A) = v(S \cup A) - v(S)$, $\forall\, A \subseteq 1..n \backslash S$ denotes A's marginal value over S.

The high-level idea of Theorem 2's proof is as follows. We define a (bipartite) directed graph among certain sets of equal size; an edge from set S to set S' shows that S has a strictly higher certain marginal value in v than S'. If v did not have HM, then this graph would have a directed cycle which is impossible.

We now exhibit a rich class of valuations that are HM, but not GS (see the full version for other classes and proofs). They attest to the richness of our HM

[11] We compare price vectors $\mathbf{p}, \mathbf{p}' \in \mathbb{R}^n$ component-wise: $\mathbf{p}' \geq \mathbf{p} \iff p'_j \geq p_j\ \forall\, j = 1..n$.

class, even when compared to the well-studied GS class. In an *order-consistent* valuation, for any $L \in 1..n$ and any sets $\{j_1, \dots, j_L\}$ and $\{j'_1, \dots, j'_L\}$, whenever $v(\{j_l\}) \geq v(\{j'_l\}), \forall\, l = 1..L$ then $v(\{j_1, \dots, j_L\}) \geq v(\{j'_1, \dots, j'_L\})$, with strict inequality if at least one single item inequality is strict.

We proceed with a quantity guarantee for HM valuations, that will be critical for guarantees on sequential revenue. No fewer items are sold for price sequence $r^1, \dots, r^d$ (regardless of which preferred bundles are bought) than in the worst-case for r^d alone, i.e. $F(r^d)$. This guarantee follows from a strong structural property, that we highlight for $d = 2$. Any set $S_1 \in \mathcal{D}(r^1)$ (i.e. preferred at a higher linear uniform price r^1) can serve as base to create sets preferred at the lower price $r^2 < r^1$ via joining any set $S_2 \in \mathcal{D}^{S_1}(r^1, r^2)$ (i.e. preferred sequentially at r^2 after buying S_1): formally, $S_2 \cup S_1 \in \mathcal{D}(r^2)$.

Theorem 3. *Fix an HM valuation v, a day $d \leq k$ and prices $r^1 > \cdots > r^d$. Let $S_\delta \in \mathcal{D}^{S_1,\dots,S_{\delta-1}}(r^1, \dots, r^\delta)$ preferred at r^δ given sets $S_1,\dots,S_{\delta-1}$ sequentially bought at $r^1,\dots, r^{\delta-1}\, \forall\, \delta = 1..d$. Then $\bigcup_{\delta=1}^{d} S_\delta \in \mathcal{D}(r^d)$ and thus $\sum_{\delta=1}^{d} |S_\delta| \geq F(r^d)$.*

We first state a property used in Theorem 3's proof. Clearly, a size j set (if any) preferred at a uniform price cannot have a higher value than another size j set.

Lemma 5. *For all prices r and sizes j, $\mathcal{D}(r) \cap \{|S| = j\}$ is either empty or $\mathcal{M}^v_j$.*

Proof (of Theorem 3). We treat the case $d = 2$; the proof for general d is similar.

Let $S_1 \in \mathcal{D}(r^1)$ be a set preferred at price r^1 and assume $|S_1| < F(r^2)$ (otherwise the claim is immediate). Let $S_2 \in \mathcal{D}^{S_1}(r^1, r^2)$ be a set preferred at price r^2 after having bought S_1 at price r^1. By Lemma 5, $S_1 \in \mathcal{M}_{\emptyset,|S_1|}$. As $F(r^2) > |S_1|$, by (HM*), $\exists S'_2 \in \mathcal{M}_{\emptyset,F(r^2)}$ a minimal set preferred at price r^2 with $S_1 \subset S'_2$. As $\mathcal{M}_{\emptyset,F(r^2)} \cap \mathcal{D}(r^2) \neq \emptyset$ (it contains S'_2), by Lemma 5, $S'_2 \in \mathcal{D}(r^2)$.

Let $u_S = v(S \cup S_1) - r^1|S_1| - r^2|S|$ be the utility from buying $S \subseteq 1..n \backslash S_1$ at r^2 after buying S_1 at r^1. As $S_2 \in \mathcal{D}^{S_1}(r^1, r^2)$, $u_{S'_2 \backslash S_1} - u_{S_2} = (v(S'_2) - r^2|S'_2|) - (v(S_2 \cup S_1) - r^2|S_2 \cup S_1|) \leq 0$. If $u_{S'_2 \backslash S_1} < u_{S_2}$ then $S_2 \cup S_1$ is preferred to S'_2 at r^2, contradicting $S'_2 \in \mathcal{D}(r^2)$. Thus $u_{S'_2 \backslash S_1} = u_{S_2}$ implying $S_2 \cup S_1 \in \mathcal{D}(r^2)$.

5 Revenue Approximation for Independent HM Valuations

We now leverage Theorem 3 towards an upper bound matching (up to a constant factor) our $C_{m,n,k} = \Omega(\frac{\log m + \log n}{k})$ lower bound. Let $L = 1 + \log m + \log n$.

Theorem 4 (upper bound). *Consider m HM valuations with maxima $H_1 \dots H_m$ and let $H = \max_{i \in 1..m} H_i$. Consider k prices $q_{x_1} = \frac{H}{2^{x_1}} \geq \cdots \geq q_{x_k} = \frac{H}{2^{x_k}}$ where $x_1 \leq \cdots \leq x_k$ are the first (lowest), ..., k-th (highest) order statistics of k iid $U[0, L]$ continuous random variables $u_1, \dots, u_k$. These prices yield expected revenue $\Omega(\frac{k}{\log m + \log n}) \sum_{i \in 1..m} H_i$. Thus $C_{m,n,k} = O(\frac{\log m + \log n}{k})$.*

If all v_i's have the same maximum ($H_i = H, \forall\, i \in 1..m$) then, as in [2], the approximation factor can be improved to $\Omega(\frac{\log n}{k})$ by using $L = \log(2n)$. Recalling Theorem 1's lower bounds, *our bounds are tight* modulo a constant factor.

Before proceeding with Theorem 4's proof, we review, motivated by Lemma 1, a natural analogue of a well-studied economic concept and relate it to H.

A valuation v's *demand curve* [2] is a step function given by $(p_l, F(p_l))_{l=0..n_v+1}$ (with $n_v \le n$) where threshold prices $0 = p_0 < p_1 < .. < p_{n_v} \le p_{n_v+1} = H$ satisfy $F(p_l) = F(p) > F(p_{l+1}), \forall p \in [p_l, p_{l+1}), \forall l = 0..n_v$. That is, for any l and any price p in $[p_l, p_{l+1}]$ the lowest size of a (preferred) bundle in $\mathcal{D}_v(p)$ is $F(p_l)$. The area A_F under v's demand curve is defined as $\sum_{l=1}^{n_v} p_l(F(p_l) - F(p_{l+1}))$.

Lemma 6. *[2] $\mathrm{A}_F = H = \max_{S \subseteq 1..n} v(S)$, i.e. v's maximum willingness to pay.*

The (worst-case) revenue $pF(p)$ from a single price p equals the part of A_F covered by p. We now generalize this to a sequence of prices: e.g., prices $r^1 > r^2$ cover a $F(r^1)r^1 + (F(r^2)-F(r^1))r^2$ part of A_F, i.e. the area of the union of two rectangles with opposite corners $(0,0)$ and $(r^i, F(r^i))$. No pricing scheme can cover more than A_F itself. However, as seen in Example 1, the area covered by a two-price sequence may be less than its revenue even for a submodular valuation.

Definition 6. *The fraction of* A_F covered *by a pricing scheme $\mathcal{P}$ with prices $p'_1 > \ldots > p'_k$ is $(1/\mathrm{A}_F) \sum_{d=1}^{k} p'_d(F(p'_d) - F(p'_{d-1}))$ where $F(p'_0) = 0$.*

We are now ready for Theorem 4's high-level proof, establishing that for HM valuations a pricing scheme $\mathcal{P}$'s revenue is at least the part of A_F covered by $\mathcal{P}$.

Proof (of Theorem 4). We proceed with one buyer; linearity of expectation will then yield the claim. Let $q_{x_1} \ge \ldots \ge q_{x_k}$ be Theorem 4's prices for H. Let set $S'_d \in \mathcal{D}^{S'_1,..,S'_{d-1}}(q_{x_1} \ldots q_{x_d})$ be bought in day d. By Theorem 3, $\sum_{\delta=1}^{d} |S'_\delta| \ge F(q_{x_d})$.

Via Lemma 8 below with $d_0 = 1, d = k, q^\delta = q_{x_\delta}$ and $x^\delta = F(q_{x_\delta})$, revenue is at least $\sum_{\delta=1}^{d} q_{x_\delta}(F(q_{x_\delta}) - F(q_{x_{\delta-1}}))$, i.e. the area covered by these prices. Theorem 5, our approximation's technical core, will yield the factor: it shows that the k random[12] prices cover well in expectation the area under the demand curve.

Theorem 5. *Consider k prices $q_{x_1} = \frac{H}{2^{x_1}} \ge \ldots \ge q_{x_k} = \frac{H}{2^{x_k}}$ where $x_1 \le \cdots \le x_k$ are the first (smallest), …, k-th (largest) order statistics of k iid continuous random variables $u_1, \ldots, u_k$ chosen uniformly at random in $[0, L]$. Then these prices cover in expectation an $\Omega(\frac{k}{\log m + \log n})$ fraction of A_F.*

Akhlaghpour et al. [1]'s recursive construction can be used to cover in expectation only an $\Omega(\frac{\log k}{\log m + \log n})$ fraction of A_F, much lower than our $\Omega(\frac{k}{\log m + \log n})$.

We proceed with Theorem 5's proof. Let $AC_1 = \mathbb{E}_{l \sim U[0,L]}[F(q_l)q_l]$ be the part of A_F covered by a price $q_l = \frac{H}{2^l}$ with l distributed uniformly on $0..L$. We know [2] $AC_1 = \int_0^L F(\frac{H}{2^x})\frac{H}{2^x}L^{-1}\mathrm{d}x \ge L^{-1}\sum_{x=1}^{L} F(\frac{H}{2^x})\frac{H}{2^x} \ge \frac{H}{4(1+\log m + \log n)}$ (note $F(H) = 0$). We recall the following facts (see e.g. [6, Chapter 2]) about order statistics.

[12] While prices of form $q_x = \frac{H}{2^x}$ also achieve the $\Theta(\log m + \log n)$ factor for $k = 1$ day [2], we can reason more easily about continuous, instead of integer, x's (scaling factors).

Lemma 7. *Let X be a* continuous *random variable in $[0, L]$ with cumulative distribution function F and probability density function (pdf) $f = F'$. Let $x_{d:k}$ be its d^{th} highest order statistic out of k independent trials. Then*

- *$x_{j:k}$'s pdf is $k\binom{k-1}{j-1}F(x)^{j-1}(1-F(x))^{k-j}f(x)$. Thus $x_d = x_{d:k}$'s ($F(x) = \frac{x}{L}, f(x) = \frac{1}{L}$) pdf is $k\binom{k-1}{d-1}x^{d-1}\frac{(L-x)^{k-d}}{L^k}$.*
- *$x_{d+1:k}$'s distribution* conditioned *on the next lowest $x_{d:k}$'s value x_d is the same as the distribution of the lowest order statistic $x_{k-d:k-d}$ out of $k-d$ trials of variable X truncated below x_d, i.e. with pdf $\frac{f(x)}{1-F(x_d)}$ for $x \in [x_d, L]$.*

Proof (of Theorem 5). The expected area covered (recall Def. 6) by all prices is

$$\begin{aligned} AC_k &= \mathbb{E}_{x_1 \dots x_k}[\sum_{d=1}^{k}(F(q_{x_d}) - F(q_{x_{d-1}}))q_{x_d}] = \mathbb{E}[\sum_{d=1}^{k}F(q_{x_d})q_{x_d}] - \mathbb{E}[\sum_{d=1}^{k-1}F(q_{x_d})q_{x_{d+1}}] \\ &= \mathbb{E}_{u_1 \dots u_k}[\sum_{d=1}^{k}F(q_{u_d})q_{u_d}] - \sum_{d=1}^{k-1}S_k^d \quad = kAC_1 - \sum_{d=1}^{k-1}S_k^d \end{aligned} \tag{4}$$

where Eq. (4) follows from the fact that the sets $\{x_1, \dots, x_k\}$ and $\{u_1, \dots, u_k\}$ coincide and we denoted $S_k^d = \mathbb{E}_{x_1 \dots x_k}[F(q_{x_d})q_{x_{d+1}}]$. We continue by upper bounding each S_k^d and then summing them up.

Using Lemma 7 (the first fact for Eqs. (5) and (6) and the second fact for Eq. (5)),

$$\mathbb{E}_{x_{d+1}|x_d}[q_{x_{d+1}}] = \int_{x_d}^{L}\tfrac{(k-d)(L-y)^{k-d-1}}{(L-x_d)^{k-d}}\tfrac{H}{2^y}\mathrm{d}y \le \int_{x_d}^{L}\tfrac{k-d}{L-x_d}\tfrac{H}{2^y}\mathrm{d}y \le 2\tfrac{k-d}{L-x_d}\tfrac{H}{2^{x_d}} \tag{5}$$

$$\text{Thus } S_k^d = \mathbb{E}_{x_1 \dots x_k}[F(q_{x_d})q_{x_{d+1}}] = \mathbb{E}_{x_d}[F(q_{x_d}) \cdot \mathbb{E}_{x_{d+1}|x_d}[q_{x_{d+1}}]]$$

$$\le \int_0^L k\tbinom{k-1}{d-1}x_d^{d-1}(L-x_d)^{k-d}L^{-k}F(\tfrac{H}{2^{x_d}})2\tfrac{k-d}{L-x_d}\tfrac{H}{2^{x_d}}\mathrm{d}x_d \tag{6}$$

$$\le 2k(k-1)\tbinom{k-2}{d-1}L^{-k}\int_0^L x_d^{d-1}(L-x_d)^{k-1-d}F(\tfrac{H}{2^{x_d}})\tfrac{H}{2^{x_d}}\mathrm{d}x_d \tag{7}$$

By summing up for all days $d = 1..k-1$, we get

$$\begin{aligned} AC_k &\ge kAC_1 - 2k(k-1)L^{-k}\sum_{d=1}^{k-1}\binom{k-2}{d-1}\int_0^L x_d^{d-1}(L-x_d)^{k-1-d}F(\frac{H}{2^{x_d}})\frac{H}{2^{x_d}}\mathrm{d}x_d \\ &\ge kAC_1 - 2k(k-1)L^{-k}\int_0^L F(\frac{H}{2^x})\frac{H}{2^x}\sum_{d=1}^{k-1}\binom{k-2}{d-1}x^{d-1}(L-x)^{(k-2)-(d-1)}\mathrm{d}x \\ &\ge kAC_1 - 2k(k-1)L^{-k}\int_0^L F(\frac{H}{2^x})\frac{H}{2^x}L^{k-2}\mathrm{d}x \\ &\ge kAC_1 - 2k(k-1)L^{-1}2AC_1 \ge k(1 - \frac{4(k-1)}{1+\log m + \log n})\frac{H}{4(1+\log m+\log n)} \end{aligned}$$

We conclude with a revenue bound given guarantees on total quantities bought.

Lemma 8. *If, at prices* $q^{d_0} > \ldots > q^d$*, at least* x^δ *items* ($x^d \geq x^{d-1} \geq \ldots \geq x^{d_0} \geq x^{d_0-1} = 0$) *are sold in total up to each day* $\delta = d_0..d$ *(e.g. at least* x^{d_0+1} *items in days* d_0 *and* d_0+1 *together) then the revenue is at least* $\sum_{\delta=d_0}^{d} q^\delta(x^\delta - x^{\delta-1})$*.*

Proof. The lowest revenue is for *exactly* x^δ items sold in day $\delta = d_0..d$ and for as few items as possible sold early, i.e. for $x^\delta - x^{\delta-1}$ items sold in day $\delta = d_0..d$.

In practice, buyers do not only have patience, but also have an influence on other buyers. We allow now a buyer's value (for any bundle) to increase depending on others' acquired bundles (but not on others' valuations). We will preserve the revenue approximation factor, despite an increased optimum revenue.

6 Positive Allocative Externalities

We now investigate revenue maximization in the presence of positive externalities, i.e. a buyer's valuation being increased by other buyers' ownership of certain items. Such influences can be subjective, e.g. resulting from peer pressure, or objective, e.g. resulting from ownership of a certain social network application. We define a new influence model via a predicate $\mathcal{I} : 1..m \rightarrow \{\texttt{false}, \texttt{true}\}$ such that $\mathcal{I}(i_0)$ only depends on seller's assignment of items to buyer i_0, e.g.

- $\mathcal{I}(i_0) = \texttt{true}$ iff buyer i_0 owns all (or, instead, at least two) items
- $\mathcal{I}(i_0) = \texttt{true}$ iff buyer i_0 owns his preferred bundle at current prices

Let $\mathcal{I}_d$ be the buyers i_0 satisfying $\mathcal{I}(i_0)$ before day d. $\mathcal{I}$ is *monotone* if $\mathcal{I}_d \subseteq \mathcal{I}_{d+1}$.

We model the valuation in day d of a buyer i as a linear mapping (depending on d only through its argument $\mathcal{I}_d \backslash \{i\}$) of i's base value

$$v_i^d(S|1..m\backslash\{i\}) = (a_i(\mathcal{I}_d\backslash\{i\})v_i(S)) \oplus b_i(\mathcal{I}_d\backslash\{i\}), \forall \text{ set } S \subseteq 1..n \quad (8)$$

where $\alpha v_i(S) \oplus \beta = \{\alpha v_i(S) \text{ if } S = \emptyset \text{ and}^{13}\ \alpha v_i(S) + \beta \text{ if } S \neq \emptyset\}$ for $\alpha, \beta \in \mathbb{R}$.

Thus, $a_i(I)$ and $b_i(I)$ measure the multiplicative and additive influences that a buyer set I (satisfying $\mathcal{I}$) have on buyer i. Say i's value for any DVD of a TV series doubles as soon as one other friend (in a set F_i) has the entire series (the predicate $\mathcal{I}$) and is then constant. Then $a_i(I) = 2 \Longleftrightarrow |I \cap F_i| \geq 1$ and $b_i(I) = 0$.

Without any influence, a valuation reduces to the base value: $a_i(\emptyset) = 1, b_i(\emptyset) = 0$. Assume a_i and b_i are non-negative, monotone and submodular[14]. Also assume that a_i, b_i, v_i are bounded: $\max_{I \subseteq 1..m\backslash\{i\}} a_i(I) = a_i(1..m\backslash\{i\}) = H^a$, $\max_I b_i(I) = b_i(1..m\backslash\{i\}) = H^b$, $\max_{S \subseteq 1..n} v_i(S) = H_i$ with $\max_{i \in 1..m} H_i = H$.

Our influence model is a distribution-free extension of single-item models [1,8]. It does not require or preclude symmetry, anonymity or a neighbor graph.

[13] Eq. (8) excludes the additive increase for $S = \emptyset$ so that $v_i^d(\emptyset|\cdot) = 0$. Also, if b_i is much larger than $a_i \max_S v_i(S)$ then a multiplicative revenue approximation is impossible: prices close to b_i are needed, rendering $\emptyset$ the preferred set, i.e. zero revenue.

[14] Submodularity (non-increasing marginal influence) is often assumed for externalities [1,8]. Positive, monotone externalities are an instance of "herd mentality".

For $a_i = 1, b_i = 0$ we recover the model before this section. Buyers are still forward-myopic and do not strategize about which items to buy today so that other buyers' values increase, thus increasing their own value etc.

With positive externalities, a natural revenue maximization approach [8] is providing certain items for free to some buyers and then charging others accordingly.

Definition 7. *The influence-and-exploit IE_k marketing strategy for $k \geq 2$ satisfies $\mathcal{I}$ (at no cost) for each buyer with probability* 0.5*, in day* 1*. Let A_1 be the set of buyers chosen in day* 1*. Independently of A_1, $k-1$ prices $q_{x_1} = \frac{H+H^b/H^a}{2^{x_1}} \geq \ldots \geq q_{x_{k-1}} = \frac{H+H^b/H^a}{2^{x_{k-1}}}$ where $x_1 \leq \cdots \leq x_{k-1}$ are the first (smallest), ..., (k–1)-th (largest) order statistics of $k-1$ iid continuous random variables $u_1, \ldots, u_{k-1}$ chosen uniformly at random in $[0, L]$. Each buyer $i \in 1..m \setminus A_1$ is offered uniform item price $H^a/3 \cdot q_{x_{d-1}}$ in day $d \geq 2$.*

Theorem 4's factor carries over, despite the affine increase in optimum revenue.

Theorem 6. *The IE_k strategy is an $O(\frac{\log m + \log n}{k})$-revenue approximation to the optimal marketing strategy for a monotone $\mathcal{I}$ over IE_k and HM base valuations.*

The price schedule $q_{x_1} \geq \ldots \geq q_{x_{k-1}}$ is (by Theorem 4) a $O(\frac{\log m + \log n}{k})$-revenue approximation given buyers' base valuations (translated by $\frac{H^b/3}{H^a/3}$). The proof establishes that the influence of other buyers (an affine mapping of a buyer's value in each day) does not result in fewer items being bought in the worst case.

7 Conclusions and Future Directions

In this paper we study prior-free revenue maximization with sequences of equal item prices. We are the first to consider combinatorial valuations for more than one item in unlimited supply in the sequential setting. We provide a sufficient condition and an algorithm improving the revenue approximation factor of an existing one-shot pricing scheme complemented by a lower bound that leverages the limited availability of price updates. We also initiate the study of revenue maximization for allocative externalities between combinatorial valuations. Several open directions appear promising to us.

The hereditary maximizers property guarantees consistency of bundles bought sequentially. We deem it of interest to find an alternative assumption, perhaps related to sequential revenue instead, that still allows revenue bounds. We assume fully patient, as opposed to instantaneous, buyers. Other patience models, e.g. arrival-departure intervals [3], may yield alternative approximations. Finally, widespread externalities in applications present many exciting open questions, both practical and theoretical, notably in multiple-item settings.

Acknowledgments. We thank Avrim Blum and Malvika Rao for detailed comments on earlier drafts of this paper, Mark Braverman for helpful discussions and Daniel Lehmann for providing us with a copy of [4].

References

1. Akhlaghpour, H., Ghodsi, M., Haghpanah, N., Mahini, H., Mirrokni, V., Nikzad, A.: Optimal iterative pricing over social networks. In: Fifth Workshop on Ad Auctions (2009)
2. Balcan, M.-F., Blum, A., Mansour, Y.: Item pricing for revenue maximization. In: Proc. ACM Conf. on Electronic Commerce, pp. 50–59 (2008)
3. Bansal, N., Chen, N., Cherniavsky, N., Rudra, A., Schieber, B., Sviridenko, M.: Dynamic pricing for impatient bidders. ACM Trans. Algorithms 6(2), 1–21 (2010)
4. Bertelsen, A.: Substitutes valuations and $M^{\natural}$-concavity. Master's thesis, Hebrew University of Jerusalem (2004)
5. Bikhchandani, S., Ostroy, J.M.: Ascending price Vickrey auctions. Games and Economic Behavior 55(2), 215–241 (2006)
6. David, H., Nagaraja, H.: Order Statistics. Wiley, Chichester (2003)
7. Gul, F., Stacchetti, E.: Walrasian equilibrium with gross substitutes. Journal of Economic Theory 87(1), 95–124 (1999)
8. Hartline, J., Mirrokni, V., Sundararajan, M.: Optimal marketing over social networks. In: Conference on the World Wide Web, WWW 2008 (2008)
9. Kleinberg, J.: Cascading behavior in networks: algorithmic and economic issues. Cambridge University Press, Cambridge (2007)
10. Lien, Y., Yan, J.: On the gross substitutes condition. Working paper (July 2007)

The Cost of Moral Hazard and Limited Liability in the Principal-Agent Problem

Felipe Balmaceda[1], Santiago R. Balseiro[2],
Jose R. Correa[1], and Nicolas E. Stier-Moses[2]

[1] Departamento de Ingeniería Industrial, Universidad de Chile, Santiago, Chile
{fbalmace,jcorrea}@dii.uchile.cl
[2] Graduate School of Business, Columbia University, New York, USA
{sbalseiro13,stier}@gsb.columbia.edu

Abstract. In the classical principal-agent problem, a principal hires an agent to perform a task. The principal cares about the task's output but has no control over it. The agent can perform the task at different effort intensities, and that choice affects the task's output. To provide an incentive to the agent to work hard and since his effort intensity cannot be observed, the principal ties the agent's compensation to the task's output. If both the principal and the agent are risk-neutral and no further constraints are imposed, it is well-known that the outcome of the game maximizes social welfare. In this paper we quantify the potential social-welfare loss due to the existence of limited liability, which takes the form of a minimum wage constraint. To do so we rely on the worst-case welfare loss—commonly referred to as the Price of Anarchy—which quantifies the (in)efficiency of a system when its players act selfishly (i.e., they play a Nash equilibrium) versus choosing a socially-optimal solution. Our main result establishes that under the monotone likelihood-ratio property and limited liability constraints, the worst-case welfare loss in the principal-agent model is exactly equal to the number of efforts available.

1 Introduction

In this paper we analyze the classical principal-agent problem as put forward by Grossman and Hart [4]. The problem entails the following contracting situation: a principal hires an agent to perform a task. The principal cares about the task's output but cannot control it directly. Instead, the output is influenced by the agent's choice of effort intensity. The principal would like to induce the agent to choose the (in his view) optimal effort intensity but since the agent incurs a cost when making effort, the principal has to compensate the agent. Because the principal cannot observe the effort intensity chosen by the agent—this is the prevailing assumption in this type of models and leads to *moral hazard*—the principal can only tie the agent's compensation to the task's output, used as a proxy of effort. This compensation scheme entails a loss since the task's output is a random variable whose distribution depends on the effort chosen by the agent. Hence, the output is not completely determined by the agent's effort intensity. If the two were perfectly correlated, the principal could infer the effort by observing the outcome.

This class of principal-agent problems has been the workhorse to understand many interesting economic phenomena such as, to name a few, the theory of insurance under

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 63–74, 2010.

moral hazard [12], the theory of managerial firms [1, 8], optimal sharecropping contracts between landowners and tenants [13], the efficiency wage theory [11], financial contracting [6], and job design and multi-tasking [5].

When both the principal and the agent are risk-neutral, the provision of a limited liability clause that restricts the exposure of the agent gives rise to an agency problem. If the principal wants to provide an incentive to the agent to work hard, he has to compensate the agent better when the realization of the task's output suggests that the effort intensity chosen by the agent was high. This imposes a gap between the marginal cost of the effort intensity experienced by the principal and the social marginal cost. Thus, the equilibrium contract will not maximize social welfare, meaning that a first-best outcome cannot be attained; instead, the constrained contract will be second-best.

In order to quantify the maximum social-welfare loss due to the existence of moral hazard and limited liability in a principal-agent setting, we rely on the concept of worst-case welfare loss, which quantifies the efficiency of a system when its players act selfishly (i.e., they play a Nash equilibrium) versus choosing a socially-optimal solution. The idea of using worst-case analysis to study non-cooperative games was introduced by Koutsoupias and Papadimitriou [7], and it is commonly referred to as the *Price of Anarchy* [9]. In our setting, the worst-case welfare loss is defined as the largest possible ratio between the social welfare of a socially-optimal solution—the sum of the principal's and agent's payoffs when the first-best effort intensity is chosen—and that of the sub-game perfect equilibrium. The worst ratio is with respect to the parameters that define an instance of the problem.

In the principal-agent setting, Babaioff, Feldman, and Nisan [2, 3] introduced a combinatorial agency problem with multiple agents performing two-effort-two-outcome tasks. The authors studied the combinatorial structure of dependencies between agents' actions, and analyzed the worst-case welfare loss for a number of different classes of action dependencies. Our model, instead, deals with a single agent and its complexity lies in handling more sophisticated tasks, rather than the interaction between agents. The goal of this article is to evaluate the worst-case welfare loss with respect to the outcome vector, the vector of agent's costs of effort, and the probability distribution of outcomes for each level of effort. The main result, shown in Theorem 1, establishes that under the monotone likelihood-ratio property and when the principal and an agent protected by limited liability are risk-neutral, the worst-case welfare loss is exactly equal to the number of efforts available. In other words, for any instance of the problem the worst-case welfare loss cannot exceed the number of efforts available and there are instances where that loss is achieved.

Our result suggests that the worst equilibrium that may arise in the finite principal-agent problem with limited liability for the agent depends on the complexity of the delegated task, as measured by the number of available efforts. When the delegated task requires the choice between two different effort intensities (e.g., shirk or work) the worst-case welfare loss is 2, while when the delegated task demands the choice of one effort intensity among E possibilities, the worst-case welfare loss is E. Thus, the worst-case welfare loss increases with the complexity of the delegated task. Our result suggests that the principal-agent paradigm that studies the consequences of moral hazard for the efficiency of contracting and organizational design is sound. The potential consequence of not dealing with a moral-hazard problem may have a non-negligible impact in the welfare of the system. For another interpretation, our results also quantify the impact of limited-liability in the utility of the principal, which is a way of measuring the inefficiency introduced by protecting the agent from carrying all the burden of the risk in the task's output.

Because the complexity of a principal-agent relationship is usually related to the number of tasks or projects rather than to the number of efforts or actions, we also study the worst-case welfare loss in an extension where there are multiple tasks. Here, the agent has to choose between working and shirking in each of several independent tasks. Surprisingly, we find that the worst-case welfare loss again equals 2, the number of efforts in each task, independently of how many tasks the agent has to work on. This confirms that, in terms of the potential welfare loss, the complexity of an agency relationship is better captured by the number of actions or efforts available rather than the number of tasks. Furthermore, it suggests that the incentive problem created by moral hazard is a natural source of economies of scope; that is, it is better to have one agent working in several different tasks than several agents working in one task each.

Most of our results arise from a characterization of the optimal wages that we provide. Working with the geometry of both the primal and the dual linear programs, we uncover the structure of the 'important' efforts, which we call *relevant*, and use them to bound the welfare of the solution to the principal-agent model with that arising when the agent chooses the socially-optimal effort.

The rest of the paper is organized as follows. In Sect. 2, we introduce the model with its main assumptions. Section 3 presents the main technical results. We start with the study of the two-effort-two-outcome case for an illustration of our techniques, continue with the general case, and present an example that shows that the lower bound is attained. We conclude with extensions in several directions in Sect. 4. For the missing proofs and details on the extensions, we refer the reader to the full version of the paper.

2 The Principal-Agent Model

In this section we describe the basic principal-agent model with $E \geq 2$ effort levels and $S \geq 2$ outcomes [4]. (Later on, in Sect. 4, we relax some of the assumptions presented below.) The agent chooses an effort $e \in \mathcal{E} \triangleq \{1, \ldots, E\}$, incurring a personal nonnegative cost of c_e. Efforts are sorted in increasing order with respect to costs; that is, $c_e \leq c_f$ if and only if $e \leq f$. Thus, a higher effort demands more work from the agent. The task's outcome depends on a random state of nature $s \in \mathcal{S} \triangleq \{1, \ldots, S\}$ whose distribution in turn depends on the effort level chosen by the agent. Each state has an associated nonnegative dollar amount that represents the principal's revenue. We denote the vector of outcomes indexed by state by $y = \{y^1, \ldots, y^S\}$. Without loss of generality, the outcomes are sorted in increasing order: $y^s \leq y^t$ if and only if $s \leq t$; hence, the principal's revenues are higher under states with a larger index. Finally, we let π_e^s be the common-knowledge probability of state $s \in \mathcal{S}$ when the agent chooses effort $e \in \mathcal{E}$. The probability mass function of the outcome under effort e is given by $\pi_e = \left\{\pi_e^1, \ldots, \pi_e^S\right\}$.

The principal can contract wages to the agent that depend on the outcome y but cannot observe the agent's chosen effort e. Indeed, the principal offers a take-it-or-leave-it contract to the agent that specifies a state-dependent wage schedule $w = \{w^1, \ldots, w^S\}$. The agent decides whether to accept or reject the offer, and if accepted, then he chooses an effort level before learning the realized state. The rational agent should accept the contract if the *individual rationality* (IR) and *limited liability* (LL) constraints are satisfied. The former specifies that the contract must yield an expected utility to the agent greater than or equal to that of choosing the *outside option*. The latter specifies that the wage must be nonnegative in every state occurring with positive probability. After accepting a contract specifying a wage schedule w, the risk-neutral agent has to choose an effort $e \in \mathcal{E}$. He does so by maximizing the expected payoff, which is given by $\pi_e w - c_e$, the difference between the expected wage and the cost incurred in the effort chosen.

Putting it all together, the principal's problem consists on choosing a wage schedule w and an effort intensity e for the agent that solve the following problem:

$$u^P \triangleq \max_{e \in \mathcal{E}, w} \pi_e(y - w) \tag{1}$$

$$\text{s.t. } \pi_e w - c_e \geq 0 \qquad \text{(IR)} \tag{2}$$

$$e \in \arg\max_{f \in \mathcal{E}} \{\pi_f w - c_f\} \qquad \text{(IC)} \tag{3}$$

$$w \geq 0\,. \qquad \text{(LL)} \tag{4}$$

The objective measures the difference between the principal's expected revenue and payment, hence computing his expected profit. Constraints (IR) and (LL) were described earlier. The *incentive compatibility* (IC) constraints guarantee that the agent will choose the principal's desired effort since he does not find it profitable to deviate from e.

Equivalently, one can formulate the principal's problem as $u^P = \max_{e \in \mathcal{E}} \{\pi_e y - z_e\} = \max_{e \in \mathcal{E}} \{u_e^P\}$. Here, we have defined z_e to be the minimum expected payment incurred by the principal so the agent accepts the contract and picks effort e. In addition, we denote by $u_e^P \triangleq \pi_e y - z_e$ the principal's maximum expected utility when effort e is implemented, and by $\mathcal{E}^P$ the set of optimal efforts for the principal, $\mathcal{E}^P \triangleq \arg\max_{e \in \mathcal{E}} \{u_e^P\}$. Exploiting that the set of efforts is finite, we can write the IC constraint (3) explicitly to obtain the *minimum payment linear program* corresponding to effort e, which we denote by MPLP(e):

$$z_e = \min_{w \in \mathbb{R}^S} \pi_e w \tag{5}$$

$$\text{s.t. } \pi_e w - c_e \geq 0 \tag{6}$$

$$\pi_e w - c_e \geq \pi_f w - c_f \quad \forall f \in \mathcal{E} \setminus e \tag{7}$$

$$w \geq 0\,. \tag{8}$$

Notice that this problem is independent of the output y.

We say that the principal *implements* effort $e \in \mathcal{E}$ when the wage schedule w is consistent with the agent choosing effort e. For a fixed effort e, (2), (3), and (4) characterize the polyhedron of feasible wages that implement e. The principal will choose a wage belonging to that set that achieves z_e by minimizing the expected payment $\pi_e w$. We are only interested in efforts that are attainable under some wage schedule, which we refer to as *feasible efforts*. An effort is feasible if the polyhedron corresponding to it is nonempty.

2.1 The Monotone Likelihood-Ratio Property

We make the assumption that the probability distributions π_e satisfy the well-known *monotone likelihood-ratio property* (MLRP). That is, $\{\pi_e\}_{e\in\mathcal{E}}$ verifies $\pi_e^s/\pi_f^s \geq \pi_e^t/\pi_f^t$ for all states $s < t$ and efforts $e < f$. The assumption of MLRP is pervasive in the literature of economics of information, and in particular in the principal-agent literature. The intuition behind it is that the higher the observed level of output, the more likely it is to come from a distribution associated with a higher effort level.

An important property of MLRP is that distributions that satisfy it also satisfy *first order stochastic dominance* (FOSD). For instance, [10] proved that $\sum_{s'=1}^{s} \pi_e^{s'} \geq \sum_{s'=1}^{s} \pi_f^{s'}$ for all states s and efforts $e < f$. A simple consequence of this that plays an important role in our derivations is that probabilities for the highest outcome S are sorted in increasing order with respect to efforts; i.e., $\pi_e^S \leq \pi_f^S$ for $e \leq f$. Note that in the case of two outcomes, MLRP and FOSD are equivalent.

2.2 Worst-Case Welfare Loss

The goal of a social planner is to choose the effort level e that maximizes the social welfare, defined as $u_e^{SW} \triangleq \pi_e y - c_e$, the sum of the welfare of the principal and the agent. The social planner is not concerned about wages, since risk neutrality ensures that wages are a pure transfer of wealth between the principal and the agent. Thus, the optimal social welfare is given by $u^{SO} \triangleq \max_{e\in\mathcal{E}}\{u_e^{SW}\}$. We denote the set of first-best efficient efforts by $\mathcal{E}^{SO} \triangleq \arg\max_{e\in\mathcal{E}}\{u_e^{SW}\}$. For analytical tractability, we will assume that the harder the agent works, the higher the social welfare in the system. In the two-outcome case, this assumption can be relaxed. In the general case, we believe that our results continue to hold without it.

Assumption 1. *The sequence of prevailing social welfare under increasing efforts is non-decreasing; i.e.,* $u_e^{SW} \leq u_f^{SW}$ *for all efforts* $e \leq f$.

For a given instance of the problem, we quantify the inefficiency of an effort e using the ratio of the social welfare under the socially-optimal effort to that under e. The main goal of the paper is to compute the worst-case welfare loss for arbitrary instances of the problem. This is defined as the smallest upper bound on the efficiency of a second-best optimal effort, which is commonly referred to as the *Price of Anarchy*[1] [9]. Therefore, the worst-case welfare loss, denoted by ρ, is defined as

$$\rho = \sup_{\pi,y,c} \frac{u^{SO}}{\min_{e\in\mathcal{E}^P} u_e^{SW}}, \tag{9}$$

[1] Actually, the price of anarchy for a maximization problem such as the one we work with in this article is often defined as the inverse of the ratio in (9). We do it in this way so ratios and welfare losses point in the same direction.

where the supremum is taken over all valid instances as described at the beginning of this section. Of course, the previous ratio for an arbitrary instance of the problem is at least one because the social welfare of an optimal solution cannot be smaller than that of an equilibrium, guaranteeing that $\rho \geq 1$. Next, we state the main result of our article that shows that under MLRP the worst-case welfare loss is bounded above by the number of efforts, and that this bound is tight.

Theorem 1. *Suppose that MLRP holds. Then, in the risk-neutral principal-agent problem with limited liability, the worst-case welfare loss ρ is exactly E.*

2.3 Preliminaries

In this section, we consider the principal's problem and reformulate it in a way that is more amenable to understand its properties, which will be useful to prove our worst-case bounds. The dual of MPLP(e), displayed in (5)-(8), is given by

$$\max_{p\in\mathbb{R}^E} \sum_{f\neq e}(c_f - c_e)p_f - c_e p_e \tag{10}$$

$$\text{s.t.} \sum_{f\neq e}(\pi_f^s - \pi_e^s)p_f - \pi_e^s p_e \leq \pi_e^s \quad \forall s \in \mathcal{S}, \tag{11}$$

$$p \leq 0\,.$$

Here, p_e is the dual variable for the IR constraint (6), while p_f is the dual variable for the IC constraint (7) for effort $f \neq e$. Notice that the null vector **0** is dual-feasible, and hence the dual problem is always feasible. Furthermore, since we only consider feasible efforts the primal is also feasible and by strong duality we have that the solution to the dual program is z_e. Notice that summing constraints (11) over $s \in \mathcal{S}$ and using that $\sum_{s\in\mathcal{S}} \pi_f^s = 1$ for all $f \in \mathcal{E}$, we get that $p_e \geq -1$. We now state some useful results.

Lemma 1. *The social welfare is at least the principal's utility; i.e., $u_e^{SW} \geq u_e^P$ for all efforts $e \in \mathcal{E}$.*

Proof. Notice that since z_e solves MPLP(e), we have that $z_e \geq c_e$ for all $e \in \mathcal{E}$. Thus, $\pi_e y - z_e \leq \pi_e y - c_e$. □

The next result stresses the importance of the agent's limited liability in the model. It is a well-known result that we state for the sake of completeness. Without the LL constraint (4), it is optimal for the principal to implement the socially-optimal effort and he captures the full social surplus, leaving no utility to the agent. As a consequence, the worst-case welfare loss is 1 meaning that, albeit unfair to the agent, the contract is efficient.

Lemma 2. *If the principal and the agent are risk-neutral and there is no limited liability constraint, the minimum expected payment z_e incurred by the principal when inducing a feasible effort e is c_e, that is, $c_e = \min_{w\in\mathbb{R}^S}\{\pi_e w$ s.t. (6), (7)$\}$.*

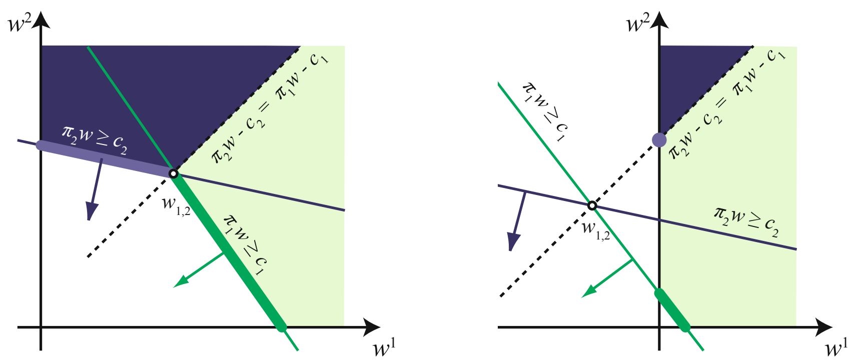
Proof. Since the effort e is feasible there exists a vector w satisfying (6) and (7). Assume for a contradiction that (6) is not tight and consider $w' = w - \mathbf{1}\varepsilon$, where $\mathbf{1}$ is the all-ones vector. Clearly w' still satisfies (7) so we can select ε so that the objective function is smaller and (6) is still feasible. □

3 Bounding the Welfare Loss

3.1 The Case of Two Efforts and Two Outcomes

In this section we look at the case with 2 efforts (such as shirk and work) and 2 states (such as fail and success), and show that the worst-case welfare loss is at most 2. This simple case is a useful exercise to gain intuition and improve the understanding of the general case. First, we provide a geometric characterization of the minimum-cost wage schedule implementing a given effort level, and compute the associated expected payments. Then, we proceed to bound the worst-case welfare loss.

Consider MPLP(2), corresponding to the agent working hard. The feasible set of wages is defined by the IR, IC and LL constraints. The IC constraint (7) ensures that the agent prefers effort 2 over 1, which can also be written as $w^2 - w^1 \geq (c_2 - c_1)/(\pi_2^2 - \pi_1^2)$. Notice that both the numerator and denominator are nonnegative. Hence, the boundary of this constraint is given by a $45°$ line, as shown by Fig. 1 which plots the feasible regions for the two efforts. The IC constraint for $e = 1$ is the same with the inequality reversed. An implication of FOSD is that the IR constraint for effort 1 is steeper than that for effort 2.



(a) $w_{1,2}$ in the first quadrant. (b) $w_{1,2}$ in the second quadrant.

Fig. 1. Feasible regions of MPLP(e) for $e \in \{1, 2\}$ (light and dark shade, respectively), according to the location of $w_{1,2}$. Optimal solutions are denoted with a bold point or segment, depending on whether they are unique or not. Arrows indicate the negative gradient of the objective function.

It will be useful to introduce the point $w_{1,2}$, defined as the intersection point between the IC constraint and the IR constraints for both efforts. This point is given by

$$w_{1,2} = \left(\frac{c_1\pi_2^2 - c_2\pi_1^2}{\pi_2^2 - \pi_1^2}, \frac{c_1\pi_2^2 - c_2\pi_1^2}{\pi_2^2 - \pi_1^2} + \frac{c_2 - c_1}{\pi_2^2 - \pi_1^2} \right).$$

The second component of this vector is nonnegative and larger than the first component because $c_2 \geq c_1$, $\pi_2^2 \geq \pi_1^2$, and $\pi_1^1 \geq \pi_2^1$.

If $w_{1,2}$ lies in the first quadrant, as in Fig. 1a, the situation is very similar to the case without liability constraints discussed earlier. Indeed, the wages $w_{1,2}$ are optimal because they satisfy all constraints and minimize the objective of MPLP. This implies that the optimal expected payment is equal to the effort's cost, and because of Assumption 1 the principal chooses $e = 2$ leaving the agent with zero surplus. The case of greater interest is when $w_{1,2}$ lies in the second quadrant, as in Fig. 1b. This occurs either when the cost of working hard is too high, or the probability of a good outcome when working hard is too low. In this case, the incentive compatible wage schedule that induces participation at the lowest cost for the principal does not satisfy the limited liability constraint. Thus, the optimal solution, attained at the intersection of the IC constraint and the vertical axis, is $w_2 = \big(0, (c_2 - c_1)/(\pi_2^2 - \pi_1^2)\big)$. The minimum expected payment for effort 2 is $z_2 = \pi_2^2(c_2 - c_1)/(\pi_2^2 - \pi_1^2)$, which is strictly larger than c_2 because the IR constraint is not binding, leaving the agent with a positive rent. The analysis for effort 1 is simpler. Under the assumption of nonnegative costs, any point that is nonnegative and for which the IR constraint is binding is optimal and attains the value c_1. Thus, the minimum expected payment equals the effort's cost, and the agent obtains zero surplus.[2]

The previous analysis will enable us to bound the worst-case welfare loss. Under Assumption 1, effort 2 is socially-optimal: $u^{SO} = u_2^{SW} \geq u_1^{SW}$. If the second-best optimal effort is 2, the worst-case welfare loss is 1. So we consider that it is second-best optimal to induce effort 1; i.e, $u_1^P \geq u_2^P$. Since the principal prefers effort 1, it must be that $z_2 > c_2$. Hence, $w_{1,2}$ must lie in the second quadrant, and $z_2 = (c_2 - c_1)\pi_2^2/(\pi_2^2 - \pi_1^2)$. Then, we have that

$$u_1^{SW} \geq u_1^P \geq u_2^P = \pi_2 y - z_2 = u_2^{SW} + c_2 - \pi_2^2 \frac{c_2 - c_1}{\pi_2^2 - \pi_1^2} = u_2^{SW} + c_1 - \pi_1^2 \frac{c_2 - c_1}{\pi_2^2 - \pi_1^2}$$

$$\geq u_2^{SW} + c_1 - \pi_1^2 \frac{(\pi_2 - \pi_1)y}{\pi_2^2 - \pi_1^2} \geq u_2^{SW} + c_1 - \pi_1 y = u_2^{SW} - u_1^{SW}, \quad (12)$$

where the inequalities follow, respectively, from Lemma 1, the principal's choice of $e = 1$, Assumption 1, and FOSD. Reshuffling terms, we have that $u_2^{SW} \leq 2u_1^{SW}$ from where the optimal social welfare cannot be better than twice the social welfare under the effort chosen by the principal. We conclude that the worst-case welfare loss is at most the number of efforts.

3.2 The General Case

We now consider the general case of an arbitrary finite number of efforts and outcomes. Here, we need to study the primal and the dual of the MPLP simultaneously. As in the previous case, we first attempt to characterize the minimum expected payments for each effort level, and then prove that the worst-case welfare loss is bounded by E.

We saw earlier that in the case of 2 efforts both of them play a role in the worst-case bound. However, in the general case only some efforts will be *relevant*. There are some other efforts, referred to as *dominated*, that although feasible will not participate

[2] This might not be the case if the limited liability constraint requires $w^2 \geq \ell$, where ℓ is large. This is discussed in the full version of the paper.

in the analysis. Relevant efforts are always preferred to dominated efforts and thus the principal will choose just from among them. This is equivalent to discarding dominated efforts from any instance and does not affect the utilities of other efforts and the efficiency metric.

In Theorem 2, we characterize the relevant efforts. We do this by observing that effort E is always relevant. From this first relevant effort, we obtain a sequence inductively observing that for any relevant effort, in the optimal solution to MPLP only the IC constraint of another relevant effort is binding. Afterwards, we prove that when a dominated effort is chosen, the principal's utility is always dominated by that of a relevant effort (hence the name 'relevant'). As before, we define the wage vector $w_{e,f}$ as the intersection of IC constraints (7) for efforts e and f with the S axis. Hence, $w_{e,f} = \big(0, \dots, 0, (c_e - c_f)/(\pi_e^S - \pi_f^S)\big)$, which is a nonnegative vector.

Theorem 2. *There exists a subsequence of relevant efforts, denoted by $\mathcal{R} = \{e_r\}_{r=1}^R \subseteq \mathcal{E}$ with $e_R = E$, such that the minimum expected payments for the principal are*

$$z_{e_1} = c_{e_1}, \quad \textit{and} \quad z_{e_r} = \pi_{e_r}^S \frac{c_{e_r} - c_{e_{r-1}}}{\pi_{e_r}^S - \pi_{e_{r-1}}^S} \geq c_{e_r} \quad \textit{for } r = 2, \dots, R.$$

Moreover, the optimal wage w_{e_r} corresponding to effort e_r is $w_{e_r, e_{r-1}}$ if $r > 1$ and $(0, \dots, 0, c_{e_1}/\pi_{e_1}^S)$ if $r = 1$.

For a dominated effort $f \notin \mathcal{R}$, let $r(f) \triangleq \min\{e \in \mathcal{R} : e > f\}$ be the smallest relevant effort greater than f. The next corollary shows that relevant efforts are sorted with respect to $z_e - c_e$ and that dominated efforts violate this order.

Corollary 1. *Relevant efforts are sorted in non-decreasing order with respect to $z_e - c_e$; that is, $z_{e_r} - c_{e_r} \leq z_{e_{r+1}} - c_{e_{r+1}}$ for all $1 \leq r < R$. Moreover, $z_f - c_f \geq z_{r(f)} - c_{r(f)}$ for any dominated effort $f \notin \mathcal{R}$.*

Proof. For the first claim observe that $z_{e_r} - c_{e_r} = \pi_{e_r} w_{e_r} - c_{e_r} \leq \pi_{e_r} w_{e_{r+1}} - c_{e_r} = \pi_{e_{r+1}} w_{e_{r+1}} - c_{e_{r+1}} = z_{e_{r+1}} - c_{e_{r+1}}$, where the inequality follows from the fact that $w_{e_{r+1}}$ is feasible for MPLP(e_r) and that w_{e_r} is the optimal solution. The second equality holds because the IC constraint between efforts e_r and e_{r+1} is binding at $w_{e_{r+1}}$.

For the second claim, let f be a dominated effort. If $f < e_{r_1}$, the result is trivial because $z_{e_{r_1}} - c_{e_{r_1}} = 0$. So, suppose that $e_r < f < e_{r+1}$. Using the dual of MPLP, as done previously, it is easy to observe that $p = -\mathbb{I}_{e_r} \pi_f^S / (\pi_f^S - \pi_{e_r}^S)$ is dual feasible for effort f, and its objective value is $(c_f - c_{e_r}) \pi_f^S / (\pi_f^S - \pi_{e_r}^S) = \pi_f^S w_{e_r,f}^S$, which by weak duality is a lower bound on z_f. Hence, $z_f \geq \pi_f^S w_{e_r,f}^S = \pi_{e_r}^S w_{e_r,f}^S + w_{e_{r+1},f}^S (\pi_f^S - \pi_{e_{r+1}}^S) + w_{e_{r+1}}^S (\pi_{e_{r+1}}^S - \pi_{e_r}^S)$. Rearranging the terms, the last expression equals $z_{e_{r+1}} + c_f - c_{e_{r+1}} + \pi_{e_r}^S (w_{e_r,f}^S - w_{e_{r+1}}^S) \geq z_{e_{r+1}} + c_f - c_{e_{r+1}}$, where the inequality follows because $w_{e_r,f} \geq w_{e_{r+1}}$. Indeed,

$$w_{e_r,f}^S = \frac{c_f - c_{e_{r+1}}}{\pi_f^S - \pi_{e_r}^S} + \frac{c_{e_{r+1}} - c_{e_r}}{\pi_f^S - \pi_{e_r}^S} = w_{e_{r+1},f}^S \frac{\pi_f^S - \pi_{e_{r+1}}^S}{\pi_f^S - \pi_{e_r}^S} + w_{e_{r+1}}^S \frac{\pi_{e_{r+1}}^S - \pi_{e_r}^S}{\pi_f^S - \pi_{e_r}^S} \geq w_{e_{r+1}}^S,$$

because $w_{e_{r+1},f} \leq w_{e_{r+1}}$ (this follows from Theorem 2) and $\pi_f^S - \pi_{e_{r+1}}^S \leq 0$. □

Relevance is central to the analysis of the principal-agent problem. Under Assumption 1, a social planner chooses effort E, a relevant effort, to maximize the social welfare. Furthermore, as a consequence of Corollary 1, there is always a relevant effort that is optimal for the principal.

Proposition 1. *There is always a relevant effort that is optimal for the principal; i.e.,* $\mathcal{E}^P \cap \mathcal{R} \neq \emptyset$.

Proof. We prove this claim by contradiction by supposing that no relevant effort is optimal for the principal. Let f be an optimal dominated effort, and consider the first next relevant effort $r(f)$. Using Corollary 1,

$$0 < u^P_f - u^P_{r(f)} = (\pi_f - \pi_{r(f)})y + z_{r(f)} - z_f \leq (\pi_f - \pi_{r(f)})y + c_{r(f)} - c_f = u^{SW}_f - u^{SW}_{r(f)},$$

which is a contradiction because Assumption 1 implies that f cannot have a larger social welfare than $r(f)$. □

Notice that the previous proposition together with Theorem 2 imply that the equilibrium of the principal-agent problem can be computed in $O(E^2 + ES)$ time, instead of solving E linear programs. The quadratic term comes from finding the relevant efforts while the second term comes from evaluating the principal's utilities for all relevant efforts.

We are now in position to prove the main result.

Theorem 3. *Assume that MLRP and Assumption 1 hold. The worst-case welfare loss for the risk-neutral principal-agent problem with limited liability is at most* E.

Proof. Under Assumption 1, it is optimal for the system that the agent chooses effort E, so $u^{SO} = u^{SW}_E$. Furthermore, by Proposition 1 the optimal strategy for the principal is to implement a relevant effort $e \in \mathcal{R}$. Note that if we remove all efforts lower than e, a consequence of Theorem 2 is that u^P_f does not change for any effort $f > e$ and u^P_e may only increase. This is because after removing the lower efforts, z_e is reduced to c_e if they were not already equal. Notice also that a dominated effort cannot become relevant after removing the efforts lower than e. Therefore, this new instance has the same the worst-case welfare loss. Thus, we do not lose any generality if we consider that it is optimal for the principal to implement effort 1; i.e., $u^P_1 \geq u^P_e$ for all $e \in \mathcal{E}$.

To lower bound the total welfare of the lowest effort, u^{SW}_1, we proceed as in (12), working exclusively with relevant efforts. To simplify notation, in the remainder of this proof we drop the r subscript and assume that all efforts are relevant. Lemma 1 and Theorem 2 imply that for any effort $e > 1$,

$$u^{SW}_1 \geq u^P_1 \geq u^P_e = \pi_e y - z_e = u^{SW}_e + c_e - \pi^S_e \frac{c_e - c_{e-1}}{\pi^S_e - \pi^S_{e-1}} = u^{SW}_e + c_{e-1} - \pi^S_{e-1} \frac{c_e - c_{e-1}}{\pi^S_e - \pi^S_{e-1}}.$$

Since $u^{SW}_e \geq u^{SW}_{e-1}$ implies that $c_e - c_{e-1} \leq \pi_e y - \pi_{e-1} y$, the last expression is bounded by

$$u^{SW}_e + c_{e-1} - \frac{\pi^S_{e-1}}{\pi^S_e - \pi^S_{e-1}}(\pi_e - \pi_{e-1})y \geq u^{SW}_e + c_{e-1} - \pi_{e-1} y = u^{SW}_e - u^{SW}_{e-1}, \quad (13)$$

where the inequality in (13) follows from MLRP because $\pi_{e-1}\pi^S_e \geq \pi_e \pi^S_{e-1}$. Summing over $e > 1$ and rearranging terms we conclude that $E u^{SW}_1 \geq u^{SW}_E$. □

This result shows that when the agent is covered against unfair situations in which he has to pay money to the principal even after having invested the effort, the fact that the principal induces the agent to implement the effort of his choice instead of a socially-optimal one is costly for the system. Indeed, the welfare loss due to limited liability and the impossibility of observing the effort exerted by the agent is bounded by the number of efforts. If we are willing to accept *the number of efforts* as a metric of the complexity of a principal-agent relationship, then the cost of coordination in the system is bigger for more complex relationships.

3.3 A Tight Instance

To wrap-up this section we construct a family of instances with 2 outcomes and E efforts whose worst-case welfare loss is arbitrarily close to the bound of E.

Fixing $0 < \varepsilon < 1$, we let the probabilities of the outcomes associated to each effort be $\pi_e = \left(1 - \varepsilon^{E-e}, \varepsilon^{E-e}\right)$ for $e \in \mathcal{E}$. Clearly, these distributions verify that $\pi_1^2 \leq \ldots \leq \pi_E^2$, and thus they satisfy MLRP. (Recall that in the case of two outcomes MLRP and FOSD are equivalent.)

Furthermore, we let $c_E = \varepsilon^{-E}$, and then set the remaining efforts so that $z_e - c_e = e - 1$ for all $e \in \mathcal{E}$. Since $z_e = (c_e - c_{e-1})\pi_e^S/(\pi_e^S - \pi_{e-1}^S)$, we obtain $c_{e-1} = c_e\varepsilon - (e-1)(1-\varepsilon)$ for $e = 2, \ldots, E$. Notice that this implies that $w_{e+1}^2 - w_e^2 = 1/\varepsilon^{E-e}$, where $w_e = \left(0, (c_e - c_{e-1})/(\pi_e^2 - \pi_{e-1}^2)\right)$ is the optimal solution to MPLP(e). Finally, let the output be $y = (0, w_E^2 + 1)$. One can prove inductively that the social utility is $u_e^{SW} = e + \sum_{i=1}^{E-e} \varepsilon^i$, and that principal's utility is $u_e^P = \sum_{i=0}^{E-e} \varepsilon^i$, for $e \in \mathcal{E}$. Hence, the instance fulfills Assumption 1 because $u_1^{SW} \leq \ldots \leq u_E^{SW}$ and the principal's utilities satisfy $u_1^P \geq \ldots \geq u_E^P$, so it is optimal for the principal to implement effort 1.

The welfare loss corresponding to this instance is given by $u_E^{SW}/u_1^{SW} = E/(1 + \sum_{i=1}^{E-1} \varepsilon^i)$, which converges to E as $\varepsilon \to 0^+$. Therefore, Theorem 3 is tight because we found a series of instances converging to a matching lower bound.

4 Generalizations of the Basic Model

The results we have provided hold true for generalizations of the basic problem introduced in Sect. 2. First, the main result is valid when the agent can incur arbitrary (potentially negative) costs for any effort, and when the utility for the outside option is arbitrary (so far it was assumed to be zero). Second, more general limited liability constraints and imposing a minimum output do not have an impact in the worst-case bounds presented earlier. In this context, we can provide more accurate bounds that depend on some other characteristics of the instance. Third, MRLP is not needed for the case of two efforts. All results remain valid without it. Fourth, considering the problem from the perspective of the principal, we can show how to adapt the worst-case bounds provided earlier and express them with respect to the principal's payoff. Fifth, in the case with two outcomes we relax Assumption 1 by showing that the sequence of social welfare utilities is unimodal, and that any effort violating that order is infeasible. Finally, when the principal hires an agent to perform multiple identical and independent tasks that follow the two-effort-two-outcome model, we can show the the worst-case welfare loss is independent of the number of tasks and equal to 2.

Acknowledgements. The research of the first and third authors was supported in part by FONDECYT through grants 1100267 and 1090050, respectively. Part of this work was done while the fourth author was visiting Universidad de Chile supported by the Millennium Institute on Complex Engineering Systems.

References

[1] Alchian, A.A., Demsetz, H.: Production, information costs, and economic organization. American Economic Review 62(5), 777–795 (1972)

[2] Babaioff, M., Feldman, M., Nisan, N.: Combinatorial agency. In: Proceedings of the 7th ACM Conference on Electronic Commerce (EC 2006), Ann Arbor, MI, pp. 18–28. ACM Press, New York (2006)

[3] Babaioff, M., Feldman, M., Nisan, N.: Free-riding and free-labor in combinatorial agency. In: Mavronicolas, M., Papadopoulou, V.G. (eds.) SAGT 2009. LNCS, vol. 5814, pp. 109–121. Springer, Heidelberg (2009)

[4] Grossman, S.J., Hart, O.D.: An analysis of the principal-agent problem. Econometrica 51(1), 7–45 (1983)

[5] Holmström, B., Milgrom, P.: Multi-task principal-agent analyses: Incentive contracts, asset ownership and job design. Journal of Law, Economics and Organization 7, 24–52 (1991)

[6] Holmström, B., Tirole, J.: Financial intermediation, loanable funds and growth. The Quarterly Journal of Economics 112(3), 663–691 (1997)

[7] Koutsoupias, E., Papadimitriou, C.H.: Worst-case equilibria. In: Meinel, C., Tison, S. (eds.) STACS 1999. LNCS, vol. 1563, pp. 404–413. Springer, Heidelberg (1999)

[8] Jensen, M.C., Meckling, W.H.: Rights and production functions: An application to labor-managed firms and codetermination. The Journal of Business 52(4), 469–506 (1979)

[9] Nisan, N., Roughgarden, T., Tardos, É., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, Cambridge (2007)

[10] Rothschild, M., Stiglitz, J.E.: Increasing risk: I. a definition. Journal of Economic Theory 2(3), 225–243 (1970)

[11] Shapiro, C., Stiglitz, J.E.: Equilibrium unemployment as a worker discipline device. American Economic Review 74(3), 433–444 (1984)

[12] Spence, M., Zeckhauser, R.: Insurance, information, and individual action. American Economic Review 61(2), 380–387 (1971)

[13] Stiglitz, J.E.: Incentives and risk sharing in sharecropping. Review of Economic Studies 41(2), 219–255 (1974)

Specializations and Generalizations of the Stackelberg Minimum Spanning Tree Game*

Davide Bilò[1], Luciano Gualà[2], Stefano Leucci[2], and Guido Proietti[1,3]

[1] Dipartimento di Informatica, Università dell'Aquila, 67010 L'Aquila, Italy
[2] Dipartimento di Matematica, Università di Tor Vergata, 00133 Roma, Italy
[3] Istituto di Analisi dei Sistemi ed Informatica, CNR, 00185 Roma, Italy
{davide.bilo,guido.proietti}@univaq.it,
guala@mat.uniroma2.it, leucci.stefano@gmail.com

Abstract. The *Stackelberg Minimum Spanning Tree* (StackMST) game is a network pricing (bilevel) optimization problem. The game is played by two players on a graph $G = (V, E)$, whose edges are partitioned into two sets: a set R of *red* edges (inducing a spanning tree of G) with a fixed non-negative real cost, and a set B of *blue* edges which are instead priced by a *leader*. This is done with the final intent of *maximizing* a revenue that will be returned for their purchase by a *follower*, whose goal in turn is to select a minimum spanning tree of G. StackMST is known to be APX-hard already when the number of distinct red costs is 2, as well as $\min\{k, 1+\ln\beta, 1+\ln\rho\}$-approximable, where k is the number of distinct red costs, β is the number of blue edges selected by the follower in an optimal pricing, and ρ is the maximum ratio between red costs. In this paper we analyze some meaningful specializations and generalizations of StackMST, which shed some more light on the computational complexity of the game. More precisely, we first show that if G is complete, then the following holds: (i) if there are only 2 distinct red costs, then the problem can be solved optimally (this contrasts with the corresponding APX-hardness of the general problem); (ii) otherwise, the problem can be approximated within $7/4 + \epsilon$, for any $\epsilon > 0$. Afterwards, we define a natural extension of StackMST, namely that in which blue edges have a non-negative *activation* cost associated, and the leader has a global activation budget that must not be exceeded, and, after showing that the very same approximation ratio as that of the original game can be achieved, we prove that if the spanning tree induced by the red edges has *radius* h (in terms of number of edges), then the problem admits a $(2h + \epsilon)$-approximation algorithm.

Keywords: Communication Networks, Minimum Spanning Tree, Stackelberg Games, Network Pricing Games.

* This work was partially supported by the PRIN 2008 research project COGENT (COmputational and GamE-theoretic aspects of uncoordinated NeTworks), funded by the Italian Ministry of Education, University, and Research.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 75–86, 2010.

1 Introduction

Leader-follower games, which were introduced by von Stackelberg in the far 1934 [12], have recently received a considerable attention from the computer science community. This is mainly due to the fact that the Internet is a vast electronic market composed of millions of independent end-users, whose actions are by the way influenced by a limited number of owners of physical/logical portions of the network (e.g., service providers). In particular, in a scenario in which the leaders can set the price for using a subset of network arcs, knowing that the followers will allocate a communication subnetwork obeying some criteria, a natural problem is that of analyzing how the leaders can optimize their pricing strategy. Games of this latter type are widely known as *Stackelberg Network Pricing Games* (SNPGs).

When only 2 players (i.e., a leader and a follower) are involved, a SNPG can be formalized as follows: We are given an either directed or undirected graph $G = (V, E)$, whose edge set is partitioned into a set R of *red* edges and a set B of *blue* edges, and an edge cost function $c : R \to \mathbb{R}^+$. The edges in B need to be priced by the leader. In the following, we assume that $n = |V|$ and $m = |R| + |B|$. Then, the leader moves first and chooses a pricing function $p : B \to \mathbb{R}^+$ for her[1] edges, in an attempt to *maximize* her objective function $f_1(p, H(p))$, where $H(p)$ denotes the decision which will be taken by the follower, consisting in the choice of a subgraph of G. This notation stresses the fact that the leader's problem is implicit in the follower's decision. Once observed the leader's choice, the follower reacts by selecting a subgraph $H(p) = (V', E')$ of G which *minimizes* his objective function $f_2(p, H)$, parameterized in p. Note that the leader's strategy affects both the follower's objective function and the set of feasible decisions, while the follower's choice only affects the leader's objective function. Quite naturally, we assume that f_1 is *price-additive*, i.e., $f_1(p, H(p)) = \sum_{e \in B \cap E'} p(e)$. This means, the leader decides edge prices having in mind that her revenue equals the overall price of her selected edges.

The most immediate SNPG is that in which we are given two specified nodes in G, say s, t, and the follower wants to travel along a *shortest path* in G between s and t (see [11] for a survey). This problem has been shown to be APX-hard [8], as well as $O(\log |B|)$-approximable [10]. For the case of multiple followers (each with a specific source-destination pair), Labbé *et al.* [9] derived a bilevel LP formulation of the problem (and proved NP-hardness), while Grigoriev *et al.* [7] presented algorithms for a restricted shortest path problem on parallel edges. Furthermore, when all the followers share the same source node, and each node in G is a destination of a single follower, then the problem is known as the *Stackelberg single-source shortest paths tree* game. In this game, the leader's revenue for each selected edge is given by its price multiplied by the number of paths – emanating from the source – it belongs to, and in [1] it was proven that finding an optimal pricing for the leader's edges is NP-hard, as soon as $|B| = \Theta(n)$.

[1] Throughout the paper, we adopt the convention of referring to the leader and to the follower with female and male pronouns, respectively.

Another basic SNPG, which is of interest for this paper, is that in which the follower wants to use a *minimum spanning tree* (MST) of G (now considered as undirected). For this game, known as *Stackelberg MST* (StackMST) game, in [5] the authors proved the APX-hardness already when the number of red edge costs is 2, and gave a $\min\{k, 1+\ln\beta, 1+\ln\rho\}$-approximation algorithm, where k is the number of distinct red costs, β is the number of blue edges selected by the follower in an optimal pricing, and ρ is the maximum ratio between red costs. In a further paper [6], the authors proved that the problem remains NP-hard even if G is planar, while it can be solved in polynomial time once that G has bounded treewidth.

Notice that all the above examples fall within the class of SNPGs handled by the general model proposed in [3], which encompasses all the cases where each follower aims at optimizing a polynomial-time network optimization problem in which the cost of the network is given by the sum of prices and costs of contained edges. Nevertheless, SNPGs for models other than this one have been studied in [2,4].

Our results. In this paper we analyze some meaningful specializations and generalizations of StackMST, which shed some more light on the computational complexity of the game. For the sake of presenting our results in a unifying framework, we start by defining the aforementioned generalized version of StackMST. First of all, notice that given any instance of StackMST, this can be simplified into an equivalent instance in which we compute a red MST of G, and then we discard all the red edges not belonging to it (see also [5]). Then, the *budgeted* StackMST game is a 2-player game defined as follows. We are given a red tree $T = (V, E(T))$ of n nodes where each edge $e \in E(T)$ has a fixed cost $c(e)$. Moreover, we are given an *activation* cost $\gamma(e)$ for each edge $e = (u, v) \notin E(T)$, and a budget Δ. The game, denoted by StackMST(γ, Δ), consists of two phases. In the first phase the *leader* selects a set F of edges to add to T such that the budget is not excedeed, i.e. $\sum_{e \in F} \gamma(e) \leq \Delta$, and then prices them with a price function $p : F \rightarrow \mathbb{R}^+$. In the second phase, the *follower* takes the weighted graph $G = (V, E(T) \cup F)$ resulting from the first phase, and computes a MST $M(F, p)$ of G. Throughout the paper, as usual we assume that when multiple optimal solutions are available for the follower, then he selects an optimal solution maximizing the leader's revenue. Then, the leader collects a revenue of $r(M(F, p)) = \sum_{e \in F \cap E(M(F,p))} p(e)$. Our goal is to find a strategy for the leader which maximizes the revenue.

Using this notation, the original StackMST game on a graph $G = (V, R \cup B)$ can be rephrased as a StackMST(γ, Δ) game in which T is any red MST of G, Δ is equal to 0, and the activation cost for an edge not in $E(T)$ is equal to 0 if it belongs to B, otherwise it is equal to any positive value. In this paper, we prove the following results:

1. StackMST$(0, 0)$ with only 2 distinct red costs can be solved optimally, where the first 0 in the argument is used to denote the fact that γ is identically equal to 0;

2. $\mathsf{StackMST}(0,0)$ can be approximated within $7/4+\epsilon$, for any $\epsilon > 0$, in general;
3. $\mathsf{StackMST}(\gamma, \Delta)$ admits a $\min\{k, 1 + \ln \beta, 1 + \ln \rho, 2h + \epsilon\}$-approximation algorithm, for any $\epsilon > 0$, where k, β and ρ are as previously defined for StackMST, and h denotes the *radius* of T w.r.t. the number of edges.

We point out that all the above problems have an application counterpart, since the $\mathsf{StackMST}(0,0)$ class of problems models the case in which the leader retains the potentiality to activate (at no cost) any missing connection in the network, while clearly result (3) complements the approximation ratio given in [5] whenever the radius of the red tree is bounded, which might well happen in practice. Finally, notice also that $\mathsf{StackMST}(0,0)$ is a specialization of the general StackMST, for which however we were not able to prove whether the problem is in P or not. Therefore, this remains a challenging open problem.

The rest of the paper is organized by providing each of the above results in a corresponding section. Due to space limitations, some proofs are omitted and result (1) is given within the paper only for the special case in which T is a path.

2 Exact Algorithm for $\mathsf{StackMST}(0,0)$ with Costs in $\{a, b\}$

In this section we present an exact polynomial time algorithm for $\mathsf{StackMST}(0,0)$ when the cost of any red edge belongs to the set $\{a, b\}$, with $0 < a \leq b$. Notice that this case is already APX-hard for StackMST. For the sake of clarity, we will present the algorithm and the analysis when the red tree is actually a path. The extension to the general case can be derived easily.

Before providing the result, we need to introduce some basic notation we will use in the rest of the paper. Let H be an undirected graph. For an edge $e \in E(H)$ we will denote by $\texttt{cycle}(H, e)$ the set of (simple) cycles in H containing edge e. Let $w : E(H) \to \mathbb{R}$ be a function on the edges of H. We define $w(H) := \sum_{e \in E(H)} w(e)$. Consider an instance $\langle T, c, \gamma, \Delta \rangle$ of $\mathsf{StackMST}(\gamma, \Delta)$. In [5], the authors proved that finding an optimal solution to the problem instance is equivalent to determining the set F of blue edges contained in the MST bought by the follower. In fact, under the assumption that F is a set of blue edges such that (V, F) is a forest and $\sum_{e \in F} \gamma(e) \leq \Delta$, then activating only edges in F and pricing each edge $e \in F$ with

$$p_F(e) := \min_{H \in \texttt{cycle}(G,e)} \max_{e' \in E(H) \cap E(T)} c(e'), \tag{1}$$

implies that F is contained in $M(F, p_F)$ as well as $r(M(F, p_F)) \geq r(M(F, p'))$ for every other pricing p' such that F is contained in $M(F, p')$. As a further remark, the authors in [5] observed that p_F can be computed in polynomial time. As a consequence, in the rest of the paper, we will focus on determining the set of leader's edges that have to be contained in the MST bought by the follower. Therefore, with a little abuse of notation, given a set F such that $\sum_{e \in F} \gamma(e) \leq \Delta$ and (V, F) is acyclic, we will denote by $r(F)$ the revenue yielded by the pricing p_F, i.e., $r(F) := r(M(F, p_F))$.

Now, we present an exact algorithm for $\mathsf{StackMST}(0,0)$ on a red path P with costs in $\{a,b\}$, with $0 < a \le b$. We call a subpath P' of P an *a-block* if P' has all edges of cost a, and P' is maximal (w.r.t. inclusion). We say that an a-block is *good* if its length is greater than or equal to 3, *bad* otherwise. Let σ be the number of bad blocks of P. The following lemma shows an upper bound to the maximum revenue r^*:

Lemma 1. $r^* \le c(P) - \min\left\{\sigma a, \left\lfloor\frac{\sigma}{2}\right\rfloor(b-a) + \left(\sigma - 2\left\lfloor\frac{\sigma}{2}\right\rfloor\right)\min\{a, b-a\}\right\}$.

Proof. Let n_a be the number of red edges of cost a. Let T^* be the tree computed by the follower w.r.t. an optimal solution. Moreover, let $B_1,\ldots,B_\sigma$ and $\hat{B}_1,\ldots,\hat{B}_{\sigma'}$ be the bad and the good blocks of P, respectively. We denote by m_i and $\hat{m}_j$ the number of edges of B_i and $\hat{B}_j$, respectively. Moreover, for an edge $e=(x,y)$, $T^*(e)$ will denote the unique path in T^* between x and y (observe that $T^*(e)$ may be the path containing only edge e). For each $i=1,\ldots,\sigma$ and $j=1,\ldots,\sigma'$, consider $T_i = \bigcup_{e\in E(B_i)} T^*(e)$ and $\hat{T}_j = \bigcup_{e\in E(\hat{B}_j)} T^*(e)$.[2] Let $\mathcal{T} = \{T_1,\ldots,T_\sigma\} \cup \{\hat{T}_1,\ldots,\hat{T}_{\sigma'}\}$. Observe that for each i,j, we have: (i) T_i and $\hat{T}_j$ are trees and every edge has cost a, (ii) $V(B_i) \subseteq V(T_i)$ and $V(\hat{B}_j) \subseteq V(\hat{T}_j)$, and (iii) $E(T^*) \cap E(B_i) \neq \emptyset$ or T_i contains at least m_i+1 edges.

Let us consider the following graph $H = (\bigcup_i V(T_i) \cup \bigcup_j V(\hat{T}_j), \bigcup_i E(T_i) \cup \bigcup_j E(\hat{T}_j))$ and let N be the number of nodes of H. Clearly, H is a forest; moreover, $N \ge \sum_{i=1}^{\sigma} m_i + \sum_{j=1}^{\sigma'} \hat{m}_j + \sigma + \sigma' = n_a + \sigma + \sigma'$ as blocks are pairwise vertex disjoint. Consider the set $\mathcal{C}$ of the connected components of H, and let $X = \{T_i \mid T_i \in \mathcal{C}, i=1,\ldots,\sigma, \forall T' \in \mathcal{T}, E(T_j) \not\subseteq E(T') \text{ and } E(T') \not\subseteq E(T_j)\}$. Let $\ell = |X|$ and let $\ell_1 = |\{T_i \mid T_i \in X, E(T^*) \cap E(B_i) \neq \emptyset\}|$. Notice that ℓ_1 is a lower bound to the number of red edges in H. Finally, let $t = |\mathcal{C} \setminus X|$. We have that the number of connected components of H is $\ell + t$, and hence H has $N - \ell - t$ edges. Now, let $Y = \{T_1,\ldots,T_\sigma\} \setminus X$. We have $|Y| = \sigma - \ell$. Moreover, since each $T_i \in Y$ has "merged" with at least one other tree, we have $t \le \sigma' + \left\lfloor\frac{\sigma-\ell}{2}\right\rfloor$. Therefore,

$$
\begin{aligned}
r^* &\le \big(N-\ell-t\big)a - \ell_1 a + \Big(n - 1 - \big(N-\ell-t\big)\Big)b \\
&\le (n_a + \sigma + \sigma' - \ell - t)(a-b) + (n-1)b - \ell_1 a \\
&= c(P) - \big(\sigma + \sigma' - \ell - t\big)(b-a) - \ell_1 a \\
&\le c(P) - \left((\sigma - \ell) - \left\lfloor\frac{\sigma-\ell}{2}\right\rfloor\right)(b-a) - \ell_1 a \\
&\le c(P) - \min\left\{\sigma a, \left\lfloor\frac{\sigma}{2}\right\rfloor(b-a) + \left(\sigma - 2\left\lfloor\frac{\sigma}{2}\right\rfloor\right)\min\{a, b-a\}\right\}. \qquad \square
\end{aligned}
$$

Now we present an exact algorithm achieving revenue equal to the upper bound of Lemma 1. The algorithm uses the following four rules. Each rule considers a subpath of P and specifies a feasible solution for the subpath, i.e. a set of blue edges within the subpath with a corresponding pricing. The solutions corresponding to the rules are shown in Figure 1.

[2] Here the union symbol denotes the union of graphs.

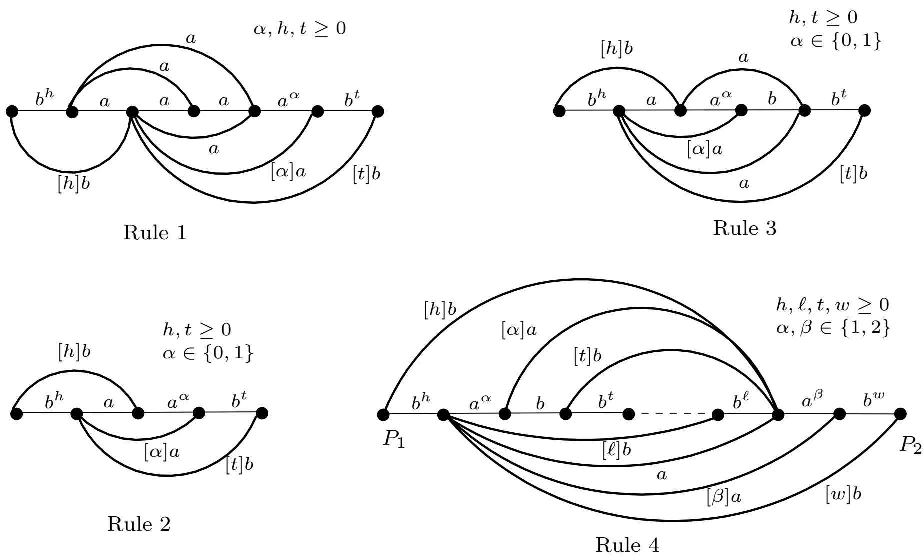


Fig. 1. Rules used by the algorithm to solve subpaths. We denote by σ^δ a path of δ edges each having a cost of σ. An edge with label $[i]\sigma$ represents i blue edges each having a price of σ. Observe that, except for Rule 2, if we run Kruskal's algorithm by giving priority to blue edges, all red edges with cost a will be discarded. Concerning Rule 2, Kruskal's algorithm selects a single red edge of cost a (of the shown subpath).

Rule 1: Let P' be a subpath of P containing only one a-block, and this a-block is good. We can obtain revenue $c(P')$ from P' by adding blue edges only within P'.

Rule 2: Let P' be a subpath of P containing only one a-block and this a-block is bad. We can compute a solution with revenue $c(P') - a$ from P'.

Rule 3: Let P' be a subpath of P containing only one a-block, this a-block is bad, and P' contains a red edge of cost b. We can obtain a revenue of $c(P') - (b - a)$ from P'.

Rule 4: Let P_1, P_2 be two edge-disjoint subpaths of P each containing only one a-block. Assume that both a-blocks are bad and P_1 contains an edge of cost b whose removal separates the two a-blocks. We can obtain a revenue of $c(P_1) + c(P_2) - (b - a)$ from P_1 and P_2.

Our algorithm is as follows. If $b \geq 3a$ then we split P into subpaths each of them containing exactly one a-block. Then we apply rule 1 or rule 2 to each subpath, depending on the a-block in the subpath is good or bad. By Lemma 2, this solution yields a revenue of $c(P) - \sigma a$.

Now, consider the case $b < 3a$. Let $B_1, \ldots, B_\sigma$ be the bad a-blocks contained in P from left to right. The algorithm splits P into subpaths such that (i) each subpath contains exactly one a-block, (ii) for every $i = 0, \ldots, \lfloor \sigma/2 \rfloor$, subpath containing B_{2i+1} has an edge of cost b incident to its right endvertex, and (iii) if σ is odd, the subpath containing B_σ has an edge of cost b incident to its left endvertex.

Let P_i be the subpath containing B_i. The algorithm uses rule 4 for every pairs of subpaths P_{2i+1}, P_{2i+2}, $i = 0, \ldots, \lfloor \sigma/2 \rfloor$, rule 1 for every subpath containing a good a-block. Finally, if σ is odd, we apply rule 3 for P_σ when $b \leq 2a$, while we use rule 2 when $b > 2a$. It is easy to see that the revenue of this solution coincides with $c(P) - \min\left\{\sigma a, \left\lfloor \frac{\sigma}{2} \right\rfloor (b-a) + \left(\sigma - 2\left\lfloor \frac{\sigma}{2} \right\rfloor\right) \min\{a, b-a\}\right\}$. Hence, from Lemma 1, we have:

Theorem 1. StackMST$(0,0)$ *can be solved in polynomial time when red edge costs are in* $\{a, b\}$.

3 StackMST$(0,0)$ Can Be Approximated within $7/4 + \epsilon$

In this section we design an algorithm that achieves an approximation ratio of $7/4 + \epsilon$. The idea of the algorithm is to partition the red tree into suitable subtrees that can be solved optimally and such that we can guarantee a revenue of at least $4/7$ of the cost of each subtree. Let $T = (V, E(T))$ be the red tree. We say that $T_1 = (V_1, E_1), \ldots, T_\ell = (V_\ell, E_\ell)$ is a partition of T into ℓ subtrees if (i) each T_i is a subtree of T, (ii) $V = \bigcup_i V_i, E(T) = \bigcup_i E_i$, and (iii) for each $i, j, i \neq j$, $E_i \cap E_j = \emptyset$. As a consequence of Equation (1), we can immediately derive the following:

Lemma 2. *Let* $T_1 = (V_1, E_1), \ldots, T_\ell = (V_\ell, E_\ell)$ *be a partition of* T *into* ℓ *subtrees. For each* i*, let* F_i *be a feasible solution for the tree* T_i*. Then* $F = \bigcup_{i=1}^{\ell} F_i$ *is a feasible solution for* T*. Moreover,* $r(F) = \sum_{i=1}^{\ell} r(F_i)$*.*

Lemma 3. *Let* T *be a tree rooted at a node* s*. There always exists a partition of* T *into* ℓ *subtrees* $T_1, \ldots, T_\ell$ *such that*

- T_ℓ *has at most 2 edges and at least one of them is incident to* s*;*
- *for every* $1 \leq j \leq \ell - 1$*,* T_j *is either (i) a path of 3 or 4 edges; or (ii) a star with at least 3 edges.*

Moreover, this partition can be found in polynomial time.

Proof. We provide a polynomial time algorithm that finds the partition of the lemma. Let h be the height of T and let $d(v)$ denote the depth of v in T, i.e. the number of edges of the path (in T) between s and v. We denote by $S(v)$ the set of the children of v. Moreover, we use $\bar{v}$ to denote the parent of v and $\bar{\bar{v}}$ to denote the parent of $\bar{v}$. We proceed in phases. In phase j, we find a subtree T_j by applying one of the rules below (we consider them in order), then we remove T_j and we move to the next phase. We stop when no rule can be applied. Let L be the set of leaves of T with depth equal to h. The rules are the following (see Figure 2):

Rule 1: if there exists a node $v \in L$ with $d(v) \geq 2$ and such that v has at least one sibling, then T_j is the star with edge set $\{(\bar{v}, \bar{\bar{v}})\} \cup \{(\bar{v}, u) \mid u \in S(\bar{v})\}$;

Rule 2: if there exists a node $v \in L$ with $d(v) \geq 2$ such that $\bar{v}$ has a sibling u and u is a leaf, then T_j is the path with edge set $\{(v,\bar{v}),(\bar{v},\bar{\bar{v}}),(\bar{\bar{v}},u)\}$;

Rule 3: if there exists a node $v \in L$ with $d(v) \geq 2$ such that $\bar{v}$ has a sibling u and u is not a leaf, then let u' be the unique child of u (u' must be unique otherwise rule 1 would apply). Then, T_j is the path with edge set $\{(v,\bar{v}),(\bar{v},\bar{\bar{v}}),(\bar{\bar{v}},u),(u,u')\}$;

Rule 4: if there exists a node $v \in L$ with $d(v) \geq 3$, then T_j is the path with edge set $\{(v,\bar{v}),(\bar{v},\bar{\bar{v}}),(\bar{\bar{v}},\bar{\bar{\bar{v}}})\}$;

Rule 5: if T is a star with at least 3 edges, then $T_j = T$.

Now, assume that the last phase is phase $\ell - 1$, then we set T_ℓ equal to the remaining tree T. If there is no edge left, we set T_ℓ equal to the empty subtree. It is easy to see that if T_ℓ is not empty, it must have at most 2 edges and one of them must be incident to s. Moreover, since each phase takes polynomial time and each T_j with $j < \ell$ contains at least one edge, the claim follows. □

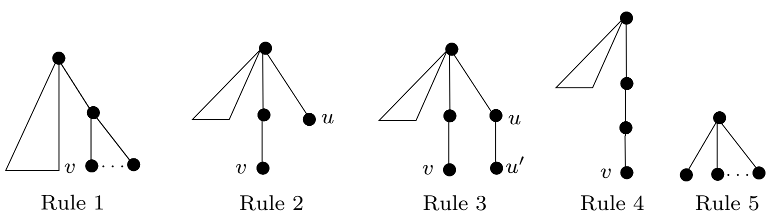


Fig. 2. The five rules of the decomposition algorithm

The following lemma holds

Lemma 4. *Let T be a star with at least 3 edges. Then we can obtain a revenue of at least $\frac{2}{3}c(T)$.*

Proof. Let s be the center of the star, and let $u_1,\ldots,u_t$ be the leaves ordered such that $c(s,u_1) \leq c(s,u_2) \leq \cdots \leq c(s,u_t)$. The set of blue edges $F = \{(u_1,u_j) \mid j = 2,\ldots,t\}$ yields a revenue of $r(F) = \sum_{j=2}^{t} c(s,u_j) \geq \frac{2}{3}c(T)$, since $t \geq 3$. □

The following lemma can be proved using case analysis

Lemma 5. *Let P be a path of 3 or 4 edges. Then we can obtain a revenue of at least $\frac{4}{7}c(P)$.*

We are now ready to prove the following:

Theorem 2. $\mathsf{StackMST}(0,0)$ *can be approximated within a factor of $7/4+\epsilon$, for any constant $\epsilon > 0$.*

Proof. W.l.o.g., we can restrict ourselves to the case $n \geq \frac{7}{2\epsilon}+2$, as for the other case we can always use the exhaustive search algorithm that tries all the possible sets of blue edges to find an optimal solution. For each v, let

$\mu(v) = \max_{u|(u,v)\in E(T)} c(u,v)$.[3] We root T at a node s minimizing μ. Then we decompose T using the algorithm given in Lemma 3 and we solve optimally each T_j with $j < \ell$. Let F_j^* be an optimal solution for each tree T_j, for every $j = 1, \ldots, \ell$. Let $F = \bigcup_{j=1}^{\ell} F_j^*$ and observe that $r(F_\ell) \geq c(T_\ell) - \min_{e\in E(T_\ell)} c(e)$. Let $e = (s,z)$ be an edge in T_ℓ and let T' the forest obtained from T by removing e. Clearly, $c(T) \leq c(T') + \mu(s)$. Let F^* denote an optimal solution for T. Clearly, $r(F^*) \leq c(T)$. As Lemma 2 together with Lemma 4 and Lemma 5 implies that $r(F) \geq 4/7c(T')$ and since $c(T') \geq \frac{1}{2}\sum_{v\in V\setminus\{s,z\}} \mu(v) \geq \frac{n-2}{2}\mu(s)$, we obtain $\frac{r(F^*)}{r(F)} \leq \frac{c(T)}{r(F)} \leq \frac{c(T')+\mu(s)}{r(F)} \leq \left(1 + \frac{2}{n-2}\right)\frac{c(T')}{r(F)} \leq 7/4 + \frac{7}{2n-4} \leq 7/4 + \epsilon$. □

4 StackMST(γ, Δ) on Trees of Bounded Radius

In this section, we study the general StackMST(γ, Δ). First, we observe that for this generalized version, the very same approximation ratio as that of the original game can be achieved as the *single-price algorithm* defined in [5] can be easily adapted to provide an approximation of $\min\{k, 1 + \ln\beta, 1 + \ln\rho\}$ for StackMST(γ, Δ) as well, where k is the number of distinct red costs, β is the number of blue edges selected by the follower in an optimal solution, and ρ is the maximum ratio between red costs.

In the remaining of the section, we focus on the case in which T is a tree of radius h measured w.r.t. the number of edges. For this case, we show that the problem remains APX-hard even for constant values of h as well as approximable within a factor of $2h + \epsilon$.

We now study StackMST(γ, Δ) when T is a tree of radius h. In the remaining of this section, we will assume that T is rooted at v_0 and has height h (corresponding to its radius). First, we observe that the reduction in [5] proving that StackMST is APX-hard even if T is a path can be modified to show that

Theorem 3. StackMST *is* APX-*hard even if T is a star.*

In the remaining of the section we will show the existence of a $(2h + \epsilon)$-approximation algorithm. The main idea of the algorithm is to reduce the problem instance into h instances in which the red trees are stars. With a little abuse of notation, in each of the h instances, the leader is sometimes allowed to activate edges which are parallel to red edges. We denote by $V_i = \{v_1, \ldots, v_{\ell_i}\}$ the set of vertices at level i and by E_i the set of edges in T going from vertices in V_i to their parents. Let T_i be a red star obtained by identifying all red edges in T but those in E_i. With a little abuse of notation, when edge (u,v) is identified, and w.l.o.g. u is the parent of v, we assume that the corresponding vertex is labeled with u. Thus, according to this assumption, we have that T_i is a star centered at v_0 with $v_1, \ldots, v_{\ell_i}$ as leaves. The cost of a red edge $e = (v_0, v)$ in T_i is $c_i(e) = c(u,v)$, where u is the parent of v in T. Let $\hat{T}_0, \hat{T}_1, \ldots, \hat{T}_{\ell_i}$ be the connected components in $T - E_i$. W.l.o.g., assume $v_i \in V(\hat{T}_i)$. Let $e_{j,q}$ be a blue edge connecting $\hat{T}_j$

[3] With a slight abuse of notation, we will write $c(u,v)$ instead of $c((u,v))$ in the rest of the paper.

and $\hat{T}_q$ with cheapest activation cost. Let $\texttt{blue}_i := \{e_{j,q} \mid j, q = 0, \ldots, \ell_i, j \neq q\}$ and let $B_i := \{\bar{e}_{j,q} := (v_j, v_q) \mid e_{j,q} \in \texttt{blue}_i\}$ be the set of blue edges the leader is allowed to activate in T_i. The activation cost of $\bar{e}_{j,q} \in B_i$ is $\gamma_i(\bar{e}_{j,q}) := \gamma(e_{j,q})$.

Let F^* be an optimal solution for the leader on input instance T and let $F_i^* := \{(v_j, v_q) \mid (u, v) \in F^*, u \in V(\hat{T}_j), v \in V(\hat{T}_q), j \neq q\}$. Let $G_i^* := (\{v_0, \ldots, v_{\ell_i}\}, F_i^*)$ and denote by $\texttt{comp}(G_i^*)$ the set of the connected components of G_i^*. We start proving an upper bound on the revenue yielded by F^*.

Lemma 6. $r(F^*) \leq c(T) - \sum_{i=1}^{h} \sum_{H \in \texttt{comp}(G_i^*)} \min_{v \in V(H)} c_i(v_0, v)$.[4]

Proof. Observe that for every $H \in \texttt{comp}(G_i^*)$ not containing vertex v_0, at least one red edge (v_0, v), for some $v \in V(H)$, has to be contained in any MST of $(V(T_i), E(T_i) \cup F_i^*)$. Thus, for some $v \in V(H)$, at least one edge (u, v) where u is the parent of v in T has to be contained in any MST of $G = (V(T), E(T) \cup F^*)$. As $c_i((v_0, v)) = c(u, v)$, the claim follows by summing over all components $H \in \texttt{comp}(G_i^*)$ for all i's. $\square$

The key idea of our algorithm is to find a set F of blue edges whose overall activation cost does not exceed the budget and such that (V, F) is a forest of stars. More precisely, for every $i = 1, \ldots, h$, the algorithm finds a set $\hat{F}_i \subseteq B_i$ such that $\sum_{e \in \hat{F}_i} \gamma_i(e) \leq \Delta$ and $\hat{G}_i := (V(T_i), \hat{F}_i)$ is a forest of stars. Let $F_i := \{e_{j,q} \mid \bar{e}_{j,q} \in \hat{F}_i\}$. Observe that (i) $G_i := (V(T), F_i)$ is a forest of stars and (ii) the overall activation cost of the edges in F_i equals the one of the edges in $\hat{F}_i$. Furthermore, using Equation (1), we can derive the following

Lemma 7. *Let* $L_i := \{v \mid v \in V(T_i), v$ *is a leaf of some star in* $\hat{G}_i\}$.[5] *Then,* $r(\hat{F}_i) \geq \sum_{v \in L_i} c_i(v_0, v)$.

Lemma 8. $r(F_i) \geq r(\hat{F}_i)$.

Lemma 9. *Let* $B' \subseteq B_i$ *and let* $U = \{v \mid v$ *is an endvertex of some edge in* $B'\}$. *There exists a polynomial time algorithm that finds two sets* F^1 *and* F^2 *such that (i)* $F^1, F^2 \subseteq B'$, *(ii) both* $(V(T_i), F^1)$ *and* $(V(T_i), F^2)$ *are forests of stars, and (iii)* $r(F^1) + r(F^2) \geq \sum_{v \in U} c_i(v_0, v)$.

Proof. Let D be the graph induced by edge set B'. Let D^j be any of the t connected component in D and let T^j be any spanning tree in D^j. As T^j is a bipartite graph, it is possible to partition the set of its vertices into two sets V_1^j and V_2^j in polynomial time. Moreover, by the connectivity of T^j, every vertex $v \in V_\ell^j$ ($\ell \in \{1, 2\}$) is adjacent to some vertex in $V_{3-\ell}^j$, and thus it is easy to find a set E_ℓ^j of edges in T^j such that $(V(D^j), E_\ell^j)$ is a forest of stars with centers in V_ℓ^j and leaves in $V_{3-\ell}^j$. Therefore, for $\ell = 1, 2$, $F^\ell = \bigcup_{j=1}^{t} E_\ell^j$ are two sets of edges satisfying (i) and (ii). Furthermore, $\bigcup_{j=1}^{t} (V_1^j \cup V_2^j) = \bigcup_{j=1}^{t} V(D^j) = V(D)$. As a consequence, from Lemma 7, (iii) is also satisfied. $\square$

[4] With a slight abuse of notation, we assume $c_i(v_0, v_0) = 0$.

[5] If a star contains only one edge, then choose exactly one of its endvertices as a leaf.

To compute $\hat{F}_i$, the algorithm does the following. Let e^i_j be a leader's edge in B_i incident to v_j with cheapest activation cost. Our algorithm uses the well-known FPTAS for the knapsack problem to compute a $(1+\epsilon/(2h))$approximate solution S_i for the instance of knapsack where each edge e^i_j is an object of profit $c_i(v_0, v_j)$ and volume $\gamma_i(e^i_j)$, and the volume of the knapsack is Δ. Denote by K_i the input instance of knapsack. Let $B' = \{e_j \mid e^i_j \in S_i\}$. The algorithm uses the decomposition algorithm described in Lemma 9 to find two subsets of edges F^1 and F^2, and then it sets $\hat{F}_i$ to F^1 if $r(F^1) \geq r(F^2)$, F^2 otherwise. The pseudocode of the algorithm is given in Algorithm 1.

Algorithm 1

```
for i = 1 to h do
    compute a (1 + ε/(2h))-approximate solution S_i for the knapsack instance K_i
    B' := {e^i_j | e^i_j ∈ S_i}
    compute F^1 and F^2 w.r.t. B' as explained in Lemma 9
    if r(F^1) ≥ r(F^2) then  F̂_i := F^1 else  F̂_i := F^2 end if
    F_i := {e_{j,q} | ē_{j,q} ∈ F̂_i}
end for
return the best of the F_i's
```

Theorem 4. *Algorithm 1 computes a* $(2h+\epsilon)$*-approximate solution in polynomial time for* StackMST(γ, Δ)*, for any constant* $\epsilon > 0$.

Proof. Let $\bar{G}_1, \ldots, \bar{G}_\ell$ be the connected components contained in G^*_i. Let $\bar{F}^*_i = \bigcup_{t=1}^{\ell} E(\bar{G}_t)$. Let ν_t be a vertex of $V(\bar{G}_t)$ such that $c_i(v_0, \nu_t) \leq c_i(v_0, v)$ for every $v \in V(\bar{G}_t)$. As $\gamma_i(e^i_j) \leq \gamma_i(e)$ for every $e \in \bar{F}^*_i$ with e incident to v_j, we have that $S^* = \bigcup_{t=1}^{\ell}\{e^i_j \mid v_j \in V(\bar{G}_t), v_j \neq \nu_t\}$ is a feasible solution of the knapsack instance K_i. Furthermore, the total profit $\texttt{profit}(S^*)$ of edges in S^* is

$$\texttt{profit}(S^*) \geq c(T_i) - \sum_{t=1}^{\ell} c_i(v_0, \nu_t).$$

Let S_i be the $(1+\epsilon/(2h))$-approximate solution of the knapsack instance K_i computed by the algorithm and let $B' := \{e^i_j \mid e^i_j \in S_i\}$. Let F^1 and F^2 be computed as explained in Lemma 9. As $F^1, F^2 \subseteq B'$, then $\sum_{e \in F^j} \gamma_i(e) \leq \Delta$, for every $j = 1, 2$. Moreover, as $r(\hat{F}_i) = \max\{r(F^1), r(F^2)\}$, we have that

$$(2+\epsilon/h) r(\hat{F}_i) \geq \big(1+\epsilon/(2h)\big)\big(r(F^1) + r(F^2)\big) \geq c(T_i) - \sum_{t=1}^{\ell} c_i(v_0, \nu_t).$$

As a consequence, because of Lemma 6 and Lemma 8, we have that

$$(2h+\epsilon) \max_{i=1,\ldots,h} r(F_i) \geq c(T) - \sum_{i=1}^{h} \sum_{H \in \texttt{comp}(H^*_i)} \min_{v \in V(G)} c_i(v_0, v) \geq r(F^*).$$

This completes the proof. □

References

1. Biló, D., Gualà, L., Proietti, G., Widmayer, P.: Computational aspects of a 2-player Stackelberg shortest paths tree game. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 251–262. Springer, Heidelberg (2008)
2. Biló, D., Gualà, L., Proietti, G.: Hardness of an asymmetric 2-player Stackelberg network pricing game. In: Electronic Colloquium on Computational Complexity (ECCC), TR09-112 (November 3, 2009)
3. Briest, P., Hoefer, M., Krysta, P.: Stackelberg network pricing games. In: Proc. of the 25th Ann. Symp. on Theoretical Aspects of Computer Science (STACS), pp. 133–142 (2008), http://drops.dagstuhl.de/opus/volltexte/2008/1340
4. Briest, P., Hoefer, M., Gualà, L., Ventre, C.: On stackelberg pricing with computationally bounded consumers. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 42–54. Springer, Heidelberg (2009)
5. Cardinal, J., Demaine, E.D., Fiorini, S., Joret, G., Langerman, S., Newman, I., Weimann, O.: The Stackelberg minimum spanning tree game. Algorithmica (2009), doi:10.1007/s00453-009-9299-y
6. Cardinal, J., Demaine, E.D., Fiorini, S., Joret, G., Newman, I., Weimann, O.: The Stackelberg minimum spanning tree game on planar and bounded-treewidth graphs. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 125–136. Springer, Heidelberg (2009)
7. Grigoriev, A., van Hoesel, S., van der Kraaij, A., Uetz, M., Bouhtou, M.: Pricing network edges to cross a river. In: Persiano, G., Solis-Oba, R. (eds.) WAOA 2004. LNCS, vol. 3351, pp. 140–153. Springer, Heidelberg (2005)
8. Joret, G.: Stackelberg network pricing is hard to approximate (2009) (manuscript)
9. Labbé, M., Marcotte, P., Savard, G.: A bilevel model of taxation and its application to optimal highway pricing. Management Science 44(12), 608–622 (1998)
10. Roch, S., Savard, G., Marcotte, P.: An approximation algorithm for Stackelberg network pricing. Networks 46(1), 57–67 (2005)
11. van Hoesel, S.: An overview of Stackelberg pricing in networks, Research Memoranda 042, Maastricht: METEOR, Maastricht Research School of Economics of Technology and Organization (2006)
12. von Stackelberg, H.: Marktform und Gleichgewicht (Market and Equilibrium). Verlag von Julius Springer, Vienna (1934)

A Novel Approach to Propagating Distrust

Christian Borgs[1], Jennifer Chayes[1], Adam Tauman Kalai[1],
Azarakhsh Malekian[2], and Moshe Tennenholtz[3]

[1] Microsoft Research New England
[2] Northwestern University, EECS Department
[3] Microsoft R and D center, Israel

Abstract. Trust propagation is a fundamental topic of study in the theory and practice of ranking and recommendation systems on networks. The Page Rank [9] algorithm ranks web pages by propagating trust throughout a network, and similar algorithms have been designed for recommendation systems. How might one analogously propagate distrust as well? This is a question of practical importance and mathematical intrigue (see, e.g., [2]). However, it has proven challenging to model distrust propagation in a manner which is both logically consistent and psychologically plausible. We propose a novel and simple extension of the Page Rank equations, and argue that it naturally captures most types of distrust that are expressed in such networks. We give an efficient algorithm for implementing the system and prove desirable properties of the system.

1 Introduction

Trust-based recommendation and ranking systems are becoming of greater significance and practicality with the increased availability of online reviews, ratings, hyperlinks, friendship links, and follower relationships. In such a recommendation system, a "trust network" is used to give an agent a personalized recommendation about an item in question, based on the opinions of her trusted friends, friends of friends, etc. There are many recommendation web sites dedicated to specific domains, such as hotels and travel, buyer-seller networks, and a number of other topics. In several of these sites, users may declare certain agents whose recommendations they trust or distrust, either by directly specifying who they dis/trust or indirectly by rating reviews or services rendered. We show how trust or distrust may be naturally propagated in a manner similar to Page Rank [9], extending the work of Andersen *et al.* [1], which was for trust alone. For simplicity, we focus on personalized recommendation systems, though our approach applies to ranking systems as well. We take, as a running example, a person that asks their trusted friends if they recommend a certain specialist. Some friends might have personal experience, while others would ask their friends, and so forth. The agent may then combine the feedback using majority vote or another aggregation scheme. Such personalized recommendations may better serve a person who does generally agree with popular opinion on a topic and may be

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 87–105, 2010.

less influenced by people of poor taste and spammers. Figure 1 illustrates two distrust networks, formally defined as follows. Among a set of nodes, N, there is a designated source $s \in N$ that seeks a positive, neutral, or negative recommendation. For $u, v \in N$, weight $w_{uv} \in [-1, 1]$ indicates the amount of trust (or distrust if negative) node u places in v, and we require $\sum_v |w_{uv}| \leq 1$ for each node u. Following Andersen *et al.*, for simplicity we consider two disjoint sets $V_+, V_- \subseteq N$,, of positive and negative voters, who are agents that have fixed positive or negative opinions about the item in question, respectively. In the case of a doctor, these may be the agents that have visited the doctor themselves. We assume that $w_{uv} = 0$ for any voter u, e.g., if they have first-hand experience they will not ask their trusted friends about the doctor. The simple random walk system [1] suggests that one consider a random walk starting at s and terminating at a voter. The recommendation is based upon whether it is more likely that the random walk terminates in a positive voter or a negative voter. In Figure 1, in case (b) clearly the recommendation should be negative, because the total trust in the positive voter is less than that in the negative voter, due to node z's distrust. However, in complex networks such as Figure 1, in case (c) it is not obvious whom to trust and distrust. It could be that w and z are trustworthy and y is not (hence y's distrust in w and z should be disregarded), or the opposite could be the case. It is not clear if there is any psychologically plausible and mathematically consistent recommendation system in general graphs.

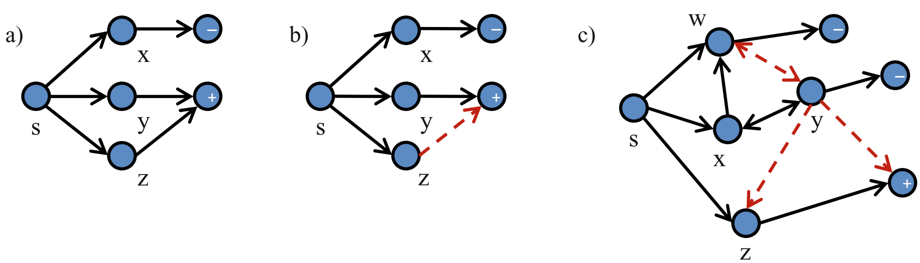


Fig. 1. Examples of (dis)trust networks. In each case, node s seeks a recommendation in $\{-, 0, +\}$. A solid/dashed edge from u to v indicates that u trusts/distrusts v. A $+/-$ indicates a voter has a fixed positive/negative opinion of the doctor (perhaps based upon first-hand experience). a) The recommendation is $+$. b) The recommendation is $-$. c) Typical networks may be cyclic and more complex.

We propose a simple solution, and justify it both through examples and the axiomatic approach, arguing that it satisfies a number of reasonable properties that one might like in a recommendation system. Moreover, it is the only system satisfying these properties. Before delving into the details, we point out that there are several types of people one might want to distrust in such a network:

1. **Bad guys.** Self-serving people, such as spammers, will recommend things based upon their own personal agendas. Distrust links can decrease their

influence on such a system. One has to be careful, however, not to simply negate whatever they suggest, since they may be strategic and say anything to achieve their goal.

2. **Careless big mouths.** Say, in your social circle, there is a vocal and opinionated person who you have discovered is careless and unreliable. Even without a trust link from you to them, that person's opinions may be highly weighted in a recommendation system if several of your trusted friends trust that person. Hence, it may be desirable for you to explicitly indicate distrust in that person, while maintaining trust links to other friends so that you can take advantage of their opinions and those of their other friends.
3. **Polar opposites.** On some two-sided topics (e.g., Democrats vs. Republicans), a graph may be largely bipartite and distrust links may indicative that one should believe the opposite of whatever the distrusted person says.

We model distrusted people as behaving in an arbitrary fashion, and seek to minimize their influence on a system. This includes the possibility that they may be self-serving adversaries (*bad guy*), but it is also a reasonable approach for people whose opinions do not generally agree (*careless*). Our approach fails to take advantage of the opinions of distrusted people (*polar*), in two-sided domains. However, many real-world topics in which people seek recommendations are multi-faceted.

We believe that it is important for trust and distrust to be propagated naturally in such a system. For example, in case of *careless* above, the distrust you indicate will not only affect the system's recommendations to you but also to people that trust you. As we argue below, propagating distrust is a subtle issue. In this paper, we present a novel and simple solution which we feel is the natural extension of the random-walk system of Andersen *et al.* [1] (similar to Page Rank [9] and other recommendation system) to distrust.

The focus of the present paper is distrust propagation in domains where there is typically little overlap in personal experiences: it is unlikely that any two people have seen a number of doctors in common or been to a number of common hotels. Ideally, one might combine collaborative filtering with trust and distrust propagation, but we leave that to future work.

1.1 The Propagation of Trust and Distrust

Figure 2 gives simple examples showing how trust and distrust may be used in recommendation systems. These examples illustrate why it is important to keep track not only of who is distrusted, but also who it is that distrusts the person. Roughly speaking, by trust propagation, we mean that if u trusts v and v trusts w, then it is as if u trusts w directly, though the weight of the trust may be diminished. Matters are more complicated in the case of distrust, which is the focus of the paper.

The ancient proverb, "the enemy of my enemy is my friend," immediately comes to mind when considering distrust propagation. In our context, it suggests that if u distrusts v and v distrusts w, then u should trust w However,

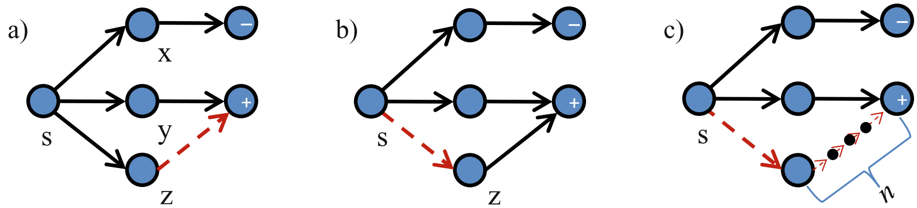


Fig. 2. Our recommendation to s is $-$ due to diminished trust in the $+$ voter. b) Our recommendation is 0 because z is untrusted, and hence we ignore z's outgoing edges. A system whose recommendation was influenced by z could be swayed arbitrarily by a careless or adversarial z.c) Here z is connected to the $+$ voter by a chain of distrust links. Again our recommendation to s is 0. The recommendation of systems obeying "the enemy of my enemy is my friend," would depend on whether n is even or odd.

this principle is inappropriate in most multifaceted domains. For example, you would probably not necessarily select a doctor on the basis that the doctor is heavily distrusted by people you distrust. Figure 2 case (c) illustrates how this maxim may lead to a recommendation that depends on whether the number of nodes in a network is even or odd, an effect of questionable psychological plausibility. Analogously, "the friend of my enemy is my enemy" would suggest that in Figure 2 case (b), one should give a negative recommendation. However, in many cases such as the medical specialist domain, it seems harsh to criticize a doctor on the basis that someone you distrust also likes that doctor. Hence, the two principles that we do recognize are: "the friend of my friend is my friend," and, "the enemy of my friend is my enemy." Other principles which we adopt are that equal amounts of trust and distrust should cancel. These principles are formalized through axioms, described later. As mentioned, a random walk is a nice toy model of how one might ask for recommendations: you ask a random friend, and if he doesn't have an opinion, then he asks a random friend, and so forth. A second justification for these systems might be called an *equilibrium* justification based upon trust scores. Namely, imagine each node has a trust score $t_u \geq 0$, which indicates how trusted the node is. One should be able to justify these scores based on the network. One easy way to do this is to have the trust in an agent be the trust-weighted sum of the declared trust from other agents. Formally,

$$t_s = 1, t_u = \sum_v t_v \cdot w_{vu} \forall u \neq s, \text{and recommendation is sign}(\sum_{u \in V_+} t_u - \sum_{u \in V_-} t_u) \quad (1)$$

The trust scores represent self-consistent beliefs about how trusted each node should be.

1.2 Prior Work on Distrust Propagation

We first argue why existing work fails to achieve our goals in the simple examples in Figure 2. Of course, each of these algorithms may have other appealing properties that our algorithm lacks, and may be more appropriate for other applications. In seminal work, Guha et al. [2] consider a number of different approaches to propagating distrust. None of them matches our behavior on simple examples like those of Figure 2. The first approach is to ignore distrust edges. (A similar approach is taken by the Eigentrust algorithm [5], where any node u which has more negative than positive interactions with a node v has the edge weight w_{uv} rounded up to 0). Of course, this approach fails to handle example 2a. Their second approach is to consider distrust at the end: after propagating trust only as in the first approach, a certain amount of distrust is "subtracted off," afterwards, from the nodes based on the steady-state trust-only scores. This is a reasonable and practical suggestion, and does agree with our behavior on the three examples in Figure 2. However, on the simple example in Figure 3 case (a), it would provide a neutral recommendation to s, since the distrust comes "too late," after equal amounts of trust have already been propagated to both voters.

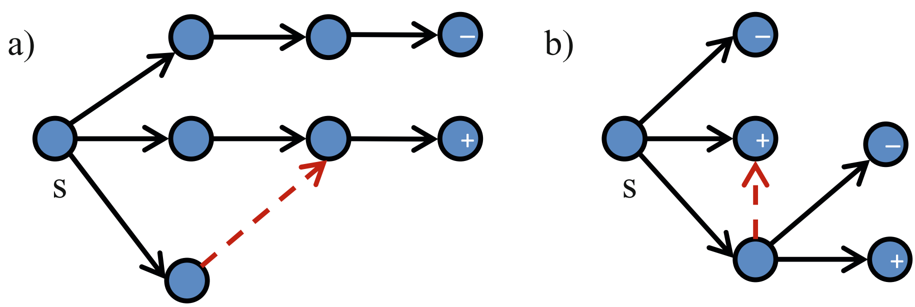


Fig. 3. Two further simple examples where our recommendation to s is $-$

The third approach of Guha et al. is to model distrust by negative weights so that $w_{vu} \in [-1, 1]$(now $\sum_u |w_{vu}| \leq 1$) and simply use the equations given above without change. This follows the "enemy of my enemy is my friend" dictim and hence has a parity dependence on n in Figure 2 case (c). It also gives a negative recommendation in Figure 2 case(b). Several others have adopted this "Plus-Minus Page Rank" approach. However, Guha et al. also consider more powerful notions of trust propagation via "co-citation," which would be interesting to add into our setting.

The PageTrust system [6], considers a random walk which traverses trust edges but "remembers" the list of all distrusted nodes seen thus far, and makes sure never to visit such a node in the remainder of the walk. Such an algorithm would assign a neutral recommendation to node s in the example of Figure 3 case(b), because of the fact that nodes taking the middle path have necessarily not observed the distrust edge. The differences in recommendations between our system and theirs can be made more dramatic. While the random walk perspective is appealing, it is not obvious how to naturally extend it to distrust because it strongly depends on the order in which nodes are visited.

1.3 Our System

Adopting the "equilibrium trust score" view of the random walk system, we propose the following modification of the Page Rank equations:

$$t_s = 1; t_u = \max(0, \sum_v t_v \cdot w_{vu}), \forall u \neq s; R = \mathrm{sgn}(\sum_{u \in V_+} t_u - \sum_{u \in V_-} t_u) \quad (2)$$

In particular, nobody can have a negative trust score: negative trust becomes 0 trust and hence a node that has a net negative incoming amount of trust is completely ignored: neither its vote nor its outgoing edges affect the system in any manner. This is consistent with our view that distrusted nodes are either to be viewed as adversarial or arbitrary, and they should be ignored. Determining which nodes should be distrusted is one challenge we face.

There are examples that show that the equations we propose need not have a unique solution. However, we show that the trust scores of the voters are unique and hence the recommendation of the system is unique. Note that in the case where $\sum_v |w_{uv}| < 1$ for each node (or equivalently where each node has a positive weight pointing back to s), a simple fixed-point argument implies a unique solution for trust scores. Also, if all weights are nonnegative, then it is straightforward to show that the voters have unique scores, by fixed-point theorems. In the case of trust only, the system is simply the random walk system of Andersen *et al.*, hence it is a proper generalization. In fact, it is somewhat surprising that no one has previously suggested this simple signed generalization of the Page Rank equations. The contributions of this paper are (1) introducing this simple modification and proving that it leads to a unique recommendation, (2) giving an efficient algorithm for computing recommendations, and (3) justifying the system using the axiomatic approach.

2 Notation and Definitions

We follow the notation and definitions of Andersen *et al.* [1] when possible. Since we consider the slightly more general case of a weighted multigraph, a voting network is a partially labeled weighted graph. The nodes represent agents and edges represent either trust or distrust between the agents. A subset of nodes, called voters, is labeled with votes of + or −. The remaining nodes are nonvoters. The recommendation system must assign, to each source nonvoter, a recommendation in $\{-, 0, +\}$.

Definition 1. *A* **voting network** *is a directed, weighted graph* $G = (N, V_+, V_-, E)$ *where* N *is a set of nodes,* $V_+, V_- \subseteq N$ *are disjoint subsets of positive and negative voters,* E *is a multiset of weighted edges over the underlying set* $N \times N \times (\mathbb{R} \setminus \{0\})$. *(Parallel edges and self-loops are allowed.) For each node* u*, the total outgoing weight magnitude is at most 1,* $\sum_{e=(u,v,w)} |w| \leq 1$.

An edge $e = (u, v, w)$ from u to v, of weight $w > 0$, indicates that u allocates a w fraction of its *say* to trusting v, and we say u trusts v. If $w < 0$, then u allocates a $-w$ fraction of its *say* to distrusting v. Since we allow parallel edges, technically an agent may trust and distrust someone, and these may cancel as covered in Axiom 1. We also allow self-loops, i.e., *self-trust*, which is covered in Axiom 5. We say that edge (u, v, w) is a **trust (distrust) edge** if w is positive (negative). We say that there is a **trust path** from u to v if there is a sequence of edges $(u, a_1, w_1), (a_1, a_2, w_2), \ldots, (a_k, v, w_k) \in E$ such that $w_i > 0$. A **path** (or u **can reach** v) exists if such a sequence exists regardless of the sign of w_i.

We denote the weight of an edge by $\omega(e)$, so that $\omega(u, v, w) = w$. With slight abuse of notation, we also define $\omega_{uv} = \sum_{(u,v,w)\in E} w$ to be the **total weight** from u to v. A node u is a **sink** if $\omega_{uv} = 0$ for all $v \in N$, and u is a **partial sink** if $\sum_v |\omega_{uv}| < 1$.

A weight of 1 from u to v means that u places complete trust in v and only v, while a weight of -1 indicates that u's influence is completely invested in diminishing v's trustworthiness. The bound on the total absolute value weight leaving a node u bounds u's total (dis)trust expressed in the system. Note that we allow the total to be strictly less than 1. This allows an agent who, for example, does not know many people on the network, to express limited (dis)trust in some agents without expressing complete trust. How much node u trusts another node v is a relative measure which is defined with respect to how much u trusts other nodes. For each node u, the weights on the outgoing edges show the fraction of his trust that u puts on each of its neighbors via the outgoing edges. As a result the total sum of these fractions should not be more than 1.

Let $n = |N|$ be the number of nodes. We denote by $V = V_+ \cup V_-$ the set of **voters** and $V^c = N \setminus V$ the set of **nonvoters**. We write $(N, V_\pm, E)$ as syntactic shorthand for (N, V_+, V_-, E).

Definition 2. *A* **recommendation system** R *takes as input a voting network* G *and* **source** $s \in V^c$ *and outputs* **recommendation** $R(G, s) \in \{-, 0, +\}$.

We denote by $\text{sgn} : \mathbb{R} \to \{-, 0, +\}$ the function that computes the sign of its input. We denote by $(x)_+$ the nonnegative part of x, i.e., $(x)_+ = \max\{0, x\}$.

3 The System

In addition to outputting the final recommendation $r \in \{-, 0, +\}$, the system outputs further information $r_+, r_- \in [0, 1]$. These numbers indicate the fraction of personalized recommendations that were positive and negative. The summary $r = \text{sgn}(r_+ - r_-)$ may be sufficient information, but it may also be useful to know the magnitude of the recommendations. For example, a neutral recommendation $r = 0$ may be arrived at because essentially no trusted parties voted, or because $r_+ = r_- = 1/2$, meaning that all trust resolved in votes, half positive and half negative. This distinction may be useful. Note that $r_+ + r_- \leq 1$.

The main idea behind the system is to compute trust scores for each node u, $t_u \geq 0$. The source, s, is defined to have trust $t_s = 1$. For the remaining $u \neq s$, the following equation must be satisfied.

$$t_u = \left(\sum\nolimits_{v \in N} \omega_{vu} t_v\right)_+ \tag{3}$$

For real $z > 0$, recall that $(z)_+ = z$, and $(z)_+ = 0$ otherwise. A trust score of 0 means a node is mistrusted. The above equations determine a unique finite value t_v for any voter v (and more generally, any node v which can reach a voter). The remainder of the nodes are irrelevant for the recommendation. The trust in positive voters is $r_+ = \sum_{v \in V_+} t_v$ and similarly $r_- = \sum_{v \in V_-} t_v$. The final recommendation is $r = \text{sgn}(r_+ - r_-)$.

Input: $G = (N, V_\pm, E)$, $s \in V^c$.
Output: positive and negative fractions $r_+, r_- \in [0, 1]$, and overall recommendation, $r \in \{-, 0, +\}$.
1. For each voter $v \in V$, remove all outgoing edges. Also remove all incoming edges to s. This makes $\omega_{vu} = 0$ and $\omega_{us} = 0$, for all $u \in N, v \in V$.
2. Remove all nodes $v \in N \setminus \{s\}$ which have no path to a voter, and remove the associated edges. That is, $N = \{s\} \cup V \cup \{u \in N \mid \exists \text{ path } e_1, \ldots, e_k \in E \text{ from } u \text{ to a voter } v\}$.
3. Solve for $x \in \mathbb{R}^N$: $x_u = 1 + \sum_v |\omega_{uv}| x_v$ for all $u \in N$.(x_u represents the expected number of steps until a walk halts when started at node u considering $|\omega_{uv}|$ as the probability of going from u to v.)
4. Solve the following linear program for t:

$$\begin{aligned} \text{minimize } \sum_u x_u \left(t_u - \sum_v \omega_{vu} t_v\right) & \text{ subject to,} \\ t_s &= 1 \\ t_u &\geq 0 \text{ for all } u \in N \\ t_u - \sum_v \omega_{vu} t_v &\geq 0 \text{ for all } u \in N \end{aligned} \tag{4}$$

5. Output $r_+ = \sum_{v \in V_+} t_v$, $r_- = \sum_{v \in V_-} t_v$, and final **recommendation** $r = \text{sgn}(r_+ - r_-)$.

Fig. 4. Abstract description of the system. Recall that $V = V_+ \cup V_-$ is the set of voters. Lemma 2 shows that (4) gives a solution to (3), and Lemma 1 shows uniqueness.

In the case of both positive and negative weights, it is not obvious whether equations (3) have any solution, one solution, or possibly multiple solutions. Moreover, it is not clear how to compute such a solution. The remainder of this section shows that they do in fact have a unique solution, which the algorithm of Figure 4 computes. It is clear that the algorithm is polynomial-time because solving linear programs can be achieved in polynomial time.

Lemma 1. *Let $G = (N, V_\pm, w)$ be a weighted voting network in which there is a path from each node to a partial sink. Then there is a unique solution $t \in \mathbb{R}^N$ satisfying $t_s = 1$ and, for each $u \neq s$, $t_u = \left(\sum_v \omega_{vu} t_v\right)_+$.*

In order to prove Lemma 1, it is helpful to understand self-loops, $\omega_{uu} \neq 0$. Any self loop can be removed with very simple changes to the trust scores.

Observation 1. *Let N be a set of nodes, and let $\overline{w} \in \mathbb{R}^{N^2}$ be such that $\sum_v |\overline{w}_{uv}| \leq 1$ and $\overline{w}_{uu} \in (-1, 1)$ for each u. Let $w \in \mathbb{R}^{N^2}$ be defined so that $\omega_{uv} = \overline{w}_{uv}/(1 - \overline{w}_{uu})$ and $\omega_{uu} = 0$ for all $u \neq v$. Then $t \in \mathbb{R}^N$ satisfies $t_u = (\sum_v \omega_{vu} t_v)_+$ if and only if the vector $\overline{t} \in \mathbb{R}^N$, where $\overline{t}_u = t_u/(1 - \overline{w}_{uu})$ for each u, satisfies $\overline{t}_u = \left(\sum_v \overline{w}_{vu}\overline{t}_v\right)_+$.*

Proof. We have that,

$$
\begin{aligned}
t_u &= \left(\sum_v \omega_{vu} t_v\right)_+ \Leftrightarrow \\
\overline{t}_u(1 - \overline{w}_{uu}) &= \left(\sum_{v \neq u} \frac{\overline{w}_{vu}}{1 - \overline{w}_{vv}} \cdot \overline{t}_v(1 - \overline{w}_{vv})\right)_+ \Leftrightarrow \\
\overline{t}_u(1 - \overline{w}_{uu}) &= \left(-\overline{w}_{uu}\overline{t}_u + \sum_v \overline{w}_{vu}\overline{t}_v\right)_+ \Leftrightarrow \\
\overline{t}_u &= \left(\sum_v \overline{w}_{vu}\overline{t}_v\right)_+ .
\end{aligned}
$$

The last two equalities are equivalent by considering two cases: $\overline{t}_u = 0$ and $\overline{t}_u > 0$. The first case is trivial, and the second case follows from the fact that $\overline{t}_u > 0$ iff $\sum_{v \neq u} \overline{w}_{vu}\overline{t}_v > 0$.

Proof ((Lemma 1)). By Observation 1, all self loops may first be removed. The proof is by induction on $n = |N|$. In the case where $n = 1$ or $n = 2$, there is trivially a unique solution. Now, consider $n > 2$. Suppose for the sake of contradiction that there are two different solutions, t and t'. Consider three cases.

Case 1. There is some node u such that $t_u = t'_u = 0$. Imagine removing u (and its associated edges) from the graph. The graph will still have the property that there is a path from each node to a partial sink, because if such a path formerly passed through u, then the node linking to u is now a partial sink. By induction hypothesis, the resulting graph has a unique solution, $\overline{t}$. However, it is easy to see that the solutions t and t' restricted to $N \setminus \{u\}$ will both remain solutions to the equations of the lemma statement. This is a contradiction because t and t' agree on $N \setminus \{u\}$ and on u as well.

Case 2. There is some node $u \neq s$ such that $t_u > 0$ and $t'_u > 0$. Similarly, we will remove the node and apply the induction hypothesis. However, in this case, when we remove the node, we will propagate trust through the node as follows. We consider the graph $\overline{G}$ with nodes $\overline{N} = N \setminus \{u\}$ and weights $\overline{w}_{vw} = \omega_{vw} + \omega_{vu}\omega_{uw}$. Note that this transformation preserves $\sum_v |\overline{w}_{wv}| \leq 1$ but does not necessarily preserve $\omega_{ww} = 0$ for each w.

We now (tediously) argue that, in $\overline{G}$, every node can reach a partial sink. In G, consider a path $u_1, u_2, \ldots, u_k$ to a partial sink u_k. Now, one difficulty is that some edge which has $\omega_{u_i u_{i+1}}$ may have $\overline{w}_{u_i u_{i+1}} = 0$. However, if this happens, we claim that u_i must be a partial sink in $\overline{G}$. To see this, consider $|w| \in \mathbb{R}_+^N$ to be the vector where $|w|_v = |w_v|$ for all v. If we had used the weights $|w|$ instead of w in G, then after propagation it would still have been the case that the sum of the weights leaving u_i is at most 1. In the signed version, after propagation the weight magnitudes are only smaller, and the weight magnitude of $\omega_{u_i u_{i+1}}$ is strictly smaller, so u_i must be a partial sink. Hence, the "zeroing out" of edges does not create a problem in terms of removing a path from a node to partial sink. Similarly, if $u = u_i$ for some i, then either the path to the partial sink remains (deleting u_i) after propagating of trust or the node u_{i-1} must have become a partial sink.

We would like to apply the induction hypothesis to $\overline{G}$. However, this graph may have self-loops and therefor is not a valid voting network. By Observation 1, though, we can remove any self loops and change the solution to the equations of the lemma by a simple predictive scalar. Hence, by induction hypothesis, the resulting graph has a unique solution, $\overline{t}$. However, it is not difficult to see that the solutions t and t' restricted to $N \setminus \{u\}$ will both remain solutions to the equations of the lemma statement. This follows because the new equations of the lemma are simply the same as the old equations along with the substitution $t_u = \sum_v \omega_{vu} t_v$, which holds in both t and t' since both $t_u, t'_u > 0$. However, since they both agree with $\overline{t}$ on $v \neq u$ and satisfy $t_u = \sum_v \omega_{vu} t_v$, they must be the same which is a contradiction.

Case 3. For each node $u \neq s$, $\text{sgn}(t_u) \neq \text{sgn}(t'_u)$

Case 3a. There are at least two nodes $u, v \neq s$ for which $\text{sgn}(t_u) = \text{sgn}(t_v)$. The idea is to (carefully) merge nodes and use induction. WLOG, say $t_u, t_v > 0$ and $t'_u = t'_v = 0$. Now, we consider merging the two nodes into one node a. That is consider a new graph with node set $\overline{N} = \{a\} \cup N \setminus \{u, v\}$ and weights the same as in G except, $\overline{w}_{xa} = \omega_{xu} + \omega_{xv}$ for each $x \in N \setminus \{u, v\}$, $\overline{w}_{ax} = (t_u \omega_{ux} + t_v \omega_{vx})/(t_u + t_v)$. It is relatively easy to see that if u or v is a partial sink, then so is a. Consider the scores $\overline{t}, \overline{t}' \in \mathbb{R}^{\overline{N}}$ which are identical to t and t' except that $\overline{t}_a = t_u + t_v$ and $\overline{t}'_a = t'_u + t'_v = 0$. It is also relatively easy to see that the conditions of the lemma are satisfied by both of these scores. However, by induction hypothesis (again we must first remove self-loops, as above, using Observation 1), this again means that $\overline{t}' = \overline{t}$, which contradicts $\overline{t} \neq \overline{t}'$.

Case 3b. There are exactly three nodes s, u, v, and $t_u = t'_v = 0$, $t'_u, t_v > 0$. If we remove all the outgoing edges of u, t should still be a valid scoring for the new network. Since we have only 3 vertices, the only incoming edge to v is from s and as a result, t_v should be equal to ω_{sv}. With the same argument, we have $t'_u = \omega_{su}$. Since $t'_v, t_u = 0$, we can conclude that:

$$\omega_{su} \cdot \omega_{uv} + \omega_{sv} \leq 0$$
$$\omega_{sv} \cdot \omega_{vu} + \omega_{su} \leq 0$$

Since $|w_e| \leq 1$, it is not hard to argue that the only way to satisfy both these equations, is to have: $\omega_{su} = \omega_{sv} > 0$ and $\omega_{uv} = \omega_{vu} = -1$. But this means that u and v are not partial sinks and also they don't have a path to a partial sink since all their weight is pointed to the other one.

Lemma 2. *The solution to step 4 of the algorithm in Figure 4 is also a solution to eq. (3).*

Proof. We first claim that the equations in step 3 have a unique (finite) solution. To see this, consider a discrete random walk which, when at node u, proceeds to each node v with probability $|\omega_{uv}|$, and with probability $1 - \sum_v |\omega_{uv}|$ halts. Hence, once the walk reaches a voter, it halts on the next step. The equations for x_u in step 3(a) represent the expected number of steps until the walk halts when started at node u. Hence there is a unique solution, and clearly $x_u \geq 1$. To see that x_u is not infinite, note that each node has a path to some voter. Let $w_{\min} = \min_{uv:\omega_{uv} \neq 0} |\omega_{uv}|$ denote the smallest nonzero weight. Then, starting at any node, after n steps, there is a probability of at least $w_{\min}^n$ of reaching a voter within n steps. Hence, the expected number of steps, i.e., "time to live," is at most $n/w_{\min}^n < \infty$.

Consider a third optimization problem.

$$\text{minimize} \sum_u x_u \left| t_u - \sum_v \omega_{vu} t_v \right| \text{ subject to:} \qquad t_s = 1, \quad \forall u: \ t_u \geq 0 \qquad (5)$$

We first claim that the solution to the above optimization problem is also a solution to eq. (3). Suppose for contradiction that the solution to (5) has $t_u - (\sum_v \omega_{vu} t_v)_+ = \Delta \neq 0$ for some node u. Next we claim that changing t_u to $t_u - \Delta$ will decrease the objective by at least $|\Delta|$. The term in the objective corresponding to u decreases by $x_u|\Delta|$. (That term will become 0 unless $\sum_v \omega_{vu} t_v < 0$, but it will still decrease by Δ nonetheless.) The term in the objective corresponding to any other v can increase by at most $x_v|\omega_{uv}\Delta|$. Hence the total decrease is at least $|\Delta|(x_u - \sum_v x_v|\omega_{uv}|) = |\Delta|$, by definition of x_u.

Next, note that (5) is a relaxation of (4) in the sense that: (a) the feasible set of (5) contains the feasible set of (4), and (b) the objective functions are exactly the same on the feasible set of (4). Also notice that a solution to (3) is a feasible point for (4). Hence, the fact that the optimum of (5) is a solution to (3) implies that any solution to (4) is also a solution to (3).

The above two lemmas show that the output of the system is unique.

4 Axioms

The following axioms are imposed upon a recommendation system, R. Two edges, $e = (u, v, w)$ and $e' = (u, v, w')$ are *parallel* if they connect the same pair of nodes (in the same direction). The first axiom states that parallel edges can be merged, if they are the same sign, or *canceled* if they have opposite signs.

Axiom 1 (Cancel/merge parallel edges). *Let $G = (N, V_\pm, E)$ be a voting network. Let $e_1 = (u, v, w_1)$, $e_2 = (u, v, w_2)$ be parallel edges. If the two edges have opposite weights, $w_1 + w_2 = 0$, then the two edges can be removed without changing the recommendation of the system. Formally, let $G' = (N, V_\pm, E \setminus \{e_1, e_2\})$. Then $R(G, s) = R(G', s)$ for all $s \in N$. If the two edges have the same sign, then the two edges can be removed and replaced by a single edge of weight $w_1 + w_2$ without changing the recommendation of the system. Formally, let $G' = (N, V_\pm, \{(u, v, w_1 + w_2)\} \cup E \setminus \{e_1, e_2\})$. Then $R(G, s) = R(G', s)$ for all $s \in N$.*

It is worth noting that this axiom (and in fact almost all of our axioms), may be used *in reverse*. For example, rather than merging two edges of the same sign, a single edge may be *split* into two edges of the same sign and same total weight, without changing the system's recommendation.

Along these lines, an easy corollary of the above axiom is that, if $\omega(e_1) + \omega(e_2) \neq 0$, then the two edges can be merged into a single edge of weight $\omega(e_1) + \omega(e_2)$, regardless of sign.

For clarity of exposition, in the further axioms, we will make statements like " changing the graph in this way does not change the systems recommendation." Formally, this means that if one considered the different graph G' changed in the prescribed manner, then $R(G, s) = R(G', s)$.

Axiom 2 (Independence of irrelevant/distrusted stuff). *For voting network $G = (N, V_\pm, E)$,*

1. Let $s \in N$. Removing an incoming edge to s doesn't change the recommendation of the system for s. Similarly, removing outgoing edges from voters doesn't change the recommendation of the system.

2. Let $u \in N$ be a node which is not reachable from s by a path of trust edges. Then removing u (and its associated edges) doesn't change the recommendation of the system for s.

3. Let $u \in N$ be a nonvoter which has no path to a voter (through any type of edges). Then removing u (and its associated edges) doesn't change the recommendation of the system for s.

The first part simply states that it doesn't matter who trusts or distrusts the source, since the source is the one looking for a recommendation. Note that the second part implies if a node has only distrust edges pointing to it, then the node is ignored as it may be removed from the system. This crucial point can also be viewed as a statement about *manipulability.* In particular, if a node is in no manner trusted, it should not be able to influence the recommendation in any manner. For the third part, if a node and all of its trust can never reach a voter, then it is irrelevant whom the edge trusts.

Axiom 3 (Cancelation). *Consider the voting network $G = (N, V_\pm, E)$ with two edges e_1, e_2, one trust and one distrust, that terminate in the same node. Suppose that there is a constant $c \in \{-, 0, +\}$ such that for any of the following modifications to G, the system's recommendation is c: (1) the two edges are both* *removed; (2) the endpoint of two edges are both redirected to a new positive voter (with no other incoming edges) without changing any weights; (3) the endpoint of two edges are both redirected to a new negative voter (with no other incoming edges) without changing any weights; and (4) these two edges are both redirected to a new voter node (whose vote may be positive or negative), and the weights of edges are both* negated. *Then the recommendation of the system (without modification) is c.*

Note that the stronger the conditions on when a pair of edges may be canceled, the weaker (and better) the above axiom is. The above axiom states that, if a pair of edges wouldn't change the system's recommendation if they were directed towards a positive or negative voter, or negated, then they cancel or are at least irrelevant. In regards to this axiom, we also would like to mention the possibility of axiomatizing the system with real-valued output $r \in \mathbb{R}$ (we chose discrete recommendations for simplicity). In this latter case, the conditions under which a pair of edges is said to cancel would be even stronger, since the recommendation must not change by any amount, arbitrarily small. The same axiomatization we have given, with the real-valued variant of the last axiom, Axiom 6, again uniquely implies our recommendation system (now with final recommendation $r = r_+ - r_-$ instead of $r = \text{sgn}(r_+ - r_-)$).

Axiom 4 (Trust Propagation). *Let $G = (N, V_\pm, E)$ be a network of trust and distrust edges. Consider nonvoter $u \neq s$ with no incoming distrust edges and no self-loop. Let $e = (u, v, w)$ be a trust edge $(w > 0)$. We can remove edge e and make the following modifications, without changing the recommendation $R(G, s)$:*

- *Renormalizaiton: replace each other edge leaving u, $e' = (u, v', w')$ by $(u, v', w'/(1 - w))$.*
- *Propagation: For each edge $e' = (x, u, w')$, replace this edge by two edges, $(x, u, w'(1 - w))$ and $(x, v, w' \cdot w)$ directly from x to v.*

The above axiom, is somewhat subtle. The basic idea behind trust propagation is that an edge from u to v of weight $w > 0$ together with an edge from x to u of weight $w' > 0$ are equivalent to an edge from x to v of weight $w' \cdot w$, because x assigns a w' fraction of its trust in u and u assigns as w fraction of its trust in v. Andersen *et al.* [1] perform this propagation by removing edge (x, u, w) and adding in the direct edge $(x, v, w \cdot w')$ and, of course, when they remove (x, u, w) they must add a new edge for each outgoing edge from u (actually, their graphs are unweighted so it is slightly different).

The additional difficulty we face, however, is that u may have outgoing distrust edges, which we do not want to specify how to propagate. Instead, we *peel off* the trust going from u to v, and replace it directly by an edge from x to v. However, since the edge leaving u only accounts for a w fraction of the total possible trust assigned by u, the remaining edges must remain in tact and rescaled. While the above axiom is admittedly involved, it is satisfied by the Random Walk system of Andersen *et al.*, and is very much in line with the notion of propagation in a random walk. It has the advantage that it provides a description of how to (carefully) propagate trust in the midst of distrust edges. We also note that a simpler axiom is possible here, more in the spirit of Andersen *et al.*, in which trust-distrust pairs can be propagated (e.g., "an enemy of my friend is my enemy").

Axiom 5 (Self trust). *Let $G = (N, V_\pm, E)$ be a voting network. Let u be a nonvoter vertex in N and e be an edge from u to itself of weight $\omega(e) > 0$. Then e can be removed and the weights of all other edges leaving u scaled by a multiplicative factor of $1/(1-\omega(e))$, without changing the recommendation of the system.*

Axiom 6 (Weighted majority). *Let $G = (N, V_\pm, E)$ be a star graph centered at $s \notin V$, $N = \{s\} \cup V$, with exactly $|V|$ edges, one trust edge from s to each voter, then the recommendation of the system is* $\operatorname{sgn}\left(\sum_{v\in V_+} \omega_{sv} - \sum_{v\in V_-} \omega_{sv}\right)$.

Note that the above axiom can be further decomposed into more primitive axioms, as is common in social choice theory. For example, Andersen *et al.* [1] use *Symmetry*, *Positive Response*, and *Consensus axioms*, which imply a majority property similar to the above.

5 Analysis

Lemma 3. *The system of Figure 4 satisfies Axioms 1–6.*

In order to prove Lemma 3, we need to show that Equation 3 or Figure 4 satisfies each of the Axioms 3– 6. Because of the lack of space, we only present the proof for Axiom 3. The rest of the proofs can be found in the full version of the paper. In the rest of the proof we assume that $G = (N, V_\pm, E)$ is given and t_v is a valid score for v using Equation (3).

Cancelation: Consider $e_1 = (u_1, v, w_1)$ and $e_2 = (u_2, v, w_2)$ and call the new voter vertex z. First note that the trust scores obtained for any network does not depend on the sign of the votes by voters. As a result, if we direct e_1 and e_2 to a positive or a negative voter, we should obtain the same vector of scores, t', for all the vertices (by uniqueness). Let $G_- = (N, V_+, V_\cup\{z\}, E \cup \{(u_1, z, w_1), (u_2, z, w_2)\} \backslash \{e_1, e_2\})$ when z has a negative vote and G_+ defined the same way but z has a positive vote. It is easy to see that $t'_z = (w_1 \cdot t'_{u_1} + w_2 \cdot t'_{u_2})_+$. We consider 3 cases here:

Case 1. $w_1 \cdot t'_{u_1} + w_2 \cdot t'_{u_2} = 0$.
In this case, t' is a valid set of scores for G as well. By uniqueness, it means that $t = t'$ and as a result, $w_1 \cdot t_{u_1} + w_2 \cdot t_{u_2} = 0$. So if we remove e_1 and e_2,t is a valid solution to the new graph and the recommendation will not change.

Case 2. $w_1 \cdot t'_{u_1} + w_2 \cdot t'_{u_2} = p > 0$.
First, one can prove that, because $p > 0$, $R(G_-, s) \le R(G, s) \le R(G_+, s)$. The idea is as follows: consider the set of scores t' for G. It is easy to see

that all vertices except v satisfy Equation 3. Define the infeasibility value of a vertex v for a given scoring t' by $\iota_{t'}(v) = |\left(\sum_{u\in N}\omega_{uv}t'_u\right)_+ - t'_v|$ and define the potential function $\phi(t') = \sum_{v\in N}\iota_{t'}(v)$.It is also easy to see that $\iota_{t'}(v) \leq p$. Now starting from t', we can reach to a feasible set of scores t as follows: Iteratively find v that has the maximum infeasibility value. Find a path from v to a sink vertex. Update the scores one by one from v to the sink vertex along this path. Call the new set of scores t. Initialize t to t' and update it as follows: First set the score of v such that $\iota_t(v) = 0$. Now, go to the next vertex along the path and based on the new scores of t update the score of the vertex such that its infeasibility value is set to 0. Note that during this score updating, $\phi(t)$ only decreases at each update. Also when we reach to a voter vertex u, if the score has been updating during this process and t'_u is changed to t_u, the sum of infeasibility values will be decreased by at least $|t_u - t'_u|$. As a result, after our iterative procedure converges, the change in the score of the voters $\sum_{v\in V_\pm}|t_v - t'_v| \leq p$ or in other words, it does not exceed the current infeasibility value over all vertices that is not more than p for t'. Now, the recommendation score for G_- is simply $\sum_{v\in V_+} t'_v - \sum_{v\in V_-} t'_v - p$ which is less than or equal to $\sum_{v\in V_+} t_v - \sum_{v\in V_-} t_v$(recommendation score for G) and that is less than or equal to $\sum_{v\in V_+} t'_v - \sum_{v\in V_-} t'_v + p$ which is the recommendation score for G_+.
Since $R(G_-, s) = R(G_+, s)$, we should have $R(G_-, s) = R(G, s) = R(G_+, s)$. Now, consider removing e_1 and e_2 from G(call the new graph G'), first note that t' is a valid set of scores for G' as well. So the recommendation score for G' is $\sum_{v\in V_+} t'_v - \sum_{v\in V_-} t'_v - p \leq \sum_{v\in V_+} t'_v - \sum_{v\in V_-} t'_v \leq \sum_{v\in V_+} t'_v - \sum_{v\in V_-} t'_v + p$. As a result, $R(G_-, s) = R(G, s) = R(G', s) = R(G_+, s)$.
Case 3. $w_1 \cdot t'_{u_1} + w_2 \cdot t'_{u_2} = -p < 0$. Imagine negating the weights of e_1 and e_2 (the axiom is invariant to this operation). Note that t' is still a valid set of scores for the new graph and by uniqueness it is the only feasible set of scores. Now, we have $-w_1 \cdot t'_{u_1} - w_2 \cdot t'_{u_2} = p > 0$ and now, we can argue as above.

The main theorem is the following.

Theorem 1. *The system of Figure 4 is the unique system satisfying Axioms 1–6.*

Proof. Let G be a voting network and $s \in N$ be a node. It suffices to show that the axioms imply that there is at most one value for $R(G, s)$, since we know, by Lemma 3 that the system of Figure 4 satisfies the axioms. The idea of the proof is to apply a sequence of changes to the graph, none of which change the recommendation $R(G, s)$. The sequence will eventually result in star graph to which we can apply Axiom 6.

First, we simplify the graph by Axiom 2: eliminating all edges to s, all edges leaving voters, and all nodes that are not reachable by trust edges from s, as well as all nodes that have no path (trust/distrust) to any voter. We then apply axiom 5 to remove any trust self-loops. We finally apply axiom 1 to merge all parallel edges.

In the body of this proof (this part), we change the graph so that there are no distrust edges pointing to nonvoters. Lemma 4, following this proof, then implies the theorem. We proceed by induction on the total number of edges to nonvoters (trust or distrust), call this k. The induction hypothesis is that, if there are at most $k-1$ edges to nonvoters, then there is a unique recommendation for the system. Say $k \geq 1$. If there are no distrust edges between nonvoters, then we are done by Lemma 4. Otherwise, let (u, v) be a distrust edge between $u, v \in V^c$ (possibly $u = v$). If $v = s$, we can apply Axiom 2 to eliminate the edge, and then use the induction hypothesis. Now there must be at least one trust edge from a nonvoter, say a, to v, otherwise v would have been eliminated. Now, imagine running the system through the algorithm of Figure 4. The simplifications we have already executed mean that no edges or nodes will be removed during Steps 1 or 2. Step 4 assigns a unique trust score to each node in the system. Now, the plan is to eliminate either edge (a, v) or edge (u, v) (or both).

Consider three cases.

Case 1: $t_a\omega_{av} + t_u\omega_{uv} = 0$. In this case, we will argue that Axiom 3 implies a unique recommendation. The reason is as follows. Consider any modification of the graph in which these two edges have been moved (and possibly negated) to a new voter. By induction hypothesis, since there are now $\leq k-1$ edges to nonvoters, the recommendation on this graph is unique and hence is equal to the recommendation given by the system of Figure 4. Also note that this system computes the unique solution to the equations (3). However, note that the same solution vector t satisfies equations in the modified graph, because moving both edges causes a change by an amount $t_a\omega_{av} + t_u\omega_{uv} = 0$ in the right hand side of any of these equations, regardless of where the edges are moved to, or even if they are both negated. Hence, the recommendation of the system is identical regardless of which voter they are moved to. Axiom 3 thus implies that there is a unique recommendation for the system.

Case 2: $t_a\omega_{av} + t_u\omega_{uv} > 0$. In this case, we can use Axiom 1 *in reverse* to split the edge (a, v) into two edges, one of weight $-t_u\omega_{uv}/t_a$ and the other of weight $\omega_{av} + t_u\omega_{uv}/t_a$. Now we can apply the same argument above on the edge (u, v) and the first of these two edges.

Case 3: $t_a\omega_{av} + t_u\omega_{uv} < 0$. In this case, we will split edge (u, v) into two, and move one of its part and edge (a, v) to a voter, thus again decreasing the total number of edges to non-voters by one. This time, we use Axiom 1 in reverse to split the edge (u, v) into two edges, one of weight $-t_a\omega_{av}/t_u$ and the other of weight $\omega_{uv} + t_a\omega_{av}/t_u$. Now, exactly as above, the pair of edges (a, v) and the (u, v) edge of weight $\omega_{uv} + t_a\omega_{av}/t_u$ exactly cancel in the system. Exactly as argued above, the recommendation of the system must be unique no matter which voter they are directed towards. Therefore, the recommendation of the system is unique.

Lemma 4. *Let $G = (N, V_{\pm}, E)$ be a voting network, and let $s \in N$. Suppose, further, that all negative edges point to voters, i.e., $\omega_{uv} \geq 0$ for all $u \in N$ and* *nonvoters $v \in V^c$. Let R be a recommendation system that satisfies Axioms Axiom 1–Axiom 6. Then there is one unique possible value for $R(G, s)$.*

Proof. First apply Axiom 2 so that the source has no incoming links. Let $S = V^c \setminus \{s\}$ be the set of nonvoters, excluding the source. In the first part of the proof, we will simplify the graph so that there are *no links between members of* S, and that all negative edges point to voters. To do this, we will repeatedly apply propagation and self-propagation, as follows. Choose any ordering on the nonvoters (besides s), $\{u_1, u_2, \ldots, u_k\} = V^c \setminus \{s\}$. In turn, for each $i = 1, 2, \ldots, k$, we will remove all links from u_i to S. First, if u_i contains a self-loop, we apply Axiom 5 to remove it. Next, for $j = 1, 2, \ldots, k$, if there is a trust edge from u_i to u_j, then we remove it using Axiom 4. This will remove all outgoing trust edges from u_i to S. The key observation here is that once all trust edges are removed from u_i to S, further steps in the above iteration will not create any edges from u_i to S. Hence, after iteration $i = k$, there will be no edges between members of S. It remains the case that all negative edges point to voters.

The graph now has some number of trust links from s to S, as well as trust and distrust links from $V^c = S \cup \{s\}$ to V. We now propagate all trust links from S to V^c, using Axiom 4. As a result, the only edges are trust links from s to $N \setminus \{s\}$, distrust links from S to V, and distrust links from s to V. We further simplify by merging any parallel edges (Axiom 1). We now proceed to remove all distrust edges. First, consider any distrust edge from s to voter $v \in V$. Since we have merged parallel edges, there cannot be any trust edges from s to v, and we have already altered the graph so that there are no other trust edges to v. Hence, by Axiom 2, we can remove v (and the edge from s to v) entirely. Next, consider any distrust edge from $u \in S$ to $v \in V$. If there is no trust edge from s to v, then again by Axiom 2, we can remove v. If there is no trust edge from s to u, then we can remove u and edge (u, v). Otherwise, there is a distrust edge from u to v and trust edges from s to u and from s to v.

Consider three cases.

Case 1: $\omega_{sv} + \omega_{su}\omega_{uv} = 0$. Now we will completely eliminate the distrust edge. We apply Axiom 4 *in reverse* to the edge from s to v, to create a trust edge from u to v and increase the weight of the edge from s to u. Now, we merge the parallel edges from u to v using Axiom 1. A simple calculation shows that the trust and distrust will exactly cancel, and no edge will remain. Hence, we have eliminated the negative edge (u, v) without creating further negative edges.

Case 2: $\omega_{sv} + \omega_{su}\omega_{uv} > 0$. Then we apply Axiom 1 *in reverse* to split edge (s, v) into two edges, one of weight $-\omega_{su}\omega_{uv} > 0$ and one of weight $\omega_{sv} + \omega_{su}\omega_{uv} > 0$. We then proceed as in Case 1 to cancel the edge from u to v with the edge from s to u of weight $-\omega_{su}\omega_{uv}$.

Case 3: $\omega_{sv} + \omega_{su}\omega_{uv} < 0$. In this case, we eliminate the trust edge from s to v, as follows. We again apply Axiom 4 in reverse to the edge from s to v, again to create a trust edge from u to v and increase the weight of the edge from s to u. Then we merge the parallel edges from u to v using Axiom 1 (in fact, we must first split the negative edge from u to v into two parts so that one may cancel the trust edge). What remains is a distrust edge from u to v, and there is no longer any trust in v. Hence, we can finally remove the node v and the associated distrust edge. Thus, in all three cases, we were able to eliminate the distrust edge without creating a new distrust edge, maintaining the special structure of the graph.

After eliminating all distrust edges, we remain with trust edges from s to $S \cup V$. By Axiom 2, we can eliminate all edges in S and any edges in V that do not have trust coming from S. Finally, we can apply Axiom 6 to get that the recommendation of the system is unique.

6 Conclusion

In conclusion, we have suggested a simple set of principles to address trust and *distrust* in recommendation systems. A guiding principle, which is apparent in Axiom 2, is that of non-manipulability by untrusted agents. This is apparent in the design of our system and axioms, and also in the features we did not include. For example, it is also natural to consider a notion of *co-trust*, which may be described as follows. Consider two agents that trust the same people. They may be viewed as similar. If one of them then votes, say, positively then a system might be inclined to give a positive recommendation to the other [2]. However, such systems would tend to be manipulable in the sense that an adversarial agent, who is completed distrusted, could influence recommendations.

Without getting into interpretation of what distrust means (such as the difference between distrust and mistrust), the key beliefs we ascribe to are that distrust should be able to cancel with trust, and that distrusted agents should be ignored. Combined with trust propagation, this gives a unique, simple recommendation system.

References

1. Andersen, R., Borgs, C., Chayes, J., Feige, U., Flaxman, A., Kalai, A., Mirrokni, V., Tennenholtz, M.: Trust-based recommendation systems: an axiomatic approach. In: WWW 2008: Proceeding of the 17th International Conference on World Wide Web, pp. 199–208 (2008)
2. Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: Proc. Int. Conf. on World Wide Web, pp. 403–412 (2004)
3. Haveliwala, T.: Topic-sensitive pagerank. In: Proc. Int. Conf. on World Wide Web, pp. 517–526 (2002)
4. Heider, F.: The Psychology of Interpersonal Relations. J. Wiley & Sons, New York (1958)
5. Kamvar, S., Schlosser, M., Garcia-Molina, H.: The EigenTrust algorithm for reputation management in P2P networks. In: Proc. Int. Conf. on World Wide Web, pp. 640–651 (2003)
6. de Kerchove, C., Dooren, P.: The Page Trust algorithm: how to rank web pages when negative links are allowed? In: Proc. SIAM Int. Conf. on Data Mining, pp. 346–352 (2008)

7. Kleinberg, J.: Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM) 46(5), 604–632 (1999)
8. Kunegis, J., Lommatzsch, A., Bauckhage, C.: The slashdot zoo: mining a social network with negative edges. In: WWW 2009: Proceedings of the 18th International Conference on World Wide Web, pp. 741–750 (2009)
9. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical Report, Stanford University (1998)

Incentives in Online Auctions via Linear Programming

Niv Buchbinder[1], Kamal Jain[2], and Mohit Singh[3]

[1] Open University, Israel
niv.buchbinder@gmail.com*
[2] Microsoft Research, Redmond, WA, USA
kamalj@microsoft.com
[3] McGill University, Montreal
mohitsinghr@gmail.com**

Abstract. Online auctions in which items are sold in an online fashion with little knowledge about future bids are common in the internet environment. We study here a problem in which an auctioneer would like to sell a single item, say a car. A bidder may make a bid for the item at any time but expects an immediate irrevocable decision. The goal of the auctioneer is to maximize her revenue in this uncertain environment. Under some reasonable assumptions, it has been observed that the online auction problem has strong connections to the classical secretary problem in which an employer would like to choose the best candidate among n competing candidates [HKP04]. However, a direct application of the algorithms for the secretary problem to online auctions leads to undesirable consequences since these algorithms do not give a fair chance to every candidate and candidates arriving early in the process have an incentive to delay their arrival.

In this work we study the issue of incentives in the online auction problem where bidders are allowed to change their arrival time if it benefits them. We derive incentive compatible mechanisms where the best strategy for each bidder is to first truthfully arrive at their assigned time and then truthfully reveal their valuation. Using the linear programming technique introduced in Buchbinder et al [BJS10], we first develop new mechanisms for a variant of the secretary problem. We then show that the new mechanisms for the secretary problem can be used as a building block for a family of incentive compatible mechanisms for the online auction problem which perform well under different performance criteria. In particular, we design a mechanism for the online auction problem which is incentive compatible and is $3/16 \approx 0.187$-competitive for revenue, and a (different) mechanism that is $\frac{1}{2\sqrt{e}} \approx 0.303$-competitive for efficiency.

1 Introduction

Online auctions in which items are sold in an online fashion with little knowledge about future bids are common in the modern environment. Consider a problem in

* This work was done while at Microsoft Research, New England.
** Part of the work done while at Microsoft Research, New England.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 106–117, 2010.

which a seller would like to put his car, a Honda civic in an excellent condition, on an auction[1]. As a first step he publishes an advertisement for the car, and defines a time frame for the sale. Assume that at future time t a potential buyer reads the advertisement, and would like to participate in the auction. The potential buyer has her value v_i for the car. However, her knowledge about the values of other potential buyers is very limited. Therefore, a reasonable assumption for her is that other buyers evaluating the car similarly to her. In particular, she believes that in a random subset of k potential buyers her value is the highest with probability $1/k$. Based on her beliefs she may now choose to arrive at any time $t' \geq t$ and then report some value v_i', possibly different than v_i if it benefits her.

Consider next the seller side of the story. The seller's knowledge about values of the potential buyers is also very limited. In particular, different people may value his Honda civic very differently. A natural model that captures such limited knowledge is an adversarial setting in which the set of values buyers have for the car are chosen arbitrarily, but that the arrival times of the buyers is a random permutation. The seller would like to design a mechanism which is incentive compatible and achieves good performance. In this work, we devise mechanisms for such an auction scenario where for any bidder, bidding truthfully and arriving at their assigned time maximizes its expected profit. Moreover, these mechanisms perform well under the criteria of both efficiency and revenue as compared to the offline VCG mechanism that sells the item to the highest bidder but charges a price of the second highest bidder [Vic61].

1.1 Auction Model

We model the online auction problem as the following mechanism design question. An auctioneer would like to sell a single item to a collection of n bidders $C = \{1, 2, \ldots, n\}$. Each bidder i has an arrival time $a_i \in [0, T]$ and a valuation v_i both of which are private information. The information given to the mechanism is only the number of bidders and the time horizon. The bidder may arrive at any time $t_i \geq a_i$. When the bidder arrives, it bids b_i for the item which may be distinct from her valuation v_i. The mechanism must then make a decision of whether to allocate the good to the bidder and at what price. All allocation decisions are irrevocable. We assume that the utility function for bidder i is the quasilinear function $v_i - p_i$ where p_i is the price faced by bidder i.

Now, we explain how the valuation and the arrival times are selected. First, an adversary chooses a set of arrival times $\{a_1, a_2, \ldots, a_n\}$ and assigns them adversarially. Then it chooses a set of values $\{v_1, v_2, \ldots, v_n\}$ and the values are matched with the arrival times using a random permutation. From the above model, each bidder makes the following reasonable assumption.

Assumption 1. *Each bidder believes that if all the bidders are sorted by their valuations then each permutation of bidders is equally likely.*

[1] The first author of this paper owns a Honda civic 2004 that he would like to sell shortly. The rest of the details may be fictional.

Unconditionally, if all the bidders are sorted by their valuations then each permutation of bidders is equally likely. What the above assumption states is that any bidder, **conditioned on her information**, still believes the above claim. Informally, this means that each bidder believes her valuation (or any other bidder's valuation) is equally likely to be the jth largest valuation for any j. Observe that the assumption is inherently ordinal and we contrast it with typical assumptions in such scenarios where it assumed that valuations are drawn independently from a fixed distribution.

We evaluate any mechanism by two criteria, *efficiency* and *revenue*. We define the outcome of a mechanism to be efficient if it allocates the good to the highest bidder and efficiency of a mechanism to be the probability with which the outcome is efficient. The revenue of a mechanism is defined to be the expected price charged by the mechanism. In the spirit of online algorithms, we compare its performance to the offline VCG mechanism that sells the item to the highest bidder but charges a price of the second highest bidder [Vic61]. We are interested in designing Bayesian incentive compatible mechanisms which ensure truthfulness with respect to both the arrival time as well as the bid. In particular, we design a mechanism where for any bidder arriving truthfully on their assigned time maximizes the expected profit given their beliefs and that other bidders are also truthful. Moreover, we also show that reporting the true valuation for the item is a dominant strategy for the bidders.

We note that ensuring truthfulness with respect to valuation is a well understood phenomenon in an offline setting [Vic61] and generalizes easily to our online model as well. The main contribution of our paper is to design an incentive compatible mechanism where arriving at their assigned time is a dominant strategy.

1.2 Results

We design a family of incentive compatible mechanisms where each mechanism in the family gives a different efficiency and revenue. Specifically, we prove the following main theorem.

Theorem 1. *For any* $0 \le \tau \le 1$*, there exists a incentive compatible online auction mechanism that is:*

- $\frac{\tau}{4} + \frac{\tau}{2} \ln \frac{1}{\tau}$*-competitive for efficiency.*
- $\frac{\tau}{2} - \frac{\tau^2}{3}$*-competitive for revenue.*

In particular, there exists a mechanism that is $3/16 \approx 0.187$*-competitive for revenue, and a (different) mechanism that is* $\frac{1}{2\sqrt{e}} \approx 0.303$*-competitive for efficiency.*

Our results are illustrated in Figure 1. The dashed lines mark the interesting values of τ which define a set of Pareto optimal mechanisms with respect to efficiency and revenue.

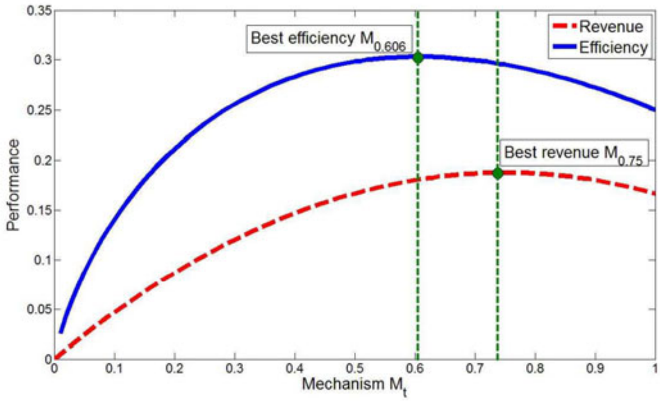


Fig. 1. The performance of the online auction mechanism as a function of τ

Techniques and connections with secretary problems. Our results are closely related to better understanding of variants of the secretary problem. In the classical secretary problem an employer would like to choose the best candidate among n competing candidates. The candidates are assumed to arrive in a random order. The secretary problem as well as many variants of it have been studied extensively in the past (See Section 1.3 for more details). Our auction mechanism is based on designing an underlying mechanisms for a variant of the secretary problem where we want to ensure that the probability that the mechanism selects the ith candidate is at least the probability of selecting the $i+1$th candidate for each i, where probability is taken over all permutations. This property in a secretary mechanism captures the inherent combinatorial structure of the auction problem where any bidder would not delay her arrival since the probability of acceptance decreases over time. We also modify our performance goals in secretary problem to mimic the goals of the auction problem. The goal of efficiency of a mechanism in the auction setting corresponds to maximizing the probability of accepting the best candidate. The other goal of maximizing revenue corresponds to maximizing the probability of hiring the best candidate while having the second best candidate appear before the best candidate. For formal definitions of the secretary model see Section 2. To obtain such mechanisms for the secretary problem, we use a recently introduced linear programming technique by Buchbinder et al [BJS10]. Buchbinder et al [BJS10] also designed new mechanisms for the secretary problem which ensure that the probability of hiring in each position is the exactly equal. In our setting, mapping the online auction problem to the secretary problem, we design new mechanisms for the probability of hiring is an non-increasing function of the position. Moreover, we also evaluate a mechanism based on its revenue apart from its efficiency. Thus the set of mechanisms obtained here differ in performance from those in Buchbinder et al [BJS10].

Finally, we believe that our novel truthfulness assumption that each bidder believes "her valuation is as good as anyone else" is very reasonable in many scenarios of lack of information, and may be useful in designing mechanisms for various other settings.

1.3 Previous Results

Recently, there has been significant work on using generalizations of secretary problems as a framework for online auctions [HKP04, Kle05, BIKK07, BIK07, BIKK08]. Incentives issues in online mechanisms have been studied in several models [LN00,HKP04,AAM03]. These works designed mechanisms where incentive issues were considered for both value and time strategies. The closest to our model is a model studied in Hajiaghayi et al [HKP04]. They studied a similar model in which an item is sold online. Bidders in their model have arrival and departure time, and the item must be allocated to a bidder by their reported departure time. The main difference of their model from our model is that they make the assumption that bidders do not receive any utility from the item if they get the item outside their arrival/departure interval. This makes the design easier since bidders who arrive early have no incentive to delay their arrival later than their departure time since they will get no utility.

The secretary problem is a well-studied problem introduced by Gardner [Gar60]. We refer the reader to the survey by Ferguson [Fer89] on the history of the problem. For our results on the secretary problem, we use the linear programming technique introduced by Buchbinder et al [BJS10] who apply the technique to the secretary problem and some of its generalizations.

2 Secretary Problem and Linear Programming

In this section, we give new mechanisms for variants of the secretary problem which form the basis for the mechanisms for the online auction problem. In the secretary problem we have a set of candidates $C = \{1, 2, \ldots, n\}$ that arrive in a random order. There is total order $\mathcal{R}$ over the set of candidates which specifies the quality of the candidates with respect to each other. The rank of the candidate is the position of the candidate in the total order $\mathcal{R}$. After interviewing a candidate, the mechanism designer learns her rank in relation to the candidates that have already been interviewed. The mechanism designer then has to take an irrevocable decision whether to hire the interviewed candidate. We study two objectives which the mechanism designer needs to maximize. The first, which we call *efficiency*, is the probability of hiring the best candidate. This goal closely relates to efficiency in the online auction scenario. The second objective, which we call *revenue*, is the probability of the event of hiring the best candidate while having the second best candidate appear before the best candidate. This objective is closely related to the revenue in the auction model. Since we want to map mechanisms for the secretary problem to incentive compatible mechanisms for the online auction problem, we want the following property to be satisfied by the secretary mechanisms. For any position $1 \leq i \leq n-1$, probability that a candidate is selected at position i is more than the probability a candidate

$$
\begin{array}{lll}
(P) & \text{(Efficiency)} \max \ \frac{1}{n} \cdot \sum_{i=1}^{n} f_i & \\
& \text{(Revenue)} \ \ \max \ \frac{1}{n(n-1)} \cdot \sum_{i=1}^{n} (i-1) \cdot f_i & \\
\text{s.t.} & & \\
\forall\, 1 \le i \le n & f_i \le i \cdot p_i & \\
\forall\, 1 \le i \le n & f_i \le 1 - \sum_{j=1}^{i-1} p_i & \\
\forall\, 1 \le i \le n-1 & p_i \ge p_{i+1} & \\
\forall\, 1 \le i \le n & f_i \ge 0, p_i \ge 0 &
\end{array}
$$

$$
\begin{array}{lll}
(D) & \min \ \sum_{i=1}^{n} x_i & \\
\text{s.t.} & & \\
\forall\, 2 \le i \le n-1 & \sum_{j=i+1}^{n} x_j - z_i + z_{i-1} \ge i y_i & \\
& \sum_{j=2}^{n} x_j - z_1 \ge y_1 & \\
& z_{n-1} \ge n y_n & \\
\text{(Efficiency)} & \forall\, 1 \le i \le n & x_i + y_i \ge \frac{1}{n} \\
\text{(Revenue)} & \forall\, 1 \le i \le n & x_i + y_i \ge \frac{i-1}{n(n-1)} \\
\forall\, 1 \le i \le n & x_i \ge 0, y_i \ge 0, z_i \ge 0 &
\end{array}
$$

Fig. 2. (P) is an LP for Maximizing efficiency/revenue with $p_i \ge p_{i+1}$. (D) is the corresponding dual LP of (P).

is selected at position $i + 1$. The above property will be crucial in establishing incentive compatibility of mechanisms for the online auction problem and therefore, we call an interview mechanism *incentive compatible* if it satisfies the above mentioned property.

In this section, we give incentive compatible mechanisms for the secretary problem and prove the following Theorem 2.

Theorem 2. *There is a mechanism $\mathcal{M}_\tau$ for each $0 \le \tau \le 1$ which is incentive compatible. The mechanism picks the best candidate with probability $\frac{\tau}{4} + \frac{\tau}{2}\ln(1/\tau)$ (efficiency) and picks the best candidate and the second best candidate appeared before the first with probability $\frac{\tau}{2} - \frac{\tau^2}{3}$ (revenue). In particular, there exists a mechanism that is $3/16 = 0.1875$-competitive for revenue, and a (different) mechanism that is 0.303-competitive for efficiency and these are optimal.*

The proof the theorem follows from mapping the feasible mechanisms for the secretary problem to feasible solutions to a linear program and then optimizing the desired objective function of efficiency or revenue. This follows the technique introduced by Buchbinder et al [BJS10]. We state the following two lemmas which will prove Theorem 2.

Lemma 1 ***(Mechanism to LP solution).*** *Let π be any incentive compatible mechanism for the secretary problem. Let p_i^π denote the probability of selecting the candidate at position i and f_i^π denote the probability of selecting the candidate at position i given that the best candidate is at position i. Then (p^π, f^π) is a feasible solution to the linear program (P). Moreover the efficiency of π is at least $\frac{1}{n} \cdot \sum_{i=1}^{n} f_i^\pi$ and the revenue is at least $\frac{1}{n(n-1)} \cdot \sum_{i=1}^{n} (i-1) \cdot f_i^\pi$.*

Proof. We first show that the solution (p^π, f^π) is a feasible solution to the linear program (P). The first two set of constraints are satisfied follows from Lemma 3.1 from Buchbinder et al [BJS10]. The last set of constraints is satisfied since π is incentive compatible for delay only strategies. Thus, probability that π of accepting a candidate at position i must be decreasing function of i.

Lemma 1 shows that the optimal solution to (P) is an upper-bound on the performance of the mechanism. The following lemma shows that every LP solution actually corresponds to a mechanism which performs as well as the objective value of the solution.

Lemma 2 ***(LP solution to Mechanism).*** *Let (p_i, f_i) for $1 \leq i \leq n$ be any feasible LP solution to (P). Then there is a mechanism π with efficiency $\frac{1}{n}\sum_{i=1}^{n} f_i$ and revenue $\frac{1}{n(n-1)} \cdot \sum_{i=1}^{n}(i-1) \cdot f_i$.*

Proof. Consider the mechanism π defined as follows. Let $r_i = \lfloor \frac{ip_i}{1-\sum_{j=1}^{i-1} p_j} \rfloor$. Then the mechanism selects the candidate at position i with probability 1 if the rank of i^{th} candidate among the candidates $1, \ldots, i$ is less than or equal to r_i. If the rank of the candidate i is $r_i + 1$ then it selects the candidate with probability $\frac{ip_i}{1-\sum_{j=1}^{i-1} p_j} - r_i$. A simple calculation shows that the probability the mechanism accepts the i^{th} candidate is exactly p_i and probability of selecting the best candidate given that it is best over all, f_i^π, is at least f_i. Hence, the efficiency of the mechanism is at least $\frac{1}{n}\sum_{i=1}^{n} f_i$. Moreover, the mechanism is incentive compatible for delay only strategies since $p_i \geq p_{i+1}$ for each i.

Let f_{ij}^π denote the probability that the mechanism accepts the candidate at position i given that it is the best and the j^{th} candidate is the second best. Then we have the following claim.

Claim. For each $i > 1$: $f_i^\pi = \frac{1}{i-1}\sum_{j=1}^{i-1} f_{i,j}^\pi$.

Proof. First, by the definition of f_i^π and $f_{i,j}^\pi$,

$$f_i^\pi = \frac{1}{n-1}\left[\sum_{j=1}^{i-1} f_{i,j}^\pi + \sum_{j=i+1}^{n} f_{i,j}^\pi\right]$$

We claim that for each $j > i$, $f_{i,j}^\pi = f_i^\pi$. The reason is that given that when $j > i$ is second best the probability the algorithm accepts i only depends on the first i numbers and **doesn't** use their values at all. The argument follows since the first $i-1$ numbers forms a random permutation. Thus we get:

$$f_i^\pi = \frac{1}{n-1}\sum_{j=1}^{i-1} f_{i,j}^\pi + \frac{n-i}{n-1} \cdot f_i^\pi$$

and so we get our claim.

Using this claim it is easy to derive the lemma since the total revenue of the mechanism is:

$$\frac{1}{n(n-1)}\sum_{i=2}^{n}\sum_{j=1}^{i-1} f^{\pi}_{i,j} = \frac{1}{n(n-1)}\sum_{i=1}^{n}(i-1)\cdot f^{\pi}_{i} \geq \frac{1}{n(n-1)}\sum_{i=1}^{n}(i-1)\cdot f_i$$

since $f^{\pi}_i \geq f_i$ for each i.

Thus solving the primal program we can derive a family of mechanisms for the problem. The mechanisms are parameterized by a real number $0 \leq \tau \leq 1$ and are as follows.

Incentive Compatible Mechanism $\mathcal{M}_\tau$:

– Let $0 \leq \tau \leq 1$. For each $1 \leq i \leq n$, while no candidate is selected, do
  - If $1 \leq i \leq \tau n$, select the i^{th} candidate with probability $\frac{i}{2\tau n - i + 1}$ if she is the best candidate so far.
  - If $\tau n < i \leq n$, select the i^{th} candidate if she is the best candidate so far.

The following claim shows that each of the mechanisms $\mathcal{M}_\tau$ is incentive compatible.

Lemma 3. *For each $1 \leq i \leq n-1$, we have $p_i \geq p_{i+1}$.*

Proof. A simple calculation shows that $p_i = \frac{1}{2\tau n}$ for each $1 \leq i \leq \tau n$ and $p_i = \frac{\tau n}{2i(i-1)}$ for each $\tau n < i \leq n$ and hence the claim holds.

By selecting τ, we obtain mechanisms with different values of efficiency and revenue. A simple calculation then yields the following lemma about the performance of the mechanisms.

Lemma 4. *The mechanism $\mathcal{M}_\tau$ for any $0 \leq \tau \leq 1$,*

– ***(Efficiency)*** *Picks the best candidate with probability $\frac{\tau}{4} + \frac{\tau}{2}\ln(1/\tau)$.*
– ***(Revenue)*** *Picks the best candidate when the second best candidate appeared before the first with probability $\frac{\tau}{2} - \frac{\tau^2}{3}$.*

Optimizing for τ, the best efficiency of $\frac{1}{2\sqrt{e}}$ is obtained when $\tau = \frac{1}{\sqrt{e}}$ while the best revenue of $\frac{3}{16}$ is obtained when $\tau = \frac{3}{4}$. Moreover, all values of τ are in the range $[\frac{1}{\sqrt{e}}, 3/4] = [0.606, 0.75]$ results in a mechanism with efficiency and revenue that is Pareto optimal.

We also show that the mechanism for efficiency and revenue are optimal by giving dual solutions to the dual linear program (D) of the corresponding value.

Lemma 5. *Let π be any mechanism which is incentive compatible. Then the efficiency of π cannot be better than $\frac{1}{2\sqrt{e}}$ and the revenue of π cannot be better than $3/16$.*

Proof. We give two dual solutions to the linear program (D) in figure 2 where the corresponding constraint for the efficiency and revenue are present. Observe that each dual solution is an upper bound on performance of any mechanism.

Efficiency. Let $\tau = \frac{1}{\sqrt{e}}$. Let $x_i = 0$ and $y_i = \frac{1}{n}$ and $z_i = i\tau \sum_{j=(\tau n+1)}^{n} \frac{1}{j} - \frac{i(i+1)}{2n}$ for $1 \leq i \leq \tau n$ and $x_i = \frac{1}{n}(1 - \sum_{j=i}^{n-1} \frac{1}{j})$ and $y_i = \frac{1}{n}\sum_{j=i}^{n-1} \frac{1}{j}$, $z_i = 0$ for $\tau n < i \leq n$. We now show that the above dual solution is feasible and has an objective value of $\approx \frac{1}{2\sqrt{e}}$. A simple calculation shows that $x_i, y_i, z_i \geq 0$ for each $1 \leq i \leq n$. We now calculate the objective value before verifying all the constraints.

$$s = \sum_{i=1}^{n} x_i = \frac{1}{n} \sum_{i=\tau n+1}^{n} (1 - \sum_{j=i}^{n-1} \frac{1}{j})$$

$$= \frac{1}{n}(n - \tau n - \sum_{j=\tau n+1}^{n-1} \sum_{i=\tau n+1}^{j} \frac{1}{j}) = \frac{1}{n}(n - \tau n - \sum_{j=\tau n+1}^{n-1} (1 - \frac{\tau n}{j}))$$

$$= \frac{1}{n}(1 + \tau n \ln \frac{n}{\tau n}) \approx \tau \ln \frac{1}{\tau} = \frac{1}{2\sqrt{e}}$$

Observe that the constraint $x_i + y_i \geq \frac{1}{n}$ is satisfied at equality for each $1 \leq i \leq n$.

We now verify the constraint $\sum_{j=i+1}^{n} x_j - z_i + z_{i-1} \geq iy_i$. For $1 \leq i \leq \tau n$, observe that $z_i = is + \frac{i(i+1)}{2n}$. Thus we have

$$\sum_{j=i+1}^{n} x_j - z_i + z_{i-1} = s - z_i + z_{i-1} =$$

$$= s - s + \frac{i(i+1) - (i-1)i}{2n} = \frac{i}{n} = iy_i$$

as required. For $\tau n + 2 \leq i \leq n - 1$, we have

$$\sum_{k=i+1}^{n} x_k - z_i + z_{i-1} = \frac{1}{n} \sum_{k=i+1}^{n} (1 - \sum_{j=k}^{n-1} \frac{1}{j}) = \frac{i}{n} \sum_{j=i}^{n-1} \frac{1}{j} = iy_i$$

The constraints for boundary cases $i = \lceil t \rceil$ and $i = n$ can be verified similarly.

Revenue. Due to lack of space we defer the proof to the full version of the paper.

3 The Online Auction Mechanism

Given the family of mechanisms in Section 2, we design mechanisms for the online auction problem which prove Theorem 1. The family of mechanisms, parameterized by parameter τ is given below. The mechanism A_τ selects two random permutations π_1 and π_2 on bidders. The permutation π_1 is used to break ties among bidders who arrive at the same time and the permutation π_2 is used to break ties among bidders who have the same valuation.

Auction mechanism A_τ**:** Let B_t be the set of agents arriving at time t.

- Order the bidders in B_t by permutation π_1. Use the valuation to define the ranks of the bidders while breaking ties according to permutation π_2.
- Feed the bidders one-by-one according to their order in B_t along with their rank to the mechanism M_τ in Section 2.
- If the mechanism decides to accept the bidder then allocate the item to that bidder.
- Set the price p for the bidder to be the highest value of any bid that arrived prior to this bidder.

Observe that the mechanism indeed satisfies the online requirement of allocating the item and setting a price for it immediately at the arrival time of the bidder. We now prove that for every τ, the mechanism A_τ given above is incentive compatible.

Lemma 6. *For any* $0 \leq \tau \leq 1$*, the mechanism* A_τ *is incentive compatible for both valuation and time arrival.*

Proof. First, we prove that the online mechanism is incentive compatible for valuation. This follows simply since the price a bidder has to pay, in case she wins the item, is the maximum price seen so far by the mechanism and is independent of her bid. Moreover, the mechanism gives the item only to the person with the highest valuation so far, therefore the mechanism is incentive compatible for valuation.

We now show that the mechanism is incentive compatible for time strategies. For simplicity, we assume that no two bidders arrive at the same time. We prove that for any bidder, conditioned on her beliefs, the expected utility of the bidder is a decreasing function of the position. Thus, the bidder has no incentive to delay her arrival time.

Consider a bidder with valuation v. Let S be a random variable of the values of the $n-1$ bidders (except the bidder we currently consider) arranged according to their arrival time. For each i, let S_i be the first i values in S, and let $v(S_i)$ be the maximal value in S_i. Let X_i be the indicator random variable that the bidder that arrived at the ith position is assigned the item. First observe that the mechanism allocates the item only to the highest bid seen so far and thus $X_i = 0$ if v is not the highest until the ith position. Therefore, the expected profit of the bidder had she arrived just before the ith arrival time is $E[(v - v(S_{i-1})) \cdot X_i]$. We next prove that the expected profit for any bidder conditioned on her beliefs is a decreasing function of the position at which the bidder appears. Since, all expectations are evaluated conditioned on the bidder's beliefs, we omit this conditioning from the notation. Formally, for each $1 \leq i \leq n-1$, we prove that

$$E[(v - v(S_{i-1})) \cdot X_i] \geq E[(v - v(S_i)) \cdot X_{i+1}] \qquad (1)$$

Observe that we have the following.

$$\begin{aligned} &E[(v - v(S_{i-1})) \cdot X_i] = E[(v - v(S_{i-1})) \cdot X_i | v > v(S_{i-1})] \cdot Pr[v > v(S_{i-1})] \\ &= E[(v - v(S_{i-1})) | v > v(S_{i-1})] \cdot Pr[X_i = 1 | v > v(S_{i-1})] \cdot Pr[v > v(S_{i-1})] \end{aligned}$$

The second equality follows by the fact that given the event that $v > v(S_{i-1})$, i.e. the bidder has the highest valuation so far, the probability of allocating the item to bidder i is independent of $v - v(S_{i-1})$. This follows since the underlying mechanism $\mathcal{M}_\tau$, and thus A_τ, does not look at the actual values but only the relative ordering when deciding whether to give the item or not to a bidder.

Conditioned on the beliefs of the bidder, we have $Pr[v > v(S_{i-1})] = 1/i$ and that the set of valuations in S_{i-1} when ordered by position form a random permutation. Thus,

$$Pr[X_i = 1 | v > v(S_{i-1})] \cdot Pr[v > v(S_{i-1})] = p_i$$

where p_i is the probability of accepting the i^{th} candidate by $\mathcal{M}_\tau$. But for mechanism $\mathcal{M}_\tau$, p_i is a decreasing function of i. Thus, it suffices to show that $E[v(S_{i-1}) | v \geq v(S_{i-1})]$ is a non-decreasing function of i. Before we prove this, we prove the following technical claim which is crucial in comparing the expected profit if the bidder arrives in position i or $i+1$. Here v_i is the random valuation of the i^{th} bidder by arrival order excluding the bidder with valuation v.

Claim. For each i, we have

$$E[v(S_{i-1}) | v > v(S_{i-1}) \ \& \ v < v_i] \leq E[v(S_{i-1}) | v > v(S_i)]$$

Proof. Let $v_2(S_i)$ be a random variable for the second maximal value in S_i.

$$\begin{aligned}
&E[v(S_{i-1}) | v > v(S_{i-1}) \ \& \ v < v_i] \\
&= E[v(S_{i-1}) | v_i > \max\{v(S_{i-1}), v\} \ \& \ v > v(S_{i-1})] \\
&= E[v(S_{i-1}) | v > \max\{S_{i-1}, v_i\} \ \& \ v_i > v(S_{i-1})] && (2) \\
&= E[v_2(S_i) | v > v(S_i) \ \& \ v_i > v(S_{i-1})] = E[v_2(S_i) | v > v(S_i)] && (3) \\
&\leq E[v(S_{i-1}) | v > v(S_i)] && (4)
\end{aligned}$$

Where equality (2) follows by the symmetry arguments on v and v_i. This is done by pairing each permutation in which $v_i > v$ to a permutation in which $v > v_i$. Second equality in (3) follows since for any permutation on v_1 to v_i the second highest value is the same. Equality (4) follows since for any permutation on the values the second highest value among the first i values is at most the highest value among the first $i-1$ values.

Now we prove the following claim which shows that $E[v(S_{i-1}) | v > v(S_{i-1})]$ is a non-decreasing function of i. This will complete the proof. Observe that

$$\begin{aligned}
&E[v(S_{i-1}) | v > v(S_{i-1})] \\
&= E[v(S_{i-1}) | v > v(S_{i-1}), v > v_i] \cdot Pr[v > v_i | v > v(S_{i-1})] \\
&+ E[v(S_{i-1}) | v > v(S_{i-1}), v < v_i] \cdot Pr[v < v_i | v > v(S_{i-1})] \\
&\leq E[v(S_{i-1}) | v > v(S_{i-1}), v > v_i] \cdot Pr[v > v_i | v > v(S_{i-1})] \\
&+ E[v(S_{i-1}) | v > v(S_{i-1}), \mathbf{v} > \mathbf{v_i}] \cdot Pr[v < v_i | v > v(S_{i-1})] && (5) \\
&= E[v(S_{i-1}) | v \geq v(S_i)] \leq E[v(S_i) | v \geq v(S_i)] && (6)
\end{aligned}$$

Inequality (5) follows by Claim 3. Inequality (6) follows since in every term we maximize over more elements.

Now, we prove the main theorem which follows directly from Lemma 6.

Proof. of Theorem 1 Given that the mechanism is incentive compatible for time strategies (Lemma 6), we get that the dominant strategy of the bidders is not to delay their arrival time. Thus, the rank given to the underlying mechanism is a random permutation of the bidders. Thus, the performance of the mechanism follows directly by Lemma 4.

References

[AAM03] Awerbuch, B., Azar, Y., Meyerson, A.: Reducing Truth-Telling Online Mechanisms to Online Optimization. In: Proc. ACM Symposium on Theory of Computing, pp. 503–510 (2003)

[BIK07] Babaioff, M., Immorlica, N., Kleinberg, R.: Matroids, Secretary Problems, and Online Mechanisms. In: Proceedings 18th ACM-SIAM Symposium on Discrete Algorithms (2007)

[BIKK07] Babaioff, M., Immorlica, N., Kempe, D., Kleinberg, R.: A Knapsack Secretary Problem with Applications. In: Charikar, M., Jansen, K., Reingold, O., Rolim, J.D.P. (eds.) RANDOM 2007 and APPROX 2007. LNCS, vol. 4627, pp. 16–28. Springer, Heidelberg (2007)

[BIKK08] Babaioff, M., Immorlica, N., Kempe, D., Kleinberg, R.: Online Auctions and Generalized Secretary Problems. SIGecom Exchange 7, 1–11 (2008)

[BJS10] Buchbinder, N., Jain, K., Singh, M.: Secretary problems via linear programming. In: Eisenbrand, F., Shepherd, F.B. (eds.) IPCO 2010. LNCS, vol. 6080, pp. 163–176. Springer, Heidelberg (2010)

[Fer89] Ferguson, T.S.: Who Solved the Secretary Problem? Statist. Sci. 4, 282–289 (1989)

[Gar60] Gardner, M.: Mathematical Games. Scientific American, 150–153 (1960)

[HKP04] Hajiaghayi, M.T., Kleinberg, R., Parkes, D.C.: Adaptive Limited-Supply Online Auctions. In: Proceedings of the 5th ACM Conference on Electronic Commerce (2004)

[Kle05] Kleinberg, R.: A Multiple-Choice Secretary Algorithm with Applications to Online Auctions. In: Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete algorithms (2005)

[LN00] Lavi, R., Nisan, N.: Competitive Analysis of Incentive Compatible On-line Auctions. In: Proc. 2nd ACM Conf. on Electronic Commerce, pp. 233–241 (2000)

[Vic61] Vickrey, W.: Counterspeculation, auctions, and competitive sealed tenders. The Journal of Finance 16(1), 8–37 (1961)

Optimal Pricing in the Presence of Local Network Effects

Ozan Candogan[1], Kostas Bimpikis[2], and Asuman Ozdaglar[1]

[1] Laboratory for Information and Decision Systems, MIT
candogan@mit.edu, asuman@mit.edu
[2] Operations Research Center and Laboratory for Information and Decision Systems, MIT
kostasb@mit.edu

Abstract. We study the optimal pricing strategies of a monopolist selling a divisible good (service) to consumers that are embedded in a social network. A key feature of our model is that consumers experience a (positive) *local network effect*. In particular, each consumer's usage level depends directly on the usage of her *neighbors* in the social network structure. Thus, the monopolist's optimal pricing strategy may involve offering discounts to certain agents[1], who have a *central* position in the underlying network. Our results can be summarized as follows. First, we consider a setting where the monopolist can offer individualized prices and derive an explicit characterization of the optimal price for each consumer as a function of her network position. In particular, we show that it is optimal for the monopolist to charge each agent a price that is proportional to her *Bonacich centrality* in the social network. In the second part of the paper, we discuss the optimal strategy of a monopolist that can only choose a single uniform price for the good and derive an algorithm polynomial in the number of agents to compute such a price. Thirdly, we assume that the monopolist can offer the good in two prices, full and discounted, and study the problem of determining which set of consumers should be given the discount. We show that the problem is NP-hard, however we provide an explicit characterization of the set of agents that should be offered the discounted price. Finally, we describe an approximation algorithm for finding the optimal set of agents. We show that if the profit is nonnegative under any feasible price allocation, the algorithm guarantees at least 88 % of the optimal profit.

1 Introduction

Inarguably social networks, that describe the pattern and level of interaction of a set of agents, are instrumental in the propagation of information and act as conduits of influence among its members. Their importance is best exemplified by the overwhelming success of online social networking communities, such as Facebook and Twitter. The ubiquity of these internet based services, that are

[1] We use the terms "agent" and "consumer" interchangeably.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 118–132, 2010.

built around social networks, has made possible the collection of vast amounts of data on the structure and intensity of social interactions. The question that arises naturally is whether firms can intelligently use the available data to improve their business strategies.

In this paper, we focus on the question of using the potentially available data on network interactions to improve the pricing strategies of a seller, that offers a divisible good (service). A main feature of the products we consider is that they exhibit a local (positive) *network effect*: increasing the usage level of a consumer has a positive impact on the usage levels of her peers. As concrete examples of such goods, consider online games (e.g., World of Warcraft, Second Life) and social networking tools and communities (e.g., online dating services, employment websites etc.). More generally, the local network effect can capture *word of mouth* communication among agents: agents typically form their opinions about the quality of a product based on the information they obtain from their peers.

How can a monopolist exploit the above network effects and maximize her revenues? In particular, in such a setting it is plausible that an optimal pricing strategy may involve favoring certain agents by offering the good at a discounted price and subsequently exploiting the positive effect of their usage on the rest of the consumers. At its extreme, such a scheme would offer the product for free to a subset of consumers hoping that this would have a large positive impact on the purchasing decisions of the rest. Although such strategies have been used extensively in practice, mainly in the form of ad hoc or heuristic mechanisms, the available data enable companies to effectively target the agents to maximize that impact.

The goal of the present paper is to characterize optimal pricing strategies as a function of the underlying social interactions in a stylized model, which features consumers that are embedded in a given social network and influencing each other's decisions. In particular, a monopolist first chooses a pricing strategy and then consumers choose their usage levels, so as to maximize their own utility. We capture the local positive network effect by assuming that a consumer's utility is increasing in the usage level of her peers. We study three variations of the baseline model by imposing different assumptions on the set of available pricing strategies, that the monopolist can implement.

First, we allow the monopolist to set an individual price for each of the consumers. We show that the optimal price for each agent can be decomposed into three components: a fixed cost, that does not depend on the network structure, a markup and a discount. Both the markup and the discount are proportional to the *Bonacich centrality* of the agent's neighbors in the social network structure, which is a sociological measure of network influence. The Bonacich centrality measure, introduced by [2], can be computed as the stationary distribution of a random walk on the underlying network structure. Hence, the nodes with the highest centrality are the ones that are visited by the random walk most frequently. Intuitively, agents get a discount proportional to the amount they influence their peers to purchase the product, and they receive a markup if they are strongly influenced by other agents in the network. Our results provide an economic foundation for this sociological measure of influence.

Perfect price differentiation is typically hard to implement. Therefore, in the second part of the paper we study a setting, where the monopolist offers a single uniform price for the good. Intuitively, this price might make the product unattractive for a subset of consumers, who end up not purchasing, but the monopolist recovers the revenue losses from the rest of the consumers. We develop an algorithm that finds the optimal single price in time polynomial in the number of agents. The algorithm considers different subsets of the consumers and finds the optimal price provided that only the consumers in the given subset purchase a positive amount of the good. First, we show that given a subset S we can find the optimal price p_S under the above constraint in closed form. Then, we show that we only need to consider a small number of such subsets. In particular, we rank the agents with respect to a weighted centrality index and at each iteration of the algorithm we drop the consumer with the smallest such index and let S be the set of remaining consumers.

Finally, we consider an intermediate setting, where the monopolist can choose one of a small number of prices for each agent. For exposition purposes, we restrict the discussion to two prices, *full* and *discounted*. We show that the resulting problem, i.e., determining the optimal subset of consumers to offer the discounted price, is NP-hard[2]. We also provide an approximation algorithm that recovers (in polynomial time) at least 88 % of the optimal revenue.

As mentioned above, a main feature of our model is the positive impact of a consumer's purchasing decision to the purchasing behavior of other consumers. This effect, known as *network externality*, is extensively studied in the economics literature (e.g., [8], [15]). However, the network effects in those studies are of *global* nature, i.e., the utility of a consumer depends directly on the behavior of the whole set of consumers. In our model, consumers interact directly only with a subset of agents. Although interaction is local for each consumer, her utility may depend on the global structure of the network, since each consumer potentially interacts indirectly with a much larger set of agents than just her peers.

Given a set of prices, our model takes the form of a *network game* among agents that interact locally. A recent series of papers studies such games, e.g., [1], [3], [7], [9]. A key modeling assumption in [1], [3] and [7], that we also adopt in our setting, is that the payoff function of an agent takes the form of a linear-quadratic function. Ballester et al. in [1] were the first to note the linkage between Bonacich centrality and Nash equilibrium outcomes in a single stage game with local payoff complementarities. Our characterization of optimal prices when the monopolist can perfectly price differentiate is reminiscent of their results, since prices are inherently related to the Bonacich centrality of each consumer. However, both the motivation and the analysis are quite different, since ours is a two-stage game, where a monopolist chooses prices to maximize her revenue subject to equilibrium constraints. Also, [3] and [7] study a similar game to the one in

[2] The hardness result can be extended to the case of more than two prices.

[1] and interpret their results in terms of public good provision. A number of recent papers ([5], [10] and [18]) have a similar motivation to ours, but take a completely different approach: they make the assumption of limited knowledge of the social network structure, i.e., they assume that only the degree distribution is known, and thus derive optimal pricing strategies that depend on this first degree measure of influence of a consumer. In our model, we make the assumption that the monopolist has complete knowledge of the social network structure and, thus, obtain qualitatively different results: the degree is not the appropriate measure of influence but rather prices are proportional to the Bonacich centrality of the agents. On the technical side, note that assuming more global knowledge of the network structure increases the complexity of the problem in the following way: if only the degree of an agent is known, then essentially there are as many different *types* of agents as there are different degrees. This is no longer true when more is known: then, two agents of the same degree may be of different type because of the difference in the characteristics of their neighbors, and therefore, optimal prices charged to agents may be different.

Finally, there is a recent stream of literature in computer science, that studies a set of algorithmic questions related to marketing strategies over social networks. Kempe et al. in [16] discuss optimal *network seeding* strategies over social networks, when consumers act myopically according to a pre-specified rule of thumb. In particular, they distinguish between two basic models of diffusion: the *linear threshold model*, which assumes that an agent adopts a behavior as soon as adoption in her neighborhood of peers exceeds a given threshold and *independent cascade model*, which assumes that an adopter *infects* each of her neighbors with a given probability. The main question they ask is finding the optimal set of initial adopters, when their number is given, so as to maximize the eventual adoption of the behavior, when consumers behave according to one of the diffusion models described above. They show that the problem of *influence maximization* is NP-hard and provide a greedy heuristic, that achieves a solution, that is provably within 63 % of the optimal.

Closest in spirit with our work, is [13], which discusses the optimal marketing strategies of a monopolist. Specifically, they assume a general model of influence, where an agent's willingness to pay for the good is given by a function of the subset of agents that have already bought the product, i.e., $u_i : 2^V \rightarrow \mathbb{R}_+$, where u_i is the willingness to pay for agent i and V is the set of consumers. They restrict the monopolist to the following set of *marketing strategies*: the seller visits the consumers in some sequence and makes a take-it-or-leave-it offer to each one of them. Both the sequence of visits as well as the prices are chosen by the monopolist. They provide a dynamic programming algorithm that outputs the optimal pricing strategy for a symmetric setting, i.e., when the agents are ex-ante identical (the sequence of visits is irrelevant in this setting). Not surprisingly the optimal strategy offers discounts to the consumers that are visited earlier in the sequence and then extracts revenue from the rest. The general problem, when agents are heterogeneous, is NP-hard, thus they consider approximation algorithms. They show, in particular, that *influence-and-exploit* strategies, that offer the product for free to a strategically chosen set A, and then offer the myopically optimal price to the remaining agents provably achieve a constant factor approximation of the optimal revenues under some assumptions on the influence model. However, this paper does not provide a qualitative insight on the relation between optimal strategies and the structure of the social network. In contrast, we are mainly interested in characterizing the optimal strategies as a function of the underlying network.

The rest of paper is organized as follows. Section 2 introduces the model. In Section 3 we begin our analysis by characterizing the usage level of the consumers at equilibrium given the vector of prices chosen by the monopolist. In Section 4 we turn attention to the pricing stage (first stage of the game) and characterize the optimal strategy for the monopolist under three different settings: when the monopolist can perfectly price discriminate (Subsection 4.1), when the monopolist chooses a single uniform price for all consumers (Subsection 4.2) and finally when the monopolist can choose between two exogenously given prices, the full and the discounted (Subsection 4.3). Finally, we conclude in Section 5. Due to space constraints all proofs are omitted and can be found in [6].

2 Model

The society consists of a set $\mathcal{I} = \{1, \dots, n\}$ of agents embedded in a social network represented by the adjacency matrix $\mathbf{G}$. The ij-th entry of $\mathbf{G}$, denoted by g_{ij}, represents the *strength* of the influence of agent j on i. We assume that $g_{ij} \in [0, 1]$ for all i, j and we normalize $g_{ii} = 0$ for all i. A monopolist introduces a *divisible* good in the market and chooses a vector $\mathbf{p}$ of prices from the set of allowable *pricing strategies* $\mathbf{P}$. In its full generality, $\mathbf{p} \in \mathbf{P}$ is simply a mapping from the set of agents to $\mathbb{R}^n$, i.e., $\mathbf{p} : \mathcal{I} \to \mathbb{R}^n$. In particular, $\mathbf{p}(i)$ or equivalently p_i is the price that the monopolist offers to agent i for one unit of the divisible good. Then, the agents choose the amount of the divisible good they will purchase at the announced price. Their utility is given by an expression of the following form:

$$u_i(x_i, \mathbf{x_{-i}}, p_i) = f_i(x_i) + x_i h_i \left(\mathbf{G}, \mathbf{x_{-i}}\right) - p_i x_i,$$

where $x_i \in [0, \infty)$ is the amount of the divisible good that agent i chooses to purchase. Function $f_i : [0, \infty) \to \mathbb{R}$ represents the utility that the agent obtains from the good, assuming that there are no network externalities, and $p_i x_i$ is the amount agent i is charged for its consumption. The function $h_i : [0, 1]^{n \times n} \times [0, \infty)^{n-1} \to [0, \infty)$ is used to capture the utility the agent obtains due to the positive network effect (note the explicit dependence on the network structure).

We next describe the two-stage *pricing-consumption* game, which models the interaction between the agents and the monopolist:

Stage 1 (Pricing) : The monopolist chooses the pricing strategy $\mathbf{p}$, so as to maximize profits, i.e., $\max_{\mathbf{p} \in \mathbf{P}} \sum_i p_i x_i - c x_i$, where c denotes the marginal cost of producing a unit of the good and x_i denotes the amount of the good agent i purchases in the second stage of the game.

Stage 2 (Consumption) : Agent i chooses to purchase x_i units of the good, so as to maximize her utility given the prices chosen by the monopolist and $\mathbf{x}_{-i}$, i.e.,

$$x_i \in \arg\max_{y_i \in [0,\infty)} u_i(y_i, \mathbf{x_{-i}}, p_i).$$

We are interested in the *subgame perfect* equilibria of the two-stage pricing-consumption game.

For a fixed vector of prices $\mathbf{p} = [p_i]_i$ chosen by the monopolist, the equilibria of the second stage game, referred to as the consumption equilibria, are defined as follows:

Definition 1 (Consumption Equilibrium). *For a given vector of prices* $\mathbf{p}$*, a vector* $\mathbf{x}$ *is a consumption equilibrium if, for all* $i \in \mathcal{I}$*,*

$$x_i \in \arg\max_{y_i \in [0,\infty)} u_i(y_i, \mathbf{x_{-i}}, p_i).$$

We denote the set of consumption equilibria at a given price vector $\mathbf{p}$ *by* $C[\mathbf{p}]$*.*

We begin our analysis by the second stage (the consumption subgame) and then discuss the optimal pricing policies for the monopolist given that agents purchase according to the consumption equilibrium of the subgame defined by the monopolist's choice of prices.

3 Consumption Equilibria

For the remainder of the paper, we assume that the payoff function of agent i takes the following quadratic form:

$$u_i(x_i, \mathbf{x_{-i}}, p_i) = a_i x_i - b_i x_i^2 + x_i \cdot \sum_{j \in \{1,\cdots,n\}} g_{ij} \cdot x_j - p_i x_i, \tag{1}$$

where the first two terms represent the utility agent i derives from consuming x_i units of the good irrespective of the consumption of her peers, the third term represents the (positive) network effect of her social group and finally the last term is the cost of usage. The quadratic form of the utility function is essential for keeping the analysis tractable, but also serves as a second-order approximation of the broader class of concave utility functions.

For a given vector of prices $\mathbf{p}$, we denote by $\mathcal{G} = \{\mathcal{I}, \{u_i\}_{i\in\mathcal{I}}, [0,\infty)_{i\in\mathcal{I}}\}$ the second stage game where the set of players is $\mathcal{I}$, each player $i \in \mathcal{I}$ chooses her strategy (consumption level) from the set $[0,\infty)$, and her the utility function, u_i has the form in (1). The following assumption ensures that in this game the optimal consumption level of each agent is bounded.

Assumption 1. *For all* $i \in \mathcal{I}$*,* $b_i > \sum_{j\in\mathcal{I}} g_{ij}$*.*

The necessity of Assumption 1 is evident from the following example: assume that the adjacency matrix, which represents the level of influence among agents, takes the following simple form: $g_{ij} = 1$ for all i, j such that $i \neq j$, i.e., $\mathbf{G}$ represents a complete graph with unit weights. Also, assume that $0 < b_i = b < n - 1$ and $0 < a_i = a$ for all $i \in \mathcal{I}$. It is now straightforward to see that given any vector of prices $\mathbf{p}$ and assuming that $x_i = x$ for all $i \in \mathcal{I}$, the payoffs of all agents go to infinity as $x \to \infty$. Thus, if Assumption 1 does not hold, in the consumption game, consumers may choose to unboundedly increase their usage irrespective of the vector of prices.

Next, we study the second stage of the game defined in Section 2 under Assumption 1, and we characterize the equilibria of the consumption game among the agents for vector of prices $\mathbf{p}$. In particular, we show that the equilibrium is unique and we provide a closed form expression for it. To express the results in a compact form, we define the vectors $\mathbf{x}, \mathbf{a}, \mathbf{p} \in \mathbb{R}^n$ such that $\mathbf{x} = [x_i]_i$, $\mathbf{a} = [a_i]_i$, $\mathbf{p} = [p_i]_i$. We also define matrix $\Lambda \in \mathbb{R}^{n \times n}$ as:

$$\Lambda_{i,j} = \begin{cases} 2b_i & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases}$$

Let $\beta_i(\mathbf{x_{-i}})$ denote the best response of agent i, when the rest of the agents choose consumption levels represented by the vector $\mathbf{x_{-i}}$. From (1) it follows that:

$$\beta_i(\mathbf{x_{-i}}) = \max\left\{ \frac{a_i - p_i}{2b_i} + \frac{1}{2b_i}\sum_{j \in \mathcal{I}} g_{ij} x_j, 0 \right\}. \tag{2}$$

Our first result shows that the equilibrium of the consumption game is unique for any price vector.

Theorem 1. *Under Assumption 1, the game $\mathcal{G} = \{\mathcal{I}, \{u_i\}_{i \in \mathcal{I}}, [0, \infty)_{i \in \mathcal{I}}\}$ has a unique equilibrium.*

Intuitively, Theorem 1 follows from the fact that increasing one's consumption incurs a positive externality on her peers, which further implies that the game involves strategic complementarities and therefore the equilibria are ordered. The proof exploits this monotonic ordering to show that the equilibrium is actually unique.

We conclude this section, by characterizing the unique equilibrium of $\mathcal{G}$. Suppose that $\mathbf{x}$ is this equilibrium, and $x_i > 0$ only for $i \in S$. Then, it follows that

$$x_i = \beta_i(\mathbf{x_{-i}}) = \frac{a_i - p_i}{2b_i} + \frac{1}{2b_i}\sum_{j \in \mathcal{I}} g_{ij} x_j = \frac{a_i - p_i}{2b_i} + \frac{1}{2b_i}\sum_{j \in S} g_{ij} x_j \tag{3}$$

for all $i \in S$. Denoting by $\mathbf{x}_S$ the vector of all x_i such that $i \in S$, and defining the vectors $\mathbf{a}_S$, $\mathbf{b}_S$, $\mathbf{p}_S$ and the matrices G_S, Λ_S similarly, equation (3) can be rewritten as

$$\Lambda_S \mathbf{x}_S = \mathbf{a}_S - \mathbf{p}_S + G_S \mathbf{x}_S. \tag{4}$$

Note that Assumption 1 holds for the graph restricted to the nodes in S, hence $I - \Lambda_S^{-1} G_S$ is invertible (see [6]). Therefore, (4) implies that

$$\mathbf{x}_S = (\Lambda_S - G_S)^{-1}(\mathbf{a}_S - \mathbf{p}_S). \tag{5}$$

Therefore, the unique equilibrium of the consumption game takes the following form:

$$\begin{aligned} \mathbf{x}_S &= (\Lambda_S - G_S)^{-1}(\mathbf{a}_S - \mathbf{p}_S), \\ \mathbf{x}_{\mathcal{I}-\mathbf{S}} &= \mathbf{0}, \end{aligned} \tag{6}$$

for some subset S of the set of agents $\mathcal{I}$. This characterization suggests that consumptions of players (weakly) decrease with the prices. The following lemma, which is used in the subsequent analysis, formalizes this fact.

Lemma 1. *Let $\mathbf{x}(\mathbf{p})$ denote the unique consumption equilibrium in the game where each player $i \in \mathcal{I}$ is offered the price p_i. Then, $x_i(\mathbf{p})$ is weakly decreasing in $\mathbf{p}$ for all $i \in \mathcal{I}$, i.e, if $\hat{\mathbf{p}}_j \geq \mathbf{p}_j$ for all $j \in \mathcal{I}$ then $x_i(\hat{\mathbf{p}}) \leq x_i(\mathbf{p})$.*

4 Optimal Pricing

In this section, we turn attention to the first stage of the game, where a monopolist sets the vector of prices. We distinguish between three different scenarios. In the first subsection, we assume that the monopolist can *perfectly price discriminate* the agents, i.e., there is no restriction imposed on the prices. In the second subsection, we consider the problem of choosing a single uniform price, while in the third we allow the monopolist to choose between two exogenous prices, p_L and p_H, for each consumer. In our terminology, in the first case $\mathbf{P} = \mathbb{R}^{|\mathcal{I}|}$, in the second $\mathbf{P} = \{(p, \cdots, p)\}$, for $p \in [0, \infty)$ and finally in the third $\mathbf{P} = \{p_L, p_H\}^{|\mathcal{I}|}$.

4.1 Perfect Price Discrimination

For the remainder of the paper, we make the following assumption, which ensures that, even in the absence of any network effects, the monopolist would find it optimal to charge individual prices low enough, so that all consumers purchase a positive amount of the good.

Assumption 2. *For all $i \in \mathcal{I}$, $a_i > c$.*

Given Assumption 2, we are now ready to state Theorem 2, that provides a characterization of the optimal prices. We denote the vector of all 1's by $\mathbf{1}$.

Theorem 2. *Under Assumptions 1 and 2, the optimal prices are given by*

$$\mathbf{p} = \mathbf{a} - (\Lambda - G)\left(\Lambda - \frac{G + G^T}{2}\right)^{-1} \frac{\mathbf{a} - c\mathbf{1}}{2}. \tag{7}$$

The following corollary is an immediate consequence of Theorem 2.

Corollary 1. *Let Assumptions 1 and 2 hold. Moreover, assume that the interaction matrix G is symmetric. Then, the optimal prices satisfy*

$$\mathbf{p} = \frac{\mathbf{a} + c\mathbf{1}}{2},$$

i.e., the optimal prices do not depend on the network structure.

This result implies that when players affect each other in the same way, i.e., when the interaction matrix G is symmetric, then the graph topology has no effect on the optimal prices.

To better illustrate the effect of the network structure on prices we next consider a special setting, in which agents are symmetric in a sense defined precisely below and they differ only in terms of their network position.

Assumption 3. *Players are symmetric, i.e.,* $a_i = a_0$*,* $b_i = b_0$ *for all* $i \in \mathcal{I}$*.*

We next provide the definition of Bonacich Centrality (see also [2]), and using it obtain an alternative characterization of the optimal prices.

Definition 2 (Bonacich Centrality). *For a network with (weighted) adjacency matrix* G *and scalar* α*, the Bonacich centrality vector of parameter* α *is given by* $\mathcal{K}(G,\alpha) = (I - \alpha G)^{-1}\mathbf{1}$ *provided that* $(I - \alpha G)^{-1}$ *is well defined and nonnegative.*

Theorem 3. *Under Assumptions 1, 2 and 3, the vector of optimal prices is given by*

$$\mathbf{p} = \frac{a_0 + c}{2}\mathbf{1} + \frac{a_0 - c}{8b_0} G\mathcal{K}\left(\frac{G+G^T}{2}, \frac{1}{2b_0}\right) - \frac{a_0 - c}{8b_0} G^T\mathcal{K}\left(\frac{G+G^T}{2}, \frac{1}{2b_0}\right).$$

The network $\frac{G+G^T}{2}$ is the average interaction network, and it represents the average interaction between pairs of agents in network G. Intuitively, the centrality $\mathcal{K}\left(\frac{G+G^T}{2}, \frac{1}{2b_0}\right)$ measures how "central" each node is with respect to the average interaction network.

The optimal prices in Theorem 3 have three components. The first component can be thought of as a nominal price, which is charged to all agents irrespective of the network structure. The second term is a markup that the monopolist can impose on the price of consumer i due to the utility the latter derives from her peers. Finally, the third component can be seen as a discount term, which is offered to a consumer, since increasing her consumption increases the consumption level of her peers. Theorem 3 suggests that it is optimal to give each agent a markup proportional to the utility she derives from the central agents. In contrast, prices offered to the agents should be discounted proportionally to their influence on central agents. Therefore, it follows that the nodes which pay the most favorable prices are the ones, that *influence* highly central nodes.

Note that if Assumption 3 fails, then Theorem 3 can be modified to relate the optimal prices to centrality measures in the underlying graph. In particular, the price structure is still as given in Theorem 3, but when the parameters $\{a_i\}$ and $\{b_i\}$ are not identical, the discount and markup terms are proportional to a weighted version of the Bonacich centrality measure, defined below.

Definition 3 (Weighted Bonacich Centrality). *For a network with (weighted) adjacency matrix* G*, diagonal matrix* D *and weight vector* $\mathbf{v}$*, the weighted Bonacich centrality vector is given by* $\tilde{\mathcal{K}}(G, D, \mathbf{v}) = (I - GD)^{-1}\mathbf{v}$ *provided that* $(I - GD)^{-1}$ *is well defined and nonnegative.*

We next characterize the optimal prices in terms of the weighted Bonacich centrality measure.

Theorem 4. *Under Assumptions 1 and 2 the vector of optimal prices is given by*

$$\mathbf{p} = \frac{\mathbf{a} + c\mathbf{1}}{2} + G\Lambda^{-1}\tilde{\mathcal{K}}\left(\tilde{G}, \Lambda^{-1}, \tilde{\mathbf{v}}\right) - G^T\Lambda^{-1}\tilde{\mathcal{K}}\left(\tilde{G}, \Lambda^{-1}, \tilde{\mathbf{v}}\right),$$

where $\tilde{G} = \frac{G+G^T}{2}$ *and* $\tilde{\mathbf{v}} = \frac{\mathbf{a}-c\mathbf{1}}{2}$.

4.2 Choosing a Single Uniform Price

In this subsection we characterize the equilibria of the pricing-consumption game, when the monopolist can only set a single uniform price, i.e., $p_i = p_0$ for all i. Then, for any fixed p_0, the payoff function of agent i is given by

$$u_i(x_i, \mathbf{x_{-i}}, p_i) = a_i x_i - b_i x_i^2 + x_i \cdot \sum_{j \in \{1,\cdots,n\}} g_{ij} \cdot x_j - p_i x_i,$$

and the payoff function for the monopolist is given by

$$\begin{array}{ll} \max_{p_0 \in [0,\infty)} & (p_0 - c)\sum_i x_i \\ s.t. & \mathbf{x} \in C[\mathbf{p}_0], \end{array}$$

where $\mathbf{p}_0 = (p_0, \cdots, p_0)$. Note that Theorem 1 implies that even when the monopolist offers a single price, the consumption game has a unique equilibrium point. Next lemma states that the consumption of each agent decreases monotonically in the price.

Lemma 2. *Let* $\mathbf{x}(\mathbf{p_0})$ *denote the unique equilibrium in the game where* $p_i = p_0$ *for all* i. *Then,* $x_i(\mathbf{p_0})$ *is weakly decreasing in* p_0 *for all* $i \in \mathcal{I}$ *and strictly decreasing for all* i *such that* $x_i(\mathbf{p_0}) > 0$.

Next, we introduce the notion of the centrality gain.

Definition 4 (Centrality Gain). *In a network with (weighted) adjacency matrix* G, *for any diagonal matrix* D *and weight vector* $\mathbf{v}$, *the centrality gain of agent* i *is defined as*

$$H_i(G, D, \mathbf{v}) = \frac{\tilde{\mathcal{K}}_i(G, D, \mathbf{v})}{\tilde{\mathcal{K}}_i(G, D, \mathbf{1})}.$$

The following theorem provides a characterization of the consumption vector at equilibrium as a function of the single uniform price p.

Theorem 5. *Consider game* $\bar{\mathcal{G}} = \{\mathcal{I}, \{u_i\}_{i\in\mathcal{I}}, [0,\infty)_{i\in\mathcal{I}}\}$, *and define*

$$D_1 = \arg\min_{i\in\mathcal{I}} H_i\left(G, \Lambda^{-1}, \mathbf{a}\right) \quad \text{and} \quad p_1 = \min_{i\in\mathcal{I}} H_i\left(G, \Lambda^{-1}, \mathbf{a}\right).$$

Moreover, let $I_k = \mathcal{I} - \cup_{i=1}^k D_i$ *and define*

$$D_k = \arg\min_{i\in I_k} H_i\left(G_{I_k}, \Lambda_{I_k}^{-1}, \mathbf{a}_{I_k}\right) \quad \text{and} \quad p_k = \min_{i\in I_k} H_i\left(G_{I_k}, \Lambda_{I_k}^{-1}, \mathbf{a}_{I_k}\right),$$

for $k \in \{2, 3 \ldots n\}$. *Then,*

(1) p_k *strictly increases in* k.
(2) *Given a* p *such that* $p < p_1$, *all agents purchase a positive amount of the good, i.e.,* $x_i(p) > 0$ *for all* $i \in \mathcal{I}$, *where* $\mathbf{x}(\mathbf{p})$ *denotes the unique consumption equilibrium at price* p. *If* $k \geq 1$, *and* p *is such that* $p_k \leq p \leq p_{k+1}$, *then* $x_i(p) > 0$ *if and only if* $i \in I_k$. *Moreover, the corresponding consumption levels are given as in* (6), *where* $S = I_k$.

Theorem 5 also suggests a polynomial time algorithm for computing the optimal uniform price p_{opt}. Intuitively, the algorithm sequentially removes consumers with the lowest centrality gain and computes the optimal price for the remaining consumers under the assumption that the price is low enough so that only these agents purchase a positive amount of the good at the associated consumption equilibrium. In particular, using Theorem 5, it is possible to identify the set of agents who purchase a positive amount of the good for price ranges $[p_k, p_{k+1}]$, $k \in \{1, \ldots\}$. Observe that given a set of players, who purchase a positive amount of the good, the equilibrium consumption levels can be obtained in closed form as a linear function of the offered price, and, thus, the profit function of the monopolist takes a quadratic form in the price. It follows that for each price range, the maximum profit can be found by solving a quadratic optimization problem. Thus, Theorem 5 suggests Algorithm 1 for finding the optimal single uniform price p_{opt}.

Algorithm 1. Compute the optimal single uniform price p_{opt}

STEP 1. Preliminaries:

- Initialize the set of *active* agents: $S := \mathcal{I}$.
- Initialize $k = 1$ and $p_0 = 0$, $p_1 = \min_{i \in \mathcal{I}} H_i(G_{\mathcal{I}}, \Lambda_{\mathcal{I}}^{-1}, \mathbf{a}_{\mathcal{I}})$
- Initialize the monopolist's revenues with $Re_{opt} = 0$ and $p_{opt} = 0$.

STEP 2.

- Let $\hat{p} = \frac{\mathbf{1}^T(\Lambda_S - G_S)^{-1}\mathbf{a}_S - c\mathbf{1}^T(\Lambda_S - G_S)^{-1}\mathbf{1}}{\mathbf{1}^T\left((\Lambda_S - G_S)^{-1} + (\Lambda_S - G_S^T)^{-1}\right)\mathbf{1}}$
- **IF** $\hat{p} \geq p_k$, let $p = p_k$.
  ELSE IF $\hat{p} \leq p_{k-1}$, let $p = p_{k-1}$ **ELSE** $p = \hat{p}$.
- $Re = (p - c)\mathbf{1}^T \cdot (\Lambda_S - G_S)^{-1}(\mathbf{a}_S - p\mathbf{1})$.
- **IF** $Re > Re_{opt}$ **THEN** $Re_{opt} = Re$ and $p_{opt} = p$.
- $D = \arg\min_{i \in S} H_i(G_S, \Lambda_S^{-1}, \mathbf{a}_S)$ and $S := S - D$.
- Increase k by 1 and let $p_k = \min_{i \in S} H_i(G_S, \Lambda_S^{-1}, \mathbf{a}_S)$.
- Return to **STEP 2** if $S \neq \emptyset$ **ELSE Output** p_{opt}.

The algorithm solves a series of subproblems, where the monopolist is constrained to choose a price p in a given interval $[p_k, p_{k+1}]$ with appropriately chosen endpoints. In particular, from Theorem 5, we can choose those endpoints, so as to ensure that only a particular set S of agents purchase a positive amount of the good. In this case, the consumption at price p is given by $(\Lambda_S - G_S)^{-1}(\mathbf{a}_S - p\mathbf{1})$ and the profit of the monopolist is equal to

$(p-c)\mathbf{1}^T \cdot (\Lambda_S - G_S)^{-1}(\mathbf{a}_S - p\mathbf{1})$. The maximum of this profit function is achieved at $\hat{p} = \frac{\mathbf{1}^T(\Lambda_S - G_S)^{-1}\mathbf{a}_S - c\mathbf{1}^T(\Lambda_S - G_S)^{-1}\mathbf{1}}{\mathbf{1}^T\left((\Lambda_S - G_S)^{-1} + (\Lambda_S - G_S^T)^{-1}\right)\mathbf{1}}$, as can be seen from the first order optimality conditions. Then, the overall optimal price is found by comparing the monopolist's profits achieved at the optimal solutions of the constrained subproblems. The complexity of the algorithm is $O(n^4)$, since there are at most n such subproblems (again from Theorem 5) and each such subproblem simply involves a matrix inversion ($O(n^3)$) in computing the centrality gain and the maximum achievable profit.

4.3 The Case of Two Prices: Full and Discounted

In this subsection, we assume that the monopolist can choose to offer the good in one of two prices, p_L and p_H ($p_L < p_H$) that are exogenously defined. For clarity of exposition we call p_L and p_H the discounted and the full price respectively. The question that remains to be studied is to which agents should the monopolist offer the discounted price, so as to maximize her revenues. We state the following assumption that significantly simplifies the exposition.

Assumption 4. *The exogenous prices p_L, p_H are such that $p_L, p_H < \min_{i \in \mathcal{I}} a_i$.*

Note that under Assumption 4, Equation (2) implies that all agents purchase a positive amount of the good at equilibrium, regardless of the actions of their peers. As shown previously, the vector of consumption levels satisfies $\mathbf{x} = \Lambda^{-1}(\mathbf{a} - \mathbf{p} + G\mathbf{x})$, and hence $\mathbf{x} = (\Lambda - G)^{-1}(\mathbf{a} - \mathbf{p})$. An instance of the monopolist's problem can now be written as:

$$\begin{array}{lll} (OPT) & \max & (\mathbf{p} - c\mathbf{1})^T(\Lambda - G)^{-1}(\mathbf{a} - \mathbf{p}) \\ & st. & p_i \in \{p_L, p_H\} \quad \text{for all } i \in \mathcal{I}, \end{array}$$

where $\Lambda \succ 0$ is a diagonal matrix, G is such that $G \geq 0$, $diag(G) = 0$ and Assumption 1 holds.

Let $p_N \triangleq \frac{p_H + p_L}{2}$, $\delta \triangleq p_H - p_N$, $\hat{\mathbf{a}} \triangleq \mathbf{a} - p_N\mathbf{1}$ and $\hat{c} \triangleq p_N - c \geq \delta$. Using these variables, and noting that any feasible price allocation can be expressed as $\mathbf{p} = \delta\mathbf{y} + p_N\mathbf{1}$, where $y_i \in \{-1, 1\}$, OPT can alternatively be expressed as

$$\begin{array}{ll} \max & (\delta\mathbf{y} + \hat{c}\mathbf{1})^T(\Lambda - G)^{-1}(\hat{\mathbf{a}} - \delta\mathbf{y}) \\ s.t. & y_i \in \{-1, 1\} \quad \text{for all } i \in \mathcal{I}. \end{array} \tag{8}$$

We next show that OPT is NP-hard, and provide an algorithm that achieves an approximately optimal solution. To obtain our results, we relate the alternative formulation of OPT in (8) to the MAX-CUT problem (see [11,12]).

Theorem 6. *Let Assumptions 1, 2 and 4 hold. Then, the monopolist's optimal pricing problem, i.e., problem OPT, is NP-hard.*

Theorem 7. *Let Assumptions 1 and 4 hold and W_{OPT} denote the optimal profits for the monopolist, i.e., W_{OPT} is the optimal value for problem OPT. Then,* *there exists a randomized polynomial time algorithm, that outputs a solution with objective value* W_{ALG} *such that* $E[W_{ALG}] + m > 0.878(W_{OPT} + m)$, *where*

$$m = \delta^2 \mathbf{1}^T A \mathbf{1} + \delta \mathbf{1}^T \left| A\hat{\mathbf{a}} - A^T \hat{c}\mathbf{1} \right| - \hat{c}\mathbf{1}^T A \hat{\mathbf{a}} - 2\delta^2 Trace(A),$$

and $A = (\Lambda - G)^{-1}$.

In the remainder of the section, we provide a characterization of the optimal prices in OPT. In particular, we argue that the pricing problem faced by the monopolist is equivalent to finding the cut with maximum weight in an appropriately defined weighted graph. For simplicity, assume that $b_i = b_0$ for all i and $\left((\Lambda - G)^{-1}\hat{\mathbf{a}} - \hat{c}(\Lambda - G)^{-T}\mathbf{1}\right) = 0$ (which holds, for instance when $\hat{\mathbf{a}} = \hat{c}\mathbf{1}$, or equivalently $\mathbf{a} - p_N \mathbf{1} = (p_N - c)\mathbf{1}$, and $G = G^T$). Observe that in this case, the alternative formulation of the profit maximization problem in (8), can equivalently be written as (after adding a constant to the objective function, and scaling):

$$\begin{aligned} \max \quad & \alpha - \mathbf{y}(\Lambda - G)^{-1}\mathbf{y} \\ s.t. \quad & y_i \in \{-1, 1\} \quad \text{for all } i \in \mathcal{I}, \end{aligned} \tag{9}$$

where $\alpha = \sum_{ij} (\Lambda - G)^{-1}_{ij}$. It can be seen that this optimization problem is equivalent to an instance of the MAX-CUT problem, where the cut weights are given by the off diagonal entries of $(\Lambda - G)^{-1}$ (see [11,12]). On the other hand observe that $(\Lambda - G)^{-1}\mathbf{1} = \frac{1}{2b_0}(I - \frac{1}{2b_0}G)^{-1}\mathbf{1}$, hence, the ith row sum of the entries of the matrix $(\Lambda - G)^{-1}$ is proportional to the centrality of the ith agent in the network. Consequently, the (i, j)th entry of the matrix $(\Lambda - G)^{-1}$, gives a measure of how much the edge between i and j contributes to the centrality of node i. Since the MAX-CUT interpretation suggests that the optimal solution of the pricing problem is achieved by maximizing the cut weight, it follows that the optimal solution of this problem price differentiates the agents who affect the centrality of each other significantly.

5 Conclusions

The paper studies a stylized model of pricing of divisible goods (services) over social networks, when consumers' actions are influenced by the choices of their peers. We provide a concrete characterization of the optimal scheme for a monopolist under different restrictions on the set of allowable pricing policies when consumers behave according to the unique Nash equilibrium profile of the corresponding game. We consider a setting of static pricing: the monopolist first sets prices and then the consumers choose their usage levels. Moreover, the game we define is essentially of complete information, since we assume that both the monopolist, as well as the consumers, know the network structure and the utility functions of the population. Extending our analysis by introducing incomplete information is an interesting direction for future research. Concretely, consider a monopolist that introduces a new product of unknown quality to a market. Agents benefit the monopolist in two ways when purchasing the product; directly by increasing her revenues, and indirectly by generating information about the product's quality and making it more attractive to the rest of the consumer pool. What is the optimal (dynamic) pricing strategy for the monopolist?

Finally, note that in the current setup we consider a single seller (monopolist), so as to focus on explicitly characterizing the optimal prices as a function of the network structure. A natural departure from this model is studying a competitive environment. The simplest such setting would involve a small number of sellers offering a perfectly substitutable good to the market. Then, pricing may be even more aggressive than in the monopolistic environment: sellers may offer even larger discounts to "central" consumers, so as to subsequently exploit the effect of their decisions to the rest of the network. Potentially one could relate the *intensity of competition* with the network structure. In particular, one would expect the competition to be less fierce when the network consists of disjoint large subnetworks, since then sellers would segment the market at equilibrium and exercise monopoly power in their respective segments.

References

1. Ballester, C., Calvo-Armengol, A., Zenou, Y.: Who's who in networks. wanted: the key player. Econometrica 74(5), 1403–1417 (2006)
2. Bonacich, P.: Power and centrality: A family of measures. The American Journal of Sociology 92(5), 1170–1182 (1987)
3. Bramoullé, Y., Kranton, R.: Public goods in networks. Journal of Economic Theory 135, 478–494 (2007)
4. Bramoullé, Y., Kranton, R., D'Amours, M.: Strategic interaction and networks. Working paper (2009)
5. Campbell, A.: Tell your friends! word of mouth and percolation in social networks. Working paper (2009)
6. Candogan, O., Bimpikis, K., Ozdaglar, A.: Optimal pricing in the presence of local network effects. Working paper (2010)
7. Corbo, J., Calvo-Armengol, A., Parkes, D.C.: The importance of network topology in local contribution games. In: Proceedings of the 3rd International Workshop on Internet and Network Economics (2007)
8. Farrell, J., Saloner, G.: Standardization, compatibility, and innovation. RAND Journal of Economics 16(1), 70–83 (1985)
9. Galeotti, A., Goyal, S.: Influencing the influencers: a theory of strategic diffusion. RAND Journal of Economics 40(3), 509–532 (2009)
10. Galeotti, A., Goyal, S., Jackson, M., Vega-Redondo, F., Yariv, L.: Network games. Review of Economic Studies 77(1), 218–244 (2010)
11. Garey, M.R., Johnson, D.S.: Computers and intractability. A guide to the theory of NP-completeness. A Series of Books in the Mathematical Sciences. WH Freeman and Company, San Francisco (1979)
12. Goemans, M., Williamson, D.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of the ACM 42(6), 1115–1145 (1995)
13. Hartline, J., Mirrokni, V., Sundararajan, M.: Optimal marketing strategies over social networks. In: Proceedings of the 17th International Conference on World Wide Web (2008)

14. Johari, R., Kumar, S.: Congestible services and network effects. Working paper (2010)
15. Katz, M., Shapiro, C.: Technology adoption in the presence of network externalities. Journal of Political Economy 94(4), 822–841 (1986)
16. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2003)
17. MacKie-Mason, J., Varian, H.: Pricing congestible network resources. IEEE Journal of Selected Areas in Communications (1995)
18. Sundararajan, A.: Local network effects and complex network structure. The BE Journal of Theoretical Economics 7(1) (2007)
19. Topkis, D.M.: Supermodularity and complementarity. Princeton University Press, Princeton (1998)

Local Dynamics in Bargaining Networks via Random-Turn Games

L. Elisa Celis[1,⋆], Nikhil R. Devanur[2], and Yuval Peres[2]

[1] University of Washington
ecelis@cs.washington.edu
[2] Microsoft Research
{nikdev,peres}@microsoft.com

Abstract. We present a new technique for analyzing the rate of convergence of local dynamics in bargaining networks. The technique reduces balancing in a bargaining network to optimal play in a random-turn game. We analyze this game using techniques from martingale and Markov chain theory. We obtain a tight polynomial bound on the rate of convergence for a nontrivial class of unweighted graphs (the previous known bound was exponential). Additionally, we show this technique extends naturally to many other graphs and dynamics.

1 Introduction

In a network bargaining game, nodes in a graph are involved in pairwise transactions with their neighbors. This type of game was introduced by Cook and Yamagishi [14] to capture the "power" of a node derived from its position in a network, and has also been used in economics to model two-sided markets [33,31]. Recently these games have been analyzed from a computational point of view, first in a centralized model [23] and later in a distributed model [3]. Analyzing simple, local dynamics that converge quickly to an equilibrium in such games was an important open problem that attracted much interest [18,22,21].

We draw a connection between network bargaining games and random-turn games. Random-turn games are a well-studied class of two-player combinatorial games in which the outcome of a coin flip determines which player moves next [25,24]. Combinatorial games can be represented as a game on a directed graph where players move a token along edges until one reaches their goal state. We transform the network bargaining game into an equivalent random-turn game which we can analyze using martingale techniques to obtain bounds on the rate of convergence. In particular, the convergence rate for the dynamics is related to the *absorption time* of the corresponding random-turn game.

We obtain a tight polynomial bound on the convergence rate for a variety of natural dynamics on a certain class of graphs. This class includes unweighted bipartite graphs with unique balanced outcomes, and the exposition is conducted in this setting for clarity. The previous bound known for any class of graphs (other than paths) was exponential.

⋆ Research supported by a NASA Graduate Fellowship.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 133–144, 2010.

Network Bargaining Game

A network bargaining game is defined on a weighted graph $G = (V', E')$ with $w' : E' \to \mathbb{R}_+$.[1] Every node in the graph is a player, and the weight of an edge represents the dollar amount available to be shared between the two adjacent players. However, each player is constrained to make at most one such sharing agreement. An *outcome* of this game is a *matching* in the graph $M \subseteq E$ and an *allocation* describing each player's profit, $f' : V' \to \mathbb{R}_+$ where for all $(uv) \in M$, we have $f'(u) + f'(v) = w'(uv)$, and for all unmatched $u \in V'$, $f'(u) = 0$.

We consider two notions of equilibrium in this game. The first (weaker) notion is that of a *stable* outcome: an outcome is stable if for all unmatched edges $(uv) \notin M$ we have $f'(u) + f'(v) \geq w'(uv)$, that is, no two adjacent players have incentive to deviate from their current matches. The second notion is that of a *balanced* outcome: an outcome is balanced if matched players divide the *surplus* equally amongst themselves. To be precise, let the *best alternate* of a node u be

$$\alpha_{f'}(u) := \max\{0, \max_{v:(uv) \in E' \setminus M} \{w'(uv) - f'(v)\}\},$$

i.e. the maximum profit a player could get from a neighbor she is not currently matched to. For every matched edge (uv) define the surplus as

$$s_{f'}(uv) = w'(uv) - (\alpha_{f'}(u) + \alpha_{f'}(v)).$$

An outcome is balanced if it is stable and for all matched edges (uv), $f'(u) = \alpha_{f'}(u) + s_{f'}(uv)/2$ and $f'(v) = \alpha_{f'}(v) + s_{f'}(uv)/2$, or equivalently, $f'(u) - \alpha_{f'}(u) = f'(v) - \alpha_{f'}(v)$. This can be seen as a generalization of Nash's bargaining solution for two players [28]. It is known that the following are equivalent: *(1)* a balanced outcome exists, *(2)* a stable outcome exists and *(3)* the matching polytope has no integrality gap [23].

Edge Balancing Dynamics

Balanced outcomes can be computed by centralized polynomial time algorithms [23], but the game is by nature distributed; individual players working on individual deals. An important open problem was to show there exist simple and natural *local dynamics* that converge quickly to a balanced outcome. We now define such dynamics with respect to a matching M and initial allocation f'.

For our dynamics, the matching M is fixed throughout. This may seem counter to the solution concept of a balanced outcome since the premise is the threat of switching partners. However, once such a threat is acknowledged, the players do not need to switch in order to bargain for their fair share. Moreover, there are distributed dynamics that find matchings [6,32] which also have a bargaining flavor in their dynamics. One can imagine a two phase approach, where in the first phase the players find a matching and in the second find a balanced outcome with the matching fixed.

[1] We reserve the notation (V, E) and w for a graph which will be used more prevalently in the random-turn game framework.

The allocations are updated synchronously, and the updates proceed in rounds. The allocation in round t is denoted by $B'_{f'}(u,t)$, the best alternatives by $\alpha_{f'}(u,t)$ and the surpluses by $s_{f'}(uv,t)$.[2] The initial allocation is $B'_{f'}(v,0) = f'(v)$. SYNCHRONOUS EDGE BALANCING is defined by the following update rule: for all $u \in V'$, $(uv) \in M$ and $t \geq 1$,

$$B'_{f'}(u,t+1) \leftarrow \alpha_{f'}(u,t) + s_{f'}(uv,t)/2.$$

Thus, the allocation for the next round is determined by "balancing" each matched edge using the allocation in the current round.

We say that an allocation f' is *ε-close to balanced* if there exists a balanced outcome B' such that $|B'(v) - f'(v)| \leq \varepsilon$ for all v, i.e. we get ε-close to a balanced outcome. Note that this is stronger than a common alternate notion of ε-balanced where $|B'_{f'}(u,t+1) - B'_{f'}(u,t)| \leq \varepsilon$, i.e. each edge is locally balanced. We wish to show SYNCHRONOUS EDGE BALANCING converges rapidly to a balanced outcome. This means that for all f', after polynomially many[3] time steps t, the allocation $B_{f'}(u,t)$ is ε-close to a balanced outcome.

Random-Turn Games

Every two-player game from Tic-Tac-Toe to Chess can be formalized as a combinatorial game on a directed graph where each turn consists of moving a token from one vertex to another along an edge [7]. Random-turn games are combinatorial games where the turns are determined by a coin flip.

We consider the following version in the main body of this paper: A RANDOM-TURN GAME consists of a directed graph $D = (V, E)$, payoff function $f : V \rightarrow [0,1]$, initial vertex v_0, and horizon $T \in \mathbb{N}$. The set V of game states contains two terminal states s and r and all payoff functions set $f(s) = 0$ and $f(r) = 1$. The game is played by Max and $Mini$ where Max's goal is to maximize the value of the end state, and $Mini$'s goal is to minimize it. Game play for horizon T is as follows: a token is initially placed at v_0 and at every step a fair coin is tossed to determine who gets to move the token. Max must always move to a predecessor of v and $Mini$ to a successor (as determined by the edge set E). We repeat until either T moves have been made, or we reach an absorbing state $\{s, r\}$. At the end of the game, $Mini$ pays Max \$$f(v)$ if the game terminates at node v. Since this is a full-information game, for any finite horizon, one can compute the optimal strategies for the two players. This defines a *value* of the game, which is the expected payoff for Max under optimal play.

Related Work and Motivation

Network bargaining games have a long history in two communities: sociology and game theory. In sociology, they are studied under the name *network exchange theory*, where the goal is to understand the power of a node as a function of its

[2] The subscript f' may be dropped when it is clear from context.

[3] Where the polynomial is in $|V|, |E|$ and $1/\log(\varepsilon)$.

position in the network (see the overview by Willer [35]). Network bargaining games as we define here were introduced by Cook and Yamagishi [14], who also introduced the notion of balanced outcomes. In fact, they also introduced local dynamics similar to what we consider in this paper, but without a theoretical analysis of the convergence of their dynamics. There have also been experimental results [13,8] which validate the relevance and applicability of this work.

In game theory, the study of bargaining can be traced back to Nash's bargaining solution [28]. Many results in this field focus on two-sided markets, which naturally give rise to the bipartite version of the network bargaining game as was introduced by Shapley and Shubik [33]. This version, known as the *assignment game*, can also be viewed as the classic Gale-Shapley stable marriage problem [19] with the addition transferable utilities. Rochford [31] defined balanced outcomes for assignment games under the name symmetrically pairwise-bargained allocations. She also showed that they are the intersection of the core and the kernel, two common solution concepts in co-operative game theory. Other solution concepts such as the nucleolus [27] have also been considered. In fact, the computability of these solution concepts has been much studied [34,29]. Other related models consider price setting as a result of a bargaining process [15].

Network bargaining games were introduced to the theoretical computer science community by Kleinberg and Tardos [23]. They gave a polynomial time algorithm to compute the set of balanced outcomes. Since then, there has been a flurry of activity: Azar, et al. [3] considered an asynchronous version of edge balancing dynamics and showed (exponential time) convergence. Other aspects of network bargaining have also been studied in the recent past [5,10,9,4,21].

We give the first polynomial time bound on local dynamics converging to a balanced outcome for any non-trivial class of graphs. The only polynomial time bound known previously was for paths. Moreover, the bounds are tight for a variety of dynamics. Independently and concurrently with our work, Kanoria, et al. [22] considered the same problem and showed convergence of a (different) dynamics to a balanced outcome. The dynamics they consider has the advantage that it does not need a matching to be known and fixed; rather, the dynamics also finds a matching. One drawback is that the outcome their process converges to is weaker (it is ε-balanced as opposed to ε-close to a balanced outcome). Additionally, the rate of convergence of their dynamics is weaker and of the form n^4/g^2 where no bounds on g are given. In fact, on many graphs where our result is tight, g could be zero.[4] Also independently and concurrently, Draief and Vojnovic [17] showed quadratic convergence of the edge balancing dynamics for the following graphs: a path, a cycle, a blossom and a bicycle. Faigle, Kern and Kuipers [18] also considered similar local dynamics for a more general class of games, but do not show bounds on the rate of convergence.

In general, analysis of the convergence of local dynamics to an equilibrium of a game is a common theme. Examples include analysis of random best response dynamics for the Gale-Shapley stable matching game [2,19]. In fact, a

[4] For instance, this occurs on any unweighted even length path.

major philosophical hypothesis of algorithmic game theory [20,12,16] is that the existence of such dynamics is crucial to validate a solution concept.

Random-turn games are natural, and many variants have been analyzed [24,25]. Most interestingly, a variant called the tug-of-war game has been found to be related to partial differential equations such as the infinity Laplacian and the p-Laplacian [30], due to which these games have received considerable attention [11,37,1].

Organization

In Section 2 we introduce our theorems, techniques and extensions. Section 3 contains a detailed analysis for unweighted bipartite graphs with unique balanced outcomes. We conclude and suggest future work in Section 4.

2 From a Bargaining Game to a Random-Turn Game

We now give a reduction from a network bargaining game to a random-turn game, the concept that lies at the heart of our results. We first restrict ourselves to unweighted bipartite graphs for clarity.

Consider a graph $G = (V', E')$ where $w'(uv) = 1$ for all $(uv) \in E'$ and V' is bipartitioned as $\{L, R\}$. Create a directed graph D as follows: let $D = (V, E)$ where V is the subset of matched vertices in L along with two special vertices, s and r. Let the set of vertices other than s and r be denoted by $\dot{V}$. Add an edge $(uv) \in E$ if $(M(u)v) \in E'$. Additionally, place an edge from s to all vertices in $\dot{V}$ and an edge from all vertices in $\dot{V}$ to r. Finally, add an (rv) edge in E if there exists an edge $(vu) \in E'$ where $u \notin M$. Similarly, add a (vs) edge if there is a $(M(v)u)$ edge with $u \notin M$. We also give an allocation $f : V \to [0, 1]$ on D, given the allocation f' on G. Define $f(v) = f'(v)$ if $v \in \dot{V}$, $f(s) = 0$ and $f(r) = 1$. See Figure 1 for an example of this reduction. Note that an allocation f' on G takes values between 0 and 1 since the edge weights all have weight 1. Thus, the definition of an allocation allows us to reconstruct f' from f, since $f'(M(v)) = 1 - f'(v)$ and $f(u) = 0$ when $u \notin M$.
The concepts (from the bargaining game described earlier) translate as follows.

- An allocation is *stable* if for all edges $(uv) \in E$, $f(u) \le f(v)$.
- Let the *best predecessor* and *successor* of a node v be $v_f^+ = \arg\max_{u:(uv)\in E}\{f(u)\}$ and $v_f^- = \arg\min_{u:(vu)\in E}\{f(u)\}$ respectively. An allocation is *balanced* if it is stable, and for all vertices $v \in \dot{V}$, $f(v) = \frac{1}{2}(f(v_f^+) + f(v_f^-))$.
- Let the allocation in round t of Synchronous Edge Balancing be $B_f(v, t)$ where $B_f(v, 0) = f(v)$. Then, balancing is equivalent to $B_f(v, t+1) = \frac{1}{2}(B_f(v^+, t) + B_f(v^-, t))$.

An interesting aspect of this reduction is the *time reversal*. By that we mean that if one considers a T-horizon Random-Turn Game and T steps of Synchronous Edge Balancing, then the first step of Synchronous Edge Balancing actually corresponds to the last step in the Random-Turn Game. In general, the t^{th} balancing step corresponds to t steps remaining in the game.

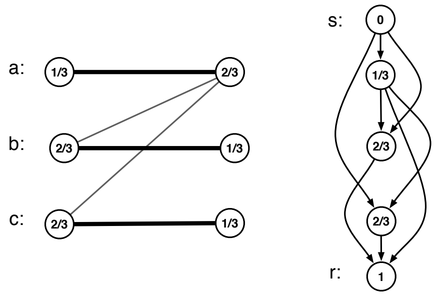


Fig. 1. An unweighted bipartite graph G and its corresponding digraph D with balanced allocations

Throughout this paper, we say a graph D is *weakly acyclic* if the only directed cycles it contains go through s or r. If a graph G reduces to a digraph D that is weakly acyclic then the balanced outcome on G is *unique*. The converse also holds for unweighted bipartite graphs.

Consider the Random-Turn Game defined by the digraph $D = (V, E)$ and the payoff function f as above. The following theorem relates the value of the Random-Turn Game to Synchronous Edge Balancing, and shows it is sufficient to analyze the convergence of the Random-Turn Game.

Theorem 1. *The value of a* Random-Turn Game *with starting vertex v and horizon T is exactly $B(v, T)$ when the directed graph is weakly acyclic.*

Let the balanced outcome be denoted by $B(v)$. For such games, we give the *optimal* rate of convergence, which is as follows. Let h be the maximum length of a path from s to r in D.

Theorem 2. *There exists a $T \in O(h^2 \log(1/\varepsilon))$ such that for all $t \geq T$ the value of the* Random-Turn Game *starting at vertex v with horizon t is within ε of $B(v)$, given that D is weakly acyclic.*

The proof of this theorem is the most technical part of the paper, and uses techniques from the theory of martingales. Recall that an allocation f' is *ε-close to balanced* if there exists a balanced outcome B' such that $|B'(v) - f'(v)| \leq \varepsilon$ for all v. We can now restate the result and the corresponding rate of convergence in Synchronous Edge Balancing. The proofs are the focus of Section 3.

Theorem 3. Synchronous Edge Balancing *on unweighted bipartite graphs with a unique balanced outcome results in an allocation that is ε-close to a balanced outcome after at most $O(|M|^2 \log(1/\varepsilon))$ rounds of the balancing process.*

This result follows directly from Theorem 2 and the fact that $h \leq |V| = |M| + 2$. Lastly, we show our result is tight.

Theorem 4. *There exist graphs G with matchings M and initial allocations such that the balancing process requires $\Omega(|M|^2 \log(1/\varepsilon))$ time to be ε-close to a balanced outcome.*

Sketch of Convergence Proof

We now give a brief sketch of the proof of Theorem 2 for the case when M is a *perfect matching* in G. Observe that if a game with finite horizon ends in an absorbing state, then the vertex payoffs don't matter. Thus one approach is to show that with high probability, a Random-Turn Game with a sufficiently large horizon ends in an absorbing state. To be precise, let $\{X_t\}$ be a sequence of vertices in a run of the Random-Turn Game under optimal play. We wish to show that for a game with sufficiently large horizon T, $X_T \in \{s, r\}$ with high probability. However, it is unclear how to analyze the behavior of X_t. Instead we show it is sufficient to analyze the related sequence of vertices $\{Y_t\}$ obtained when Max plays optimally, but $Mini$ plays as if the payoff function was B. We show $B(Y_t)$, the value of the balanced outcome of vertex Y_t, is a supermartingale. Moreover, we know that it is bounded in $[0, 1]$ and show that its conditional variance is at least $1/h^2$. These suffice to prove the desired bound on the absorption time.

Extensions

To summarize, the approach outlined to prove convergence of Synchronous Edge Balancing is as follows: reduce it to convergence of a Random-Turn Game (Theorem 1) and show bounds on this game (Theorem 2). The first part of this approach can be extended naturally to show convergence (but not rates) for may variants of the dynamics and general graphs. For *non-bipartite graphs* we maintain a vertex in D for each matched vertex in G. If the graph is *weighted* we use running payoffs in the random-turn game. *Damped dynamics* correspond to lazy random-turn games. And if we wish vertices to be *individual rational*, then the corresponding *capped dynamics* are captured by a random-turn game where the players are allowed to quit. This list is far from exhaustive, but illustrates the flexibility and robustness of our technique and is discussed further in the full version of this paper.

3 Rate of Convergence

We begin with the proof of Theorem 1. We recall some notation: given an allocation f, $v_f^+ = \arg\max_{u:uv \in E}\{f(u)\}$ and $v_f^- = \arg\min_{u:vu \in E}\{f(u)\}$. The allocation in round t of Synchronous Edge Balancing is $B_f(v, t)$ (we now drop the subscript f for convenience). The updates are, $B(v, t + 1) = \frac{1}{2}(B(v^+, t) + B(v^-, t))$ where v^+ and v^- are defined with respect to $B(v, t)$. Theorem 1 says that $B(v, T)$ is the value of the Random-Turn Game starting at vertex v with horizon T. The proof is by induction on T. We first strengthen the inductive hypothesis to assume the *optimal strategies* for Max and $Mini$ are to choose v^+ and v^- respectively. We refer to this strategy as the *balancing strategy.*

Theorem 5. *Given a* Random-Turn Game *with horizon T, the optimal strategy for either player is the balancing strategy.*

Proof (Theorems 1 and 5). The proof is by a joint induction on the horizon t to prove (a) $B(v,t)$ is the value of the game and (b) the optimal strategy when $t+1$ moves remain is the balancing strategy.

In the base case, $t=0$. To show (a), note that the expected payoff of the game for Max at node v is exactly $B_f(v,0)=f(v)$ since there are no moves to be made. To show (b), consider the horizon $t+1=1$ at a given node v. In this case, optimal moves for Max and $Mini$ are clearly v_f^+ and v_f^- respectively, since the payoff at the end of this turn will be the terminal payoff of the game.

For the inductive step, let us assume that for all v and some $t \in \mathbb{N}$, the value of the game of horizon $t-1$ is $B_f(v,t-1)$, and in the t horizon game the bargaining strategy is optimal. To prove (a) we note that the latter statement implies Max will move to $v^+_{B(v,t-1)}$ if he wins the coin toss and $Mini$ will move to $v^-_{B(v,t-1)}$ if she wins the coin toss. From the first part of the inductive hypothesis we know $B(v,t-1)$ is the expected payoff for Max in the $t-1$ horizon game. Thus, the expected payoff of the game for Max under optimal play in the t horizon game is $\frac{1}{2}(B(v^+,t-1)+B(v^-,t-1)) = B(v,t)$. To prove (b), consider the $t+1$ horizon game. Under optimal play, Max wishes to maximize his expected payoff, and $Mini$ wishes to minimize the expected amount she has to pay. Assume we are at vertex u, and recall that Max must move to a predecessor of u and $Mini$ to a successor. Since there are t steps remaining after the initial step, an optimal strategy for Max ($Mini$) will maximize (minimize) the expected payoff $B_f(v,t)$. Thus, if Max wins the toss he will move to $v^+_{B(v,t)}$ and if $Mini$ wins it she will move to $v^-_{B(v,t)}$, which is precisely the balancing strategy. □

We now give the proof of Theorem 2 for the case where we have a perfect matching. Note that with the assumptions of the theorem, this implies D is *strongly acyclic*; i.e. it does not contain cycles of any kind. We briefly explain the technical extension for non-perfect matchings at the end of this section. The main idea behind the proof is to first reduce the analysis to showing that a particular sequence $\{Y_t\}$ (of vertices in V) gets absorbed at $\{s,r\}$ with high probability, and then show this happens in polynomial time using techniques from the theory of martingales.

Proof (Theorem 2). Consider two allocations, f and g such that $f(v) \le g(v)$ for all v. We show in Lemma 1 that $B_f(v,t) \le B_g(v,t)$ for all v,t. Hence, if we consider the initial allocations

$$\mathbf{0}(v) = \begin{cases} 0 \text{ if } v \neq r; \\ 1 \text{ otherwise.} \end{cases} \qquad \text{and} \qquad \mathbf{1}(v) = \begin{cases} 1 \text{ if } v \neq s; \\ 0 \text{ otherwise.} \end{cases},$$

we have $B_{\mathbf{0}}(v,t) \le B_f(v,t) \le B_{\mathbf{1}}(v,t)$ for all v,t, and f. Thus, it suffices to prove that $B_{\mathbf{0}}(v,T) \ge B(v) - \varepsilon$ and $B_{\mathbf{1}}(v,T) \le B(v)+\varepsilon$ for $T \in O(h^2 \log(1/\varepsilon))$. We will prove the latter, and the proof for the former follows exactly with the roles of $Mini$ and Max reversed and the payoff function $\mathbf{0}$ instead of $\mathbf{1}$.

Consider the game with payoff function $\mathbf{1}$ and horizon T where $T \in O(h^2 \log(1/\varepsilon))$. Consider the sequence of vertices $\{X_t\}$ with $X_0 = v$ that occurs if $Mini$ and Max play optimally. From Theorem 1,

$$B_{\mathbf{1}}(v,t) = \mathbb{E}\mathbf{1}(X_t). \tag{1}$$

Now consider the half-optimal sequence $\{Y_t\}$ with $Y_0 = v$, where Max plays optimally for the payoff function $\mathbf{1}$ and $Mini$ plays optimally for the payoff function B. For the game with payoffs $\mathbf{1}$ Max's expected payoff is only higher. That is

$$\mathbb{E}\mathbf{1}(X_t) \le \mathbb{E}\mathbf{1}(Y_t). \tag{2}$$

Our key result in Lemma 3 shows that for any function f, $\mathbb{E}_v[|f(Y_T) - B(Y_T)|] \le \varepsilon$. (The proof of this lemma follows by showing convergence of the sequence $\{Y_t\}$.) If we take $f = \mathbf{1}$ and note that $\mathbf{1}(Y_t) \ge B(Y_t)$, we get

$$\mathbb{E}\mathbf{1}(Y_T) \le \mathbb{E}B(Y_T) + \varepsilon. \tag{3}$$

Now consider the sequence $\{Z_t\}$ with $Z_0 = v$ that occurs when $Mini$ and Max play optimally for the payoff function B. The expected payoff for Max with payoff function B is higher in $\{Z_t\}$ than in $\{Y_t\}$. Thus

$$\mathbb{E}B(Y_T) \le \mathbb{E}B(Z_T). \tag{4}$$

Finally, we show in Lemma 2 that

$$\mathbb{E}B(Z_T) = B(v). \tag{5}$$

From (1) – (5), it follows that $B_{\mathbf{1}}(v,T) \le B(v) + \varepsilon$ as desired. □

Lemma 1. *The balancing process is monotonic, namely if $f(v) \le g(v)$ for all $v \in V$, then $B_f(v,t) \le B_g(v,t)$ for all v, t.*

Lemma 2. *The value of a* Random-Turn Game *with function $f = B$ is equal to B for all horizons $T \in \mathbb{N}$.*

This Lemma follows from Theorem 1 and the observation that B is a fixed point of Synchronous Edge Balancing. A detailed proof of both lemmas can be found in the full version of this paper.

Lemma 3. *Consider the expected payoff for Max in the half-optimal chain $\{Y_t\}$ defined in the proof of Theorem 2. For sufficiently large t, the expected payoff for Max with payoff function f is close to the balanced outcome B. Specifically, $\mathbb{E}_v[|f(Y_T) - B(Y_T)|] \le \varepsilon$ when $T \ge 4h^2 \log(1/\varepsilon)$.*

Proof. Clearly if $Y_t \in \{s, r\}$, then the game has ended and $f(Y_t) - B(Y_t) = 0$. Additionally, the difference $|f(Y_t) - B(Y_t)|$ is at most 1 since $f(v), B(v) \in [0,1]$ for all $v \in V$. Thus, the expected difference $\mathbb{E}_v[|f(Y_t) - B(Y_t)|]$ is at most the probability that Y_t has not been absorbed.

Let us now show this probability is bounded by ϵ, namely $\mathbb{P}r_v[Y_t \notin \{s,r\}$ for $t \ge 4h^2 \log(1/\varepsilon)] \le \varepsilon$ for all $v \in V$. The main convergence is shown in Lemma 4 which says that the probability that $Y_t \notin \{s, r\}$ for $t = 4h^2$ is at most $\frac{1}{4}$. Since the statement holds for all $v \in V$, if we are not at s or r after $4h^2$ time steps we can simply apply the lemma again. Thus, after $4h^2 \log(1/\varepsilon)$ time steps, the probability that we are not at s or r is $(\frac{1}{4})^{\log(1/\varepsilon)} = 4^{\log \varepsilon} \le \varepsilon$. □

Lemma 4. *$\mathbb{P}r_v[Y_t \notin \{s,r\}$ for $t \geq 4h^2] \leq \frac{1}{4}$ for all $v \in V$ where h is the height of D[5] and $\{Y_t\}$ is the half-optimal chain defined above.*

Proof. Let the absorption time be $\tau = \min\{t : Y_t \in \{s,r\}\}$. Note that $\mathbb{P}r[Y_t \notin \{s,r\}$ for some $t \geq 4h^2] = \mathbb{P}r[\tau \geq 4h^2]$. We show that $\mathbb{E}[\tau] \leq h^2$. Then by Markov's inequality, $\mathbb{P}r[\tau \geq 4h^2] \leq \frac{1}{4}$ as desired.

Consider the sequence $\{\Psi_t\} = \{B(Y_t)\}$. In the half-optimal chain $\{Y_t\}$, Max plays suboptimally and $Mini$ plays optimally according to payoff function B (see the proof of Theorem 2). Hence $B(Y_t)$ is an upper bound on the expected payoff for Max at time t, and therefore $\{\Psi_t\}$ is a supermartingale.[6]

Now consider the quadratic chain $\Phi_t = 2\Psi_t - \Psi_t^2 + t\sigma^2$ where σ^2 is a lower bound on the conditional variance of Ψ_t. We show that Φ_t is also a supermartingale (Lemma 5). Therefore, since $\Phi_t \geq 0$, the optional stopping theorem[7] gives $\mathbb{E}[\Phi_\tau] \leq \Phi_0 \leq 1$. The bounds on Ψ_t also imply that $2\Psi_t - \Psi_t^2 \geq 0$, and hence we get $\mathbb{E}[\Phi_\tau] \geq \mathbb{E}[\tau]\sigma^2$,and $\mathbb{E}[\tau] \leq \frac{1}{\sigma^2}$. By Lemma 6, we know that we can take $\sigma^2 = \frac{1}{h^2}$, so $\mathbb{E}[\tau] \leq h^2$ as required. □

Lemma 5. *Given a supermartingale $0 \leq \Psi_t \leq 1$ with conditional variance at least σ^2, the quadratic chain $\Phi_t = 2\Psi_t - \Psi_t^2 + t\sigma^2$ is a supermartingale.*

Lemma 6. *The variance of a step in $\{\Psi_t\}$ is at least $\sigma^2 = 1/h^2$.*

The proofs of these lemmas can be found in the full version of this paper.

When M is not a perfect matching, D is a weakly acyclic digraph with a cycles through s and/or r. Any vertex in such a cycle must take value exactly 0 or 1 in the balanced outcome, thus these cycles can be treated as absorbing states. Hence, we can first analyze the mixing time of the cycle using spectral techniques[8], and then apply the theorems above to get the same time bound.

4 Conclusion and Future Work

We reduced the problem of analyzing the convergence of local dynamics for a network bargaining game to that of a random-turn game. With this reduction we bring all the machinery from the analysis of random processes, especially the theory of Markov chains and martingales, to the analysis of local dynamics. We used these techniques to give the optimal bound on unweighted graphs with a unique balanced outcome. Prior to this work, there was no effective technique known to analyze such dynamics, and the best bound known on any non-trivial class of graphs was exponential.

Our work opens up a promising line of approach to analyze many variants of local dynamics on general graphs. The most immediate is perhaps to bound the convergence rate for weighted graphs. The difficulty with our current analysis is

[5] The *height* is the length of the longest path from s to r.

[6] Recall that a supermartingale is a sequence $\{a_t\}$ in which $a_t \geq \mathbb{E}[a_{t+1}|a_t]$.

[7] See Theorem 10.10 (d) in [36].

[8] See Chapter 12 in Peres, et al. [26] for an exposition on spectral techniques.

that the supermartingale Ψ_t we used in the unweighted case is unbounded when there are weights. We believe a different supermartingale that does not suffer from this drawback could give the appropriate bound.

The most significant technical hurdle arises when D is cyclic. In this case, the game may never end, since it might get stuck in a *stalemate*, where the players travel in a cycle indefinitely. Thus, a bound on the absorption time of the game does not suffice – we must analyze the behavior on the cycle separately by internally considering its *mixing time*, and externally treating it as an absorbing state.[9] However, the details of such an analysis remain unclear.

A final important direction is to obtain tight polynomial bounds for dynamics which find both the matching and the balanced outcome simultaneously. One approach would be to combine the dynamics by Kanoria et. al. [22] with our techniques to attain a tight polynomial rate of convergence.

References

1. Workshop on new connections between differential and random turn games, pde's and image processing (2009)
2. Ackermann, H., Goldberg, P.W., Mirrokni, V.S., Röglin, H., Vöcking, B.: Uncoordinated two-sided matching markets. In: EC, pp. 256–263 (2008)
3. Azar, Y., Birnbaum, B., Celis, L.E., Devanur, N.R., Peres, Y.: Convergence of local dynamics to balanced outcomes in exchange networks. In: FOCS (2009)
4. Azar, Y., Devanur, N., Jain, K., Rabani, Y.: Monotonicity in bargaining networks. In: SODA (2010)
5. Bateni, M.H., Hajiaghayi, M.T., Immorlica, N., Mahini, H.: Network bargaining with capacity constraints. In: ICALP (2010)
6. Bayati, M., Borgs, C., Chayes, J.T., Zecchina, R.: Belief-propagation for weighted b-matchings on arbitrary graphs and its relation to linear programs with integer solutions. SIAM J. in Discrete Math. (to appear, 2010)
7. Berlekamp, E.R., Conway, J.H., Guy, R.K.: Winning ways for your mathematical Plays. Academic Press, London (1982)
8. Chakraborty, T., Judd, S., Kearns, M., Tan, J.: A behavioral study of bargaining in social networks. In: EC (2010)
9. Chakraborty, T., Kearns, M.: Bargaining solutions in a social network. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 548–555. Springer, Heidelberg (2008)
10. Chakraborty, T., Kearns, M., Khanna, S.: Nash bargaining: Algorithms and structural results. In: EC (2009)
11. Charro, F., García Azorero, J., Rossi, J.D.: A mixed problem for the infinity Laplacian via tug-of-war games. Var. & Par. Differential Eq. 34(3), 307–320 (2009)
12. Chen, X., Deng, X.: Settling the complexity of two-player Nash equilibrium. In: FOCS, pp. 261–272 (2006)
13. Cook, K.S., Emerson, R.M., Gillmore, M.R., Yamagishi, T.: The distribution of power in exchange networks: Theory and experimental results. American Journal of Sociology 89, 275–305 (1983)

[9] This proposed approach is a generalization of our analysis of non-perfect matchings in unweighted graphs.

14. Cook, K.S., Yamagishi, T.: Power in exchange networks: a power-dependence formulation. Social Networks 14(3-4), 245–265 (1992)
15. Corominas-Bosch, M.: Bargaining in a network of buyers and sellers. Journal of Economic Theory 115(1), 35–77 (2004)
16. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. In: STOC, pp. 71–78 (2006)
17. Draief, M., Vojnovic, M.: Bargaining dynamics in exchange networks. In: The 48th Annual Allerton Conference on Communication, Control, and Computation, Monticello, IL (September 2010)
18. Faigle, U., Kern, W., Kuipers, J.: An efficient algorithm for nucleolus and prekernel computation in some classes of TU-games. Ph.D. thesis, Memorandum No. 1464, University of Twente (1998)
19. Gale, D., Shapley, L.S.: College admissions and the stability of marriage. The American Mathematical Monthly 69(1), 9–15 (1962)
20. Jain, K.: A polynomial time algorithm for computing an Arrow-Debreu market equilibrium for linear utilities. SIAM J. Comput. 37(1), 303–318 (2007)
21. Kanoria, Y.: An FPTAS for bargaining networks with unequal bargaining powers. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 282–293. Springer, Heidelberg (2010)
22. Kanoria, Y., Bayati, M., Borgs, C., Chayes, J.T., Montanari, A.: Fast convergence of natural bargaining dynamics on exchange networks. In: SODA (to appear, 2011)
23. Kleinberg, J., Tardos, E.: Balanced outcomes in social exhchange networks. In: STOC (2008)
24. Lazarus, A., Loeb, D.E., Propp, J.G., Stromquist, W.R., Ullman, D.H.: Combinatorial games under auction play. Games and Economic Behavior 27 (1999)
25. Lazarus, A., Loeb, D.E., Propp, J.G., Ullman, D.H.: Richman games. Games of No Chance 29, 439–449 (1996)
26. Levin, D.A., Peres, Y., Wilmer, E.L.: Markov Chain Mixing Times. Academic Mathematical Society, London (2009)
27. Maschler, M.: The bargaining set, kernel, and nucleolus. In: Aumann, R.J., Hart, S. (eds.) Handbook of Game Theory with Economic Applications, vol. 1, ch. 18, pp. 591–667. Elsevier, Amsterdam (1992)
28. Nash, J.: The bargaining problem. Econometrica 18, 155–162 (1950)
29. Paulusma, D.: Complexity Aspects of Cooperative Games. Ph.D. thesis, University of Twente, Enschede, the Netherlands (2001)
30. Peres, Y., Schramm, O., Sheffield, S., Wilson, D.B.: Tug-of-war and the infinity laplacian. Journal of the AMS 22, 167–210 (2009)
31. Rochford, S.C.: Symmetrically pairwise-bargained allocations in an assignment market. Journal of Economic Theory 34(2), 262–281 (1984)
32. Sanghavi, S., Malioutov, D.M., Willsky, A.S.: Linear programming analysis of loopy belief propagation for weighted matching. In: NIPS (2007)
33. Shapley, L.S., Shubik, M.: The assignment game I: The core. Int. Journal of Game Theory 1(2), 111–130 (1972)
34. Solymosi, T., Raghavan, T.E.S.: An algorithm for finding the nucleolus of assignment games. Int. Journal of Game Theory 23(2), 119–143 (1994)
35. Willer, D. (ed.): Network Exchange Theory. Praeger (1999)
36. Williams, D.: Probability with Martingales. Cambridge University Press, Cambridge (1991)
37. Yu, Y.: Maximal and minimal solutions of an Aronsson equation: L^∞ variational problems versus the game theory. Var. & Par. Differential Eq. 37, 63–74 (2010)

Selective Call Out and Real Time Bidding

Tanmoy Chakraborty[1,⋆], Eyal Even-Dar[4,⋆⋆], Sudipto Guha[1],
Yishay Mansour[2], and S. Muthukrishnan[3]

[1] University of Pennsylvania, Philadelphia, PA
{tanmoy,sudipto}@cis.upenn.edu
[2] Google Israel and The Blavatnik School of Computer Science,
Tel-Aviv University, Tel-Aviv, Israel
mansour.yishay@gmail.com
[3] Google Research, 76 Ninth Ave, New York, NY
muthu@google.com
[4] evendar.eyal@gmail.com

Abstract. Ads on the Internet are increasingly sold via ad exchanges such as RightMedia, AdECN and Doubleclick Ad Exchange. These exchanges allow real-time bidding, that is, each time the publisher contacts the exchange, the exchange "calls out" to solicit bids from ad networks. This solicitation introduces a novel aspect, in contrast to existing literature. This suggests developing a joint optimization framework which optimizes over the allocation and well as solicitation.

We model this selective call out as an online recurrent Bayesian decision framework with bandwidth type constraints. We obtain natural algorithms with bounded performance guarantees for several natural optimization criteria. We show that these results hold under different call out constraint models, and different arrival processes. Interestingly, the paper shows that under MHR assumptions, the expected revenue of generalized second price auction with reserve is constant factor of the expected welfare. Also the analysis herein allow us prove adaptivity gap type results for the adwords problem.

1 Introduction

A dominant form of advertising on the Internet involves *display* ads; these are images, videos and other ad forms that are shown on a web page when viewers navigate to it. Each such showing is called an *impression*. Increasingly, display ads are being sold through exchanges such as RightMedia, AdECN and DoubleClick Ad Exchange. On the arrival of an impression, the exchange solicits bids and runs an auction on that particular impression. This allows *real time bidding* where ad networks can determine their bids for each impression individually in real time (for an example, see [24]), and more importantly where the creative (advertisement) can be potentially produced on-the-fly to achieve

⋆ Part of this work was done while the authors were visiting Google Research.
⋆⋆ Part of this work was done while the author was at Google Research.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 145–157, 2010.

better targeting [22]. This potential targeting comes hand in hand with several challenges. The Exchange and the networks face a mismatch in infrastructure and capacities and objectives. From an infrastructure standpoint, the volume of impressions that come to the exchange is very large comparison to a smaller ad network limited in servers, bandwidths, geographic location preferences. This implies a bound on the number of auctions the network can participate in effectively. A network would prefer to be solicited only on impressions which are of interest to it, and in practice use a descriptive languages to specify features of impressions (say, only impressions from NY). However this is an offline feature and runs counter to the attractiveness of real time bidding. Therefore the exchange has to "call out" to the networks selectively, simultaneously trying to balance the objective of soliciting as many networks as possible and increasing total value, as well as not creating congestion. This leads to a host of interesting questions in *developing a joint optimization framework that optimizes over the allocation objective as well as the decisions to solicit the bids.*

1.1 Selective Call Out: The Model

Let n be the number of ad networks $1, 2 \ldots n$. We assume that impressions arrive from a fixed (unknown) distribution over a finite set U_I, and that there exists a finite set of bid values U_B, where $L = \max\{u|\ u \in U_B\}$. In the following, ad networks will be indexed by $i \in \{1, 2 \ldots n\}$, impressions by $j \in U_I$ and bid values by $k \in U_B$. The problem setting involves several steps:

1. An impression (or keyword) j, assumed drawn from a distribution $\mathcal{D}$, comes to the exchange. There may be *multiple slots* associated with a single impression, corresponds to text ads being blocked together, different locations in the page, which are often characterized by different *discount rate*. Let there be M slots, with discount rates $1 \geq \varrho_1 \geq \varrho_2 \ldots \geq \varrho_M \geq 0$. If a bidder bids v, then it is assumed that the bid for the ℓ^{th} slot is $v\varrho_\ell$. The case of $M = 1$ is common and correspond to a basic pay-per-impression mechanism with discount rate 1.
2. Given an impression j, the bid of ad network i for impression j is drawn from a fixed distribution $\mathbf{V}_{ij}$ such that the bid is v with probability p_{ijv}. Note that the bids of different networks are likely to be correlated based on the perceived value of the impression, however, conditioned on j, the specific dynamics of different bidders can be construed to be independent. We assume that the exchange has learned or can predict these p_{ijv} given the impression j. This is an assumption similar to estimation processes used by search engines to predict click through rates.
3. The exchange decides on the subset S_j of networks to call out, subject to the *Call-Out constraints*, which roughly bounds the rate at which the exchange can send impressions to solicit bid from an ad network. This decision is executed before seeing the next impression.

To define a specific problem in the above framework, we need to specify (i) an objective function (ii) a model for call out constraints, and (iii) the comparison class. The **goal** is to design a call-out policy that satisfies (ii) and is near optimal in the objective function (i), when compared to other algorithms in class (iii). We discuss the instantiations of (i),(ii), and (iii) in the following.

(i) Objective Functions. We consider three different objective functions. (a) *Total Value*: The sum of the maximum bids in S_j over the arriving impressions j. (b) *GSP-Reserve* (defined above) and (c) *Revenue under posted price mechanism (take-it-or-leave-it prices).* All the quantities are in expectation. Total value corresponds to the welfare. The GSP with an uniform reserve price is a common mechanism used in these settings. Posted Prices (different networks may get different prices) are also used in this context. Note, unlike [8,4], the mechanism is *parallel* posted price because all the prices are posted first.

(ii) Call Out Constraint Models. The simplest model for the call out constraints is a model where one impression arrives at the exchange at each time step and if the total number of arrivals is m, ad network i can be solicited at most times for some known $\rho_i \leq 1$. We will refer to this as *time average model under uniform arrival* – which describes the constraints at the outgoing and incoming sides of the exchange respectively. Most of the paper will focus on this model – *primarily because we can show that other common models reduce to this variant.* The non-initiated reader can skip the description of these models and proceed to (iii). On the incoming side of the exchange, standard practice is to assume bursty (Poisson) arrivals. We consider this generalization. On the outgoing side, the simple model allow the possibility that the call-outs to a network are made on contiguous subset of impressions. This misses the original goal that the ad network would receive the impressions at a "smooth" rate. A common model used for behavior is the *token bucket model* [26]. A token bucket has two parameters, bucket size σ and token generation rate ρ. The tokens represent sending rights, and the bucket size is the maximum number of tokes we can store. The tokens are generated at a rate of ρ per unit time, but the number of tokens never exceeds σ. In order to send, one needs to use a token, and if there are no tokens, one can not send. The output stream of a (σ, ρ) token bucket can be handled by a buffer of size σ and a time average rate of ρ – the buffer is initially full. Unlimited buffer size corresponds to the time average model.

(iii) Comparison Class. Given the call out constraints, we define the class of admissible policies. An *admissible call out policy* specifies (possibly with randomization), for each arriving impression j, the subset S_j of ad networks to call out, while satisfying all call out constraints over the entire sequence of impressions. The policy bases its decision on the prior information about the bid distributions, and has no knowledge of the actual bid values. In the case of GSP-Reserve mechanism, the call out policy also decides the reserve price for each impression (and likewise for the case of a posted price mechanism). Our comparison class is the set of all admissible call out policies which know the bid distributions for every impression, but do not know the actual realization of the bids. The performance of a policy is measured as the expected (over the bid distributions and the impression arrivals) objective value obtained per arriving impression, when impressions are drawn from $\mathcal{D}$. *A policy is α-approximate if it achieves at least α times the performance of the optimal policy for the corresponding objective.*

1.2 Our Results, Roadmap and Related Work

We provide three algorithms LP-VAL, LP-GSP, and LP-POST for the objectives discussed for the time average uniform arrival model. We then prove that the results translate naturally to other constraint models. Recall, L is the largest possible bid. The algorithms will have a natural two-phase approach where we use the t initial impressions as a sample as exploration and subsequently use/exploit this algorithm (see Section 2 for more discussion). We show that:

Theorem 1. *Suppose the optimal policy has expected total value at least $\delta > 0$. For any $\epsilon > 0$,* LP-VAL *with a sample of $t = \tilde{O}(\frac{n^2 L}{\delta\epsilon})$ impressions gives a $(1 - \frac{1}{e} - \epsilon)$-approximate policy.*

Theorem 2. *Suppose the optimal GSP-Reserve policy has expected revenue at least $\delta > 0$. For any $\epsilon > 0$,* LP-GSP *with a sample of $t = \tilde{O}(\frac{n^2 L}{\delta\epsilon})$ impressions gives a $O(1)$-approximate policy, if all bid distributions satisfy the* monotone hazard rate (MHR) *property. Moreover, the call outs of the policy derived from* LP-GSP *are identical to those of the policy derived from* LP-VAL.

Theorem 3. *Suppose the optimal posted price policy has expected revenue at least $\delta > 0$. For any $\epsilon > 0$,* LP-POST *with a sample of $t = \tilde{O}(\frac{n^2 L}{\delta\epsilon})$ impressions gives a $(1 - \frac{1}{e} - \epsilon)$-approximate policy.*

In particular, we show that when every bidder is solicited – GSP-Reserve achieves a revenue that is $O(1)$ factor of optimal welfare, when all bid distributions satisfy the MHR property. This is a common distributional assumption in economic theory, and is satisfied by many distributions [3]. This result is in the same spirit as (but immediately incomparable to) the result in [4], which relates the optimum sequential posted price revenue to the optimal welfare under the same assumptions. We are unaware of such results about GSP-Reserve. We do **not** need the MHR assumption unlike the result in [4] for sequential posted prices, since the comparison classes are different (the prices are posted in parallel). We discuss the realization of the above algorithms for different network models next.

Theorem 4. *For a given distribution on impressions $\mathcal{D}$, suppose we have an α-approximate policy for an objective which is additive given the allocations and the realizations (and is at least $\delta > 0$) in the time average uniform arrival call out model. Let $\sigma_i > \sigma$ $\forall i$. Then we can convert the policy to a $(\alpha - \frac{1}{\sigma - 1})$-approximate policy in the token bucket model. The result extends to Poisson arrivals.*

Comparison with the Adwords Model: The call out optimization framework is similar to the online ad allocation framework for search ads, or the Adwords problem [19,5,7], and its stochastic variants [9,27]. However there are significant differences which we discuss below. The Adwords problem is posed in the deterministic setting where the expected revenue is treated as a known deterministic reward of allocating an impression j to an advertiser i. The call out framework has no deterministic analogue; the rationale of the exchange is that the bids and the participation are not known. In this regard, the call-out framework is similar to the Bayesian mechanism design [21].

This unknown participation has broad conceptual implications, first of which is the notion of "adaptivity gap". An ad-allocation policy may choose to react to realization of the random variables. This aspect is central in the exchange setting. Interestingly, the analysis in this paper also allows us to consider adaptivity gap for in the Adwords setting, if the allocation policy is allowed to adapt to the occurence or absence of a click (instead of using a deterministic quantity which is the product of the click-through-rate and the cost of a click). This is a significant issue for low click through rates and large bids, such that a payout affects the budget substantially. To the best of our knowledge, no analysis of adaptivity gap existed for the adwords problem previously,

Second, many objective functions such as generalized second price with reserve (henceforth GSP-Reserve), for one or multiple slots, have very different behaviors in the Bayesian and deterministic settings. Consider running Myerson's (or similar) mechanism on the expected bids instead of the distributions. For *known* deterministic bids, reserve prices can be made equal to the bid, and are not useful. Strong lower bounds hold for GSP without reserves [1]. Note that in GSP-Reserve we announce an uniform reserve price, *before* the bids are solicited.

Third, the notion of a comparison class in case of call out optimization framework requires more care than the Adwords framework. In the setting of these large exchanges, a comparison class with full foreknowledge of all information (in particular, the realization of the bids) is unrealistic. Moreover, the realizations of the bids depends on the networks which are called out, and two different strategies that call out to two different subsets will have completely different information. Thus to compare two algorithms, we should compare their expected outcome – *but each algorithm is allowed to be adaptive.*

We use Lagrangian decoupling techniques for separable convex optimization pioneered by Rockafellar [25]. This has been used in the stochastic variants of the Adwords problem in [9,27]. But the similarity ends there. *The different possible objectives of the call-out framework are not convex.* In fact, in the case of optimizing revenue in GSP (with reserve) or in posted price mechanisms, the objective is not submodular for all prices (as in welfare maximization). Submodular maximization with linear constraints has been studied, and while good approximation algorithms exist [6,17], they are inherently offline – the key aspect of call out optimization is that the decision has to be made in an online fashion. The same is true for sequential posted price mechanisms analyzed in [8,4] (albeit with more general matroid setting), the posted prices in the call out setting need to be announced in parallel (and the eventual allocation is sequential).

Other Related Work: A combination of stochastic and online components appear in many different settings [14,15,2,13] which are not immediately relevant to the call-out problem. We note that the bandwidth-like constraints (where the constraint is on a parameter different than the obtained value, as is the case for call-outs) has not been studied in the bandit setting (see [16,23]). Finally, bidding and inventory optimization problems [11,10,20], are not immediately relevant.

Roadmap: We summarize the results on online stochastic convex optimization in Section 2. We subsequently discuss the the total value problem in Section 3. We discuss the GSP-Reserve problem in Section 4. The posted price problem is discussed in the full version. The token bucket model and other arrival assumptions are also discussed in the full version.

2 Preliminaries

Consider a maximizing a "separable" linear program (LP) $\mathcal{L}$ defined on Q *global* constraints with right hand side b_i, such that the Lagrangian relaxation produced by the transferring these constraints to the objective function decouples into a collection of independent non-negative smaller LPs $\mathcal{L}_j$ over n' variables and *local* constraints. This implies that the objective function of $\mathcal{L}$ is a weighted linear combination of $\mathcal{L}_j$. The uniqueness of the optimum solutions for $\mathcal{L}_j$ implies that $\mathcal{L}$ reduces to finding the Lagrangian multipliers. The unique solution is achieved by adding "small perturbations", see Rockafellar [25]. However, this approach only provides a certificate of optimality and a solution, once we are given the Lagrangians. The approach does not give us an algorithm to find the Lagrangian multipliers themselves. Devanur and Hayes [9] showed that if the smaller LPs could be sampled with the same probability as their contribution to the objective of $\mathcal{L}$, and the derivatives of the Lagrangian can be bounded, then the Lagrangians derived from a small number of samples (suitably scaled) can be used to solve the overall LP. The weighted sampling reduces to the prefix of the input if the $\mathcal{L}_j$s arrive in random order (see [12]). This was extended to convex programs in [27]. The number of sample bound requires several (easy) Lipschitz type properties:

1. The optimum value of $\mathcal{L}$ is at least $\delta > 0$ and the optimum solution of $\mathcal{L}_j$ is at most R.
2. For each setting of the Lagrangians, every $\mathcal{L}_j$ has a unique optimum solution.
3. Reducing b_i by a factor of $1 - \epsilon$ reduces $\mathcal{L}$ by at most $(1 - \epsilon)$.
4. $\mathcal{L}$ does not change by more than a constant times $1 + \epsilon$ if we alter the value of the optimum Lagrangian multipliers by a factor of $1 + \epsilon$.

Theorem 5. *Sample $t = \tilde{O}(\frac{n'QR}{\delta\epsilon})$ of the smaller linear programs and consider the linear program $\mathcal{L}'$ which corresponds to the union of these smaller linear programs and suitably scaled global constraints. If we use the optimum Lagrangian multipliers corresponding to the global constraints of $\mathcal{L}'$ to solve the decoupled instances of $\mathcal{L}_j$ as they are available (in an online fashion) then we produce a $1 + \epsilon$ approximation to the optimum solution of $\mathcal{L}$.*

In the setting of Adwords, the smaller LPs correspond to the arrival of an impression, and the associated assignment. Thus the stochastic framework of the Adwords problem is obviously of relevance to the call out optimization framework. However the focus shifts on solving the smaller LPs which encode the call out decision. A nice outcome of the approach is a simple two phase algorithm;

an *Exploration phase* where the samples are drawn, and an *Exploitation phase* where the Lagrangian multipliers are used. If Q, n', R are small, then the exploration phase can be (relatively) short and this yields a natural algorithm. Thus the goal of the rest of the paper would be to formulate separable convex relaxations and achieve the mentioned properties.

3 The Total Value Problem

In this section, we prove Theorem 1, and describe LP-VAL. Let q_j denote the probability that impression j arrives. We shall add infinitesimal random perturbations to p_{ijv} which shall not affect the performance of any policy but ensure p_{ijv} are in general positions, that is, any combination of them will almost surely create a non-singular matrix.

The LP Relaxation: Let x_{ij} be the (conditional) probability that advertiser i was called out on impression j. Let $y_{ijv\ell}$ be the probability that advertiser i bid the value t and was assigned the slot ℓ (also conditioned on j). The constraints are named as $A(x, y)$ and $B(x, y)$ as shown.

$$LP1 = \max \sum_j q_j \sum_v \sum_i \sum_\ell v\varrho_\ell y_{ijv\ell} \quad \text{s.t.} \quad \left.\begin{array}{ll} \sum_j q_j x_{ij} & \leq \rho_i \\ \sum_i \sum_v y_{ijv\ell} \leq 1 & \\ \left.\begin{array}{ll} x_{ij} & \leq 1 \\ \sum_\ell y_{ijv\ell} & \leq p_{ijv} x_{ij} \\ x_{ij}, y_{ijv\ell} & \geq 0 \end{array}\right\} B(x,y) & \end{array}\right\} A(x,y)$$

Decoupling: Let λ_i^* be the optimum Lagrangian variable for the constraint $\sum_j q_j x_{ij} \leq \rho_i$. $LP1$ then decouples to smaller LPs, $LP2(j, \boldsymbol{\lambda_i^*})$ subject to the constraints $A(x, y)$, that is, $LP1 = LP1(\boldsymbol{\lambda_i^*}) = \sum_i \lambda_i^* \rho_i + \sum_j q_j LP2(j, \boldsymbol{\lambda_i^*})$ where $LP2(j, \boldsymbol{\lambda_i^*}) = \max\left(\sum_v \sum_i \sum_\ell v\varrho_\ell y_{ijv\ell} - \sum_i \lambda_i^* x_{ij}\right)$

Solving $\mathbf{LP2}(\mathbf{j}, \boldsymbol{\lambda_i^*})$. We begin by considering the dual. Let $\tau_{j\ell}$ be the dual of the constraint $\sum_i \sum_v y_{ijv\ell} \leq 1$. Let ξ_{ijv} correspond to the dual of the constraint $\sum_\ell y_{ijv\ell} \leq p_{ijv} x_{ij}$. Let ζ_{ij} correspond to the dual of $x_{ij} \leq 1$.

$$DualLP2(j, \boldsymbol{\lambda_i^*}) = \min \sum_i \zeta_{ij} + \sum_\ell \tau_{j\ell} \quad \text{s.t.} \quad \begin{array}{ll} \tau_{j\ell} + \xi_{ijv} & \geq v\varrho_\ell \\ \zeta_{ij} - \sum_v \xi_{ijv} p_{ijv} & \geq -\lambda_i^* \\ \tau_{j\ell}, \xi_{ijv}, \zeta_{ij} & \geq 0 \end{array}$$

Lemma 1. *Let $\tau_{j\ell}^*$ be the optimum dual variables for LP2. Then (i) For all ℓ there exists i and $v \geq \tau_{j\ell}^*/\varrho_\ell$ s.t. $\xi_{ijv}^* = v\varrho_\ell - \tau_{j\ell}^*$ (ii) $\tau_{j\ell}^*/\varrho_\ell$ is non-increasing.*

Proof. For every ℓ there must be some i, v such that we have $\tau_{j\ell}^* + \xi_{ijv}^* = v\varrho_\ell$. Otherwise we can keep decreasing $\tau_{j\ell}^*$, keeping all other variables the same and contradict the optimality of the dual solution. Now $\xi_{ijv}^* \geq 0$ and the condition on t follows. The condition corresponds to the set of points (v, ξ_{ijv}^*) in the two dimensional (x, y) plane being above the lines $\{y = \varrho_\ell x - \tau_{j\ell}^*\}$. For the second part, consider $\tau_{j\ell}^*$ and the i, v such that we have $\tau_{j\ell}^* + \xi_{ijv}^* = v\varrho_\ell$. Define t to be the support of ℓ. Let $v \geq \tau_{j\ell}^*/\varrho_\ell$ be the largest such support of ℓ. Consider $\tau_{j(\ell-1)}^*$. We have $(\tau_{j(\ell-1)}^* + \xi_{ijv}^*)/\varrho_{\ell-1} \geq v = (\tau_{j\ell}^* + \xi_{ijv}^*)/\varrho_\ell$. But $\varrho_{\ell-1} \geq \varrho_\ell$ and thus $\tau_{j\ell}^*/\varrho_\ell$ are non-increasing in ℓ. Moreover, if $\varrho_\ell = \varrho_{\ell-1}$ then $\tau_{j\ell}^* = \tau_{j(\ell-1)}^*$.

Decoupling LP2$(\mathbf{j}, \boldsymbol{\lambda}_i^*)$ itself. Consider $LP2(j, \boldsymbol{\lambda}_i^*)$ with the Lagrangians $\tau_{j\ell}^*$. The problem decomposes under the constraints $B(x, y)$, to $LP2(j, \boldsymbol{\lambda}_i^*) = \sum_\ell \tau_{j\ell}^* + \sum_i LP3(j, \boldsymbol{\lambda}_i^*, \boldsymbol{\tau}_{j\ell}^*, i)$ where $LP3(j, \boldsymbol{\lambda}_i^*, \boldsymbol{\tau}_{j\ell}^*, i) = \max\left(\sum_\ell \sum_v \left[v\varrho_\ell - \tau_{j\ell}^*\right] y_{ijv\ell} - \lambda_i^* x_{ij}\right)$.

Lemma 2. *Define* $\ell(v) = \arg\max_{\ell'} \left\{\varrho_{\ell'} v - \tau_{j\ell'}^* | \varrho_{\ell'} v > \tau_{j\ell'}^*\right\}$ *and* $\ell(t) = M+1$ *if the set is empty. Set* $y_{ijv\ell}^* = p_{ijv}$ *if* $\ell = \ell(v)$ *and* 0 *otherwise. If* $\sum_v \sum_\ell v\varrho_\ell y_{ijv\ell}^* \geq \lambda_i^*$ *we set* $x_{ij} = 1$ *and* $y_{ijv\ell} = y_{ijv\ell}^*$. *Otherwise we set* $x_{ij} = y_{ijv\ell} = 0$.

Proof. $LP3(j, \boldsymbol{\lambda}_i^*, \boldsymbol{\tau}_{j\ell}^*, i)$ is optimized at $x_{ij} = 1$ or $x_{ij} = 0$. This is because if $0 < x_{ij} < 1$ and $\sum_\ell \sum_v \left[v\varrho_\ell - \tau_{j\ell}^*\right] y_{ijv\ell} - \lambda_i^* x_{ij} > 0$ then we can multiply all the variables by $1/x_{ij}$ and have a better solution. If the latter condition is not true then $x_{ij} = 0$ is an equivalent solution. If $x_{ij} = 1$ the optimal setting for $y_{ijv\ell}$ is $y_{ijv\ell}^*$. (Note that $y_{ijv\ell}^*$ is uniquely determined for a fixed t.) Thus the overall optimization follows from comparing the $x_{ij} = 1$ and $x_{ij} = 0$ case.

Interpretation and the Call Out Algorithm: Given $\{\tau_{j\ell}^*\}_{\ell=1}^M$, the distribution $\{p_{ijv}\}$ for i, is divided into at most $M + 1$ pieces (some of the pieces can be a single point) given by the upper envelope (the constraint max) of the lines $\{\varrho_\ell x - \tau_{j\ell}^*\}_{\ell=1}^M$ and the line $y = 0$, in the x–y coordinate plane. Intuitively, seeing the value $x = t$, if the upper envelope corresponds to the equation $\varrho_\ell x - \tau_{j\ell}^*$ then we are "interested" in the slot ℓ. If the weighted (by ϱ_ℓ) sum of interests, given by $\sum_v v\varrho_\ell y_{ijv\ell}^*$ exceeds λ_i^*, then it is beneficial to call out i. We call out based on this condition and allocate the slots in decreasing order of bids.

Analysis: The LP2 solution satisfies: $LP1 = \sum_{i,v} vp_{ijv}\varrho_{\ell(v)}$ and $\sum_{i,v:\ell(v)=\ell} p_{ijv} = 1$. For each slot ℓ, let $w_i(\ell) = \sum_{v:\ell(v)=\ell} vp_{ijv}$ and $u_i(\ell) = \sum_{v:\ell(v)=\ell} p_{ijv}$. Order the i in non-increasing order of $w_i(\ell)/u_i(\ell)$ inside the slot. If we call out to i and get t, then for the sake of analysis we will consider its contribution to slot $\ell(t)$ only. Moreover, we stop the contribution to a slot ℓ if any any of the i return a value t with $\ell(t) = \ell$. The best M ordered bids outperform the analyzed contribution in every scenario. Therefore it suffices to bound the contribution.

Lemma 3. *Suppose we are given a set of independent variables* Y_i *such that* $Pr[Y_i \neq 0] = u_i$ *and* $E[Y_i] = w_i$. *Consider the random variable* Y *corresponding to the process which orders the variables* $\{Y_i\}$ *in non-increasing order of* w_i/u_i, *and stops as soon as the first non-zero value is seen. Then* $E[Y] = \sum_i \prod_{i'<i}(1 - u_{i'})w_i \geq \sum_i w_i \left(1 - e^{-1}\right)$.

Proof. Let $F(\{(w_i, u_i)\}) = \sum_i \prod_{i'<i}(1 - u_{i'})w_i$. Let $\lambda = \sum_i w_i / \sum_i u_i$. Given the sequence $\{(w_i, u_i)\}$ where w_i/u_i are non-increasing, if there exists an i such that $w_i/u_i \neq w_{i+1}/u_{i+1}$, then define a new sequence $\{(w_i', u_i)\}$ as follows:

$$w_{i'}' = \begin{cases} w_{i'} & \text{if } i' \neq i, i+1 \\ w_i - \Delta & \text{if } i' = i \\ w_{i+1} + \Delta & \text{if } i' = i+1 \end{cases} \qquad \text{where} \qquad \Delta = \frac{\frac{w_i}{u_i} - \frac{w_{i+1}}{u_{i+1}}}{\frac{1}{u_i} + \frac{1}{u_{i+1}}}$$

Note that $\sum_i w_i = \sum_i w'_i$ and w'_i/u_i remains non-increasing. Now $F(\{(w_i, u_i)\}) - F(\{(w'_i, u_i)\}) = \prod_{i'<i}(1-u_{i'})\Delta - \prod_{i'<i+1}(1-u_{i'})\Delta = \prod_{i'<i}(1-u_{i'})u_i\Delta > 0$. Thus, we can repeatedly perform the above steps till we get a sequence such that w'_i/u_i remains the same for all i and $\sum_i w_i = \sum_i w'_i$. Clearly $w'_i = \lambda u_i$ in this case. The function F continues to decrease, $F(\{(w_i, u_i)\}) \geq F(\{(w'_i, u_i)\})$ and

$$F(\{(w'_i, u_i)\}) = \sum_i \prod_{i'<i}(1-u_{i'})\lambda u_i = \lambda\left(1 - \prod_i (1-u_i)\right) \geq \lambda\left(1 - e^{-\sum_i u_i}\right)$$

But $\frac{1}{x}(1-e^{-x})$ is decreasing over $[0, 1]$ and the worst case is $x = 1$.

In slot ℓ (renumbering the advertisers in the order of $w_i(\ell)/u_i(\ell)$) we get an expected reward of $w_i(\ell)$ if we reach i. But the events are independent in a particular slot. Thus the expected reward in a slot is bounded by (using independence and Claim 3) to be $(1-e^{-1})$ times $\sum_i w_i(\ell)$. We now apply linearity of expectation across the slots – *observe that the events across the slots are quite correlated*. The expected reward is at least $(1-e^{-1})$ times $\sum_\ell \sum_i w_i(\ell) = LP1$. Theorem 1 follows from Lemmas 2 and 3 and the application of Theorem 5.

4 Generalized Second Price with Reserve (GSP-Reserve)

The call outs for this problem would be exactly the same as the algorithm in Section 3. *We will however adjust the reserve prices.* The reserve price will be the same for all the advertisers being called out on that impression. In fact either we will run a single slot auction with a reserve price, or simply GSP for the M slots. The decision will depend on the LP solution found for this specific impression (and the contributions of different parts of the LP). Recall that the bid distribution $\mathbf{V}_{ij}$ of advertiser i on impression j is assumed to satisfy the MHR property. We use the following:

Lemma 4. *(Lemma 3.3 in [4]) For any random variable V following an MHR distribution, let $v^* = \arg\min_v\{v | v\Pr[V \geq v] \geq \frac{1}{2}\sum_{v' \geq v} v' \Pr[V = v']\}$. Then $\Pr[V \geq v^*] \geq e^{-2}$.*

The next lemma is a restatement of Lemma 2 and the subsequent analysis.

Lemma 5. *Given an impression $j(t)$, and define $v_1(t) = \max_{\ell:\varrho_1 \neq \varrho_\ell}(\tau^*_{j1} - \tau^*_{j\ell})/(\varrho_1 - \varrho_\ell)$. The call out to a set $S(t) \neq \emptyset$, ensures that $\sum_i \sum_{v \geq v_1(t)} p_{ijv} \geq 1$.*

Definition 1. *Given an impression $j(t)$, and the call out decision to a set $S(t) \neq \emptyset$ at time t, let $v^*_m(i,t) = \min\{v | 2v\Pr[V_{ij} \geq v] \geq \sum_{v' \geq v} v' p_{ijv'}\}$ and $\Psi(t) = \{i | i \in S(t)$ and $v_1(t) \leq v^*_m(i,t)\}$. Note that using Lemmas 4, and 5, we have $|\Psi(t)| \leq \lfloor e^2 \rfloor = 7$ since $i \in \Psi(t)$ contributes a probability mass of at least e^{-2}.*

Lemma 6. *Given an impression $j(t)$, and the call out decision to a set $S(t) \neq \emptyset$ at time t, we can set a single threshold $v^*(t) \geq v_1(t)$ such that if we set a reserve price $v^*(t)$ for a single slot then the revenue (ignoring the multiplicative discount factor ϱ_1) is at least $\frac{1}{4(7e^2+1)} \sum_{i \in S(t)} \sum_{v \geq v_1(t)} v p_{ijv}$.*

Proof. Let $\sum_{i\in S(t)}\sum_{v\geq v_1(t)} vp_{ijv} = Z$. We have two cases, (i) $\sum_{i\in\Psi(t)}\sum_{v\geq v_1(t)} vp_{ijv} \geq 7e^2Z/(7e^2+1)$ or (ii) otherwise. In case (i), pick the $i\in\Psi(t)$ such that $\sum_{v\geq v_1(t)} vp_{ijv}$ is maximized, which is at least $e^2Z/(7e^2+1)$ since $|\Psi(t)|\leq 7$. Let V_{ij} be the random variable that corresponds to the bid of advertiser i on impression j. Now since $v^*_m(i,t)\geq v_1(t)$ we have that

$$\sum_{v\geq v^*_m(i,t)} vp_{ijv} \geq \Pr\left[V_{ij}\geq v^*_m(i,t)|V_{ij}\geq v_1(t)\right]\sum_{v\geq v_1(t)} vp_{ijv}$$
$$\geq \Pr\left[V_{ij}\geq v^*_m(i,t)\right]\sum_{v\geq v_1(t)} vp_{ijv} \leq \frac{1}{e^2}\sum_{v\geq v_1(t)} vp_{ijv}$$

which is at least $Z/(7e^2+1)$. Now, if we set $v^*(t)=v^*_m(i,t)$ then just from i we have $\sum_{i\in S(t)}\sum_{v\geq v^*(t)} p_{ijv} \geq \frac{1}{2}\sum_{v\geq v^*_m(i,t)} vp_{ijv}$ and therefore in this case the lemma is true.

In case (ii), we have $\sum_{i\in S(t)\setminus\Psi(t)}\sum_{v\geq v_1(t)} vp_{ijv}\geq Z/(7e^2+1)$. But for each $i\in S(t)\setminus\Psi(t)$ we have $v_1(t)\sum_{v\geq v_1(t)} p_{ijv}\geq\frac{1}{2}\sum_{v\geq v_1(t)} vp_{ijv}$ and as a consequence, $v_1(t)\sum_{i\in S(t)\setminus\Psi(t)}\sum_{v\geq v_1(t)} p_{ijv}$ is at least $Z/(2(7e^2+1))$. Consider setting $v^*(t)=v_1(t)$. Let $p=\sum_{i\in S(t)\setminus\Psi(t)}\sum_{v\geq v_1(t)} p_{ijv}$. Since $p\leq 1$ (from definition of $v_1(t)$, see Lemma 5) the probability of sale is at least $(1-\frac{1}{e})p$ which is bounded below by $p/2$. The Lemma follows in this case as well.

Lemma 7. *Given an impression $j(t)$, and the call out decision to a set $S(t)\neq\emptyset$ at time t, consider (i) If $\sum_{i\in S(t)}\sum_v v\varrho_{\ell(v)}y^*_{ijv\ell(v)}\geq 3\sum_{i\in S(t)}\sum_{v\geq v_1(t)} v\varrho_1 y^*_{ijv1}$ then call-out to $S(t)$ and run regular GSP. (ii) Otherwise call-out to $S(t)$ and run a single slot auction with the threshold $v^*(t)$ given by Lemma 6. This algorithm gives a revenue which is $\Omega(1)$ factor of the LP bound on efficiency which is given by $\sum_v\sum_{i\in S(t)}\sum_v v\varrho_\ell y^*_{ijv\ell(v)}$. Note that the call out decisions are based on optimizing the total value/ efficiency of the slots, and thus are feasible.*

Proof. Let the non-increasing ordered list of values that are returned for a time step be $a_1(t)$. Suppose we are in case (i). Then the revenue of GSP is at least $\sum_{r=1}^M \varrho_r a_{r+1}(t)$. Now since ϱ_r are decreasing, and $a_r(t)$ are non-increasing, we have $\sum_{r=1}^M\varrho_r a_{r+1}(t)\geq\sum_{r=1}^M\varrho_r a_r(t)-\varrho_1 a_1(t)$, which implies that

$$\mathbf{E}\left[\sum_{r=1}^M \varrho_r a_{r+1}(t)\right]\geq\mathbf{E}\left[\sum_{r=1}^M\varrho_r a_r(t)\right]-\varrho_1\mathbf{E}\left[a_1(t)\right] \quad (1)$$

We know from Section 3 that $\mathbf{E}\left[\sum_{r=1}^M\varrho_r a_r(t)\right]\geq\left(1-\frac{1}{e}\right)\sum_{i\in S(t)}\sum_v v\varrho_\ell y^*_{ijv\ell(v)}$. We observe that $\mathbf{E}\left[a_1(t)\right]\leq\sum_{i\in S(t)}\sum_{v\geq v_1(t)} vy^*_{ijv1}$. This is easily seen if we write an LP for the maximum value seen (this LP is for analysis only). Let x_{iv} be the probability that $i\in S(t)$ is the maximum with value v. Then (we drop the index j for convenience):

$$\mathbf{E}\left[a_1(t)\right]\leq LPMAX=\max\sum_i\sum_v vx_{iv} \qquad \text{s.t.} \qquad \begin{array}{l}\sum_i\sum_v x_{iv}\leq 1\\ x_{iv}\leq p_{iv}\\ x_{iv}\geq 0\end{array}$$

The optimum solution of $LPMAX$ is $x^*_{iv} = p_{iv}$ for $v > \tau$ and $x^*_{iv} \leq p_{iv}$ for one i and $v = \tau$. Here τ is the optimum dual variable for the constraint $\sum_{i\in S(t)} \sum_v x_{iv} \leq 1$. Note that $\sum_{i\in S(t)} \sum_v x^*_{iv} = 1$. For $v > v_1(t)$ we have $y^*_{ijv1} = p_{ijv}$ and $v < v_1(t)$ we have $y^*_{ijv1} = 0$. Moreover $\sum_{i\in S(t)} \sum_{v\geq v_1(t)} y^*_{ijv1} = 1$. Likewise for $v > \tau$ we have $x^*_{iv} = p_{iv}$ and $v < \tau$ we have $x^*_{iv} = 0$ and $\sum_{i\in S(t)} \sum_{v\geq\tau} x^*_{iv} = 1$.

Suppose that $\tau < v_1(t)$. We arrive at a contradiction because $\sum_{i\in S(t)} \sum_{v\geq\tau} x^*_{iv} > \sum_{i\in S(t)} \sum_{v\geq v_1(t)} y^*_{ijv1} = 1$ which implies that we are exceeding the probability mass of 1 for the maximum. On the other hand if $\tau > v_1(t)$, then we again have a contradiction that $\sum_{i\in S(t)} \sum_{v\geq v_1(t)} y^*_{ijv1} > \sum_{i\in S(t)} \sum_{v\geq\tau} x^*_{iv} = 1$ which implies $\{y^*_{ijv1}\}$ were not feasible.

As a consequence, $\tau = v_1(t)$ and for $v > \tau = v_1(t)$ we have $x^*_{iv} = y^*_{ijv1} = p_{ijv} = p_{iv}$. For $v < \tau = v_1(t)$ we have $x^*_{iv} = y^*_{ijv1} = 0$. Therefore $\mathbf{E}\left[a_1(t)\right] \leq LPMAX = \sum_{i\in S(t)} \sum_v v x^*_{iv} = \sum_{i\in S(t)} \sum_v v y^*_{ijv1}$ as claimed. Applying this claim to Equation 1, and the fact that $\sum_{i\in S(t)} \sum_v v y^*_{ijv1} \leq \frac{1}{3} \sum_{i\in S(t)} \sum_v v \varrho_\ell y^*_{ijv\ell(v)}$, we get

$$\mathbf{E}\left[\sum_{r=1}^{M} \varrho_r a_{r+1}(t)\right] \geq \left(1 - \frac{1}{e} - \frac{1}{3}\right) \sum_{i\in S(t)} \sum_v v \varrho_\ell y^*_{ijv\ell(v)}$$

Thus in this case the expected revenue is $\Omega(1)$ of the LP bound on the efficiency.

Suppose we are in case (ii). By Lemma 6 we are guaranteed an expected revenue of $\Omega(1)$ times $\varrho_1 \sum_{i\in S(t)} \sum_{v\geq v_1(t)} v p_{ijv} \geq \sum_{i\in S(t)} \sum_{v\geq v_1(t)} v \varrho_1 y^*_{ijv1} \geq \frac{1}{3} \sum_{i\in S(t)} \sum_v v \varrho_{\ell(v)} y^*_{ijv\ell(v)}$. In this case also the expected revenue is $\Omega(1)$ of the LP bound; the lemma follows.

Proof. (Of Theorem 2). Let App be the policy that approximately maximizes the efficiency. Let OPT_G be the optimum GSP-Reserve policy. Let OPT_B be the optimum policy which maximizes the total value. Given a policy P let Gsp(P) denote the expected revenue of the policy if the charged as GSP, BestKW(P) denote the expected (weighted) efficiency. Then for any policy Gsp$(P) \leq$ BestKW(P). Let R(App) be the revenue of the policy in Lemma 7. Therefore, for some absolute constant $\alpha \geq 1$,

$$\textsc{Gsp}(\text{OPT}_G) \leq \textsc{BestKW}(\text{OPT}_G) \leq \textsc{BestKW}(\text{OPT}_B) \leq LP1 \leq \alpha\ R(\textsc{App})$$

The theorem follows (again appealing to Theorem 5).

References

1. Azar, Y., Birnbaum, B.E., Karlin, A.R., Nguyen, C.T.: On revenue maximization in second-price ad auctions. In: Fiat, A., Sanders, P. (eds.) ESA 2009. LNCS, vol. 5757, pp. 155–166. Springer, Heidelberg (2009)
2. Babaioff, M., Immorlica, N., Kempe, D., Kleinberg, R.: Knapsack secretary problem with applications. In: Charikar, M., Jansen, K., Reingold, O., Rolim, J.D.P. (eds.) RANDOM 2007 and APPROX 2007. LNCS, vol. 4627, pp. 16–28. Springer, Heidelberg (2007)

3. Barlow, R.E., Marshall, A.W., Proschan, F.: Properties of probability distributions with monotone hazard rate. The Annals of Mathematical Statistics 34(2), 375–389 (1963)
4. Bhattacharya, S., Goel, G., Gollapudi, S., Munagala, K.: Budget constrained auctions with heterogeneous items. In: STOC (2010), http://arxiv.org/abs/0907.4166
5. Buchbinder, N., Jain, K., Naor, J.: Online primal-dual algorithms for maximizing ad-auctions revenue. In: Arge, L., Hoffmann, M., Welzl, E. (eds.) ESA 2007. LNCS, vol. 4698, pp. 253–264. Springer, Heidelberg (2007)
6. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a submodular set function subject to a matroid constraint (extended abstract). In: Fischetti, M., Williamson, D.P. (eds.) IPCO 2007. LNCS, vol. 4513, pp. 182–196. Springer, Heidelberg (2007)
7. Chakrabarty, D., Goel, G.: On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and gap. In: Proceedings of FOCS, pp. 687–696 (2008)
8. Chawla, S., Hartline, J., Malec, D., Sivan, B.: Sequential posted pricing and multiparameter mechanism design. In: STOC (2010), http://arxiv.org/abs/0907.2435
9. Devanur, N., Hayes, T.: The adwords problem: Online keyword matching with budgeted bidders under random permutations. In: Proc. 10th Annual ACM Conference on Electronic Commerge, EC (2009)
10. Ghosh, A., McAfee, P., Papineni, K., Vassilvitskii, S.: Bidding for representative allocations for display advertising. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 208–219. Springer, Heidelberg (2009)
11. Ghosh, A., Rubinstein, B.I.P., Vassilvitskii, S., Zinkevich, M.: Adaptive bidding for display advertising. In: WWW, pp. 251–260 (2009)
12. Guha, S., McGregor, A.: Approximate quantiles and the order of the stream. In: Proc. of PODS, pp. 273–279 (2006)
13. Guha, S., Munagala, K.: Adaptive uncertainty resolution in bayesian combinatorial optimization problems. ACM Transactions of Algorithms (to appear)
14. Kleinberg, J., Rabani, Y., Tardos, E.: Allocating bandwidth for bursty connections. SIAM J. Of Computing 30(1), 171–217 (2001)
15. Kleinberg, R.: A multiple-choice secretary problem with applications to online auctions. In: Proceedings of SODA, pp. 630–631 (2005)
16. Kleinberg, R., Slivkins, A., Upfal, E.: Multi-armed bandits in metric spaces. In: STOC, pp. 681–690 (2008)
17. Kulik, A., Shachnai, H., Tamir, T.: Maximizing submodular set functions subject to multiple linear constraints. In: SODA, pp. 545–554 (2009)
18. Marchetti-Spaccamela, A., Vercellis, C.: Stochastic on-line knapsack problems. Mathematical Programming 68, 73–104 (1995)
19. Mehta, A., Saberi, A., Vazirani, U., Vazirani, V.: Adwords and generalized on-line matching. In: FOCS, pp. 264–273 (2005)
20. Muthukrishnan, S.: Adexchanges: Research issues. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 1–12. Springer, Heidelberg (2009)
21. Myerson, R.B.: Optimal auction design. Math. of OR 6(1), 58–73 (1981)
22. Nolet, M.: http://www.mikeonads.com/2009/08/30/rtb-part-i-what-is-it/
23. Pandey, S., Olston, C.: Handling advertisements of unknown quality in search advertising. In: Proceedings of NIPS, pp. 1065–1072 (2006)

24. P. release, http://www.doubleclick.com/insight/blog/archives/doubleclick-advertising-exchange/announcing-the-new-doubleclick-adexchange.html
25. Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton (1970)
26. Tanenbaum, A.S.: Computer Networks, 3rd edn. Prentice-Hall, Englewood Cliffs (1996)
27. Vee, E., Vassilvitskii, S., Shanmugasundaram, J.: Optimal online assignment with forecasts. In: EC (2010)

Approximation Schemes for Sequential Posted Pricing in Multi-unit Auctions

Tanmoy Chakraborty[1,⋆], Eyal Even-Dar[4,⋆⋆], Sudipto Guha[1], Yishay Mansour[2], and S. Muthukrishnan[3]

[1] University of Pennsylvania, Philadelphia, PA
{tanmoy,sudipto}@cis.upenn.edu
[2] Google Israel and The Blavatnik School of Computer Science, Tel-Aviv University, Tel-Aviv, Israel
mansour.yishay@gmail.com
[3] Google Research, 76 Ninth Ave, New York, NY
muthu@google.com
[4] evendar.eyal@gmail.com

Abstract. We design algorithms for computing approximately revenue-maximizing *sequential posted-pricing mechanisms (SPM)* in *K-unit auctions*, in a standard *Bayesian model*. A seller has K copies of an item to sell, and there are n buyers, each interested in only one copy, and has some value for the item. The seller posts a price for each buyer, using Bayesian information about buyers' valuations, who arrive in a sequence. An SPM specifies the ordering of buyers and the posted prices, and may be *adaptive* or *non-adaptive* in its behavior.

The goal is to design SPM in polynomial time to maximize expected revenue. We compare against the expected revenue of optimal SPM, and provide a polynomial time approximation scheme (PTAS) for both non-adaptive and adaptive SPMs. This is achieved by two algorithms: an efficient algorithm that gives a $(1 - \frac{1}{\sqrt{2\pi K}})$-approximation (and hence a PTAS for sufficiently large K), and another that is a PTAS for constant K. The first algorithm yields a non-adaptive SPM that yields its approximation guarantees against an optimal adaptive SPM – this implies that the *adaptivity gap* in SPMs *vanishes* as K becomes larger.

1 Introduction

We consider the following **Sequential Posted Pricing** problem in a *K-unit auction.* There is a single seller with K identical copies of a single item to sell, to n prospective buyers. Each buyer is interested in exactly one copy of the item, and has a value for it that is unknown to the seller. The buyers arrive in a sequence, and each buyer appears exactly once. The arrival order may be chosen by the seller. The seller quotes a price for the item to each arriving buyer, and may quote different prices to different buyers. Assuming that buyers are rational, a buyer buys the item if the price quoted to him is less than his value for the

⋆ Part of this work was done while the authors were visiting Google Research.
⋆⋆ Part of this work was done while the author was at Google Research.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 158–169, 2010.

item, and pays the quoted price to the seller. This process stops when either K buyers have bought the item or when all buyers have arrived and left.

We focus on *pricing and ordering strategies* in the above model, called *sequential posted-price mechanisms (*__SPM__*s)*, that maximize the seller's expected revenue. Posted price mechanisms are clearly incentive compatible, and commonly used in practice. We design strategies in a *Bayesian framework*, where each buyer draws his value of the item from a distribution. These value distributions are known to the seller, and are used in designing the mechanism.

SPMs were recently studied in the general context of *Bayesian single-parameter mechanism design (BSMD)*, which includes our K-unit auction, by Chawla et. al. [7]. They designed efficiently computable SPMs for various classes of BSMD problems and compared their expected revenue to that of the optimal auction mechanism, which was given by Myerson [13]. For the K-unit auction, they showed that their SPM guarantees $(1-1/e)$-fraction of the revenue obtained by Myerson's auction. Bhattacharya et. al. [4] (as well as [7]) also used sequential item pricing to approximate optimal revenue, when the seller has multiple distinct items. However, the SPM computed by their algorithms may *not* be the optimal SPM, *i.e.* there may exist SPMs with greater expected revenue. Given that SPMs are quite common in practice, we focus in this paper on efficiently computing an optimal SPM.

Our Results. The results in [7] immediately imply a $(1-1/e)$-approximation for the problem of computing optimal SPMs in K-unit auction. We strictly improve this bound. We design two different algorithms – the first is a polynomial time algorithm that gives $(1-\frac{1}{\sqrt{2\pi K}})$-approximation, and is meant for large values of K, and the second is a polynomial time approximation scheme (PTAS) for constant K. Combining these two algorithms yield a polynomial time approximation scheme for the optimal SPM problem, for all values of K: if $K > \frac{1}{2\pi\epsilon^2}$, run the first algorithm, else run the second algorithm. Recall that a PTAS is an algorithm that, for any given constant $\epsilon > 0$, yields $(1-\epsilon)$-approximation in polynomial time (the exponent of the polynomial should be a function of ϵ only, and independent of input size).

Note that a sequential posted pricing strategy can be adaptive – it can alter its prices and the ordering of the remaining buyers based on whether the current buyer buys the item. We shall call such strategies as *Adaptive SPMs, or* __ASPM__*s*, while SPM shall refer to a non-adaptive pricing and ordering strategy. Clearly, the expected revenue from an optimal ASPM is at least that from an optimal SPM. Our first algorithm outputs an SPM, but our proof shows that it gives the same approximation guarantee of $(1-\frac{1}{\sqrt{2\pi K}})$ against an optimal ASPM. This yields a corollary that the *adaptivity gap asymptotically vanishes as K increases.* On the other hand, it is easy to construct instances with $K=2$, such that there is a constant factor *adaptivity gap*, *i.e.* gap in expected revenue between optimal SPM and ASPM. We design a third algorithm that outputs an ASPM, and is a PTAS for computing an optimal ASPM, for constant K. Again, combining this result with our first algorithm, we obtain a PTAS for the optimal ASPM problem, for all values of K. Adaptive PTAS with multiplicative approximation is rare to find in stochastic optimization problems. For example, an adaptive

PTAS for the stochastic knapsack problem has been developed very recently [3]. The theorem below summarizes our results.

Theorem 1. *There is a PTAS for computing a revenue-maximizing SPM in K-unit auctions, for all K. The same result holds for ASPMs.*

Our Techniques. The first algorithm is based on a linear programming (LP) relaxation of the problem, such that the optimal solution to the LP upper bounds the expected revenue from any ASPM. We show that this LP has an optimal integral solution, from which we construct a pricing for the buyers. The buyers are ordered simply in decreasing order of prices – it is easy to see that this is an optimal ordering policy given the prices. The LP formulation implies that if there were no limit on the number of copies the seller can sell, then the expected revenue obtained from this pricing would be equal to the LP optimum, and *at most K copies of the item are sold in expectation.* However, the algorithm is restricted to selling at most K copies in all realizations, and the result follows by bounding the loss due to this hard constraint. The interesting property we find is that *this loss vanishes as K increases.* It should be noted that an LP-based approach is used in [4]; however, they consider a more general problem with multiple distinct items, and their analysis yielded no better than constant approximation factors.

The second algorithm uses a dynamic programming approach, which is common in the design of approximation schemes. We make some key observations that reduce the problem to an *extended version of the generalized assignment problem (GAP) [14] with constant number of bins*, which has polynomial time algorithm (polynomial in the size of bins and number of items) using dynamic programming [8]. The main observation is that in any SPM, if we pick a contiguous subsequence of buyers to whom there is very small probability of selling even a single copy, and arbitrarily permute this subsequence, the resulting SPM will have almost the same expected revenue as the original SPM. This observation drastically cuts down the number of configurations that we have to check before finding a near-optimal SPM.

The third algorithm for computing ASPM is a generalization of the second algorithm, but it must now approximate a decision tree, that may branch at every step based on whether a copy is bought, instead of an SPM sequence. The key observation in this case is that there exists a near-optimal decision tree that does not branch too often, and the problem again reduces to an extension of GAP with constant number of bins.

Other Related Work. Sequential item pricing for combinatorial auctions has also been studied in prior-free settings, where no knowledge about the buyers' valuation is assumed (*eg.* [1,6]). These results compare the revenue obtained to the optimal social welfare, primarily due to lack of a better upper bound, and get no better than logarithmic approximation results. Maximizing welfare via truthful mechanisms in prior-free settings have been studied for K-unit auctions [9,10] and other combinatorial auctions [11,12]. Bayesian assumptions provide better upper bounds, and has led to constant approximation against optimal revenue for any auction [4,7]. Independently, a result similar to ours for large K in K-unit auctions has been recently proved [15]. But Bayesian assumptions can lead to tighter upper bounds on optimal sequential pricing, and that is our main contribution. A parallel posted-price approach has been used in a more complex repeated ad auction setting to get constant approximation [5].

2 Preliminaries

In a K-unit auction, there is a single seller who has K identical copies of a single item, and wish to sell these copies to n prospective buyers $B_1, B_2 \ldots B_n$. Each buyer B_i is interested in one copy of the item, and has value v_i for it. v_i is drawn from a distribution specified by cumulative distribution function (cdf) F_i that is known to the seller. The values of different buyers are *independently* drawn from their respective distributions. Without loss of generality, we assume that $K \leq n$.

Definition 1. *Let* $\mathbf{p}_{iv}$ *denote the probability that* B_i *has value* v *for the item. Let* $\tilde{p}_{iv}$ *denote the probability that* B_i *has value at least* v*. We shall call it the* success probability *when* B_i *is offered price* v*. Clearly* $\tilde{p}_{iv} = \sum_{v' \geq v} \mathbf{p}_{iv'}$*.*

We assume, for all our results, that each value distribution is discrete, with at most L distinct values in its *support* (*i.e.* these values have non-zero probability mass). Let U_{V_i} be the support set of values for the distribution of B_i, and let $U_V = \bigcup_{i=1}^{n} U_{V_i}$. We shall also assume that L is polynomial in n, and that $\tilde{p}_{iv}$ is an integral multiple of $\frac{1}{10n^2}$ for all i, v. These assumptions are without loss of generality for obtaining PTAS for optimal SPM or ASPM (discussion deferred to full version).

Definition 2. *A sequential posted-price mechanism (SPM) is a mechanism which considers buyers arrive in a sequence, and offers each of them a take-it-or-leave-it price: the buyer may either buy a copy at the quoted price or leave, upon which the seller makes an offer to another buyer. Each buyer is given an offer at most once, and the process ends when either all* K *copies have been sold, or there is no buyer remaining.*

An SPM specifies the entire sequence of buyers and prices before the process begins. In contrast, an adaptive sequential posted-price mechanism (ASPM) may decide the next buyer based on which of the current and past buyers accepted their offered prices.

Note that there can be no adaptive behavior when $K = 1$, since the process stops with the first accepted price. Thus an ASPM can be specified by a *decision tree*: each node of the tree contains a buyer and a price to offer. Each node may have multiple children. The selling process starts at the root of the tree (*i.e.* offers the price at the root to the buyer at the root), and based upon whether a sale occurs at the root, moves to one of the children of the root, and continues inductively. The process stops when either K items have been sold, or n buyers have appeared on the path in the decision tree traversed by the process – the latter nodes are the leaves of the decision tree.

It is easy to see that the decision of an optimal ASPM at any node of the tree should depend only on the number of copies of the item left and the remaining set of buyers (the latter is solely determined by the node reached by the process). Thus, each node has at most K children, at most one each for the number of copies left. Note that an ASPM may not adapt immediately to a sale – it may move to a fixed buyer regardless of the outcome. Such a node will only have a single child. Without loss of generality, we shall represent an ASPM such that each non-leaf node either has a single child or K children (some of which may even be infeasible). The latter nodes are called *branching nodes*. In this context, an SPM is simply an ASPM whose decision tree is a path.

SPM and ASPM are incentive compatible: a buyer B_i buys the item if and only if its value v_i is equal to or greater than the price offered to it, and pays only the quoted price to the seller.

Definition 3. *The revenue $R(v_1, v_2 \dots v_n)$ obtained by the seller for a given SPM is the sum of the payments made by all the buyers, which is a function of the valuations of the buyers. The expected revenue of an SPM or ASPM is computed over the value distributions $\mathbf{E}_{v_i \sim F_i} R(v_1, v_2 \dots v_n)$. An* optimal *SPM or ASPM is an SPM (respectively, ASPM) that gives the highest expected revenue among all SPMs (respectively, ASPMs).*

Let the expected revenue of an optimal SPM (or ASPM) be OPT. *An α-approximate SPM (or ASPM, respectively), where $\alpha \leq 1$, has expected revenue at least $\alpha\mathsf{OPT}$.*

2.1 Basic Results

An SPM must specify an ordering of the buyers as well as the prices to offer to them. It is worth noting that if either one of these tasks is fixed, the other task becomes easy. Proofs of many lemmas have been deferred to the full version due to lack of space.

Lemma 1. *Given take-it-or-leave-it prices to offer to the buyers, a revenue-maximizing SPM with these prices simply considers buyers in the order of decreasing prices. Given an ordering of buyers, one can compute in polynomial time a revenue-maximizing ASPM that uses this ordering (and only adapts the offered prices).*

3 LP-Based Algorithm for Large K

In this section we present our first algorithm that yields us an approximation factor that improves as K increases, and implies a vanishing adaptivity gap. The following theorem summarizes our result.

Theorem 2. *For all $K \geq 1$, if a seller has K units to sell, there exists an SPM whose expected revenue is at least $1 - \frac{K^K}{K!e^K} \geq 1 - \frac{1}{\sqrt{2\pi K}}$ fraction of the optimal ASPM. This SPM can be computed in polynomial time.*

As a first step to our algorithm, we add random infinitesimal perturbation to the values $v \in U_{V_i}$ and the associated probability values $\mathbf{p}_{iv}$, so that almost surely, U_{V_i} are disjoint, and further, all the values and probabilities are in *general position*. Intuitively, this property is used in our algorithm to break ties.

Consider any ASPM $\mathcal{P}$, that may even be randomized. Consider the event E_{iv} that B_i is offered the item at price v, and accepts the offer. Let y_{iv} denote the probability of that E_{iv} occurs when $\mathcal{P}$ is implemented. Let x_{iv} denote the probability that B_i was offered price v when $\mathcal{P}$ is implemented. Note that both probabilities are taken over the value distributions of the buyers, as well as internal randomization of $\mathcal{P}$. Naturally, we must have $y_{iv} \leq \tilde{p}_{iv} x_{iv}$. Also, by linearity of expectation, the expected revenue obtained by $\mathcal{P}$ is $\sum_{i=1}^{n} \sum_{v \in U_V} v y_{iv}$. Moreover, $\sum_{i=1}^{n} \sum_{v \in U_V} y_{iv}$ is the expected number of copies of the item sold by the seller, and this quantity must be at most K. Finally, the mechanism enforces that each buyer is offered a price at most once in any realization, and hence in expectation,*i.e.* $\sum_{v \in U_V} x_{iv} \leq 1$.

Viewing x_{iv} and y_{iv} as variables depending upon the selected ASPM, optimum of the following linear program LP-K-SPM provides an upper bound to the expected revenue from any ASPM, since any ASPM provides feasible assignment to the variables. Our algorithm involves computing an optimal solution to this program with a specific structure, and use the solution to construct an SPM.

$$
\begin{aligned}
\text{LP-K-SPM} = \quad & \max \sum_{i=1}^{n} \sum_{v \in U_V} v y_{iv} \\
y_{iv} \quad & \leq \tilde{p}_{iv} x_{iv} \qquad \forall i \in [1,n], v \in U_V \\
\textstyle\sum_{v \in U_V} x_{iv} \quad & \leq 1 \qquad \forall i \in [1,n] \\
\textstyle\sum_{i=1}^{n} \sum_{v \in U_V} y_{iv} & \leq K \\
& y_{iv}, x_{iv} \geq 0
\end{aligned}
$$

Lemma 2. *Assuming that the points in U_{V_i} and the probabilities $\tilde{p}_{iv}$ have been perturbed infinitesimally, and so are in general position, there exists an optimal* structured solution x^*_{iv}, y^*_{iv} *of* LP-K-SPM*, computable in polynomial time, such that:*

1. *for all i, v, $y_{iv} = \tilde{p}_{iv} x_{iv}$.*
2. *for each i there is exactly one v such that $x_{iv} > 0$. Let $v(i)$ denote the value for which $x_{iv(i)} > 0$.*
3. *There exists at most one i such that $1 \leq i \leq n$ and $0 < x_{iv(i)} < 1$. If such $i = i'$ exists, then $v(i') = \min_{i=1}^{n} v(i)$.*

Our algorithm for computing an SPM is as follows: Compute an optimal structured solution of LP-K-SPM. Then construct an SPM where we offer price $v(i)$ to B_i, and consider buyers in order of decreasing $v(i)$.

3.1 Approximation Factor

It remains to analyze the approximation factor of our algorithm. Let the order of decreasing prices be $B_{\pi(1)}, B_{\pi(2)} \ldots B_{\pi(n)}$. For $1 \leq i < n$, let Z_i be a two-valued random variable that is $v(\pi(i)) = z_i$ with probability $\tilde{p}_{\pi(i)v(\pi(i))} = u_i$, and 0 otherwise. To define Z_n, note that $x^*_{\pi(n)v(\pi(n))}$ in the structured optimal solution may not have been 1, so let Z_n be $v(\pi(n)) = z_n$ with probability $x^*_{\pi(n)v(\pi(n))}\tilde{p}_{\pi(n)v(\pi(n))} = u_n$ and 0 otherwise. If $Z = \sum_{i=1}^n Z_i$, then $\mathbf{E}[Z]$ is the optimum of the LP solution mentioned in Lemma 2. The revenue of the algorithm, however, is at least equal to the sum of the first K variables in the sequence $Z_1, Z_2 \dots Z_n$ that are non-zero. Let this sum be denoted by the random variable Z'. Note that $z_1 \geq z_2 \geq \dots \geq z_n$, and $\sum_{i=1}^n u_i \leq K$. The following lemma immediately implies Theorem 2.

Lemma 3. $\mathbf{E}[Z'] \geq (1 - \frac{K^K}{K!e^K})\mathbf{E}[Z] \geq (1 - \frac{1}{\sqrt{2\pi K}})\mathbf{E}[Z]$.

Proof. Let $\alpha(i) = z_i u_i$. Let the probability that we reach Z_i in the sequence before finding K non-zero variables, be given by the function $f(i, \boldsymbol{u})$ (this function is independent of $z_1, z_2 \dots z_n$), where $\boldsymbol{u} = (u_1, u_2 \dots u_n)$. Then $\mathbf{E}[Z'] = \sum_{i=1}^n f(i, \boldsymbol{u})\alpha(i)$, while $\mathbf{E}[Z] = \sum_{=1}^n \alpha(i)$. Observe that $f(i, \boldsymbol{u})$ is monotonically decreasing in i. We shall narrow down the the instances on which $\mathbf{E}[Z']/\mathbf{E}[Z]$ is minimized.

Claim. Given an instance comprising variables $Z_1, Z_2 \dots Z_n$ such that $z_i > z_{i+1}$, one can modify it to construct another instance $\tilde{Z}_1, \tilde{Z}_2 \dots \tilde{Z}_n$ such that $\mathbf{E}[Z']/\mathbf{E}[Z]$ decreases.

Thus, we can restrict our attention to instances where $z_1 = z_2 = \dots = z_n = z^*$ (say). Without loss of generality, we let $z = 1$, so that $Z_1, Z_2 \dots$ are Bernoulli variables, and $Z' = \min\{Z, K\}$. Note that the ordering of the variables do not influence Z'. The next step is to show that if we *split* the variables, keeping $\mathbf{E}[Z]$ unchanged, $\mathbf{E}[Z']$ can only decrease.

Claim. Let $Z_1, Z_2 \dots Z_n$ be Bernoulli variables, such that the *success probability* is $\mathbf{Pr}[Z_j = 1] = u_j$. Suppose that we modify the set of variables by removing Z_i from it and adding two Bernoulli variables $\tilde{Z}_i$ and $\hat{Z}_i$ to it, where $\mathbf{Pr}\left[\tilde{Z}_i = 1\right] = \tilde{u}_i > 0$ and $\mathbf{Pr}\left[\hat{Z}_i = 1\right] = \hat{u}_i > 0$, and $\tilde{u}_i + \hat{u}_i = u_i$. Then $\mathbf{E}[Z'] = \mathbf{E}[\min Z, K]$ decreases or remains unchanged due to this modification, while $\mathbf{E}[Z] = K$ remains unchanged.

Assume that the success probabilities of the Bernoulli variables are all rational – since rational numbers form a dense set in reals, this shall not change the lower bound we are seeking. Then, there exists some large integer N such that all the probabilities are integral multiples of $1/N$. Further, we can choose an arbitrarily large N for this purpose. Now, split each variable that has success probability t/N into t variables, each with success probability $1/N$. The above claim implies that $\mathbf{E}[Z']/\mathbf{E}[Z]$ can only decrease due to the splitting. Thus, it remains to lower bound $\mathbf{E}[Z']/K$ for the following instance, as $N \to \infty$: KN Bernoulli variables, each with success probability $1/N$.

For this final step, we use the well-known property that the sum of Bernoulli variables with infinitesimal success probabilities approach the *Poisson* distribution with the same mean. In particular, if P is a Poisson variable with mean

K, then the total variation distance between Z and P is at most $(1 - e^{-K})/N$ (see *e.g.* [2]), which tends to zero as $N \to \infty$. Thus, we simply need to find $\mathbf{E}[\min P, K]/K$, and this is the lower bound on $\mathbf{E}[Z']/\mathbf{E}[Z]$ that we are seeking. It can be verified that $\mathbf{E}[\min P, K] = K(1 - \frac{K^K}{K!e^K})$ which proves the lemma.

4 PTAS for Constant K

We now define an optimization problem called ExtGAP, and our PTAS for both SPM and ASPM for constant K will reduce to solving multiple instances of this problem.

ExtGAP: Suppose there are n objects, and each object has L *versions*. Let version j of object i have profit p_{ij} and size $s_{ij} \leq 1$. Also, suppose there are C bins $1, 2 \ldots C$, where bin ℓ has size s_ℓ and a *discount factor* γ_ℓ . The goal is to *place* versions of objects to bins, such that:

1. Each object can be placed into a particular bin at most once, as a unique version. If object i is placed as version j into bin ℓ, then it realizes a profit of $\gamma_\ell p_{ij}$ and a size of s_{ij}.
2. Each object can appear in multiple bins, as different versions. However, there is a given collection $\mathbb{F}_C$ of *feasible subsets* of bins $1, 2 \ldots C$. The set of bins that an object is placed into must be a feasible subset.
3. The sum of realized sizes of objects placed into any bin ℓ must be less than s_ℓ.

The profit made by an assignment of object version to bins, that satisfy all the above conditions, is the sum of realized profits by all objects placed in the bins. The goal is to find an assignment that maximizes the profit.

Lemma 4. *For all objects and versions i, j, let s_{ij} be a multiple of $1/M$ for some fixed $M \geq 2$. Then an optimal solution to ExtGAP can be found in time $(ML)^{O(C)}n$.*

4.1 PTAS for Computing SPM

Theorem 3. *There exists a PTAS for computing an optimal SPM for constant K, yielding $(1 - \epsilon)$-approximation in running time $\left(\frac{nk}{\epsilon}\right)^{poly(k,\epsilon^{-1})}$,.*

We shall, without loss of generality, give a $(1 - ck\epsilon)$-approximation, and this will imply the above theorem: putting $\epsilon = \epsilon'/ck$ will yield a $(1 - \epsilon')$-approximation.

We first establish some definitions that we shall use. Let a *segment* refer to a sequence of some buyers and prices offered to these buyers – we shall refer to parts of an SPM as segments. Let the *undiscounted contribution* $\mathcal{V}(B_i)$ of a buyer B_i, when offered price $x(B_i)$, be $\alpha(B_i) = x(B_i)\tilde{p}_{ix(B_i)}$, while its *weight* be $\tilde{p}_{ix(B_i)}$, its success probability. Undiscounted contribution $\mathcal{V}(S)$ of a segment S is the sum of undiscounted contributions of buyers in the segment, and the weight of the segment is the sum of their weights.

Given an SPM, let $dis(B)$ denote the probability that the selling process reaches buyer B. The *real contribution* of a buyer to the expected revenue is $\alpha(B)dis(B)$, and the expected revenue of the SPM is the sum of the real contributions of all the buyers. More generally, let $\gamma_\ell(B)$ denote the probability that B_i is reached with at least ℓ items remaining. Then $dis(B) = \gamma_1(B)$. The discount factor $dis(S)$ of a segment S, whose first buyer is B, is defined to be $dis(B)$. Similarly, we define $\gamma_\ell(S) = \gamma_\ell(B)$.

We present our algorithm through a series of structural lemmas, each of which follows quite easily from the preceding lemmas. The first step towards our algorithm is that we can restrict our attention to truncated SPMs.

Lemma 5. *There exists an SPM of total weight at most $K \log \frac{K}{\epsilon}$, where each buyer has discount factor at least ϵ, that gives an expected revenue of at least $(1-\epsilon)\mathsf{OPT}$. We shall refer to SPMs that satisfy this condition as* truncated*.*

We can now restrict ourself to approximating an optimal truncated SPM. The following definition of a permutable segment will be crucial to the description of our algorithm.

Definition 4. *We shall call an SPM segment* permutable *if either:*

1. *its weight is at most $\delta = \frac{\epsilon^3}{20K^3}$. We shall refer to such a permutation segment as a* small buyers segment*.*
2. *it has a single buyer, possibly of weight more than δ. In this case, we shall refer to this buyer as a* big buyer*.*

Any SPM can clearly be decomposed into a sequence of permutable segments and big buyers. Moreover, any truncated SPM can be decomposed into a sequence of at most $C = O(\frac{K \log \frac{K}{\epsilon}}{\delta})$ permutable segments. This is because if the permutable segments are maximally chosen, then two consecutive permutable segments in the decomposition either have at least one big buyer between them, or their weights must add up to more than δ (otherwise, the two segments can be joined to create one permutable segment).

Lemma 6. *The probability of selling at least one copy of the item in a small buyers permutable segment that has weight s is at least $s - s^2$. The probability of selling at least $t \geq 1$ copies (assuming that at least t copies are left as inventory) in such a segment is at most s^t. So the probability of selling exactly one copy is at least $s - 2s^2$.*

Lemma 7. *Consider a permutable segment of weight s appearing in an SPM, and let its discount factor be γ. Then the discount factor of the last buyer in the segment is at least $\gamma(1-s)$. If the undiscounted contribution of the segment is α, then the real contribution of buyers in this segment to the expected revenue is at least $\alpha\gamma(1-\delta)$ and at most $\alpha\gamma$.*

The above lemma shows that the real contribution of a segment can be approximated by the product of its discount factor and its undiscounted contribution, which does not depend on the exact buyers, their relative ordering or prices in that segment. We next show that the discount factor of a segment, given a decomposition of an SPM into permutable segments, can also be approximated as a function of the approximate sizes of preceding segments.

Lemma 8. *Given an SPM, that can be decomposed into an ordering of permutable segments $S_1, S_2 \ldots$. Let S_i be a small buyers segment. Let s be the weight of S_i.*

Then $\gamma_\ell(S_i)(1-s)+\gamma_{\ell+1}(S_i)s+4s^2 \geq \gamma_\ell(S_{i+1}) \geq \gamma_\ell(S_i)(1-s)+\gamma_{\ell+1}(S_i)s-2s^2$.

Lemma 9. *Given any SPM decomposed into $Q \leq C$ permutable segments $S_1, S_2 \ldots$, such that the weight of S_i is between $s_i+\tau$ and $s_i-\tau$ for all $1 \leq i \leq n'$, where $\tau = \delta/20C$. Consider an alternate SPM (with possibly different buyers), that has n' buyers, and the i^{th} buyer in the segment has weight s_i. Let $\rho(\ell, i)$ be the probability that the i^{th} buyer is reached in the alternate SPM with at least ℓ items remaining. Then*

$$\rho(\ell, i) - 12(\delta^2+\tau)i) \leq \gamma_\ell(S_i) \leq \rho(\ell, i) + 12(\delta^2+\tau)i).$$

If the SPM is truncated, then $dis(S_i) = \gamma_1(S_i) \geq \epsilon$, and since $i \leq Q \leq C$, $\delta = \frac{\epsilon^3}{20K^3}$ and $\tau \leq \delta/20C$, so we can get a multiplicative guarantee $\rho(1,i)(1-\epsilon) \leq dis(S_i) \leq \rho(1,i)(1+\epsilon)$.

We shall refer to the following as a *configuration*: An ordering of up to C permutable segments, where each permutable segment is specified only by the weight of the segment and big buyer respectively, each weight being a multiple of $\tau = \frac{\delta}{20C}$. Note that the configuration does NOT specify which buyer belongs to which segment, or the individual weights of the buyers. This is because a configuration is specified by at most C positive integers (weight of each segment is specified by a positive integer $z < \frac{1}{\tau}$, which indicates that the weight is $z\tau$). We shall represent a configuration z as an ordered tuple of integers $(z_1, z_2, z_3 \ldots)$. *Note that there are at most $(\frac{1}{\tau})^{O(C)} = (\frac{K}{\epsilon})^{O(K)}$ distinct configurations.* We say that an SPM has configuration z if it can be decomposed into an ordering of permutable segments $S_1, S_2 \ldots$ such that S_i has weight at least $(z_i-1)\tau$ and at most $z_i\tau$.

For any given configuration z, the expected revenue of an SPM with configuration z can be approximated, up to a factor of $(1-\delta)(1-2\epsilon)$ by a linear combination of the undiscounted contribution of the permutable segments, *where the coefficients of the linear combination depend only on z*. The coefficients are the discount factors, which can be computed by looking at an alternate SPM with a buyer for each segment, such that the i^{th} buyer has weight $z_i\tau$. This is a direct conclusion of Lemma 9 and Lemma 7. The discount factors of each buyer in the alternate SPM can be easily computed in $O(CK)$ time using dynamic programming. Let $A_z(i)$ denote the discount factor of the i^{th} buyer in the alternate SPM corresponding to z.

For any configuration z, we compute prices for the buyers, and a division of buyers into permutable segments $S_1, S_2 \ldots$ such that S_i has weight *at most* $z_i\tau$, and $\sum_i A_z(i)\mathcal{V}(S_i)$ is maximized (it is not necessary to include all buyers). This is precisely an instance of ExtGAP, where each buyer is an object, the different possible prices and the corresponding success probabilities create the different versions, and the sizes of the bins are given by z, and the feasible subsets for an object simply being that each object can get into at most one bin. This can be solved as per Lemma 4. The solution may not saturate every bin, and hence may not actually belong to configuration z. However, for any two configurations $z = (z_1, z_2, z_t)$ and $z' = (z'_1, z'_2 \ldots z'_t)$, such that $z_i \leq z'_i \; \forall 1 \leq i \leq t$, we have $A_z(i) > A_{z'}(i)$. So the SPM formed by concatenating $S_1, S_2 \ldots$ in that order generates revenue at least $(1 - 3\epsilon)$ times the revenue of the optimal sequence that has configuration z.

Thus our algorithm is to find an SPM for each configuration, using the algorithm for ExtGAP, and output the best SPM among them as the solution.

4.2 PTAS for Computing ASPM

Theorem 4. *There exists a PTAS for computing an optimal SPM, for any constant K. The running time of the algorithm is $\left(\frac{nk}{\epsilon}\right)^{(k\epsilon^{-1})^{O(k)}}$, and gives $(1-\epsilon)$-approximation.*

As mention in Section 2, an ASPM is specified by a decision tree, with each node containing a buyer and an offer price. We extend some definitions used for SPMs to ASPMs. The *weight of a node* is the success probability at this node conditioned on being reached. A *segment* in an ASPM is a contiguous part of a path (that the selling process might take) in the decision tree. A segment is called *non-branching* if all but possibly the last node are non-branching. Other definitions such as weight and contribution of a segment are identical. A *permutation segment* is a non-branching segment satisfying properties as defined earlier (Definition 4). The *discount factor* of a node (or a segment starting at this node, or a subtree rooted at this node) is the probability that the node is reached in the selling process.

Consider any ASPM whose tree is decomposable into D non-branching segments, each of weight at most H. (Note that $D = 1$ for an SPM.) Then the entire tree of a truncated ASPM decomposes into $C = O(DH/\delta)$ permutable segments. We shall refer to such ASPMs as *C-truncated ASPMs.* A configuration for a C-truncated ASPM shall now list the weights of at most C permutable segments and also specify a *tree structure* among them, *i.e.* the parent segment of each segment in the decision tree. Moreover, since each path can have no more than C segments, it is sufficient to specify the weights to the nearest multiple of $\tau = \delta/20C$, to get the discount factor of each segment with sufficient accuracy. So there are $(C/\tau)^{O(C)} = C^{O(C)}$ configurations for C-truncated ASPMs.

For each configuration, we can use ExtGAP (with C bins) to compute an ASPM that is at least $(1 - \epsilon)$ times the revenue of an optimal ASPM with that configuration, as before. The discount factor of each permutable segment in the configuration can be computed with sufficient accuracy, similar to Lemma 9. Iterating over all possible configurations, we can find a near-optimal C-truncated

ASPM. Solving ExtGAP requires time exponential in the number of bins (see Lemma 4), so the entire running time of the above algorithm is $\left(\frac{nkC}{\epsilon}\right)^{O(C)}$). Lemma 10 gives the required non-trivial characterization.

Lemma 10. *There exists an ASPM with the following properties:*

1. *Its expected revenue is at least* $(1-\epsilon)$ *times the expected revenue of the optimal ASPM.*
2. *The decision tree is decomposable into* $D = (K/\epsilon)^{O(K)}$ *non-branching segments.*
3. *Each non-branching segment in the tree has weight at most* $H = (K/\epsilon)^{O(1)}$.
4. *Each path in the tree consists of at most* $(K/\epsilon)^{O(1)}$ *permutable segments.*

References

1. Balcan, M.F., Blum, A., Mansour, Y.: Item pricing for revenue maximization. In: ACM Conference on Electronic Commerce, pp. 50–59 (2008)
2. Barbour, A.D., Hall, P.: On the rate of poisson convergence. Mathematical Proceedings of the Cambridge Philosophical Society 95, 473–480 (1984)
3. Bhalgat, A., Goel, A., Khanna, S.: Improved approximation algorithms for stochastic knapsack problems. In: SODA (2011)
4. Bhattacharya, S., Goel, G., Gollapudi, S., Munagala, K.: Budget constrained auctions with heterogeneous items. In: STOC, pp. 379–388 (2010)
5. Chakraborty, T., Even-Dar, E., Guha, S., Mansour, Y., Muthukrishnan, S.: Selective call out and real time bidding. In: WINE (2010)
6. Chakraborty, T., Huang, Z., Khanna, S.: Dynamic and non-uniform pricing strategies for revenue maximization. In: FOCS, pp. 495–504 (2009)
7. Chawla, S., Hartline, J.D., Malec, D.L., Sivan, B.: Multi-parameter mechanism design and sequential posted pricing. In: STOC, pp. 311–320 (2010)
8. Chekuri, C., Khanna, S.: A polynomial time approximation scheme for the multiple knapsack problem. SIAM J. Comput. 35(3), 713–728 (2005)
9. Dobzinski, S., Dughmi, S.: On the power of randomization in algorithmic mechanism design. In: FOCS, pp. 505–514 (2009)
10. Dobzinski, S., Nisan, N.: Mechanisms for multi-unit auctions. In: ACM Conference on Electronic Commerce, pp. 346–351 (2007)
11. Dobzinski, S., Nisan, N., Schapira, M.: Truthful randomized mechanisms for combinatorial auctions. In: STOC, pp. 644–652 (2006)
12. Feige, U.: On maximizing welfare when utility functions are subadditive. In: STOC, pp. 41–50 (2006)
13. Myerson, R.B.: Optimal auction design. Mathematics of Operations Research 6, 58–73 (1981)
14. Shmoys, D.B., Tardos, É.: An approximation algorithm for the generalized assignment problem. Math. Program. 62, 461–474 (1993)
15. Yan, Q.: Mechanism design via correlation gap. In: SODA (2011)

Truthful Mechanisms for Exhibitions

George Christodoulou[1,2], Khaled Elbassioni[1], and Mahmoud Fouz[3]

[1] Max-Planck-Institut für Informatik, Saarbrücken, Germany
[2] Cluster of Excellence "Multimodal Computing and Interaction", Universität des Saarlandes
[3] FR Informatik, Universität des Saarlandes, Saarbrücken, Germany
gchristo,elbassio,fmahmoud@mpi-inf.mpg.de

Abstract. We consider the following combinatorial auction: Given a range space $(U, \mathcal{R})$, and m bidders interested in buying only ranges in $\mathcal{R}$, each bidder j declares her bid $b_j : \mathcal{R} \rightarrow \mathbb{R}_+$. We give a deterministic truthful mechanism, when the valuations are single-minded: when $\mathcal{R}$ is a collection of fat objects (respectively, axis-aligned rectangles) in the plane, there is a truthful mechanism with a $1+\epsilon$- (respectively, $\lceil \log n \rceil$)-approximation of the social welfare (where n is an upper bound on the maximum integral coordinate of each rectangle). We also consider the non-single-minded case, and design a randomized truthful-in-expectation mechanism with approximation guarantee $O(1)$ (respectively, $O(\log m)$).

1 Introduction

In a combinatorial auction, there are m bidders competing on a finite set of k items for sale. The preferences of a player over the different subsets of items are expressed via a valuation function, that assigns to every subset of items a non-negative real value. An important, well-studied objective is to allocate the items to the bidders in a way that maximizes the *social welfare*, i.e., the sum of the valuations of the players on the allocated subsets.

We consider auctions where the bundles of items of interest have a geometric interpretation, i.e., they form connected geometric objects. Our work is inspired by the following applications: An owner of a space considers renting her land to exhibitors who will participate in a certain exhibition (say a computer show). The exhibitors bid on certain subsets of the space, and the owner has to decide which parts of the land she should allocate to which bidder, and how she should charge each winner. Typically, the owner may only allow bidders to bid on regions of certain shape (say squares or rectangles), and each bidder has a bid that depends on the location of the region (for instance, central regions or regions close to the entry of the whole exhibition can be more valuable to the bidder). A similar situation arises in advertisements: there is a certain space on a screen that can be used for displaying ads. A number of bidders compete for the total space, and have their individual valuations for each region on the screen[1]. Again, it is natural to expect the regions of interest to be squares or rectangles.

[1] See, for instance, "http://www.milliondollarhomepage.com"

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 170–181, 2010.

In order to capture the above scenarios, we consider the following combinatorial auction: Let $U \subseteq \mathbb{R}^2$ be a set of points in the plane. Given m customers who are interested in buying subsets of U, each customer declares her bids on certain subsets of U (called ranges). Based on the bids, the auctioneer has to decide a feasible allocation of subsets to customers, and a payment to be charged to every winner (who gets allocated a non-empty subset).

In the above applications, it is natural to consider the situation where the possible subsets are connected regions in the plane, and moreover, those that are axis-aligned, or have some sort of *fatness*, such as squares or discs.

All the participants in an auction are selfish agents whose only goal is to maximize their utility, i.e., they want to obtain the bundle of items that maximizes the value minus the price. Therefore, they will try to manipulate the mechanism by misreporting their true values if this will increase their utility. In order to neutralize the effects of selfishness, a standard desired property of a mechanism that determines the allocation and payment is *truthfulness*, or *incentive-compatibility.* We look for truthful mechanisms, where the best strategy of each bidder is to report his true valuation. At the same time, we are interested in maximizing the *social welfare*, i.e., the sum of the valuations of the winners on their allocated subsets. The celebrated VCG mechanism [7, 9, 18], achieves both goals, but it runs in exponential time for most interesting scenarios. We consider polynomial-time mechanisms that approximate the optimal social welfare. The quality of an allocation is measured by the ratio of its total value to the optimal total value.

An important, well-studied special class of valuation functions, is the class of *single-minded valuations* (SM), where each bidder is interested in obtaining a particular subset of items. Lehmann et al. [14] showed that when the bidders are single-minded, there exists a truthful auction with approximation ratio $\sqrt{k}$, where k is the number of items. They also proved that this is the best ratio possible, even disregarding strategic issues, like truthfulness, unless NP = ZPP.

Our main contribution is the design of truthful mechanisms with approximation guarantees for two natural geometric settings of single-minded bidders; where the regions are either *axis-aligned rectangles* or *fat* objects. Intuitively, fat objects do not contain long and skinny parts; they are a well-known generalization of many common geometric objects like discs and squares. For instance, in the aforementioned motivating example on exhibitions, the regions of interest are usually rectangular with bounded width-to-height ratio; hence, they are fat. Apart from these results for the single-minded case, we also provide randomized truthful-in-expectation mechanisms for the non-single-minded case (non-SM).

The paper is organized as follows. We discuss related work in Section 1.1, and we give an overview of our results along with a comparison to previous work in Section 1.2. In Section 2, we give a precise problem formulation and describe some preliminaries. Finally, in Sections 3 and 4, we describe our results in detail.

1.1 Related Work

The work that is most related to ours is that by Babaioff and Blumrosen [2]. Motivated by similar applications, the authors in [2] study the single-minded version of the above problem. They generalize the greedy mechanism of Lehmann et al. [14] to obtain truthful mechanisms whose approximation guarantees are parameterized by the *aspect ratio* of the regions under interest, defined as the maximum ratio between the diameter and width of any object[2]. Two different informational settings are considered in [2]. In the *Known Single-Minded model (KSM)*, the auctioneer knows the actual range of each bidder but not the true valuation. In the more general *Unknown Single-Minded model (USM)*, both the ranges and valuations are private information of each bidder. The truthful mechanisms obtained in [2] have approximation guarantees $O(R^{4/3})$ in the USM model and $O(R)$ in the KSM model, where R is the maximum aspect ratio of the objects. These approximation guarantees are improved to $O(R)$ in case of arbitrary rectangles in the USM model, and $O(\log R)$ in case of axis-parallel rectangles in the KSM model. The latter result is obtained by using the bid-monotonic algorithms of Khanna et al. [12], which we discuss below.

In [12], the authors consider the *rectangle packing* problem. Here, given a set of m weighted rectangles in the plane, the problem is to find a disjoint collection of at most p rectangles and maximum weight. Note that without the restriction on the cardinality of the set, the problem corresponds to assigning disjoint rectangles to customers such that the social welfare is maximized. The authors of [12] assume that n rectangles are given on an $n \times n$ grid (i.e., they have integral coordinates) and obtain an $O(\log n)$ approximation algorithm that runs in time $O(n^2 p + np \log n)$. In contrast, our approach (which is similar to the one in [1]) for rectangles in the SM case only assumes that the rectangles have integral x-coordinates in $[0, n]$[3]. It achieves an approximation ratio of $\lceil \log n \rceil$, and runs (with a slight modification for the stated variant) in time $O(m \log m + mp \log n)$, where m is the number of rectangles. Note that m could be much smaller than the 'width' n of the plane.

Our approach for the non-SM case is based on rounding the LP-relaxation for the social welfare maximization (SWM) problem and then resorting to the general results of Lavi and Swamy [13]. Motivated by secondary spectrum auctions, and independently from our work, Hoefer et al. [10] considered a more general setting for the non-SM case, in which the feasible allocations are determined by a *conflict graph*, which is assumed to have a small *inductive independence number*[4]. They obtain randomized truthful mechanisms based also on an LP formulation of an extension of the SWM problem, combined with the results in [13]. We note that the intersection graph of a set of fat objects has a small

[2] More precisely, the aspect ratio of an object is the ratio between the maximum distance between any two points in the object and the minimum length of a projection of the object along any direction; equivalently, it is the ratio between the diameters of the minimum enclosing and maximum enclosed disc.

[3] Actually, it is already sufficient to assume that the rectangles lie in $[0, n] \times \mathbb{R}$ and have a minimum width of at least 1.

[4] A graph is said in [10] to have an inductive independence number ρ if there exists an ordering on the vertices s.t., for each vertex v, the subgraph induced on the neighbors of v, that precede v in the order, has independence number at most ρ.

inductive independence number, and hence some of the results in [10] can also be adapted to our setting.

1.2 Results and Techniques

Our main results concern single-minded valuations. We show that there is a truthful mechanism with a $(1+\epsilon)$-approximation of the social welfare, provided that the interesting regions (that the customers bid on) are fat ranges in $[0,1]^d$. This result is best possible since the SWM problem is already NP-hard for the setting of fat ranges [8, 11]. When the interesting regions are axis-aligned rectangles with integral x-coordinates, the approximation ratio will be $\lceil \log n \rceil$, assuming an upper bound of n on the maximum integral x-coordinate for each rectangle. We remark that, when $n = poly(m)$, getting a better bound than $O(\log n)$ will mean to get a better approximation guarantee than $O(\log m)$ for the SWM problem for rectangle ranges, which is a standing open problem[5].

Theorem 1. *Let $\epsilon > 0$ be an arbitrary constant. There is a polytime deterministic truthful mechanism, in the single-minded case, with approximation ratio $1+\epsilon$ for β-fat ranges with $\beta = O(1)$ and running time $m^{O(\epsilon^{-d+1})}$. For axis-aligned rectangles, there is a truthful mechanism, in the single-minded case, with approximation guarantee $\lceil \log n \rceil$ and $O(m \log(mn))$ running time.*

Here, β-fatness is a measure for how fat the object is [6]; for instance, a disc has a constant fatness, while a line segment has unbounded fatness. A precise definition is given in Section 2.

The results of Theorem 1 improve some of the results in [2]. For the case of constant aspect ratio $R = O(1)$, the mechanism of [2] achieves a constant approximation for the KSM information model, while we are able to show a truthful PTAS in the more general case of the USM model. It is important to emphasize that we consider only the USM model here; the mechanism is not aware of neither the true sizes and places of the objects nor their true values. In particular, for the fat ranges case, we assume that we are given a priori a square of size L in which all figures are guaranteed to lie inside. We also assume that the fatness parameter β is a known constant. For axis-parallel rectangles, we assume that all possible rectangles have integral coordinates, ranging from 1 to n along one of the principal directions. In that sense, our USM model is a bit weaker than the USM model in [2], but much stronger than the KSM model, as once we know the exact sizes and positions of the rectangles, we can compute the premises required in our model. It is not hard to see that the definition we use for fatness, given precisely below, is more general than bounding the aspect ration; in other words, $R = O(1)$ implies $O(1)$-fatness. Thus, we get a truthful PTAS when $R = O(1)$. For the case of axis-parallel rectangles, we strengthen the $O(\log R)$ result for the KSM model in [2]. In particular, in the above USM model, we get a truthful mechanism with a $\lceil \log n \rceil$-approximation ratio.

[5] Recently, Chalermsook and Chuzhoy [5] gave an $O(\log \log m)$-approx. algorithm for unit valuations. However, their result does not seem to extend to general valuations.

For the analysis of the algorithms in the single-minded case, we introduce a general framework for mechanism design and give sufficient conditions for the *monotonicity* of mechanisms that follow the framework. Roughly speaking, monotonicity is essentially a sufficient condition for truthfulness and ensures that when a range is shrunk or its value increased, then it remains in the solution if it was there before the change. The framework captures all algorithms that first decompose the instance into several (smaller) instances, solve each instance independently, and then return the best of all these solutions. Both for the fat ranges and the rectangles, the algorithms presented fit the framework and are shown to satisfy the sufficient conditions for monotonicity. The framework might also be of independent interest to show monotonicity of other mechanisms.

For the fat-ranges case, we modify Chan's algorithm [6] for packing d-dimensional fat ranges to get a monotone PTAS. For the rectangles case, we use a natural decomposition technique to partition the set of rectangles into different 'levels'. An instance is then formed by taking all rectangles of one level and, in addition, extensions of all rectangles of higher levels. Each instance is then solved by projecting all its rectangles to one line and solving the corresponding interval packing problem by dynamic programming to optimality. Although in [12], the authors also decompose the problem to instances of the interval packing problem, the decomposition we use is simpler and yields a better running time.

For the case of general valuations, that we refer to as the *non-single-minded* (non-SM) case, the results of Lavi and Swamy [13] allow us to derive randomized truthful-in-expectation mechanisms from LP-rounding algorithms for approximating the optimum social welfare. By developing such rounding algorithms, we obtain truthful-in-expectation mechanisms with the following guarantees.

Theorem 2. *There is a polytime truthful-in-expectation mechanism, in the non-SM case, with approx. ratios* 4β *for* β*-fat ranges, and* $4 \log m$ *for rectangles.*

2 Problem Definition and Preliminaries

There is an extensive literature on the design of truthful mechanisms for combinatorial auctions. In this section, we will only give the basic definitions needed in this paper. For an excellent introduction, we refer the reader to the book by Nisan et al. [15] and the references therein.

Let $[m] = \{1, \ldots, m\}$. Let $(U, \mathcal{R})$ be a range space, defined by a set U and a collection $\mathcal{R}$ of subsets of U, called *ranges*. Given m bidders, we assume that bidder j declares her bid $b_j(r)$ over every possible range $r \in \mathcal{R}$. The true valuation of the bidder j on range r will be denoted by $v_j(r)$. It is naturally assumed that the empty range $\emptyset \in \mathcal{R}$, and that $v_j(\emptyset) = 0$ for all j. In the single-minded case, each bidder is only interested in a single range $r_j \in \mathcal{R}$, that is, $v_j(r) = v_j(r_j)$ for all $r \supseteq r_j$ and $v_j(r) = 0$ for all $r \not\supseteq r_j$.

In this paper, U will be a set of points in d-dim. Euclidean space, and $\mathcal{R}$ a collection of connected regions. By the *size* of a region, we mean the side length of its smallest enclosing hypercube. More specifically, we will consider rectangles with integral x-coordinates ranging from 1 to n and *fat* objects in $[0, 1]^d$.

There are several definitions of fatness in the literature [3, 6, 17]. We use the following definition by Chan [6]. Recall that a *box* is the generalization of a square to higher dimensions. In what follows we assume that boxes are axis-aligned.

Definition 1 ([6]). *Let $\beta > 0$ be a constant. A collection C of ranges is β-*fat *if for any ℓ and any box B of size ℓ, we can choose β points $H(B)$ s.t. every range intersecting B and has size at least ℓ contains one point in $H(B)$.*

E.g., axis-aligned squares have fatness of $\beta = 4$ since for any box B of size ℓ and any such square S of size at least ℓ that intersects B, S must contain one of the four corners of B. On the other hand, line segments have unbounded fatness.

The social welfare maximization problem (SWM) is to find the optimum *integer* solution of the following linear program:

$$\max \sum_{\substack{j \in [m] \\ r \in \mathcal{R}}} v_j(r) x_{j,r} \tag{P}$$

$$\text{s.t.} \sum_{\substack{j \in [m] \\ r \in \mathcal{R}: u \in r}} x_{j,r} \leq 1 \qquad \text{for all } u \in U \tag{1}$$

$$\sum_{r \in \mathcal{R}} x_{j,r} = 1 \qquad \text{for all } j \in [m] \tag{2}$$

$$x_{j,r} \geq 0 \qquad \text{for all } j \in [m], r \in \mathcal{R}.$$

Informally, we want to find an allocation that maximizes the total valuations (i.e., the social welfare) while making sure that (1) each item is only assigned to one bidder and (2) each bidder is only assigned at most one range.

A *mechanism* takes as an input the set of bids $\{b_j : j \in [m]\}$ and outputs (i) a feasible allocation, that is, a 0/1-vector $\tilde{x}$ (or in other words an assignment of a (possibly empty) range r_j to each bidder $j \in [m]$) satisfying (1) and (2); and (ii) a payment $p : [m] \to \mathbb{R}_+$, that is, an amount $p_j \geq 0$ that is charged to bidder j, for all $j \in [m]$. The mechanism is said to satisfy *individual rationality* if it results in a non-negative utility for each bidder, that is, $v_j(r_j) - p_j \geq 0$, if the mechanism allocates r_j to bidder j, for $j = [m]$. For that, we assume that the bids are also single-minded, i.e., similar to the valuations, for all bidders $j \in [m]$ there exists a range $r'_j \in \mathcal{R}$ such that for all $r \supseteq r'_j$ we have $b_j(r) = b_j(r'_j)$ and for all $r \not\supseteq r'_j$, $b_j(r) = 0$. Thus, we can specify b_j by a range r_j and $b_j(r_j)$ and can write $(r_j, b_j(r_j))$ for a bid. The mechanism is said to be *truthful* if a bidder cannot improve his utility, under the mechanism, by bidding something different from his true valuation, regardless of the other players' bids. Formally, if the mechanism outputs the allocation-payment (r_j, p_j) for bidder $j \in [m]$, given the vector of bids (b_j, b_{-j}), and it outputs $(\hat{r}_j, \hat{p}_j)$ given the vector of bids (v_j, b_{-j}), where b_{-j} denotes the vector of bids of all other bidders $j' \neq j$, then the mechanism will be truthful if it satisfies $v_j(\hat{r}_j) - \hat{p}_j \geq v_j(r_j) - p_j$, for all j and all v_j, b_j, b_{-j}.

For the single-minded case, a sufficient condition for truthfulness is *monotonicity* and *critical payment*. The following formulation is adopted from [15, Lemma 11.9] and slightly adjusted to our notation.

Lemma 1 ([15]). *A mechanism for single-minded bidders in which losers pay 0 is incentive compatible if it satisfies the following two conditions: (i)* Monotonicity: *A bidder j who wins with bid $(r_j, b_j(r_j))$ keeps winning for any $b'_j > b_j$ and for any $r \subset r_j$ (for any fixed settings of the other bids); (ii)* Critical Payment: *A bidder j who wins pays the minimum value needed for winning: the infimum of all values b such that (r_j, b) still wins.*

We note that usually it is easy to define prices that satisfy the critical payment condition. Thus, the main difficulty in obtaining truthful mechanisms lies in ensuring monotonicity.

For randomized mechanisms, the mechanism is said to be *truthful-in-expectation* if $\mathbb{E}[v_j(\hat{r}_j) - \hat{p}_j] \geq \mathbb{E}[v_j(r_j) - p_j]$ for all $j \in [m]$ and all v_j, b_j, b_{-j}, where the expectation is taken over the random choices made by the algorithm.

The approximation guarantee of the mechanism is the ratio between the social welfare given by the mechanism $\sum_{j\in[m]} v_j(r_j)$ and the optimal social welfare, defined as the optimal integral solution of LP (P). Lavi and Swamy [13] showed that an "LP-based" approximation algorithm for SWM can be used to get a truthful-in-expectation mechanism of the same approximation guarantee.

Theorem 3 ([13]). *An LP-based α-approximation algorithm for the SWM problem can be used to obtain a truthful-in-expectation mechanism with approximation guarantee α.*

3 The Single-Minded Case

In this section, we introduce a general framework for mechanism design in the single-minded case and give sufficient conditions for monotonicity of such mechanisms. We will then apply our framework to the two cases where the ranges are fat objects and rectangles.

3.1 A General Framework for Monotone Mechanisms

The SWM problem can be thought of as finding a maximum-weight packing among a set of ranges with given weights $b(1), \ldots, b(m)$. Several existing algorithms for solving the packing problem can be put into the following framework:

(F1) k ordered instances $\mathcal{R}_1, \ldots, \mathcal{R}_k$ are obtained from the original input $\mathcal{R}$;
(F2) an algorithm $\mathcal{A}$ is used to solve each instance $\mathcal{R}_i$ independently, returning a packing $\mathcal{R}'_i$;
(F3) the packing $\mathcal{R}' = \text{argmax}_i\{b(\mathcal{R}'_i)\}$ with maximum weight is returned, where ties are broken according to the order of the instances.

The following lemma describes sufficient conditions for such an algorithm to be monotone. In these conditions, we will consider changing one range or its corresponding bid, while all other ranges and bids are kept the same.

Lemma 2. *An algorithm that satisfies (F1)-(F3) is monotone if it satisfies further the following conditions (with $\mathcal{A}$ denoting the algorithm from (F2)):*

(C1) If a range is shrunk or its bid increased, the order of the instances is unaffected and no new instances are created.
(C2) Algorithm $\mathcal{A}$ is monotone.
(C3) If a range r is shrunk or its bid increased, then for all instances $\mathcal{R}_i$, the total weight of the solution returned by $\mathcal{A}$ on $\mathcal{R}_i$ can not decrease if r is contained in the solution (after the change) and remains the same otherwise.

Remark 1. Note that (C3) is always satisfied if we assume that the algorithm $\mathcal{A}$ is (1) *monotone in its output value*, i.e., if a range is shrunk or its bid increased, $\mathcal{A}$ returns a solution of total weight at least as large as the weight of the solution returned before the change, and (2) for all instances $\mathcal{R}_i$, changing range r can only result in the addition or drop of r from instance $\mathcal{R}_i$, i.e., all other ranges remain in exactly the same instances as before.

In the next subsections, we apply this general framework to the special cases of ranges of bounded fatness and axis-aligned rectangles, to prove Theorem 1.

3.2 The Fat Ranges Case

We describe now how to modify Chan's algorithm [6] for packing m d-dimensional fat ranges to get a monotone PTAS for this case. By scaling, we can assume that all objects lie in $[0, 1]^d$. We define the size of a range r to be the size of the smallest bounding box $B(r)$. We first sketch Chan's algorithm, which follows the framework (F1)-(F3). The idea is to choose the instances $\mathcal{R}_j$ such that they can be solved to optimality using dynamic programming. For that, the space is divided recursively by the following procedure: start with a hypercube containing all the ranges, then partition every non-empty hypercube recursively into 2^d equally sized hypercubes. Such a division is represented by means of a 2^d-dimensional tree (for $d = 2$ it is called a *quadtree*). Each instance $\mathcal{R}_j$ is obtained from a (different) shifted version of this basic division by including all ranges into $\mathcal{R}_j$ that are 'large' w.r.t. the smallest cell in which they are contained. Each instance is then solved to optimality and the one of maximum value is returned.

The dynamic program (DP) introduces for each cell $\mathcal{C}$ and disjoint subcollection $\mathcal{B}$ of the ranges crossing the boundary of $\mathcal{C}$ the table entry $pack[\mathcal{C}, \mathcal{B}]$, which is defined as the maximum weight of a subcollection $\mathcal{B}'$ of ranges that lie completely inside $\mathcal{C}$, such that $\mathcal{B} \cup \mathcal{B}'$ is a disjoint collection. For a collection $\mathcal{B}'$ of ranges, let $\mathcal{B}'|_{\partial\mathcal{C}}$ be the subsets of ranges from $\mathcal{B}'$ crossing the boundary of $\mathcal{C}$. Let $\mathcal{C}_1, \ldots, \mathcal{C}_{2^d}$ denote the children of cell $\mathcal{C}$ in the tree. The table is filled bottom-up by the easily verifiable formula: $pack[\mathcal{C}, \mathcal{B}] = \max_{\mathcal{B}'} \Big(\sum_{i=1}^{2^d} pack[\mathcal{C}_i, (\mathcal{B}' \cup \mathcal{B})|_{\partial\mathcal{C}_i}] + b(\mathcal{B}') \Big)$, where $b(\mathcal{B}')$ is the sum of bids of all ranges in $\mathcal{B}'$, and where the maximum is over all disjoint subcollections $\mathcal{B}' \subseteq \bigcup_i \mathcal{R}|_{\partial\mathcal{C}_i} \setminus \mathcal{R}|_{\partial\mathcal{C}}$ s.t. $\mathcal{B}' \cup \mathcal{B}$ is a disjoint collection.

The problem with such a straightforward approach is that the size of the table might be too large since we might have to consider an exponential number of disjoint subcollections $\mathcal{B}$ for some cell $\mathcal{C}$. To overcome this problem, we only consider ranges that are large with respect to their (smallest) enclosing cell.

Definition 2 ([6]). *A range of size ℓ is k-*aligned *if it is inside a tree cell of size at most $k\ell$.*

The key observation now is that when all objects are β-fat and k-aligned, the packing problem can be solved *exactly* in polytime using dynamic programming. The reason is that any feasible solution to the packing problem can have at most $K = 2\beta dk^{d-1}$ objects that intersect the boundary of any cell and there are only $O(m)$ many relevant cells in the bottom-up approach. Thus, for every such cell we consider only collections $\mathcal{B}'$ with $|\mathcal{B}'| \leq 2^d K$ ranges. Hence, the table will have at most $m^{O(K)}$ entries. Note that K is a constant if k and d are constants. In summary, we have the following lemma.

Lemma 3 ([6]). *If all ranges in C are β-fat and k-aligned, then the packing problem can be solved in $m^{O(\beta dk^{d-1})}$ time.*

Chan now considers $O(k)$ shifts of the basic division, each of which defines an instance $\mathcal{R}_j$ by removing all ranges that are not k-aligned w.r.t. that shift. Each instance is solved to optimality by the above DP, and the packing of maximum value is returned. Choosing k to be roughly d/ϵ, a $1+\epsilon$-approximation is achieved.

Theorem 4 ([6]). *Given a collection of m $O(1)$-fat objects in $\mathbb{R}^d$, the above algorithm gives a $1+\epsilon$-approximation to the packing problem in $m^{O(\epsilon^{-d+1})}$ time.*

A truthful algorithm. In order to make the algorithm monotone, we show how to ensure that all conditions of Lemma 2 are met. First, we order the instances given by the shifts in an arbitrary, but fixed order. Second, we need to ensure that, for each shifted instance, the (quadtree) partitioning is *independent* of the ranges. This can be achieved by assuming that all ranges in the instance lie in a fixed range, say $[0, 1)^d$. So the initial cell will be the unit hypercube. We keep partitioning a cell until it does not contain any range in its interior. Even though the number of cells is not polynomial in m, it is easy to see that one can still implement the DP in polytime (assuming fixed d), by 'zooming' into the relevant cells (those that are intersected by at least one range).

Since the partitioning of the instances is independent from the ranges, (C1) is immediately satisfied. In order to satisfy (C2) and (C3) we do *not* drop any ranges from the instances even if they are not k-aligned, and modify the DP accordingly. In particular, we now have a table entry $pack[\mathcal{C}, \mathcal{B}]$ for every cell $\mathcal{C}$ and every disjoint subcollection $\mathcal{B} \subseteq \mathcal{R}|_\mathcal{C}$ of ranges that either cross the boundary of $\mathcal{C}$ or are completely inside $\mathcal{C}$, *and* have cardinality at most K. In contrast to the DP before, the maximum in the above recurrence is now over all subcollections of ranges in $\mathcal{H}(\mathcal{C}, \mathcal{B})$, where $\mathcal{H}(\mathcal{C}, \mathcal{B})$ is the set of disjoint subcollections of ranges $\mathcal{B}' \subseteq \mathcal{R}|_\mathcal{C} \setminus \mathcal{R}|_{\partial\mathcal{C}}$ such that $\mathcal{B}' \cup \mathcal{B}$ is a disjoint collection and $|\mathcal{B}'| \leq K$:

$$pack[\mathcal{C}, \mathcal{B}] = \max_{\mathcal{B}' \in \mathcal{H}(\mathcal{C},\mathcal{B})} \Big(\sum_{i=1}^{2^d} pack[\mathcal{C}_i, (\mathcal{B}' \cup \mathcal{B})|_{\mathcal{C}_i}] + b(\mathcal{B}') \Big). \tag{3}$$

Note that in contrast to the original DP we do not only consider ranges that lie on the boundary of the child cells of $\mathcal{C}$ but all ranges that lie (completely) in $\mathcal{C}$. Despite these modifications, we still have $m^{O(K)}$ relevant table entries.

Clearly, the solution computed by the modified DP is no worse than the solution computed by the original DP (since we just search over a larger space). Still, we have to ensure that the DP is monotone (i.e., satisfies (C2)). For that we specify how to break ties during the computation of $pack[\mathcal{C}, \mathcal{B}]$ in (3). Consider an arbitrary but fixed order on all ranges: $r_1, \ldots, r_m$. Note that during the computation (3), we consider all optimal subcollections $\mathcal{B}' \in \mathcal{H}(\mathcal{C}, \mathcal{B})$. Whenever there is more than one such subcollection we return the subcollection that forms the lexicographically smallest ordered sequence of ranges.

Lemma 4. *Let $\epsilon > 0$. The modified algorithm for fat ranges returns a $1+\epsilon$-approximate, monotone solution and runs in time $m^{O(\epsilon^{-d+1})}$.*

In order to ensure that the algorithm is truthful, we still have to specify the critical payment of each winner. We use a similar payment as in the VCG mechanism (see e.g. [15]). Let W_j be the maximum value returned by the DP over all instances if we remove the object r_j and W_j' the value if in addition to r_j we also remove all objects that intersect r_j. Clearly, $W_j' \leq W_j$. Then define the payment of bidder j to be $p_j = W_j - W_j'$. To see why this definition indeed yields the critical value for bidder j, note that if bidder j bids less, no solution of maximum value over all instances contains r_j, so bidder j does not win. On the other hand, if bidder j bids $p_j + \epsilon$ for any $\epsilon > 0$, every solution of maximum value must contain r_j, so bidder j wins for sure. Finally, individual rationality follows as in the VCG mechanism, since for a winner j, $b_j + W_j' \geq W_j$.

3.3 The Rectangles Case

We now consider the case when the given objects are axis-aligned rectangles with integral x-coordinates assuming values from 1 to n. We assume that n is a power of 2; otherwise we can extend the interval to the nearest power of 2. In that case, it is easy to verify that we achieve an approximation ratio of $\lceil \log n \rceil$.

Any set of rectangles $r_1, \ldots, r_m$ can be partitioned into at most $k = \log m$ sets, which can be interpreted as lying at different "levels". The rectangles at level $\ell \in [k]$ can be further partitioned into $h = 2^{\ell-1}$ sets, such that all rectangles in one set intersect one vertical line, while every pair of rectangles from two different sets are disjoint (see, e.g., [1]). We will denote by $\mathcal{L}_\ell$ the set of these $2^{\ell-1}$ vertical lines at level ℓ. To make this decomposition independent of the rectangles themselves, which is needed for monotonicity, we give the decomposition explicitly. We define $\mathcal{L}_i$ to be the set of vertical lines $x = n/2^i, x = 3 \cdot n/2^i, \ldots, x = (2^i - 1) \cdot n/2^i$. Clearly, we get $k = \log n$ levels. We assign each rectangle to the smallest level such that one of its lines intersects the rectangle. Note that each rectangle is intersected by exactly one vertical line of its level.

Given a set of rectangles $\{r_1, \ldots, r_m\}$, we iterate the following two steps, for $\ell \in [k]$: First, we project every rectangle whose level is at least ℓ onto the nearest vertical line at level ℓ. This gives a set of intervals on each vertical line in $\mathcal{L}_\ell$. The bid of a bidder j on a given projection is the same as on the original rectangle. Second, we apply a monotone DP for the maximum weight-independent set problem on the set of intervals on each vertical line in $\mathcal{L}_\ell$. The resulting independent set of intervals corresponds to a disjoint set of rectangles $\mathcal{R}_\ell$. At the end we output the set of rectangles $\mathcal{R}_\ell$ that achieves the highest total bid, i.e., maximizes $\sum_{r\in\mathcal{R}_\ell} b_j(r)$. In case of ties, we choose the $\mathcal{R}_\ell$ at the *lowest* level. It is straightforward to verify that this algorithm is a special case of the general framework described in Section 3.1, so monotonicity follows from Lemma 2. Finally each winner is charged the critical payment of the projection that made him a winner.

Lemma 5. *The above procedure for rectangles is monotone, gives an approximation guarantee of* $\log n$ *and has a running time of* $O(m\log(mn))$*.*

4 The Non-Single-Minded Case

In this section, we show that in the non-single-minded case, we get a truthful-in-expectation mechanism with approximation guarantee $O(1)$ (resp., $O(\log m)$) for fat objects (resp., axis-aligned rectangles.) The idea is to use Theorem 3. For that, we show that the integrality gap of the LP (P) is $O(1)$ for fat objects and $O(\log m)$ for rectangles (and there is an algorithm verifying this).

We give a randomized algorithm that with high probability returns an integral solution $\tilde{x}$ such that $b^T\tilde{x} = \Omega(b^Tx)$, where b denotes the (column) vector of bids and x denotes the optimal fractional solution This algorithm can be derandomized using the work of [16]. Combining the results yields Theorem 2.

The fat ranges case. Assume all ranges are β-fat. We apply randomized rounding with alteration (see, e.g., [4]). Let $(x_{j,r}:\ j\in[m], r\in\mathcal{R})$ be any feasible solution for (P). Let $\gamma\in(0,1)$ be a constant to be specified later. We define the rounded solution $\tilde{x}$ by its *winner set* W, obtained by the following procedure. First, for every bidder j, we choose a range $r=r_j$ with probability $\gamma x_{j,r}$ if r is non-empty and with probability $1-\gamma(1-x_{j,r})$ if $r=\emptyset$. Second, let $W=\emptyset$, and $r_1,r_2,\ldots,r_m$ be the ranges selected in Step 1 in *non-increasing* order of size. For $j=1,\ldots,m$, if r_j does not intersect any range r_i with $i\in W$, add j to W.

By construction, the set $(r_j\ :\ j\in W)$ is a valid allocation. It remains to prove the approximation guarantee of $O(1)$.

Lemma 6. $\mathbb{E}[\sum_{j\in W} b_{j,r_j}] \geq \frac{1}{4\beta}\sum_{j\in[m],r\in\mathcal{R}} b_{j,r}x_{j,r}$.

The rectangles case. We use a similar technique as in the previous section. However, since rectangles can have unbounded fatness, we refine the previous technique by using the levelwise decomposition described in Section 3.3. Let $(x_{j,r}:\ j\in[m], r\in\mathcal{R})$ be any feasible solution for (P), and $\gamma=\frac{1}{2}$. We define the rounded solution $\tilde{x}$ by its winner set W, obtained by the following procedure.

1. For every bidder j, we choose a rectangle $r=r_j$ with probability $\gamma x_{j,r}$ if r is non-empty and with probability $1-\gamma(1-x_{j,r})$ if $r=\emptyset$.
2. Let $r_1,r_2,\ldots,r_m$ be the rectangles selected in Step 1, ordered from top to bottom (breaking ties arbitrarily).
3. Consider a levelwise decomposition of these rectangles into $\log m$ levels. Pick a level $i\in\{1,2,\ldots,\log m\}$ at random. Let S be the set of rectangles in level i.

4. Let $W = \emptyset$. For $j \in S$, if r_j does not intersect any r_i, $i \in W$, add j to W.

Lemma 7. $\mathbb{E}[\sum_{j \in W} b_{j,r_j}] \geq \frac{1}{4 \log m} \sum_{j \in [m], r \in \mathcal{R}} b_{j,r} x_{j,r}$.

Acknowledgments. We thank Rene Sitters for suggesting the application, and Rajiv Raman and Rene Sitters for helpful discussions.

References

[1] Agarwal, P.K., van Kreveld, M.J., Suri, S.: Label placement by maximum independent set in rectangles. Comput. Geom. 11(3-4), 209–218 (1998)
[2] Babaioff, M., Blumrosen, L.: Computationally-feasible truthful auctions for convex bundles. GEB 63(2), 588–620 (2008)
[3] de Berg, M., Gudmundsson, J., Katz, M., Levcopoulos, C., Overmars, M., van der Stappen, A.F.: TSP with Neighborhoods of varying size. J. of Algorithms 57, 22–36 (2005)
[4] Chakrabarti, A., Chekuri, C., Gupta, A., Kumar, A.: Approximation algorithms for the unsplittable flow problem. Algorithmica 47(1), 53–78 (2007)
[5] Chalermsook, P., Chuzhoy, J.: Maximum independent set of rectangles. In: SODA, pp. 892–901 (2009)
[6] Chan, T.M.: Polynomial-time approximation schemes for packing and piercing fat objects. J. Algorithms 46(2), 178–189 (2003)
[7] Clarke, E.H.: Multipart pricing of public goods. Public Choice, 17–33 (1971)
[8] Fowler, R.J., Paterson, M.S., Tanimoto, S.L.: Optimal packing and covering in the plane are NP-complete. IPL 12, 133–137 (1981)
[9] Groves, T.: Incentives in teams. Econometrica, 617–631 (1973)
[10] Hoefer, M., Kesselheim, T., Vöcking, B.: Approximation algorithms for secondary spectrum auctions. CoRR abs/1007.5032 (2010)
[11] Imai, H., Asano, T.: Finding the connected components and a maximum clique of an intersection graph of rectangles in the plane. J. Algorithms 4, 310–323 (1983)
[12] Khanna, S., Muthukrishnan, S., Paterson, M.: On approximating rectangle tiling and packing. In: SODA, pp. 384–393 (1998)
[13] Lavi, R., Swamy, C.: Truthful and near-optimal mechanism design via linear programming. In: FOCS, pp. 595–604 (2005)
[14] Lehmann, D., O'Callaghan, L.I., Shoham, Y.: Truth revelation in approximately efficient combinatorial auctions. JACM 49(5), 577–602 (2002)
[15] Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, Cambridge (2007)
[16] Sivakumar, D.: Algorithmic derandomization via complexity theory. In: STOC, pp. 619–626 (2002)
[17] Stappen, A.: Motion Planning amidst Fat Obstacles. Ph.D. dissertation, Utrecht University, the Netherlands (1994)
[18] Vickrey, W.: Counterspeculation, auctions and competitive sealed tenders. Journal of Finance, 8–37 (1961)

A Truthful Constant Approximation for Maximizing the Minimum Load on Related Machines

George Christodoulou[1,3], Annamária Kovács[2,⋆], and Rob van Stee[3,⋆⋆]

[1] Cluster of Excellence "Multimodal Computing and Interaction", Saarland University, Saarbrücken, Germany
gchristo@mpi-inf.mpg.de
[2] Department of Informatics, Goethe University, Frankfurt am Main, Germany
panni@cs.uni-frankfurt.de
[3] Max Planck Institute for Informatics, Saarbrücken, Germany
vanstee@mpi-inf.mpg.de

Abstract. Designing truthful mechanisms for scheduling on related machines is a very important problem in single-parameter mechanism design. We consider the covering objective, that is we are interested in maximizing the minimum completion time of a machine. This problem falls into the class of problems where the optimal allocation can be truthfully implemented. A major open issue for this class is whether truthfulness affects the polynomial-time implementation.

We provide the first constant factor approximation for deterministic truthful mechanisms. In particular we come up with a $2 + \varepsilon$ approximation guarantee, significantly improving on the previous upper bound of $\min(m, (2 + \varepsilon)s_m/s_1)$.

1 Introduction

Algorithmic Mechanism Design studies scenarios where there is an optimization problem at hand, but selfish agents control some input parameters. These parameters are unknown to the optimizer and are *private* values of the agents. Moreover, the agents might be only interested in satisfying their own interests and therefore they might have incentive to misreport their values, if this can lead to an output or solution that they prefer. In order to elicit the missing information, the mechanism design approach uses side payments to motivate the agents to reveal their true values. Roughly speaking, a *mechanism* consists of two components: an algorithm that takes as input the reported values, and returns a solution of the optimization problem, and a payment algorithm that hands out payments to the agents. Each agent's goal is to maximize her utility, that is the payment she gets minus her actual value on the solution. A mechanism is *truthful* if it is in the best interest of each agent to report truthfully.

Given a class of problems, the challenge is to characterize the objective functions that one can truthfully optimize/approximate. Under this framework, scheduling is a very natural and well-studied setting to explore the boundaries of truthful implementation. On the one hand, the algorithmic techniques that have been developed are very

⋆ Research supported by the German Research Foundation (DFG), project KO 4083/1-1.
⋆⋆ Research supported by the German Research Foundation (DFG), project STE 1727/3-2.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 182–193, 2010.

broad, and the question is to what extent those techniques can be applied to the design of truthful mechanisms. On the other hand scheduling is conceptually similar to a combinatorial auction, a setting that is very important in economics, and therefore insights can be transfered from one problem to the other. In a scheduling setting, there are m machines and n tasks, and each machine is controlled by an agent that has as private values the processing times it needs to execute the tasks. The algorithmic goal is to allocate the jobs to the machines so that some objective (most commonly the makespan) is optimized. In the unrelated machines setup the processing times for each machine are expressed via a vector of size m, while for the related machines setup they are expressed via a single parameter, the speed of the machine.

A natural question that arises in many single-parameter settings is: what is the approximability of *polynomial-time* truthful mechanisms? Taking a problem that one can solve exactly with a truthful mechanism, can one also achieve the best possible approximation guarantee, or does truthfulness have a negative computational impact? Is the class of polynomial-time truthful mechanisms less powerful with respect to approximation, compared to the class of polynomial-time non-truthful algorithms? For makespan minimization such a separation does not exist. Dhangwatnotai et al. [8] showed a randomized truthful-in-expectation PTAS and later Christodoulou and Kovács [7] showed a deterministic truthful PTAS that is the best one can achieve even with non-truthful approximation algorithms [13,10].

In order to explore further the performance of truthful mechanisms in single-parameter problems, we focus on the covering objective for scheduling on related machines, that is we are interested in maximizing the minimum completion time over all machines. This objective is important in settings where a system is only alive if all of its components are alive. One can think of the jobs as batteries with varying capacities, or hard-drives of various sizes that we want to use as a backup medium [16]. The covering problem is also closely related to the max-min fairness problem, where we want to distribute indivisible goods to players so as to maximize the minimum valuation.

Mu'alem and Schapira [14] showed that maximizing the minimum load for unrelated machines cannot be approximated within a constant factor by a deterministic truthful mechanism.[1] On the other hand, using the arguments in [1], one can show that for related machines the optimal allocation is truthful, although not efficient. For the non-strategic version, Epstein and Sgall showed a PTAS [10].

The question we address in this paper is: 'What is the best deterministic, polynomial-time, truthful approximation mechanism that one can design for the covering problem?' We provide the first deterministic truthful mechanism with constant approximation for the covering objective. In particular, we obtain an approximation guarantee of $2 + \varepsilon$.

Related work. The non-strategic version of the problem has been extensively studied in the past in various contexts for online and approximations algorithms. For identical machines, Woeginger [17] designed a polynomial time approximation scheme (PTAS) and gave tight results for deterministic online algorithms. Azar and Epstein [3]

[1] In fact the authors showed this for the combinatorials auctions setup where the agents are utility maximizers, while in the scheduling setting the agents are cost minimizers. However, a simple modification of their argument works for scheduling as well.

studied the randomized online setting. Furthermore, for the case where jobs arrive in non-increasing order and also the optimal value is known in advance, they gave a deterministic 2-competitive online algorithm NEXT COVER.

In [4], a PTAS was designed for related machines, and later this was generalized to capture a large class of objective functions in [10]. Epstein and van Stee [11] provide a PTAS and also an FPTAS for constant number of related machines which they then use as a subroutine for a truthful FPTAS, while Efraimidis and Spirakis [9] show an FPTAS for the more general case of unrelated machines. Dhangwatnotai et al. [8] provide a randomized truthful-in-expectation PTAS. Epstein and van Stee [11] also give a monotone approximation algorithm with approximation ratio $\min(m, (2+\varepsilon)s_m/s_1)$ where $\varepsilon > 0$ can be chosen arbitrarily small and s_i is the (real) speed of machine i.

The max-min fairness problem has been studied intensively in recent years, see for instance [2,5,6,12] and references therein.

Our results and techniques. For any positive $\varepsilon < 1/5$, we show a $(2+5\,\varepsilon)$-approximation, *monotone* algorithm for the covering problem. Monotonicity of a scheduling algorithm means that whenever a single machine (agent) increases his reported speed (asuming that the other speeds are unchanged), the machine receives not less total job-size, than with the original speed. As known from the classic work of Myerson [15], and completed with the payment scheme given there, this yields a truthful mechanism, that is computable in polynomial running time for constant ε. With this result we significantly improve upon the previous best approximation ratio of $\min(m, (2+\varepsilon)s_m/s_1)$ given in [11].

As a standard technique applied in all approximation schemes for related scheduling [13,10,8,7], we define a directed acyclic graph with vertices representing possible job-sets allocated to single machines. Relative to the total size of any given set, we distinguish *normal* and *tiny* jobs in the set. We consider a special form of schedules, where the whole sequence of machines is partitioned into *segments*, each segment having either sets of (nearly) only normal jobs, or sets of only tiny jobs. The allocation of jobs *within* segments must adhere to strict regulations, which allow for both good approximation and polynomial-time optimization.

We could exploit some of the ideas used for the monotone PTAS for related machine scheduling (with the makespan objective) [7], while defining an essentially different type of allocation. We point out that the current result is not a straightforward adaptation of [7]. In fact, we were unable to find such an adaptation for maximizing the cover: although in many aspects the setting is symmetric to that of makespan minimization, this symmetry breaks when handling the tiny jobs.[2] On the positive side, striving for the weaker approximation ratio admits a very simple and technically less demanding construction than in [7].

[2] The difficulty lies roughly in the fact that in case of makespan minimization a machine that becomes bottleneck loses (many or all of) its tiny jobs, while in case of maximizing the cover, a bottleneck machine might collect all tiny jobs from faster machines. This makes exact optimization with our methods impossible, since the *exact* workload of a set of tiny jobs is not known.

2 Preliminaries

The input is given by a set P_I of n input jobs, and a vector σ of input speeds $\sigma_1 \leq \ldots \leq \sigma_m$. We round up every input speed to the nearest integral power of $1+\varepsilon$. Denoting the respective rounded speeds by s_i, we have $s_1 \leq \ldots \leq s_m$. We use the interval notation (e.g., $[1, m]$) for a set of consecutive machine indices. The letters p or q are used to denote jobs, as well as the respective job sizes in a given formula. We fix a nondecreasing order $p_1 \leq p_2 \leq \ldots \leq p_n$ of all input jobs. If $Q = \{q_1, q_2, \ldots, q_j\}$ is an arbitrary job set, then the *weight* or *workload* of Q is $|Q| = \sum_{r=1}^{j} q_r$. An allocation of the jobs to the machines is an *(ordered) partition* $(P_1, P_2, \ldots, P_m)$ of the jobs into m *sets*. We search for an output where the workloads $|P_i|$ are in non-decreasing order. We assume w.l.o.g. that $n \geq m$, since otherwise the cover is trivial. We are only interested in allocations where $P_i \neq \emptyset$ for $i \in [1, m]$ (otherwise the approximation ratio is ∞).

Our graph-algorithm outputs a schedule of optimum cover over a restricted type of job partitions. We name these partitions *segmented partitions*, because the output partition can be subdivided into *segments*, each consisting of consecutive job sets of the partition. Every job allocated in some earlier segment precedes all jobs allocated in any later segment, with respect to the fixed job order. The allocation of jobs within a partition segment will have to adhere to one of two forms: *smooth allocation*, or *canonical allocation*.

A *smooth allocation* of a set of consecutive jobs $P = \{p_j, p_{j+1}, \ldots, p_k\}$ into r sets is the partition segment output by the following *smoothing procedure*: We construct a fractional allocation into r sets of equal workloads of size $|P|/r$ (we assume $p_k \ll |P|/r$). We start with the smallest job p_j, add jobs in the fixed order, and cut a job into two whenever the total workload reached $|P|/r$. We continue with the next set, and the remaining part of the divided job, and so on. Next, we allocate each job that was cut into two, to the first one of its two sets. Finally, we order the job partition in increasing order of workloads.

Before we turn to *canonical allocations*, we need to fix the constants δ and ρ, and classify the input job sizes accordingly. For a desired approximation bound of $2 + 5\varepsilon$, we choose a $\delta \ll \varepsilon$.[3] For ease of exposition, we will assume that $(1+\delta)^t = 2$ for some $t \in \mathbb{N}$. Furthermore, we define ρ as the unique integer power of 2 in $(\delta/8, \delta/4]$.

Definition 1. *If p denotes (the size of) a job, then $\overline{p}$ denotes this job rounded up to the nearest integral power of* $(1+\delta)$. *A job p is in the* job class C_l, *iff* $\overline{p} = (1+\delta)^l$.

A *canonical allocation* within a segment means that the sets have non-decreasing workloads, moreover that jobs that belong to the same job class appear in increasing order over the sets of the segment. Given an arbitrary partition $(Q_1, Q_2, \ldots, Q_r)$ of some *subset* $Q \subseteq P_I$ of the jobs, the *canonization procedure* constructs a partition $(Q'_1, Q'_2, \ldots, Q'_r)$ of Q with canonical allocation.[4] In effect, this procedure

(A) permutes jobs *within job classes*, and thus *perturbs* each set Q_i, so that the *perturbed set* $\widetilde{Q}_i$ has a workload in $[|Q_i|/(1+\delta), |Q_i|(1+\delta)]$; and then

(B) sorts the perturbed $\widetilde{Q}_i$ sets by increasing workloads to obtain $(Q'_1, \ldots, Q'_r)$.

[3] $12\delta < \varepsilon$ suffices.

[4] The procedure appeared in the full version of [7].

3 Segmented Partitions

In the following we introduce *magnitudes*, and make the definition of segmented partitions exact. As the main result of the section, we show in Theorem 1 that for arbitrary input with rounded speeds, a segmented partition of cover within a factor of $\frac{1-\varepsilon}{2}$ of the optimum exists. As done previously in [10,8,7], we associate a *magnitude* w_i, an integer power of 2, with each set P_i in the partition. The set with the associated magnitude will also be denoted by (P_i, w_i). Magnitudes are used to focus on the relevant job sizes when representing job sets with the help of integer arrays. We will require $w_i/5 < |P_i| \leq w_i$ $(i \in [1, m])$, and $w_1 \leq w_2 \leq \ldots \leq w_m$.

Definition 2. *A job p is* tiny wrt. magnitude w_i, *if $p \leq \rho \cdot w_i$. A jobset (P_i, w_i) is sandy, if all jobs in P_i are tiny wrt. w_i. A jobset (P_i, w_i) is* normal*, if it has at least one non-tiny job, and a (possibly empty) consecutive sequence of the largest tiny jobs wrt. w_i.*

Note that the property of being a tiny job for some magnitude either holds for a whole class of jobs or for none of them, as $\rho \cdot w_i$ is an integer power of 2. Next we define the two allowed sorts of partition segments: one for sandy sets, and another one for normal sets.

Definition 3. *A* sandy segment *consists of sandy sets of equal magnitude (P_i, w), $(P_{i+1}, w), \ldots, (P_h, w)$ with a smooth allocation of consecutive jobs (of size at most $\rho \cdot w$).*

Definition 4. *A* normal segment *of a partition consists of normal sets of nondecreasing magnitudes (P_i, w_i), $(P_{i+1}, w_{i+1}), \ldots, (P_h, w_h)$, with a canonical job allocation. The union of the sets contains consecutive jobs of the ordered input.*

Definition 5. *A* segmented partition *is a partition of the input jobs into m jobsets P_i of non-decreasing workloads $|P_i|$. It is subdivided into partition segments each of which is either sandy or normal; if job p precedes job q in the fixed ordering, then p belongs to the same segment as q, or to an earlier segment.*

Theorem 1. *Let $0 < \varepsilon < 1/5$ and $\delta \ll \varepsilon$ be fixed. Given an arbitrary set of n input jobs in a fixed non-decreasing order, and m non-decreasing input speeds that are integral powers of $(1 + \varepsilon)$, there exists a segmented partition having a cover of at least $(1 - \varepsilon)\frac{\text{OPT}}{2}$, where* OPT *means the optimum cover.*

Proof. Let both the machine speeds and the job sizes be indexed in non-decreasing order. It is easy to show (see the full paper for details) that the greedy procedure that takes the (integral) jobs in the given order, and fills each machine until it has finish time at least $\text{OPT}/2$, will not run out of jobs before the last machine is filled. Obviously, the same holds, if each machine i is filled up to some given finish time $f_i \leq \text{OPT}/2$.

Now we construct the segmented partition for the given input. We start by a greedy integral allocation which is shown in Figure 1. Note that the pre-magnitudes defined in Step 1 are increasing in i. Observe that those machines that were filled up to $C = \text{OPT}/2$ have normal jobsets (call them normal machines), and those filled only to $(1 - \varepsilon/2) \cdot C$ have sandy sets (call them sandy machines), with respect to the pre-magnitudes w_i'. The

Input: job set P_I, machine speeds $s_1 \leq \cdots \leq s_m$, optimal cover OPT.

1. Let $C = \text{OPT}/2$. For machine i of speed s_i let the *pre-magnitude* w'_i (a power of 2) be uniquely defined so that $2s_iC \leq w'_i < 4s_iC$.
2. Allocate jobs in the fixed increasing order as follows.
   For $i = 1$ to $m - 1$ do
   (a) Assign jobs to the current machine until the finish time is at least $(1 - \varepsilon/2)C$.
   (b) If the last (largest) job on the machine has size more than $\rho w'_i$, continue assigning jobs until the finish time is at least C.
3. Assign the remaining jobs to machine m.

Output: job assignment $Q_1, \ldots, Q_m$ with cover of at least $(1 - \varepsilon/2)\text{OPT}/2$.

Fig. 1. Greedy integral allocation procedure

only possible exception to this is machine m, which is always filled up to at least C, but might contain a sandy set (in this case it is a sandy machine).

Within machines of equal speed, zero or more sandy machines are followed by zero or more normal machines, since such machines have the same pre-magnitude. Each such sequence of sandy machines (i.e., of the same machine speed) will be a sandy segment. The remaining maximal sequences of normal machines, possibly spanning over different machine speeds will be the normal segments. Next, we redistribute the jobs *within* each segment in order to fulfill the conditions of Definitions 3 and 4 and prove the approximation bound of $(1 - \varepsilon) \cdot C = (1 - \varepsilon) \cdot \frac{\text{OPT}}{2}$. The sets of this final allocation will be denoted by $P_1, P_2, \ldots, P_m$.

Sandy segments. Consider first a segment consisting of all the sandy machines of the same speed s and having the same pre-magnitude $w' < 4Cs$. Using $\rho w' < 4\rho Cs \leq \delta Cs$, such machines have a workload of at least $(1 - \varepsilon/2) \cdot C \cdot s$ and at most $(1 - \varepsilon/2 + \delta) \cdot C \cdot s$. We now apply the smoothing procedure (see Section 2) to these machines and the jobs assigned to them. Then for each machine i of the segment we have

$$(1 - \varepsilon/2 - \delta) \cdot C \cdot s \leq |P_i| \leq (1 - \varepsilon/2 + 2\delta) \cdot C \cdot s. \tag{1}$$

The obtained partition on the segment adheres to Definition 3, and the cover is higher than $(1 - \varepsilon) \cdot C$; the pre-magnitudes can remain the valid magnitudes w_i of the jobsets.

An exceptional case occurs when machine m is (sandy and) is part of the segment. In this case the upper bound in (1) might fail, and the common magnitude of the segment needs to be increased acccordingly. However, then the segment is the very last one, and Claim 3 and the theorem still holds.

Normal segments. Consider now an arbitrary normal segment $(Q_s, \ldots, Q_t)$. We create a canonical allocation by running the canonization procedure (see the last lines of 185). Since sorting cannot decrease the cover [10], the cover remains above $C/(1 + \delta) > (1 - \varepsilon)C$. We also need to find proper magnitudes for machines in the normal segment. Before doing this, we conclude the main line of the proof by showing that the workloads $|P_i|$ are increasing in i. All other conditions of Definition 5 hold by construction.

Claim. The workloads $|P_i|$ are increasing in i.

Proof. Clearly, the workloads within each segment are increasing, since the segments have either a canonical or a smooth allocation. Next we show that they are increasing over the whole schedule. First, we compare a normal set P_i with a sandy set P_j of a *preceding* sandy segment. Assume that $P_i = \widetilde{Q}_{i'}$, for some $s_j \le s_{i'}$, where i' and i are in the same normal segment. We saw in (1) that for the sandy set $|P_j| \le (1 - \varepsilon/2 + 2\delta) \cdot C \cdot s_j$, whereas for the normal set $\frac{1}{(1+\delta)} \cdot C \cdot s_{i'} \le \frac{1}{(1+\delta)} \cdot |Q_{i'}| \le |\widetilde{Q}_{i'}| = |P_i|$. Using $1 - \varepsilon/2 + 2\delta < 1 - \delta < \frac{1}{1+\delta}$, this proves $|P_j| < |P_i|$.

Assume now that P_i is a sandy set, and P_j is either a jobset in a preceding sandy segment, or the perturbed $\widetilde{Q}_{j'}$ set of a preceding normal segment. Let $s = s_j$ in the first, and $s = s_{j'}$ in the second case, respectively. In both cases, by construction $s \le s_i/(1+\varepsilon)$. Furthermore, all the jobs in P_i, and in sets of previous segments have size of at most $\rho w_i \le \delta C s_i$, which implies the bound

$$|P_j| \le (C \cdot s + \delta C s_i)(1+\delta) \le C \cdot s_i \cdot \left(\frac{1}{1+\varepsilon} + \delta\right)(1+\delta)$$

for both cases. Using the lower bound for $|P_i|$ from (1) and the fact that $(\frac{1}{1+\varepsilon} + \delta)(1 + \delta) < 1 - \varepsilon/2 - \delta$ for $\delta < \varepsilon/12$, we obtain $|P_j| < |P_i|$. □

Finally, we define magnitudes w_i. Fix a normal segment, and let w'_0 be the smallest pre-magnitude in this segment. For each set P_i of the segment, we define the magnitude as $w_i = \max\{w'_0, 2^{\lceil \log |P_i| \rceil}\}$. For these magnitudes $w_i/5 < |P_i| \le w_i$ holds.

Claim. The magnitudes are increasing over the whole schedule, and the (P_j, w_j) are normal sets.

Proof. The magnitudes are increasing within the segment because the workloads are increasing. Furthermore, if the magnitude w_j of some set $P_j = \widetilde{Q}_i$ is larger than the pre-magnitude w'_i of Q_i, then $2s_iC \le w'_i \le w_j/2 < |\widetilde{Q}_i| < (1+\delta)|Q_i|$, whereas $|Q_i|$ had to reach (only) a workload of s_iC. Thus, the last job in Q_i is at least as big as (roughly) the sum of all other jobs in Q_i, and in particular for $\delta < 1/5$ we obtain:

$(*)$ If $w_j > w'_i$ then Q_i contains a job of size at least $\frac{w_i}{5(1+\delta)}$.

Since in the subsequent *sandy* segment (of magnitude w) this jobsize is tiny by definition, we have $w_i/[5(1+\delta)] < \rho w$. Therefore, the magnitudes are increasing over the whole partition.

We show that the (P_j, w_j) are normal sets. Let $P_j = \widetilde{Q}_i$. Recall that (Q_i, w'_i) is normal by the definition of normal machines. With respect to the new magnitude w_j, there is at least one normal job in the set. This is clear if $w_j \le w'_i$, and follows from $(*)$ if $w_j > w'_i$. Assume now that the machine also contains tiny jobs with respect to w_j. Note that since the jobs are consecutive (disregarding perturbation) in each set, every other set has either only tiny jobs or only normal jobs with respect to w_j. However, any set in the same normal segment that has only tiny jobs with respect to w_j, must have a magnitude less than w_j (since each set does have a normal job for its own magnitude) and so (since magnitudes are increasing) it is a set P_k for some $k < j$. By the definition of canonical allocations and normal sets, this proves the normality of P_j. □

This proves the existence of a segmented partition with cover at least $(1-\varepsilon)\frac{\text{OPT}}{2}$. □

4 Graph Construction

In this section we construct a directed acyclic graph, depending on the set of input jobs P_I. The vertices represent either *normal jobsets*, or *sandy partition segments*. An arc between two vertices should indicate that the corresponding sets or segments can be neighbors in the partition (e.g., that some normal set P_i can be followed by a certain sandy segment $(P_{i+1},\ldots,P_k)$). A given input speed vector $(s_1,\ldots,s_m)$, determines a weight on each graph vertex, meaning the (minimum) finish time induced by the workload(s) $|P_i|$. A path, leading over some $P_1,P_2,\ldots,P_m$, that maximizes the minimum weight over its vertices, represents an optimal solution among all segmented partitions.

The above outlined technique was introduced by Hochbaum and Shmoys [13] for a PTAS for related scheduling, and has been used for (monotone) approximation schemes for related scheduling [10,4,8,7]. Based on this previous work, our graph construction (adapted for segmented partitions) is straightforward. As a difference to all of the known PTAS algorithms, the notion of segmented partitions allows for a pure and *exact* representation of the jobsets, and a very plain graph structure. Of course, we pay for this simplification with a loss of a factor 2 in the approximation ratio.

Set configurations. *Set configurations* are used to represent normal jobsets. Each set configuration α is a triple $\alpha=(w,\boldsymbol{n}^o,\boldsymbol{n}^1)$, where w is the magnitude of the set, and the vectors $\boldsymbol{n}$ are *size vectors*. If the configuration is supposed to define the set P_i, this is done by the two size vectors defining the cumulative jobsets $\bigcup_{k=1}^{i-1}P_k$, and $\bigcup_{k=1}^{i}P_k$, respectively. Thus, size vectors represent sets of jobs of size between ρw and w (in fact, a prefix set of each job class), and a prefix subset of the tiny jobs of size at most ρw. They are indexed by the integers from $\lambda=\log\rho w$ to $\Lambda=\log w$, and have nonnegative integer coordinates. The entry n_l for some $l\in(\lambda,\Lambda]$ means that the cumulative jobset contains exactly the first n_l jobs of the class C_l. Observe that this is an adequate representation of canonical allocations, where jobs within each class appear in the fixed order. Finally, the coordinate n_λ stands for some *prefix subset* $\{p_1,p_2,\ldots,p_{n_\lambda}\}$ of all the jobs of size at most ρw. We can speak of a *valid set configuration* only if a handful of conditions are fulfilled. For instance, by the definition of normal sets it is required that either $\boldsymbol{n}^o_\lambda=\boldsymbol{n}^1_\lambda$ (no tiny jobs), or that $\boldsymbol{n}^1_\lambda$ is the number of jobs of size at most ρw (the largest tiny jobs are all in the set). Also, $w/5<|P|\le w$ must hold, and can be easily checked. The rest, like $\boldsymbol{n}^o\le\boldsymbol{n}^1$, and other bounds on the coordinates, are straightforward, and will not be detailed here. For an illustration see Figure 2.

We bound the number of different valid set configurations. Every size vector has

$$\log_{1+\delta}w-\log_{1+\delta}\rho w=\log_{1+\delta}1/\rho=\mathcal{O}(1/\delta\cdot\log(1/\rho))$$

integer coordinates between 0 and n. Each possible pair of size vectors determines a set, which has at most 3 possible valid magnitudes. Therefore, for constant δ there is a polynomial number of different set configurations.

Segment configurations. A *segment configuration* $\beta=(w,r,n^o,n^1)$ stands for a sandy segment, and has altogether four positive integer entries. This tuple defines a smooth allocation of the jobs $\{p_{n^o+1},p_{n^o+2},\ldots,p_{n^1}\}$ into r sets of magnitude w. Notice that the jobs are distributed evenly over the segment, *regardless* of the machine speeds. Moreover, at any point of the calculation, the sets of the segment $P_i,P_{i+1},\ldots,P_{i+r-1}$

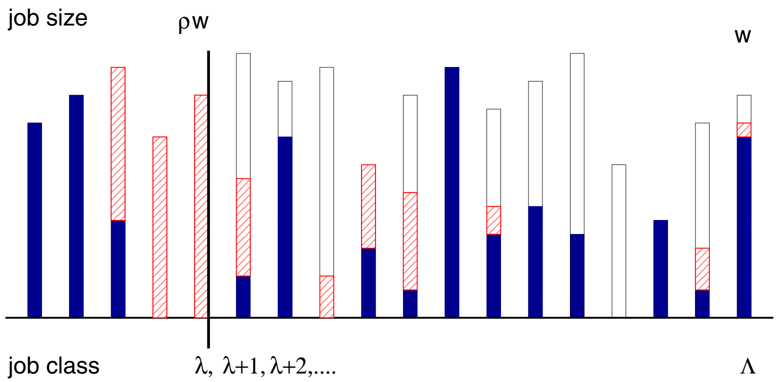


Fig. 2. A set configuration: the thin rectangles represent job classes; the solid part belongs to the set $\bigcup_{k=1}^{i-1} P_k$, and the striped part to set P_i. Note that P_i has a contiguous set of the largest tiny jobs. The sets P_i which we construct in Theorem 1 (see Figure 1) always contain jobs of *consecutive* classes, where each class except the first and last class is completely contained in P_i.

(of increasing workloads) can easily be computed. In order to have a valid configuration, the conditions $p_{n^1} \leq \rho w$, and $w/5 < |P| \leq w$ (for each set P) must hold. The number of different valid segment configurations is bounded by $3mn^2$.

The directed graph $\mathcal{G}(V, A)$. The vertex set V of $\mathcal{G}$ has $m + 2$ layers. Each layer $i \in [1, m]$ contains a vertex (i, α) for every valid set configuration α, and a vertex (i, β) for every valid segment configuration $\beta = (w, r, n^o, n^1)$ for which $i + r \leq m + 1$. Recall that for any configuration on level i, the entry $\boldsymbol{n}^o$ (or n^o) uniquely determines the cumulative jobset $\bigcup_{k=1}^{i-1} P_k$. Similarly, the entry $\boldsymbol{n}^1$ (resp. n^1), encodes the cumulative set $\bigcup_{k=1}^{i} P_k$ (resp. $\bigcup_{k=1}^{i+r-1} P_k$)). We add a source vertex s on layer 0, that stands for the empty jobset (say, with $n^1 = 0$), and a sink vertex t on layer $m+1$ for the complete jobset (say, with $n^o = n$).

Next we define the arc set A. There is an arc between two configurations if and only if they satisfy all of the following conditions. From a set configuration (i, α) all arcs lead into (set or segment) configurations of layer $i + 1$. From a segment configuration (i, β), all arcs lead into (set or segment) configurations of layer $i + r$. Obviously, a necessary condition for an arc between two configurations is that the $\boldsymbol{n}^1$ or n^1 entry of the first one should represent the same jobset as the $\boldsymbol{n}^o$ or n^o entry of the second one. Finally, it is required that the magnitudes w are nondecreasing along every arc, and similarly, the workloads of the represented sets must be nondecreasing (here for set configurations $|P_i|$ is meant, and for segment configurations we consider $|P_i|$ for incoming arcs, and $|P_{i+r-1}|$ for outgoing arcs).

The first two arc conditions ensure that any (s, t)-path of the graph induces a *partition* of the input jobs into m sets. Due to the fact that here the graph vertices represent jobsets *exactly*, (as opposed to different rounding techniques applied in previous work), the following statement is straightforward:

Proposition 1. *There is a one-to-one correspondence between segmented partitions of the input jobs P_I into m sets, and the directed (s, t)-paths in graph $\mathcal{G}$.*

Finish times. Note that both segmented partitions and the graph $\mathcal{G}(V, A)$ were defined independently of the speed vector. Now for given (rounded) input speeds $s_1 \leq s_2 \leq \ldots \leq s_m$, we can assign a *finish time* $f(v)$ (a weight) to every vertex $v \in V$ of the graph. For a vertex $v = (i, \alpha)$ with a set configuration α representing set P_i, the finish time is $f(v) = \frac{|P_i|}{s_i}$. If $v = (i, \beta)$ with induced jobsets $P_i, \ldots, P_{i+r-1}$, the (minimum) finish time is defined as

$$f(v) = \min \left\{ \frac{|P_k|}{s_k} \mid i \leq k < i + r \right\}.$$

5 Monotone Algorithm for Covering

Once the graph $\mathcal{G}$ is constructed, the problem boils down to finding an (s, t)-path of maximum cover. This can be done by a standard dynamic programming algorithm which we call MAXPATH. Because tie-breaking rules are crucial for monotonicity, we fix an arbitrary (e.g., lexicographical) linear order $\prec$ over all valid (set and segment) configurations.

The monotone algorithm MONCOVER is presented in Figure 3. Since for constant ε the number of different configurations is polynomial in n and m, the size of $\mathcal{G}$ is polynomial, and the algorithm runs in polynomial time. Let OPT_σ and OPT_s denote the optimal cover values with the original and the rounded speeds, respectively. Clearly, we have $\text{OPT}_\sigma/(1+\varepsilon) \leq \text{OPT}_s$, since this ratio holds for the cover of every fixed allocation. Moreover, by Theorem 1 and Proposition 1, the output of MONCOVER has a cover of at least $\text{OPT}_s(1-\varepsilon)/2$. Altogether we obtain that the cover of the output is at least

$$\frac{\text{OPT}_\sigma}{2} \cdot \frac{(1-\varepsilon)}{(1+\varepsilon)} \geq \frac{\text{OPT}_\sigma}{2+5\varepsilon}.$$

Theorem 2. *Algorithm* MONCOVER *is monotone.*

Proof. The proof is analogous to part of the monotonicity proof in [7]. With given input speed vector $\sigma_1, \sigma_2, \ldots, \sigma_m$, let the output of MONCOVER be $P_1, P_2, \ldots, P_m$. We assume that for some machine $i \in [1, m]$, the speed σ_i is increased to $\sigma' > \sigma_i$, the

Input: job set P_I, machine speeds $\sigma_1 \leq \cdots \leq \sigma_m$, and $\varepsilon \in (0, 1/5)$.

1. Fix an appropriate $\delta < \varepsilon/12$, fix a nondecreasing order of the jobs, determine the job classes C_l, and construct the graph $\mathcal{G}(V, A)$.
2. Round up each speed σ_i to s_i, the nearest integral power of $(1 + \varepsilon)$.
3. Using the rounded speeds, compute the finish time $f(v)$ of every graph vertex $v \in V$.
4. Compute the optimal (s, t)-path (of maximum cover) of $\mathcal{G}$ with procedure MAXCOVER.
5. Output the job partition $P_1, P_2, \ldots, P_m$ determined by the path.

Output: a partition of the input jobs with a cover within a factor $2 + 5\varepsilon$ of the optimum cover.

Fig. 3. The algorithm MONCOVER

new rounded speed being s', and show that with this new input the algorithm allocates at least as much workload to the machine, as with speed σ_i.

We start with a couple of simple observations. Since the machines are indexed in increasing order of speed (breaking ties by some fixed machine order), the new index i' of the machine is at least i. If $s' = s_i$ (the rounded speed remains the same), then the output of the algorithm is exactly the same, and the allocated workload will be $|P_{i'}| \geq |P_i|$, and the theorem holds. Further, it is enough to prove the theorem for the case when $s' = (1+\varepsilon)s_i$, and the machine index does not change, i.e., i was the highest index of speed s_i, and becomes the lowest index of speed $s'_i = s' = (1+\varepsilon)s_i$. For all other cases the proof easily follows by 'continuously' increasing σ_i to σ'.

Observe that the graph $\mathcal{G}(V, A)$ constructed in step 1 does not change, and $f'(v) \leq f(v)$ holds for the new finish time $f'(v)$ of each vertex $v \in V$. Now we turn to procedure MaxPath. Because the finish times cannot increase, the minimum of the finish times over any path in $\mathcal{G}$ cannot increase. In particular, for every vertex v the optimum cover $opt(v)$ over all (v, t)-paths cannot increase either, i.e., $opt'(v) \leq opt(v)$ holds, where $opt'()$ denotes the new optimum. Note that the optimal (v, t)-path itself might change.

If the path which is output by MaxPath is the same for both input speeds, then the theorem holds. So, let $s, v_1, v_2, \ldots, v_r = t$ be the output path with speed s_i, and $s, v'_1, v'_2, \ldots, v'_{r'} = t$ be the output path with speed s'_i, and k be the minimum index s.t. $v_k \neq v'_k$. That is, $v_k = succ(v_{k-1})$ for the first, and $v'_k = succ'(v_{k-1})$ for the second input. Since no vertex could increase its $opt()$ value, in the second input v'_k could improve its relative position to v_k only due to $opt'(v_k) < opt(v_k)$. In particular, the path $v_k, v_{k+1}, \ldots, v_r$, decreased its cover from $opt(v_k)$ to *at most* $opt'(v_k)$ when s_i increased. That is, i must have become a bottleneck machine, and the minimum finish time over $v_k, v_{k+1}, \ldots, v_r$ became $|P_i|/s'_i = f'(v_q)$, where machine i is represented by the configuration of vertex v_q in the path. So, we have $|P_i|/s'_i = f'(v_q) \leq opt'(v_k)$. On the other hand, $opt'(v_k) \leq opt'(v'_k)$, since v'_k was selected over v_k, and $opt'(v'_k) \leq |P'_i|/s'_i$, because P'_i is determined by the new optimal path. Putting it together, we obtain $|P_i| \leq |P'_i|$. □

6 Conclusions

The question whether there is a monotone PTAS for related scheduling with cover optimization, remains open. The same holds for minimizing the L_p-norm of finish times for any $p > 1$. While for the respective (non-strategic) problems the classic PTAS, as well as the randomized monotone PTAS are easy to adapt [10,8], the same does not seem to hold concerning the deterministic monotone PTAS.

References

1. Archer, A., Tardos, É.: Truthful mechanisms for one-parameter agents. In: Proc. 42nd Annual Symposium on Foundations of Computer Science, pp. 482–491 (2001)
2. Asadpour, A., Saberi, A.: An approximation algorithm for max-min fair allocation of indivisible goods. In: Proc. 39th Annual ACM Symp. Theory of Comp. (STOC), pp. 114–121. ACM, New York (2007)

3. Azar, Y., Epstein, L.: On-line machine covering. In: Burkard, R.E., Woeginger, G.J. (eds.) ESA 1997. LNCS, vol. 1284, pp. 23–36. Springer, Heidelberg (1997)
4. Azar, Y., Epstein, L.: Approximation schemes for covering and scheduling on related machines. In: Jansen, K., Rolim, J.D.P. (eds.) APPROX 1998. LNCS, vol. 1444, pp. 39–47. Springer, Heidelberg (1998)
5. Bateni, M., Charikar, M., Guruswami, V.: Maxmin allocation via degree lower-bounded arborescences. In: Proc. 41st Annual ACM Symp. Theory of Comp., pp. 543–552. ACM, New York (2009)
6. Chakrabarty, D., Chuzhoy, J., Khanna, S.: On allocating goods to maximize fairness. In: Proc. 50th Annual IEEE Symp. on Found. of Comp. Sci. (FOCS), pp. 107–116. IEEE Computer Society, Los Alamitos (2009)
7. Christodoulou, G., Kovács, A.: A deterministic truthful PTAS for scheduling related machines. In: Proc. 21st SIAM Symp. on Disc. Algs. (SODA), pp. 1005–1016. SIAM, Philadelphia (2010)
8. Dhangwatnotai, P., Dobzinski, S., Dughmi, S., Roughgarden, T.: Truthful approximation schemes for single-parameter agents. In: Proc. 49th IEEE Symp. on Found. of Comp. Sci., FOCS (2008)
9. Efraimidis, P.S., Spirakis, P.G.: Approximation schemes for scheduling and covering on unrelated machines. Theoretical Computer Science 359(1-3), 400–417 (2006)
10. Epstein, L., Sgall, J.: Approximation schemes for scheduling on uniformly related and identical parallel machines. Algorithmica 39(1), 43–57 (2004)
11. Epstein, L., van Stee, R.: Maximizing the minimum load for selfish agents. Theoretical Computer Science 411(1), 44–57 (2010)
12. Feige, U.: On allocations that maximize fairness. In: Proc. 19th annual ACM-SIAM Symp. Discr. Algs. (SODA), pp. 287–293. SIAM, Philadelphia (2008)
13. Hochbaum, D.S., Shmoys, D.B.: A polynomial approximation scheme for scheduling on uniform processors: Using the dual approximation approach. SIAM Journal on Computing 17(3), 539–551 (1988)
14. Mu'alem, A., Schapira, M.: Setting lower bounds on truthfulness. In: Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1143–1152 (2007)
15. Myerson, R.B.: Optimal auction design. Mathematics of Operations Research 6, 58–73 (1981)
16. Skutella, M., Verschae, J.: A robust PTAS for machine covering and packing. In: de Berg, M., Meyer, U. (eds.) ESA 2010. LNCS, vol. 6346, pp. 36–47. Springer, Heidelberg (2010)
17. Woeginger, G.J.: A polynomial time approximation scheme for maximizing the minimum machine completion time. Operations Research Letters 20(4), 149–154 (1997)

Braess's Paradox in Large Sparse Graphs

Fan Chung and Stephen J. Young

Department of Mathematics
University of California, San Diego
9500 Gilman Drive
La Jolla, CA 92093
{fan,s7young}@math.ucsd.edu

Abstract. Braess's paradox, in its original context, is the counter-intuitive observation that, without lessening demand, closing roads can improve traffic flow. With the explosion of distributed (selfish) routing situations understanding this paradox has become an important concern in a broad range of network design situations. However, the previous theoretical work on Braess's paradox has focused on "designer" graphs or dense graphs, which are unrealistic in practical situations. In this work, we exploit the expansion properties of Erdős-Rényi random graphs to show that Braess's paradox occurs when $np \geq c\log(n)$ for some $c > 1$.

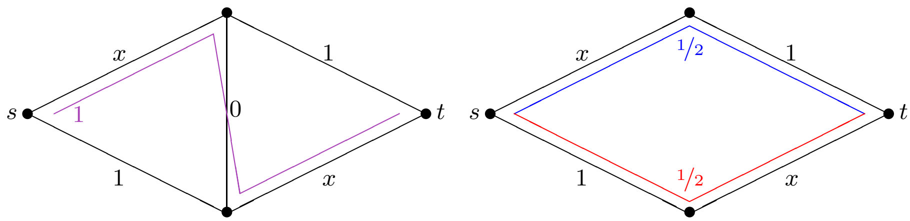
Keywords: Braess's paradox, price of anarchy, random graphs, selfish routing.

1 Introduction

In 1968 Dietrich Braess observed that there were road networks such that if the travellers were behaving selfishly it was possible to improve everyone's travel time by *removing* roads, even roads with extremely fast travel times. Specifically, he considered the situation illustrated in Figure 1 in the case of routing one unit of flow from s to t. As we can see in Figure 1(a) when the users behave selfishly all of the flow passes through the zero latency central edge and the total latency is 2. However, as we can see in Figure 1(b), by removing the central zero latency edge, the selfish routing will spread the flow uniformly over the paths between s and t, resulting in an overall latency of $\frac{3}{2}$.

Since its discovery Braess's paradox has spawned a significant amount of work aimed at understanding the full implications of the paradox, both theoretically [3,5,11,12] and via anecdotal observations [2,7]. In many ways the recent trend towards studying the "Price of Anarchy" [8,10] has its roots in the discovery of Braess's paradox. However, these "worst case" analyses via designer instances give little insight into the practical consequences of Braess's paradox. Although the anecdotal evidence indicates that Braess's paradox can occur in real world, it gives little to no feeling for how prevalent or severe the paradox can be in real world networks. Recently, Valiant and Roughgarden [16] began to answer this question by providing the first proof that Braess's paradox could

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 194–208, 2010.



(a) Original Network with equilibrium latency 2

(b) Modified network with equilibrium latency 3/2

Fig. 1. Braess's Paradox

occur in a large class of non-designer graphs. Specifically, they showed that in sufficiently dense instances of Erdős-Rényi random graphs with affine latency functions satisfying certain mild conditions, Braess's paradox occurs with high probability (that is, with probability $1 - \mathcal{O}(n^{-c})$ for some $c > 0$). In this work we extend their results to almost all conected Erdős-Rényi random graphs. That is, with similar mild conditions on the latency function, Braess's paradox occurs with high probability in Erdős-Rényi random graphs with expected degree at least a $c' \log(n)$ for some $c' > 1$.

We consider a single commodity flow on an undirected Erdős-Rényi random graph G with a designated sink s and source t with latency functions $\ell = \{\ell_e\}_{e \in E(G)}$ associated to each edge. Letting $\mathcal{P}$ be the set of simple s-t paths in G, a *flow* is a function $f\colon \mathcal{P} \longrightarrow \mathbb{R}^{\geq 0}$. The flow on an edge e with respect to a flow f is $f_e = \sum_{P \in \mathcal{P}_e} f(P)$, where $\mathcal{P}_e$ is the set of paths in $\mathcal{P}$ containing the edge e. We will say an edge is *flow carrying* if $f_e > 0$, similarly, we will say a vertex v is flow carrying if it is incident to some flow carrying edge. The *traffic rate*, or value, of a flow f, is $R = \sum_{P \in \mathcal{P}} f(P)$. The *latency of a path* P with respect to a flow f, which we will denote as $\ell_P(f)$, is $\sum_{e \in E(P)} \ell_e(f_e)$. Notice that the latency of a path depends on the value of the flow on an edge, not the value of the flow on the particular path under consideration.

We concern ourselves with the case of routing an infinitely divisible flow from s to t such that each "unit" of flow behaves selfishly. That is, given a fixed traffic rate R we are interested in the properties of those flows such that there is no incentive for any "unit" of flow to change the path it is on. Alternatively, we may say that such a flow is at Nash equilibrium. One way of characterizing such flows is that for every pair $P_1, P_2 \in \mathcal{P}$, if $f(P_1) > 0$, then $\ell_{P_1}(f) \leq \ell_{P_2}(f)$ [17]. In particular, this implies that all flow carrying paths have the same latency at Nash equilibrium. It has been shown that every selfish routing network has a Nash equilibrium flow and further, all flow carrying paths in all possible Nash equilibrium flows have the same latency [1]. Thus, we may define $\mathcal{L}(G, \ell, R)$ as the common latency of all flow carrying paths at Nash equilibrium in the graph G with the latency functions ℓ at the traffic rate R. Given this notation, Braess's paradox may be restated as the observation that there exists an instance (G, ℓ, R) and a subgraph G' of G, such that $\mathcal{L}(G', \ell, R) < \mathcal{L}(G, \ell, R)$. For a given instance, define the *Braess ratio* of the instance, denoted $\rho(G, \ell, R)$, as

$$\max_{G' \subseteq G} \frac{\mathcal{L}(G, \ell, R)}{\mathcal{L}(G', \ell, R)}. \tag{1}$$

Note that the Braess ratio is specification of the price of anarchy to the context of this context.

1.1 Previous Work

Recently, driven by the obvious practical applications there has been some work attempting to answer the question of whether networks can be designed to avoid Braess's paradox. A, perhaps, more important question is whether Braess's paradox can be exploited to improve the performance of already existent real world networks. Roughgarden emphatically answers these questions in the negative in [13], by showing that unless P = NP there is not an $\frac{n}{2}$ approximation algorithm to determining a subnetwork which achieves a Braess ratio larger than 1. Further, even if the latency functions are restricted to be affine, there is still not a $\frac{4}{3}$ approximation algorithm, and this result is tight [14]. This leads naturally to the following important practical question: Is Braess's paradox a prevalent phenomenon or, like the exponential examples for the simplex method [6,15], is it an academic curiosity that can be ignored in practice? The recent work of Valiant and Roughgarden [16] has begun to address this fundamental question. In order to state and understand their results, we need the following definition. A pair of distributions $\mathcal{A}$ and $\mathcal{B}$ is *reasonable* if

- $\mathcal{A}$ has bounded support $[A_{\min}, A_{\max}]$ with $A_{\min} > 0$,
- there is some closed interval I_A of positive length, such that for every nontrivial subinterval $J \subseteq I_A$, $\mathbb{P}(\mathcal{A} \in J) > 0$, and
- there is some interval $I_B = [0, \eta]$, with $\eta > 0$, such that for every nontrival subinterval $J \subseteq I_B$, $\mathbb{P}(\mathcal{B} \in J) > 0$.

Within this context they were able to show the following theorem.

Theorem 1 ([16]). *Let $p = \Omega\left(n^{-\frac{1}{2}+\zeta}\right)$ be an edge sampling probability with $\zeta > 0$ and let $\mathcal{A}$ and $\mathcal{B}$ be reasonable distributions. Let G be an Erdős-Rényi random graph with edge probability p and let $\ell_e = a_e f_e + b_e$ for all edges, where (a_e, b_e) is distributed as $\mathcal{A} \times \mathcal{B}$. There is a constant $\rho = \rho(\zeta, \mathcal{A}, \mathcal{B}) > 1$ such that, with high probability the instance (G, ℓ), admits a choice of traffic rate R such that the Braess ratio of the instance is at least ρ.*

Given the importance and self evident "correctness" of Nash equilibrium flows in a practical context (especially in the context of automobile traffic) it is unsurprising that there has been significant previous work on the properties of such flows. We collect a few of the more useful results in the following lemma.

Lemma 2. *Given an instance (G, ℓ, R) and an Nash equilibrium flow f for each vertex v define $d_s(v)$ as the shortest path from v to s with respect to the latencies $\ell(f)$. Define $d_t(f)$ analogously. The following properties then hold for all Nash equilibrium flows f.*

1. *If f is a Nash equilibrium flow for traffic rate R on the network G with latencies ℓ, then for every vertex v we have $d_s(v) + d_s(t) \geq \mathcal{L}(G, \ell, R)$ with equality if v is a flow carrying vertex.*
2. *[13] If f is a flow achieving traffic rate r for the instance (G, ℓ) then for all edges $e = \{u, v\}$, $d_s(v) - d_s(u) \geq \ell_e(f_e)$ with equality if and only if equality holds whenever $f_e > 0$.*
3. *[4,9] For every network G and strictly increasing latency functions ℓ, $\mathcal{L}(G, \ell, R)$ is continuous and strictly increasing function of R.*
4. *[13] There is a Nash equilibrium flow f so that the set of edges with $f_e > 0$ is acyclic when considered as a directed graph.*

Now by part (3) of this lemma, if the latency functions are all strictly increasing, there is a function $R_\ell^G(\mathcal{L})$ which gives the unique value R so that the latency of a Nash equilibrium flow on the network G with latencies ℓ is precisely $\mathcal{L}$. When the underlying graph is clear, we will denote this simply as $R_\ell(\mathcal{L})$.

1.2 Our Contribution

In the work of Valiant and Roughgarden the critical structural property they use in their proofs is that if $p \gg n^{-\frac{1}{2}+\zeta}$ for some $\zeta > 0$, then $\mathcal{G}(n, p)$ has polynomially many disjoint paths of length two between any two vertices. Whereas if p is $\mathcal{O}\left(n^{-\frac{1}{2}+o(1)}\right)$ there are clearly not even polynomially many paths of length two between any two vertices. More importantly, very few, if any real world networks share this property. The primary contribution of our work is to generalize their methodology to rely on a more prevalent real world property, expansion.

In Section 2 we analyze an idealized network and reveal two key properties of Nash equilibria that will be used in our proof. Specifically, we observe that at particular traffic rate in this idealized network every internal vertex is at the same latency distance from s and further, every internal vertex is equidistant from s and t. In the more general case, this breaks down into two claims. The first, which we prove in Section 3, is that for any two internal vertices u and v, their latency distance to s differs by at most a fixed constant δ. Valiant and Roughgarden prove a similar result for dense Erdős-Rényi random graphs in their δ-lemma using a specialized case of our expansion based argument. The second claim is that the internal vertices are equidistant from s and t. Clearly, this cannot hold if the internal vertices do not all have the same latency distance to s, however, as we show in Section 4 the latency distance for internal vertices is roughly balanced between exiting s and entering t. We note that Valiant and Roughgarden's proof of the balance lemma would suffice at this point, however, their proof depends on a somewhat unnatural and cumbersome discretization argument which we are able to completely avoid. Finally, in Section 5 we show, to our knowledge, the first proof that Braess's paradox can occur in sparse, non-designer graphs with the following theorem.

Theorem 3. *Let G be an Erdős-Rényi random graph on n vertices with edge probability p. Let $\mathcal{A}$ and $\mathcal{B}$ be reasonable distributions and let all latency functions have the form $\ell_e(f_e) = a_e f_e + b_e$ where (a_e, b_e) is distributed according to $\mathcal{A} \times \mathcal{B}$. There are constants $\delta > 0$, $c > 1$, and $\rho > 1$ such that, if $\mathbb{P}\left(\mathcal{B} \leq \frac{\delta}{\log(n)}\right) pn \geq c \log(n)$, then there is a flow rate R such that the instance (G, ℓ, R) has Braess's ratio at least ρ with high probability.*

2 Sketch of Ideas via a Motivating Example

In many ways the ideas behind the our proofs are motivated by the following example. Let G be a graph consisting of s, t, and a complete bipartite graph $K_{n,n}$ such that s and t are each adjacent to every vertex on opposite sides of the bipartition. That is, $(\Gamma(s), \Gamma(t))$ forms the bipartition of $K_{n,n}$. Define the latency function such that every edge in the $K_{n,n}$ has latency 0 and every edge adjacent to s or t has latency function $ax + b$ where (a, b) is distributed as $\mathcal{A} \times \mathcal{B}$. See Figure 2(a). Since the latencies in $K_{n,n}$ are all 0, we can explicitly calculate the flow given two values, c and $\mathcal{L}$, where c is the value of $d_s(v)$ for any vertex in the $K_{n,n}$ and $\mathcal{L}$ is the overall latency of the network. Specifically, the total flow is

$$R = \sum_{\substack{v \in \Gamma(s) \\ b_{\{s,v\}} \leq c}} \frac{c - b_{\{v,s\}}}{a_{\{v,s\}}} = \sum_{\substack{u \in \Gamma(t) \\ b_{\{s,u\}} \leq \mathcal{L} - c}} \frac{\mathcal{L} - c - b_{\{u,s\}}}{a_{\{u,s\}}}. \tag{2}$$

We note that this implies that, up to lower order terms, $c = \mathcal{L} - c$ with high probability.

Now, if $\mathcal{A}$ and $\mathcal{B}$ are reasonable, then let $I_\mathcal{A}$ and $I_\mathcal{B}$ be the two intervals witnessing their reasonableness. As in [16] choose $A_1 < A_2$ arbitrarily in the

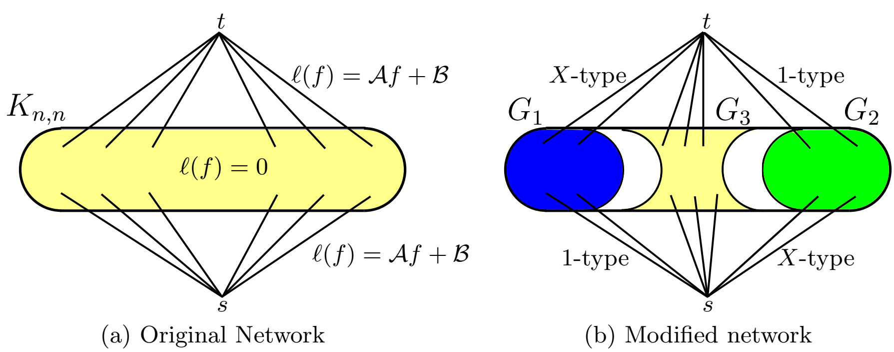


Fig. 2. Motivating Example

interior $I_{\mathcal{A}}$ and B in the interior of $I_{\mathcal{B}}$. Letting $\mathcal{B}_{\leq B}$ be $\mathcal{B}$ restricted to be at most B, we then have that in our motivating example

$$R_\ell(2B) = \frac{B - \mathbb{E}\left[\mathcal{B}_{\leq B}\right]}{\mathbb{E}\left[\mathcal{A}\right]} \mathbb{P}(\mathcal{B} \leq B)\, n \tag{3}$$

up to low order terms. Now we want to partition $K_{n,n}$ into three parts, yielding graphs G_1, G_2, and G_3 so that $R_\ell^{G_1}(2B) + R_\ell^{G_2}(2B) + R_\ell^{G_3}(2B) > R_\ell^{G}(2B)$. If we succeed in this, then the graph $G' = G_1 \cup G_2 \cup G_3$ is a proper subgraph of G which can route more flow than G at the same latency. Then, since the latency function is strictly increasing by Lemma 2, this implies that Braess's paradox occurs. In order to do that we will again mimic the work of Valiant and Roughgarden [16] and partition the vertices into three classes based on the latency function of the edge adjacent with s or t. Specifically, fix $\epsilon > 0$ so that $(1-\epsilon)A_2 > A_1$ and let E_1 be the edges with latency function $ax + b$ such that $a \leq A_1$ and $b \in (B, (1+\epsilon)B)$. These are the 1-*type* edges. Similarly, define the X-*type* edges as those where the latency function $ax + b$ is such that $a \in ((1-\epsilon)A_2, A_2)$ and $b \leq \epsilon B$ and denote the set of such edges by E_X. We will then choose the partition of the $K_{n,n}$ to force the use of 1-type edges which are underutilized in the routing in G. To that end, define G_1 as the graph induced by $\{s,t\} \cup \{v \in \Gamma(s) \mid \{v,s\} \in E_1\} \cup \{u \in \Gamma(t) \mid \{u,t\} \in E_X\}$. Further define G_2 as the graph induced by $\{s,t\} \cup \{v \in \Gamma(s) \mid \{v,s\} \in E_X\} \cup \{u \in \Gamma(t) \mid \{u,t\} \in E_1\}$. Finally, G_3 is the graph induced by $\{s,t\}$ and the vertices not in G_1 or G_2. In other words, G_1 contains all the 1-type edges adjacent to s and all the X-type adjacent to t, G_2 contains all the X-type edges adjacent to s and all the 1-type adjacent to t, and the remaining edges are in G_3. See Figure 2(b). Now if the probability of being a 1-type edge is greater than the probability of being a X-type edge, randomly move 1-type edges (and their associated vertices) to G_3 so that the expected degrees of s and t are the same in G_1 and G_2. By performing the analogous operation if the probability of being an X-type edge is greater than the probability of being a 1-type edge, we may assume that the expected degrees of s and t in the same in each of G_1, G_2, and G_3. That is, letting D be the expected degree of s in G_1, we have $D = \mathbb{E}\left[\Gamma_{G_1}(s)\right] = \mathbb{E}\left[\Gamma_{G_2}(s)\right] = \mathbb{E}\left[\Gamma_{G_1}(t)\right] = \mathbb{E}\left[\Gamma_{G_2}(t)\right]$.

Now consider $R_\ell^{G_3}(2B)$. Since the distribution of latencies are the same at s and t in G_3, for every $v \in \Gamma_{G_3}(\{s,t\})$, $d_s(v) = B$ up to lower order terms. Similarly, when considering $R_\ell^{G}(2B)$ we have for every vertex $v \in \Gamma_G(\{s,t\})$, $d_s(v) = B$ up to lower order terms as well. Thus, the flow on edges in present in G_3 is the same (again, up to low order terms) as the flow on the corresponding edges in G. Furthermore, this also implies that the 1-type edges in G have asymptotically zero flow. Hence, we have that asymptotically $R_\ell^{G}(2B) - R_\ell^{G_3}(2B) \leq \frac{B}{(1-\epsilon)A_2} D$. Now we return to G_1. By similar arguments as above there exists some constant c such that for every 1-type edge e in G_1, $a_e f_e + b_e = c$ and for every X-type edge e', $a_{e'} f_{e'} + b_{e'} = 2B - c$. Summing over all 1-type and X-type edges in G_1 we have, asymptotically,

$$
\begin{aligned}
2BD &= \sum_{v \in \Gamma_{G_1}(s)} a_{\{v,s\}} f_{\{v,s\}} + b_{\{v,s\}} + \sum_{u \in \Gamma_{G_1}(t)} a_{\{u,t\}} f_{\{u,t\}} + b_{\{u,t\}} && (4)\\
&\leq \sum_{v \in \Gamma_{G_1}(s)} A_1 f_{\{v,s\}} + (1+\epsilon)B + \sum_{u \in \Gamma_{G_1}(t)} A_2 f_{\{u,t\}} + \epsilon B && (5)\\
&= (A_1 + A_2)\, R_\ell^{G_1}(2B) + (1+2\epsilon)BD. && (6)
\end{aligned}
$$

Solving, we have that $R_\ell^{G_1}(2B) \geq \frac{(1-2\epsilon)B}{A_1+A_2}D$. Similarly, $R_\ell^{G_2}(2B) \geq \frac{(1-2\epsilon)B}{A_1+A_2}D$ and thus, if

$$\frac{2(1-2\epsilon)B}{A_1+A_2}D > \frac{B}{(1-\epsilon)A_2}D. \tag{7}$$

then Braess's paradox occurs with high probability in this example. By rearranging, we get that this condition is equivalent to

$$2(1-2\epsilon)(1-\epsilon) > 1 + \frac{A_1}{A_2}. \tag{8}$$

Since $A_1 < A_2$, there is some $\epsilon > 0$ which makes this inequality true and thus Braess's paradox occurs.

We notice that there are two key observations that allow Braess's paradox to occur in this example. The first is that for any internal vertices u and v, $d_s(u) = d_s(v)$. This observation is mimicked by the δ-lemma, which will be proved in Section 3, which says that $|d_s(u) - d_s(v)| \leq \delta$ for some small $\delta > 0$. The next key observation is that $\mathcal{L} - c = c$ and so the latency is balanced exiting s and entering t, which corresponds naturally to the balance lemma which we prove in Section 4.

3 Small Latency Separates Interior Vertices

The basis for Valiant and Roughgarden's proof of the δ-lemma is that if p is $\Omega\left(n^{-\frac{1}{2}+\zeta}\right)$ with $\zeta > 0$, then there are many paths of length two between any pair of vertices. This structural property allows the flow to be spread out among many internally disjoint paths, yielding a relatively small increase in latency between any two internal vertices. Although no similar property holds for sparser Erdős-Rényi graphs, the following expansion property will provide a sufficient analog.

Lemma 4. *Let G be an Erdős-Rényi random graph on n vertices with edge probability p. There is some $c > 1$ such that, if $np > c\log(n)$, then with high probability every subset U of the vertices is such that $|\Gamma(U)| \geq \left(\frac{e-1}{e} - o(1)\right)\min\{np|U|, n\}$*

Roughly speaking, this lemma allows us to view the graph through the lens of a series of s-t cuts, each of which has a relatively large number of edges crossing the cut. This large number of edges crossing the cut allows for the flow from s to t to be spread out and guarantees that there are some low latency edges crossing the cut. Then, by moving the endpoints of these low latency edges across the cut, we get a new s-t cut and can repeat this procedure. However, there are two primary difficulties in applying this methodology. The first is that the initial cut can have a relatively small number of edges compared to the subsequent cuts, and the second is that at the some point the number of edges crossing the cut begins to decline. We deal with first by allowing the first cut to use edges with higher latency in order to get an initial "core" of vertices with which to start the method. The second difficulty is dealt with by building cuts from s and from t, and recognizing that when those cuts are close to colliding there are a large number of short (in fact, length at most two) internally disjoint paths between the two sets. This last step is the only step needed in the denser case dealt with in [16].

Lemma 5. *Let G be an Erdős-Rényi random graph on n vertices with edge probability p. Let $\mathcal{A}$ and $\mathcal{B}$ be reasonable distributions and let all latency functions have the form $\ell_e(f_e) = a_e f_e + b_e$ where (a_e, b_e) is distributed according to $\mathcal{A} \times \mathcal{B}$. For any sufficiently small fixed $\delta > 0$, there are some constants $c > 1$ and $n_0 > 0$ such that, if $n > n_0$, $np > c\log(n)$ and $\mathbb{P}\left(\mathcal{B} \leq \frac{\delta}{\log(n)}\right) pn \geq 4$, then for any two flow carrying vertices u, v other than s and t in the instance $(G, \ell, R_\ell(2B))$, we have $|d_s(u) - d_s(v)| \leq 7\delta$ and $|d_t(u) - d_t(v)| \leq 7\delta$ with high probability.*

Proof. Since the case where p is a constant was resolved by Valiant and Roughgarden in [16] we may assume without loss of generality that $p \to 0$. Let v_s be a flow carrying vertex that minimizes $d_s(v_s)$ and let v_t be a flow carrying vertex that maximizes $d_s(v_t)$. Since, for flow carrying vertices v, $d_s(v) + d_t(v) = 2B$, in order to show the lemma it suffices to show that $d_s(v_t) - d_s(v_s) \leq 7\delta$ for sufficiently large n. We note that the amount of flow entering v_s is at most $\frac{d_s(v_s)}{A_{\min}} \leq \frac{2B}{A_{\min}}$. Further, since by Chernoff bounds the $\deg(s) \leq \frac{3}{2}np$ with high probability, $R_\ell(2B) \leq \frac{3Bnp}{A_{\min}}$. Additionally, by Chernoff bounds, with high probability there are least $\frac{2}{3}\mathbb{P}(\mathcal{B} \leq \delta)\, np$ edges e adjacent to v_s with $b_e \leq \delta$. At most half of these have flow more that $2\frac{2B}{A_{\min}}\frac{1}{\frac{2}{3}\mathbb{P}(\mathcal{B}\leq\delta)np}$, thus there are at least $\frac{1}{3}\mathbb{P}(\mathcal{B} \leq \delta)\, np$ vertices v such that $d_s(v) \leq d_s(v_s) + A_{\max}\frac{6B}{\mathbb{P}(\mathcal{B}\leq\delta)np} + \delta$. With this in mind, let $c_0 = d_s(v_s) + A_{\max}\frac{6B}{\mathbb{P}(\mathcal{B}\leq\delta)np} + \delta$ and let $U_0 = \{v \in V(G) \mid d_s(v) \leq c_0\}$ and note that $|U_0| \geq \frac{1}{3}\mathbb{P}(\mathcal{B} \leq \delta)\, np$.

We now will inductively define a sequence U_i and c_i such that $U_i \subset U_{i+1}$ and $c_i < c_{i+1}$, stopping when $|\Gamma(U_i)| \geq \frac{3n}{5}$. Suppose then that U_i and c_i are defined and that $|\Gamma(U_i)| < \frac{3n}{5}$ and let $\gamma = \frac{\delta}{\log(n)}$. Then, by Lemma 4 and noting that $\frac{3}{5} < 1 - \frac{1}{e}$, with high probability $|\Gamma(U_i) \backslash U_i| \geq \frac{3}{5}np\,|U_i|$ for sufficiently large n. Furthermore, (again by Chernoff bounds) with probability at least $1 - e^{-\frac{5}{24}\mathbb{P}(\mathcal{B}\leq\gamma)np|U_i|} > 1 - e^{-\frac{\mathbb{P}(\mathcal{B}\leq\gamma)np|U_i|}{5}}$ there are at least $\frac{1}{2}\mathbb{P}(\mathcal{B} \leq \gamma)\, np\,|U_i|$ vertices in $\overline{U_i}$ (the compliment of U_i) that are connected to a vertex in U_i by an edge e with $b_e \leq \gamma$. Let U_i' be the set of such vertices and let E_i be a set of witnesses for membership in U_i'. That is, for every vertex $v \in U_i'$ there is a unique edge $e \in E_i$ so that $e \in U_i \times \{v\}$ and $b_e \leq \gamma$. Now since $(U_i, \overline{U_i})$ is a cut and we may assume that the Nash equilibrium flow is cycle free, there is $R_\ell(2B) \leq \frac{3Bnp}{A_{\min}}$ flow crossing the cut from U_i to $\overline{U_i}$. But then at most half of the edges in E_i have flow greater than $2\frac{3Bnp}{A_{\min}|U_i'|}$ and in particular at least half of the vertices $v \in U_i'$ have

$$d_s(v) \leq c_i+\gamma+A_{\max}\frac{6Bnp}{A_{\min}\left|U_i'\right|} \leq c_i+\gamma+\frac{12A_{\max}Bnp}{A_{\min}np\left|U_i\right|} = c_i+\gamma+\frac{12A_{\max}B}{A_{\min}\left|U_i\right|}. \quad (9)$$

Thus we define $c_{i+1} = c_i + \gamma + \frac{12A_{\max}B}{A_{\min}|U_i|}$ and $U_{i+1} = \{v \in V(G) \mid d_s(v) \leq c_{i+1}\}$. By the above we have that

$$|U_{i+1}| \geq \left(\frac{1}{4}\mathbb{P}(\mathcal{B} \leq \gamma)\, np + 1\right)|U_i| \geq \left(\frac{1}{4}\mathbb{P}(\mathcal{B} \leq \gamma)\, np + 1\right)^i |U_0|\,. \quad (10)$$

If i^* is the first i such that $|\varGamma(U_i)| \geq \frac{3n}{5}$, then this implies that

$$i^* \leq \frac{\log(\frac{3n}{5|U_0|})}{\log(\frac{1}{4}\mathbb{P}(\mathcal{B} \leq \gamma)\, np + 1)} \leq \frac{\log(\frac{9}{5\mathbb{P}(\mathcal{B}\leq\delta)p})}{\log(\frac{1}{4}\mathbb{P}(\mathcal{B} \leq \gamma)\, np + 1)}. \quad (11)$$

Since, by assumption $\mathbb{P}(\mathcal{B} \leq \gamma)\, np \geq c\log(n)$ for some $c > 1$, we have $\frac{1}{p} \leq \frac{n}{\log(n)}$, and thus $i^* \leq \log(n)$ for sufficiently large n. As a consequence, we have that

$$\begin{aligned}
c_{i^*} &\leq c_0 + \gamma i^* + \sum_{i=0}^{i^*}\frac{12A_{\max}B}{A_{\min}\left|U_i\right|} && (12)\\
&\leq c_0 + \frac{\delta}{\log(n)}\log(n) + \sum_{i=0}^{\log(n)}\frac{12A_{\max}B}{A_{\min}\left(\frac{1}{4}\mathbb{P}(\mathcal{B} \leq \gamma)\, np + 1\right)^i |U_0|} && (13)\\
&= c_0 + \delta + \frac{12A_{\max}B}{A_{\min}\left|U_0\right|}\sum_{0}^{\log(n)}\left(\frac{1}{4}\mathbb{P}(\mathcal{B} \leq \gamma)\, np + 1\right)^{-i} && (14)\\
&\leq \left(d_s(v_s) + \frac{6BA_{\max}}{\mathbb{P}(\mathcal{B} \leq \delta)\, np} + \delta\right) + \delta + \frac{36A_{\max}B}{A_{\min}\mathbb{P}(\mathcal{B} \leq \delta)\, np}\sum_{i=0}^{\infty}2^{-i} && (15)\\
&= d_s(v_s) + 2\delta + \frac{BA_{\max}\left(6A_{\min} + 72\right)}{A_{\min}\mathbb{P}(\mathcal{B} \leq \delta)\, np}. && (16)
\end{aligned}$$

At this point it worth noting that the failure probability in the recursive construction of U_{i^*} is at most

$$\begin{aligned}
\sum_{i=0}^{i^*} e^{-\frac{\mathbb{P}(\mathcal{B}\leq\gamma)np|U_i|}{5}} = \sum_{i=0}^{i^*} e^{-\frac{\mathbb{P}(\mathcal{B}\leq\gamma)np\left(\frac{1}{4}\mathbb{P}(\mathcal{B}\leq\gamma)np+1\right)^i|U_0|}{5}} &\leq \sum_{i=0}^{i^*} e^{-\frac{2^{i+2}|U_0|}{5}} && (17)\\
&\leq \sum_{i=0}^{i^*} e^{-\frac{4}{15}2^i\mathbb{P}(\mathcal{B}\leq\delta)np} && (18)\\
&\leq \log(n)e^{-\frac{4}{15}\mathbb{P}(\mathcal{B}\leq\delta)np}. && (19)
\end{aligned}$$

Thus, with high probability, this construction method succeeds.

In a similar way, we can define $c'_{j^*} = d_s(v_t) - 2\delta - \frac{BA_{\max}(6A_{\min}+72)}{A_{\min}\mathbb{P}(\mathcal{B}\leq\delta)np}$ and $V_{j^*} = \left\{v \mid d_s(v) \geq c'_{j^*}\right\}$ and have that $|\Gamma(V_{j^*})| \geq \frac{3n}{5}$. Without loss of generality, we may assume that $c'_{j^*} - c_{i^*} > 0$ and $V_{j^*} \cap U_{i^*} = \varnothing$. Now since $|\Gamma(U_{i^*})| + |\Gamma(V_{j^*})| \geq \frac{6n}{5}$ there are at least $\frac{n}{10}$ edge disjoint paths of length at most 2 between U_{i^*} and V_{j^*}. Furthermore, by Chernoff bounds, this implies that with high probability there are at least $\frac{1}{12}\mathbb{P}(\mathcal{B} \leq \delta)^2 n$ such paths where all edges e on the path have $b_e \leq \delta$. Now, at most half of those paths have flow more that $2\frac{12R_\ell(2B)}{\mathbb{P}(\mathcal{B}\leq\delta)^2 n}$ and thus

$$c'_{j^*} - c_{i^*} \leq 2\delta + \frac{48A_{\max}R_\ell(2B)}{\mathbb{P}(\mathcal{B} \leq \delta)^2 n} \leq 2\delta + \frac{144A_{\max}Bp}{A_{\min}\mathbb{P}(\mathcal{B} \leq \delta)^2}. \tag{20}$$

Putting all the pieces together we have that $d_s(v_s) - d_s(v_t) \leq 6\delta + \frac{144BA_{\max}}{\mathbb{P}(\mathcal{B}\leq\delta)^2}p + \frac{BA_{\max}(12A_{\min}+144)}{A_{\min}\mathbb{P}(\mathcal{B}\leq\delta)np}$, which, for sufficiently large n, is at most 7δ. □

This control on the spread of flow carrying vertices leads to the following control on the spread of all vertices other than s and t.

Corollary 6. *Let G be an Erdős-Rényi random graph on n vertices with edge probability p. Let $\mathcal{A}$ and $\mathcal{B}$ be reasonable distributions and let all latency functions have the form $\ell_e(f_e) = a_e f_e + b_e$ where (a_e, b_e) is distributed according to $\mathcal{A} \times \mathcal{B}$. For any sufficiently small fixed $\delta > 0$ there are some constants $c > 1$ and $n_0 > 0$ such that, if $n > n_0$ and $\mathbb{P}\left(\mathcal{B} \leq \frac{\delta}{\log(n)}\right) np \geq c\log(n)$, then for any two vertices u, v other than s and t in the instance $(G, \ell, R_\ell(2B))$, we have $|d_s(u) - d_s(v)| \leq 8\delta$ and $|d_t(u) - d_t(v)| \leq 8\delta$ with high probability.*

Proof. Let u and v be arbitrary vertices in $\mathcal{G}(n,p) - \{s,t\}$. Now if c is large enough then $\mathcal{G}(n,p) - \{s,t\}$ restricted to those edges e where $b_e \leq \frac{\delta}{\log(n)}$ is connected and has diameter at most $\log(n)$ with high probability. Let P be a path from u to v in the restricted graph and suppose without loss of generality that $d_s(u) \leq d_s(v)$. If the path P contains at most one flow carrying vertex, then none of the edges along the path carry any flow and thus $d_s(v) \leq d_s(u) + \log(n)\frac{\delta}{\log(n)} = d_s(u) + \delta$ by Lemma 2, and the claim follows. Thus there are at least two flow carrying vertices on the path P. Let u_f be the closest (in terms of the path) flow carrying vertex to u and similarly for v_f. Let g_u be the number of edges on P between u and u_f and similarly for g_v. Since none of the first g_u edges carry flow there is a path from u to t of length at most $g_u\frac{\delta}{\log(n)} + d_t(u_f) = g_u\frac{\delta}{\log(n)} + 2B - d_s(u_f)$. Further, since $d_s(u) + d_t(u) \geq 2B$ we have $2B - d_s(u) \leq d_t(u) \leq g_u\frac{\delta}{\log(n)} + 2B - d_s(u_f)$. Arguing analogously $d_s(v) \leq g_v\frac{\delta}{\log(n)} + d_s(v_f)$. Combining the inequalities we have

$$2B + d_s(v) - d_s(u) \leq (g_u + g_v)\frac{\delta}{\log(n)} + d_s(v_f) - d_s(u_f). \tag{21}$$

By assumption $d_s(u) \leq d_s(v)$ and by Lemma 5, $|d_s(v_f) - d_s(u_f)| \leq 7\delta$, thus $|d_s(v) - d_s(u)| \leq (g_u + g_v)\frac{\delta}{\log(n)} + 7\delta \leq 8\delta$. A similar argument shows that $|d_t(v) - d_t(u)| \leq 8\delta$. □

At this point it is worth noticing that the condition that $\mathbb{P}\Big(\mathcal{B} \leq \frac{\delta}{\log(n)}\Big)\, pn$ is likely necessary for this method to work as otherwise $\mathcal{L}(G, \ell, R) \to \infty$ even when $R = 0$.

4 Equal Distance Separates the Interior from s and t

In order to prove the balance lemma, Valiant and Roughgarden discretize the space of latency functions $a_e f_e + b_e$ into collections where $a_e \in [i\tau, (i+1)\tau)$ and $b_e \in [j\tau, (j+1)\tau)$ for some pair (i, j) and a fixed small constant τ. They then show that if the latency cost for leaving s is significantly more than the latency cost for entering t, this implies that the flow on an edge in collection (i, j) leaving s is at least a constant factor larger (independent of i and j) than the flow entering t on an edge in collection (i, j). As this implies that more flow leaves s than enters t, this clearly is a contradiction yielding the balance lemma. We proceed in a similar manner, except we note that the random variables under consideration are all bounded and thus by applying the law of large numbers, we may side step the discretization argument in favor of a more direct argument.

Lemma 7. *Let G be an Erdős-Rényi random graph on n vertices with edge probability p. Let $\mathcal{A}$ and $\mathcal{B}$ be reasonable distributions and let all latency functions have the form $\ell_e(f_e) = a_e f_e + b_e$ where (a_e, b_e) is distributed according to $\mathcal{A} \times \mathcal{B}$. For any sufficiently small fixed $\delta > 0$ there are some constants $c > 1$ and $n_0 > 0$ such that, if $n > n_0$ and $\mathbb{P}\Big(\mathcal{B} \leq \frac{\delta}{\log(n)}\Big)\, np \geq c\log(n)$, then for any vertex v other than s and t in the instance $(G, \ell, R_\ell(2B))$, we have $d_s(v) \leq B + 10\delta$ with high probability.*

Proof. We proceed by contradiction. Suppose that v is a vertex, other than s and t, and that $d_s(v) > B + 10\delta$. By Corollary 6, we have that for every vertex u other then s and t, $d_s(u) > B + 2\delta$. Furthermore, since for every flow carrying vertex w, $d_s(w) + d_t(w) = 2B$, this implies that every flow carrying vertex has $d_t(w) < B - 2\delta$. Now consider the flow leaving s. Let e an edge adjacent to s, then we have that $a_e f_e + b_e > B + 2\delta$. Further, this implies that if $b_e \leq B + 2\delta$, then $f_e > \frac{B + 2\delta - b_e}{a_e} > 0$. Let E_s be the set of such edges. Now for any constant $\epsilon' > 0$, we have by Chernoff bounds that $|E_S| \geq (1 - \epsilon')\mathbb{P}(\mathcal{B} \leq B + 2\delta)\, pn$ with high probability (since $B + 2\delta$ is a constant and $pn \geq c\log(n)$). Furthermore, since the range of $\mathcal{A}$ is bounded and we are concerned only with a bounded range of $\mathcal{B}$, we may apply the law of large numbers to get convergence. In particular, if we let $\mathcal{B}_{[x,y]}$ be the random variable $\mathcal{B}$ conditioned on $x \leq \mathcal{B} \leq y$, we have that

$$R_\ell(2B) \geq \sum_{e \in E_s} \frac{B + 2\delta - b_e}{a_e} \tag{22}$$

$$= \left(B + 2\delta - \mathbb{E}\left[\mathcal{B}_{[0,B+2\delta]}\right] - o(1)\right)\left(\mathbb{E}\left[\frac{1}{\mathcal{A}}\right] + o(1)\right)|E_s| \tag{23}$$

$$\geq (1 - \epsilon')\frac{B + 2\delta - \mathbb{E}\left[\mathcal{B}_{[0,B+2\delta]}\right] - o(1)}{\mathbb{E}\left[\mathcal{A}\right] + o(1)}\mathbb{P}(\mathcal{B} \leq B + 2\delta)\, pn. \tag{24}$$

Applying a similar argument to the edges adjacent to t we have that with high probability,

$$R_\ell(2B) \leq (1+\epsilon')\frac{B-2\delta-\mathbb{E}\left[\mathcal{B}_{[0,B-2\delta]}\right]+o(1)}{\mathbb{E}\left[\mathcal{A}\right]-o(1)}\mathbb{P}(\mathcal{B}\leq B-2\delta)\,pn. \tag{25}$$

Thus in order to provide the contradiction, it suffices to show that for sufficiently large n

$$\frac{(1-\epsilon')\frac{B+2\delta-\mathbb{E}\left[\mathcal{B}_{[0,B+2\delta]}\right]-o(1)}{\mathbb{E}[\mathcal{A}]+o(1)}\mathbb{P}(\mathcal{B}\leq B+2\delta)\,pn}{(1+\epsilon')\frac{B-2\delta-\mathbb{E}\left[\mathcal{B}_{[0,B-2\delta]}\right]+o(1)}{\mathbb{E}[\mathcal{A}]-o(1)}\mathbb{P}(\mathcal{B}\leq B-2\delta)\,pn} > 1. \tag{26}$$

Observing that ϵ' is arbitrary, this is equivalent to showing

$$\frac{\left(B+2\delta-\mathbb{E}\left[\mathcal{B}_{[0,B+2\delta]}\right]\right)\mathbb{P}(\mathcal{B}\leq B+2\delta)}{\left(B-2\delta-\mathbb{E}\left[\mathcal{B}_{[0,B-2\delta]}\right]\right)\mathbb{P}(\mathcal{B}\leq B-2\delta)} > 1. \tag{27}$$

We note that, for positive a,b,c, $\frac{a+c}{b+c}>1$ if and only if $\frac{a}{c}>1$. Thus, by adding $\mathbb{E}\left[\mathcal{B}_{[0,B-2\delta]}\right]\mathbb{P}(\mathcal{B}\leq B-2\delta)$ to the numerator and denominator, we have that demonstrating Equation 27, is equivalent to showing that

$$\frac{(B+2\delta)\,\mathbb{P}(\mathcal{B}\leq B+2\delta)-\mathbb{E}\left[\mathcal{B}_{[B-2\delta,B+2\delta]}\right]\mathbb{P}(|\mathcal{B}-B|\leq 2\delta)}{(B-2\delta)\,\mathbb{P}(\mathcal{B}\leq B-2\delta)} > 1 \tag{28}$$

Observing that $(B+2\delta)\,\mathbb{P}(\mathcal{B}\leq B+2\delta)\geq(B-2\delta)\,\mathbb{P}(B-2\delta)$, we can rearrange this to

$$0 < \left(B+2\delta-\mathbb{E}\left[\mathcal{B}_{[B-2\delta,B+2\delta]}\right]\right)\mathbb{P}(|\mathcal{B}-B|\leq 2\delta)+4\delta\mathbb{P}(\mathcal{B}\leq B-2\delta)\,. \tag{29}$$

But this follows immediately from the choice of B and δ, and the reasonableness of $\mathcal{B}$. □

5 Braess's Paradox Occurs in Erdős-Rényi Graphs

As in Section 2 and in Valiant and Roughgarden's work [16], in order to form the more efficient subnetwork we will classify edges adjacent to s and t as either 1-type, X-type or unclassified. We will then use these classifications to create a graph G' so that $R_\ell^{G'}(2B(1-\mu)) > R_\ell^{G}(2B)$ for some μ. In the construction that follows it will be convenient to suppose that $p \ll \frac{1}{\sqrt{n}}$, and in particular, that $\Gamma(s)\cap\Gamma(t)=\varnothing$ with high probability. We will now construct G' be partitioning the internal vertices of G into three sets and considering only edges that are induced by one of the sets or are incident to one of s and t.

To that end, let p_1 be the probability that an edge is 1-type, that is, the latency function $a_e f_e + b_e$ satisfies that $b_e \in (B,(1+\epsilon)B)$ and $a_e \leq A_1$. Similarly, define p_X as the probability that an edge is X-type, in other words, $b_e \leq \epsilon B$ and $a_e \in ((1-\epsilon)A_2, A_2)$. Now define $p^* = \min\{p_1, p_X\}$. With this notation we create three collections of vertices as follows:

– For each $v \in \Gamma(s) - \{t\}$ if the edge $\{s,v\}$ is 1-type assign v to the set V_{1X} with probability $\frac{p^*}{p_1}$. If it is X-type assign v to the set V_{X1} with probability $\frac{p^*}{p_X}$. Otherwise assign v to V_U.
– For each $u \in \Gamma(t) - \{s\}$, if the edge $\{u,t\}$ is 1-type assign u to the set V_{X1} with probability $\frac{p^*}{p_1}$. If it is X-type assign u to the set V_{1X} with probability $\frac{p^*}{p_X}$. Otherwise assign u to V_U.
– For each $v \notin \Gamma(s) \cup \Gamma(t) - \{s,t\}$ assign v uniformly at random to one of V_{1X}, V_{X1}, or V_U.

Define the graph G_{1X} as the subgraph of G induced by $V_{1X} \cup \{s,t\}$ excluding the edge $\{s,t\}$. Similarly define G_{X1} and G_U with the edge $\{s,t\}$ allowed to be present in G_U. It is important to note that each edge adjacent to s or t appears in precisely one of the three graphs. Thus, letting $G' = G_{1X} \cup G_{X1} \cup G_U$, we have $\deg_{G'}(s) = \deg_G(s)$ and $\deg_{G'}(t) = \deg_G(t)$.

Now in a similar manner as Lemma 5, Lemma 7 and Corollary 6 we have the following results

Lemma 8. *Let G be an Erdős-Rényi random graph on n vertices with edge probability p. Let $\mathcal{A}$ and $\mathcal{B}$ be reasonable distributions and let all latency functions have the form $\ell_e(f_e) = a_e f_e + b_e$ where (a_e, b_e) is distributed according to $\mathcal{A} \times \mathcal{B}$. For any sufficiently small fixed $\delta > 0$ there are some constants $c > 1$ and $n_0 > 0$ such that if $n > n_0$, $\mathbb{P}\left(\mathcal{B} \leq \frac{\delta}{\log(n)}\right) np \geq c\log(n)$, and G_{1X}, G_{X1}, G_U are defined as above, then the instance $\left(G', \ell, R_\ell^{G'}(2B(1-\mu))\right)$ satisfies that:*

1. *For any vertices v and u other than s and t, both in one of G_{1X}, G_{X1}, or G_U, $|d_s(v) - d_s(u)| \leq 8\delta$, and*
2. *for any vertex v other than s and t in G_U, $B - 10\delta \leq d_s(v)$,*

with high probability.

With this lemma in hand we can proceed to the proof of our main theorem.

Proof (Theorem 3). Our goal will be to show that there is some $\delta > 0$ and $\mu \in (0,1)$ such that $R_\ell^{G'}(2B(1-\mu)) > R_\ell(2B)$. Since the latency of a Nash equilibrium flow is strictly increasing in the flow if all the latency functions are affine by Lemma 2, this implies that $\mathcal{L}\left(G', \ell, R_\ell^{G'}(2B)\right) \leq (1-\mu)\mathcal{L}\left(G, \ell, R_\ell^G(2B)\right)$ and thus the Braess's ratio is at least $(1-\mu)^{-1}$ at the flow rate $R_\ell^G(2B)$. First, we will consider the difference in flow between G and G' over the subgraph G_U and then we will consider the differences in flow over G_{1X} and G_{X1}. For a given edge e (adjacent to s or t), denote by f_e the flow on the edge in G and denote by f'_e the flow on the edge in G'.

Since, for ϵ sufficiently small p^* is a constant that is strictly less than $\frac{1}{2}$, there is some constant c^* such that p^*pn and $(1-2p^*)pn$ are at least $c^*\log(n)$. Thus, we have that for any fixed constant $\epsilon^* > 0$, if N is the neighbor of s or T in G_{1X} or G_{X1}, then $||N| - p^*pn| \leq \epsilon p^*pn$. Similarly, if N is the neighborhood of s or t in G_U, then $||N| - (1-2p^*)pn| \leq \epsilon^*(1-2p^*)pn$ with high probability.

Suppose e is adjacent to s in G_u. Then by the Lemma 7, we have that $a_e f_e + b_e \leq B + 10\delta$, and in particular $f_e \leq \frac{B+10\delta-b_e}{a_e}$. Similarly, by Lemma 8 we have $a_e f'_e + b_e \geq B - 10\delta$ and $f'_e \geq \frac{B-10\delta-b_e}{a_e}$ and thus

$$f_e - f'_e \leq \frac{B + 10\delta - b_e}{a_e} - \frac{B - 10\delta - b_e}{a_e} = \frac{20\delta}{a_e} \leq \frac{20\delta}{A_{\min}}. \tag{30}$$

Summing over the neighbors of s in G_U we have that there is at most $(1+\epsilon^*)(1-2p^*)pn\frac{20\delta}{A_{\min}}$ more flow along those edges in G than in G'.

Now let e_s be adjacent to s in G_{1X} and let e_t be adjacent to t in G_{1X}. Then we have that

$$\begin{aligned} 2B(1-\mu) &\leq a_{e_s} f'_{e_s} + b_{e_s} + 8\delta + a_{e_t} f'_{e_t} + b_{e_t} & (31)\\ &\leq A_1 f'_{e_s} + (1+\epsilon)B + 8\delta + A_2 f'_{e_t} + \epsilon B. & (32)\end{aligned}$$

Thus $B - 2\mu B - 2\epsilon B - 8\delta \leq A_1 f'_{e_s} + A_2 f'_{e_t}$. Summing over all choices of e_s and e_t we get

$$\frac{|\Gamma_{G_{1X}}(s)|\,|\Gamma_{G_{1X}}(t)|\,(B - 2\epsilon B - 2\mu B - 8\delta)}{A_1\,|\Gamma_{G_{1X}}(t)| + A_2\,|\Gamma_{G_{1X}}(s)|} \leq R_\ell^{G_{1X}}(2B(1-\mu))\,. \tag{33}$$

In particular, $\frac{(1-\epsilon^*)^2}{1+\epsilon^*}\frac{B-2\mu B-2\epsilon B-8\delta}{A_1+A_2}p^*pn \leq R_\ell^{G_{1X}}(2B(1-\mu))$. Similarly, we have that $\frac{(1-\epsilon^*)^2}{1+\epsilon^*}\frac{B-2\mu B-2\epsilon B-8\delta}{A_1+A_2}p^*pn \leq R_\ell^{G_{X1}}(2B(1-\mu))$.

Finally, consider the flow in G on the edges adjacent to s that appear in either G_{1X} or G_{X1}. For the edges in that appear in G_{1X} the flow is at most $\frac{10\delta}{A_{\min}}$ by Lemma 7. Similarly, the flow in G for an edge that appears in G_{X1} is at most $\frac{B+10\delta}{(1-\epsilon)A_2}$.

Thus. letting $\epsilon, \epsilon^*, \delta, \mu \to 0$, we have that in this limit

$$\frac{R_\ell^{G'}(2B(1-\mu)) - R_\ell^{G}(2B)}{pn} \geq \frac{2B}{A_1 + A_2} - \frac{B}{A_2} > 0. \tag{34}$$

where the last inequality follows from the fact that $A_1 < A_2$. Thus, by continuity, for $\epsilon, \epsilon^*, \mu, \delta$ sufficiently small, $R_\ell^{G'}(2B(1-\mu)) > R_\ell^{G}(2B)$ and Braess's paradox occurs with Braess ratio $(1-\mu)^{-1}$. □

Now the proof of Valiant and Roughgarden [16] deals with the case where $p \gg n^{-\frac{1}{2}+\zeta}$ for some $\zeta > 0$ and our proof deals with the case where $p \ll n^{-\frac{1}{2}}$, leaving a small gap between these results. The only difficulty in extending our results to cover the gap is dealing with that fact that within this range of p there is a positive probability of $\Gamma(s) \cap \Gamma(t) \neq \varnothing$. However, this gap can be closed by using the more complicated partitioning scheme used by Valiant and Roughgarden which appropriately deals with vertices in $\Gamma(s) \cap \Gamma(t)$. Thus, with an appropriate choice of distribution for the latency function, there is a flow for which Braess's paradox occurs with high probability in $\mathcal{G}(n,p)$ almost down to the connectivity threshold. In fact, it is plausible that this work could be expanded to a broader class of expanders than $\mathcal{G}(n,p)$ under certain degree conditions.

References

1. Beckmann, M., McGuire, C.B., Winsten, C.B.: Studies in the economics of transportation. Yale University Press, London (1956)
2. Fisk, C., Pallottino, S.: Empirical evidence for equilibrium paradoxes with implications for optimal planning strategies. Transportation Research Part A: General 15(3), 245–248 (1981)
3. Frank, M.: The Braess paradox. Math. Programming 20(3), 283–302 (1981)
4. Hall, M.A.: Properties of the equilibrium state in transportation networks. Transportation Science 12(3), 208–216 (1978)
5. Kameda, H.: How harmful the paradox can be in the braess/cohen-kelly-jeffries networks. In: Proceedings of IEEE Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2002, vol. 1, pp. 437–445 (2002)
6. Klee, V., Minty, G.J.: How good is the simplex algorithm? In: Inequalities, III (Proc. Third Sympos., Univ. California, Los Angeles, Calif. (1969); dedicated to the memory of Theodore S. Motzkin), pp. 159–175. Academic Press, New York (1972)
7. Kolata, G.: What if they closed 42d street and nobody noticed? The New York Times (December 25, 1990)
8. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. In: Meinel, C., Tison, S. (eds.) STACS 1999. LNCS, vol. 1563, pp. 404–413. Springer, Heidelberg (1999)
9. Lin, H., Roughgarden, T., Tardos, É.: A stronger bound on Braess's paradox. In: Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms (electronic), pp. 340–341. ACM, New York (2004)
10. Papadimitriou, C.: Algorithms, games, and the internet. In: Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing (electronic), pp. 749–753. ACM, New York (2001)
11. Pas, E.I., Principio, S.L.: Braess' paradox: Some new insights. Transportation Research Part B: Methodological 31(3), 265–276 (1997)
12. Penchina, C.M.: Braess paradox: Maximum penalty in a minimal critical network. Transportation Research Part A: Policy and Practice 31(5), 379–388 (1997)
13. Roughgarden, T.: On the severity of Braess's Paradox: designing networks for selfish users is hard. J. Comput. System Sci. 72(5), 922–953 (2006)
14. Roughgarden, T., Tardos, É.: How bad is selfish routing? J. ACM (electronic) 49(2), 236–259 (2002)
15. Spielman, D.A., Teng, S.-H.: Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time. J. ACM (electronic) 51(3), 385–463 (2004)
16. Valiant, G., Roughgarden, T.: Braess's paradox in large random graphs. Random Structures Algorithms (to appear)
17. Wardrop, J.: Some theoretical aspects of road traffic research. Proceedings of the Institution of Civil Engineers, Part II 1(36), 352–362 (1952)

False-Name-Proofness in Social Networks

Vincent Conitzer[1], Nicole Immorlica[2], Joshua Letchford[1],
Kamesh Munagala[1], and Liad Wagman[3]

[1] Duke University
{conitzer,jcl,kamesh}@cs.duke.edu
[2] Northwestern University
nickle@eecs.northwestern.edu
[3] Illinios Institute of Technology
lwagman@stuart.iit.edu

Abstract. In mechanism design, the goal is to create rules for making a decision based on the preferences of multiple parties (agents), while taking into account that agents may behave strategically. An emerging phenomenon is to run such mechanisms on a social network; for example, Facebook recently allowed its users to vote on its future terms of use. One significant complication for such mechanisms is that it may be possible for a user to participate multiple times by creating multiple identities. Prior work has investigated the design of *false-name-proof* mechanisms, which guarantee that there is no incentive to use additional identifiers. Arguably, this work has produced mostly negative results. In this paper, we show that it is in fact possible to create good mechanisms that are robust to false-name-manipulation, by taking the social network structure into account. The basic idea is to exclude agents that are separated from trusted nodes by small vertex cuts. We provide key results on the correctness, optimality, and computational tractability of this approach.

1 Introduction

Recently, Facebook, Inc. decided to allow its users to vote on its future terms of use [19]. While the result was not binding,[1] this vote represents a new phenomenon that is likely to become more prominent in the future: agents participating in an election or other mechanism through a social networking site. Holding an election among the users of a social networking site introduces some issues that do not appear in regular elections. Perhaps the foremost such issue, and the one that we will focus on, is that it is generally easy for a user to create additional accounts/identities, allowing her to vote multiple times. This can compromise the legitimacy of the election and result in a suboptimal alternative being chosen.

The topic of designing elections or other mechanisms for settings where it is easy to create multiple identities and participate multiple times has already received some attention. The primary approach has been to design mechanisms that are *false-name-proof* [15,16], meaning that an agent never benefits from participating more than once.

[1] The result would have been binding if at least 30% of all active users had voted, a seemingly impossibly high turnout in this context.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 209–221, 2010.

(This is analogous to the better-known concept of *strategy-proofness*, meaning that an agent never benefits from misreporting her preferences. In fact, false-name-proofness is often defined in a way that subsumes strategy-proofness.) Unfortunately, existing results on false-name-proofness are quite negative, especially in voting contexts. For the case where additional identities can be created at zero cost, a general characterization of false-name-proof voting mechanisms has been given [5]; this characterization implies that for the special case where there are only two alternatives, the best we can do is the *unanimity* mechanism. This mechanism works as follows: if all voters agree on which alternative is better, that alternative is chosen; but if there is any disagreement (no matter in which proportions), then a fair coin is flipped to decide between the alternatives. This is an extremely negative result, since the mechanism is almost completely unresponsive to the votes.[2] Several ways to circumvent such negative results have been proposed, such as assuming that creating additional identities comes at a small cost [14] or considering a model in which it is possible to verify some of the identities [4].

These prior results do not consider any social network structure that may hold among the identities. Rather, these earlier results can be thought of as applying to settings where a user creates an account for the sole purpose of casting a vote (or bid, etc.), so that no social network structure is specified. We will show in this paper that by using the social network structure in the mechanism, it is possible to obtain much more positive results, because fake identities will look suspect in the social network (graph) structure. To give some intuition, consider John Doe, who has a legitimate account on the social networking site. In order to cast more votes, he can create several other identities (*false names*), such as Jane Jones and Jimmy Smith. Among the accounts that he controls, he can create any network structure by linking them to each other. However, if the other users behave legitimately, then he will not be able to link his additional accounts to any of the other users' identities (since, after all, they have never heard of Jane Jones or Jimmy Smith); he will only be able to get his friends to link to his legitimate identity (John Doe). This results in an odd-looking social network structure, where his legitimate identity constitutes a vertex cut in the graph, whose removal separates the fake identities from the rest of the graph.

In the remainder of this paper, we generalize the intuition afforded in the above scenario, giving a notion of when a node is "suspect" based on small vertex cuts that separate it from the trusted nodes. In Section 2, we formally define the setting that we will focus on. In Section 3, we discuss false-name-proofness and provide a sufficient condition for guaranteeing it. In Section 4, we discuss how to find all suspect nodes when trusted nodes are given exogenously to the algorithm. Then, in Section 5, we extend our analysis to settings in which we do not have trusted nodes initially, but we can actively verify nodes. We give both correctness and optimality results. The full version of this paper includes all the proofs and some additional examples, as well as

[2] The literature on false-name-proof voting mechanisms is quite recent: earlier work on false-name proofness considered other settings, such as *combinatorial auction* mechanisms, where multiple items are for sale at the same time. Unfortunately, here, too, there are strong impossibility results, including a result that states that under certain conditions, from the perspective of a worst-case efficiency ratio, it is impossible to significantly outperform the simple mechanism that sells all items as a single bundle [8].

simulation results for random graph models, in which we investigate how many vertices will typically be regarded as suspect (exogenous case) or how many need to be verified (endogenous case).

Related Work. The basic intuition that the creation of false identities in a social network results in suspiciously small vertex cuts has previously been explored in several papers, in peer-to-peer networks [18,17] and web spam detection [2,3,6,7,13].

The work on fraud in peer-to-peer networks attempts to thwart Sybil attacks in which one or more malicious users obtain multiple identities in order to out-vote legitimate users in collaborative tasks like Byzantine failure defenses. These papers propose protocols that ensure that *not too many* false identities are accepted. While this may be sufficient to thwart certain Sybil attacks in decentralized distributed systems, it can still leave incentives for an agent to create multiple identities, especially in applications such as elections in which the electorate is about evenly divided. Furthermore, a major hurdle in the Sybil attack research is that any protocol must be decentralized. In contrast, in this paper, we follow the stricter approach of guaranteeing that the creation of false identities is always weakly suboptimal, corresponding to the standard approach in the mechanism design literature. On the other hand, we allow our mechanisms to be centralized, as we envision them being run by the proprietor of the social network who has access to the network structure.

Fraud is also prevalent in the world wide web where users sometimes create fake webpages and links with the sole intent of boosting the PageRank of given website(s). Several researchers have considered using link structure to combat spam [2,3,6,7,13]. In SpamRank [2,3], the authors assume that a node is suspect if the main contribution to its PageRank is generated from a small set of supporting nodes (see also [6]). Our focus on small vertex cuts can be interpreted as an extreme version of the conditions proposed in SpamRank. An alternative approach, as taken by TrustRank [7] and Anti-TrustRank [13], assumes the existence of an oracle (e.g., a human being) which is able to determine the legitimacy of any given website. Calls to the oracle are, however, expensive, and so the main task in the protocol is to select a seed set of pages. The protocol then guesses the legitimacy of the remaining pages based on their connectivity to the seed set. In particular, the protocol assumes that legitimate pages rarely point to illegitimate ones, and hence the illegitimate pages are those that are "approximately isolated." Again, this approach is similar to our approach at a high level; the selection of the seed set corresponds to our verification policy (discussed later in the paper), and the condition of approximate isolation corresponds to the condition of small vertex cuts in our work. Despite these similarities, the particulars of the model and definitions are quite different, as these protocols are designed to combat fraudulent attacks in PageRank, whereas our goal is to prevent fraudulent attacks in voting or other mechanisms.

2 Setting

Our results can be applied to any mechanism design domain, but for the sake of concreteness, it may be helpful to think about the simple setting in which m agents must select between two alternatives. Each agent has a strict preference for one alternative over the other. The mechanism designer wishes to make a socially desirable choice, i.e., select an alternative that is beneficial for society as a whole. The majority rule, in which the alternative preferred by more voters wins, would be ideal; unfortunately, the majority rule will result in incentives to create false names, if naïvely applied.

Agents are arranged in a social network consisting of n nodes where $m \leq n$. Each agent i has a legitimate account in the social network, corresponding to a node v_i^t, as well as a (possibly empty) set of illegitimate accounts V_i^f. There is an arbitrary graph structure among the legitimate nodes in the social network—that is, we impose no structure on the subgraph induced by the legitimate nodes $\{v_i^t\}_{i \in \{1,\ldots,m\}}$.

In the most basic version of our model, we assume that no two manipulating agents can work together, so that an agent can only link her illegitmate nodes to each other and to her own legitimate node. Hence, for any $i \neq j$, there are no edges between V_i^f and $\{v_j^t\} \cup V_j^f$. However, for each agent i, we allow an arbitrary graph structure on $\{v_i^t\} \cup V_i^f$.

In the more general version of our model, we assume that up to k agents can collude together. (The basic model is the special case where $k = 1$.) That is, the agents $1, \ldots, m$ are partitioned into coalitions $S_j \subseteq \{1, \ldots, m\}$, with $|S_j| \leq k$ for each j. Let $V_{S_j}^f$ be the set of all illegitimate nodes used by S_j, that is, $V_{S_j}^f = \bigcup_{i \in S_j} V_i^f$, and let $V_{S_j}^t$ be the set of all legitimate nodes used by S_j, that is, $V_{S_j}^t = \bigcup_{i \in S_j} \{v_i^t\}$. Two distinct coalitions cannot link their illegitimate nodes to each other, so that for any $i \neq j$, there are no edges between $V_{S_i}^f$ and $V_{S_j}^t \cup V_{S_j}^f$. However, for each coalition S_i, we allow an arbitrary graph structure on $V_{S_i}^t \cup V_{S_i}^f$.

To summarize, our social network setting consists of

- a set of m agents denoted $\{1, \ldots, m\}$,
- a set of m legitimate nodes, one for each agent, denoted $V^t = \{v_1^t, \ldots, v_m^t\}$,
- a collection of m (possibly empty) sets of illegitimate nodes, one for each agent, denoted $\{V_1^f, \ldots, V_m^f\}$,
- a partition of the agents $\{1, \ldots, m\}$ into subsets S_j, where $|S_j| \leq k$ (the no-collusion case corresponds to $k = 1$), such that for any i, j, there are no edges between $V_{S_i}^f$ and $V_{S_j}^t \cup V_{S_j}^f$ (apart from this, the graph structure can be arbitrary).

Some of the nodes in the graph will be *trusted*. For example, the mechanism designer may personally know the agents corresponding to these nodes in the real world. This is a case in which trust is *exogenous*, that is, we have no control over which agents are trusted: the trusted agents are given as part of the input. Later in the paper, we will consider settings where we can, with some effort, *verify* whether any particular node is legitimate (for example, by asking the node for information that confirms that there is a corresponding agent in the real world). Nodes that pass this verification step become trusted nodes; this is a case of *endogenous* trust. It should be noted that, in either case, we do *not* assume that a trusted node will refrain from creating additional identifiers. That is, the only sense in which the node is trusted is that we know it corresponds to a real agent.

The mechanisms that we consider in this paper operate as follows. A *suspicion policy* is a function that takes as input the social network graph $G = (V, E)$ as well as a set T of trusted nodes, $T \subseteq V^t \subseteq V$; and as output labels every node in V as either "deemed legitimate" or "suspect." Generally, all the nodes in T will be deemed legitimate, but others may be deemed legitimate as well based on the network structure. Subsequently, all the nodes that have been deemed legitimate get to participate (*e.g.*, vote) in a standard mechanism f (*e.g.*, the majority rule), and based on this an outcome is chosen. (In this context, we only consider *anonymous* mechanisms that treat all nodes that get to participate identically.) In the case where nodes become trusted through verification, we also have a *verification policy* that takes G as input and determines which nodes to verify.

We consider a game played between the mechanism designer and the agents (more precisely, the coalitions S_j). First, the mechanism designer announces her mechanism, consisting of f and the suspicion policy (and, in the case where trust is obtained through verification, a verification policy). Then, each coalition S_j creates its illegitimate nodes $V^f_{S_j}$, as well as the edges that include these nodes (they can only have edges to other nodes in $V^f_{S_j}$, and to $V^t_{S_j}$). Note that the coalitions do *not* strategically determine edges between legitimate nodes in this game: in order to focus on false-name manipulation, only the creation of false nodes and their edges is modeled in the game. Also note that the mechanism designer, when announcing her mechanism, is unaware of the true graph as well as which agents are in coalitions together.

After obtaining the social network graph (and, possibly, some exogenously trusted nodes), the mechanism designer runs (1) (possibly) the verification policy and (2) the suspicion policy. The designer subsequently asks the nodes that have been deemed legitimate to report their preferences, and then finally runs (3) the standard mechanism f on these reported preferences, to obtain the outcome.

Whether this results in incentives for using false names depends on all of the components (1), (2), and (3), and each one individually can be used to make the whole mechanism false-name-proof. For example (for component 3), if f is by itself false-name-proof, then even if we verify no nodes and deem every node legitimate, there is still no incentive to engage in false-name manipulation. The downside of this approach is that we run into all the impossibility results from the literature on designing false-name-proof mechanisms. Similarly (for component 1), if we verify all nodes and then only deem the trusted nodes (the ones that passed the verification step) legitimate, there is no incentive to use false names. Of course, this generally results in far too much overhead. In this paper, we will be interested in suspicion policies (component 2) that by themselves guarantee that there is no incentive to use false names. For this, we heavily rely on the social network structure. In the first part of the paper, we do not consider verification policies—we take which nodes are trusted as given exogenously.

3 False-Name-Proofness

To define what it means for a suspicion policy to guarantee false-name-proofness, we first need to define some other properties. The next two definitions assume that a coalition can be thought of as a single player with coherent preferences; this is reasonable in the sense that if there is internal disagreement within the coalition, this will only make it more difficult for them to manipulate the mechanism.

Definition 1. *A standard mechanism f is* k-strategy-proof *if it is a dominant strategy for every coalition of size at most k to report truthfully.*

Definition 2. *A standard (anonymous) mechanism f* satisfies k-voluntary participation *if it never helps a coalition of size at most k to use fewer identifiers.*

Because the coalitions play a game with multiple stages, it is important to specify what we assume the coalitions learn about each other's actions in earlier stages—that is, what are the information sets in the extensive form of the game? Specifically, when a coalition reports its preferences to f, what does the coalition know about the nodes and edges created by other coalitions? We assume that a coalition learns nothing about other coalitions' actions, except that the coalition can (possibly) make inferences about what others have done based on which of its own nodes have been deemed legitimate. Thus, it is assumed that each coalition is rational and has perfect recall, but also that it does not have any other way of observing what other coalitions have done.

Definition 3. *We say that the* Limited Information Assumption (LIA) *holds if, for every coalition S_j, for every two nodes*[3] ν_1, ν_2 *in the extensive form of the game (where S_j is about to report preferences to f), the following holds. If S_j has taken the same node-and-edge creation actions at ν_1 and ν_2, and the same nodes have been deemed legitimate for S_j at ν_1 and ν_2, then these nodes are in the same information set—that is, S_j cannot distinguish them.*

It should be emphasized that LIA does not specify the information sets exactly—it is merely an *upper bound* on how much the coalitions learn about each other's actions. Specifically, we can also require the coalitions to report preferences for nodes *before* informing them exactly which of these nodes have been deemed legitimate. In an extreme special case of this (for which our results still hold), we can consider the situation where a coalition must create nodes and edges and report preferences for its nodes *at the same time*, making the game a single-stage game. In this case, when a coalition is reporting preferences, it clearly knows nothing about what the other coalitions have done at all, since they are moving at the same time. This is equivalent to saying that a coalition first creates nodes and edges, and then reports preferences for these nodes but without learning anything (including which of these nodes have been deemed legitimate). This is consistent with LIA: it just means that even more nodes in the game tree are in the same information set than is strictly required by LIA.

We now define what it means for a suspicion policy to guarantee false-name-proofness.

Definition 4. *A suspicion policy Π* guarantees false-name-proofness for coalitions of size at most k *if, under the LIA assumption, the following holds. For any standard (anonymous) mechanism f that is k-strategy-proof and satisfies k-voluntary participation, if we combine Π with f, then for any true social network structure on V^t, for any initial trusted nodes $T \subseteq V^t$, and for any partition of V^t into coalitions S_j of size at most k each, it is a dominant strategy for each coalition to set $V^f_{S_j} = \emptyset$ and report truthfully.*

[3] These are not to be confused with the nodes in the network.

A Sufficient Condition for Guaranteeing False-Name-Proofness. We now provide a sufficient condition for guaranteeing false-name-proofness.

Definition 5. *A suspicion policy Π is k-*robust *if, for any true social network structure on V^t, for any initial trusted nodes $T \subseteq V^t$, and for any partition of V^t into coalitions S_j of size at most k each, we have the following. For every coalition S_j, for every profile of actions taken by the other coalitions:*

1. *The actions of S_j (in terms of creating new nodes and edges) do not affect which of the other coalitions' identifiers ($V \setminus (V^t_{S_j} \cup V^f_{S_j})$) are deemed legitimate.*
2. *The number of identifiers in $V^t_{S_j} \cup V^f_{S_j}$ that are deemed legitimate is maximized by setting $V^f_{S_j} = \emptyset$.*

Theorem 1. *If a suspicion policy Π is k-robust, then it guarantees false-name-proofness for coalitions of size at most k.*

4 Exogenously Given Trusted Nodes

We begin by studying the case where the trusted nodes T are given exogenously. This could correspond to the case where the mechanism designer personally knows the owners of some of the nodes on the network, or perhaps these nodes have already been successfully verified in an earlier stage. Later in the paper, we will study the case where there are no exogenously given trusted nodes, so that we have to decide which nodes to verify. Given G and T, the next step is to determine which nodes to label as "suspect," based on the fact that they are not well connected to trusted nodes. We will make our suspicion policy precise shortly, but first we illustrate the basic idea on a small example. We recall that k denotes the maximum size of a coalition of colluding agents. Figure 1 gives an example of a network with two exogenously given trusted nodes, for the case where $k = 1$. As the figure illustrates, nodes that are separated from the trusted nodes by a vertex cut of size 1 could be false identities created by the node on the vertex cut in order to manipulate the outcome of the mechanism. Hence, they are deemed *suspect*.

In the following subsections, we first define our suspicion policy precisely and prove that it has several nice properties, including guaranteeing false-name-proofness. We then prove that this policy is optimal in the sense that any other suspicion policy with these properties would label more nodes as suspect. Finally, we give a polynomial-time algorithm for determining whether nodes are deemed legitimate or not under this policy.

The Suspicion Policy. One natural approach is to label as suspect every node v that is separated from all the trusted nodes by a vertex cut of size at most k (this cut may include some of the trusted nodes). After all, such a node v may have been artificially created by a coalition of nodes corresponding to its vertex cut. On the other hand, for a node v that is *not* separated from the trusted nodes by any vertex cut of size at most k, there is no coalition of nodes that could have artificially created v. While this reasoning is correct, it turns out that, to guarantee false-name-proofness, it is not sufficient to label *only* the nodes separated from the trusted nodes by a vertex cut of size at most k as suspect. The reason is that this approach may still leave an incentive for a coalition to

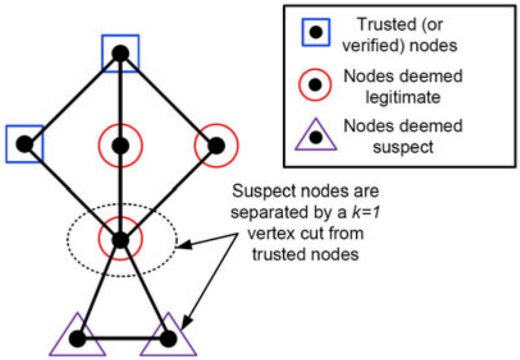


Fig. 1. Example network. The nodes correspond to identities (user accounts), and the edges correspond to (say) friendship relations between the identities. The mechanism designer, at this point for exogenous reasons, considers certain nodes "trusted" (marked by squares), that is, she is sure that they are not false names. The nodes marked with triangles are separated from the trusted nodes by a vertex cut of size one (indicated by the dotted ellipse). As a result, it is conceivable that these nodes are false names, created by the agent corresponding to the vertex-cut node; hence, they are labeled *suspect*. The remaining nodes are not separated from the trusted nodes by a vertex cut of size one, and as a result they are deemed *legitimate* (marked by circles).

create false nodes: not because these false nodes will be deemed legitimate, but rather because it may prevent *other* nodes from being labeled as suspect. We first observe a fundamental property of nodes being separated from the trusted nodes by a vertex cut of size at most k.

Lemma 1 (cf. Menger [11]). *For an initially untrusted node v, the following two statements are equivalent.*

1. *v is not separated from the initially trusted nodes by a vertex cut of size at most k (which may include initially trusted nodes).*
2. *There exist $k+1$ vertex-disjoint paths from (distinct) initially trusted nodes to v.*

The problem with the approach above is that a coalition may use false nodes that will be labeled suspect, but that help create paths to other nodes that will be deemed legitimate as a result. The solution is to apply the procedure *iteratively*, in each stage removing the nodes that are separated from all the trusted nodes by a vertex cut of size at most k, until convergence.

Definition 6. *Let r take as input $G = (V, E)$ and $T \subseteq V$, and as output produce the subgraph G' of G that results from removing those nodes in $V - T$ that are separated from the trusted nodes T by a vertex cut of size at most k (as well as removing the edges associated with these nodes). These vertex cuts are allowed to include nodes in T. Let $G = G^{(0)}, G' = G^{(1)}, G^{(2)}, \ldots, G^{(n_{G,T})}$ be the sequence of graphs that results from applying r iteratively on $(G^{(i)}, T)$, where $n_{G,T}$ is the smallest number satisfying $G^{(n_{G,T})} = G^{(n_{G,T}-1)}$ (note this sequence must converge as the set of nodes* *in successive iterations is nonincreasing). Then our suspicion policy Π_k^*, when applied to (G,T), deems all the nodes in $G^{(n_{G,T})}$ legitimate, and all the other nodes in G suspect.*

In each iteration, the procedure for computing Π_k^* removes *all* the nodes that are at that point separated from all the trusted nodes by a vertex cut of size at most k. This corresponds to eliminating nodes in a particular order. One may wonder if the result would be any different if we eliminated nodes in a different order, for example, in one iteration removing only a subset of the nodes that are at that point separated from all the trusted nodes by a vertex cut of size at most k, before continuing to the next iteration. This is analogous to the notion of path independence of iterated strict dominance in game theory: no matter in which order we eliminate strictly dominated strategies, in the end we obtain the same set of remaining strategies [10]. (This is in contrast to iterated *weak* dominance, where the order of elimination does affect the final remaining strategies.) We will show a similar path independence result for removing nodes in our setting. To do so, we first define the class of suspicion policies that correspond to *some* order; then we show that the class has only one element, namely, Π_k^*.[4]

Definition 7. *Let $\mathbf{\Pi}_k$ be the class of all suspicion policies that correspond to a procedure where:*

- *In each iteration, some subset of the nodes that are at that point separated from all the trusted nodes by a vertex cut of size at most k is eliminated from the graph;*
- *This subset must be nonempty when possible;*
- *When no additional nodes can be eliminated, the remaining nodes are exactly the ones deemed legitimate.*

Lemma 2. *The class $\mathbf{\Pi}_k$ consists of a singleton element Π_k^*,* i.e., $\mathbf{\Pi}_k = \{\Pi_k^*\}$.

We now show that our policy Π_k^* guarantees false-name-proofness for coalitions of size at most k.

Lemma 3. *Let $G = (V,E)$ be a graph and let $T \subseteq V$ be the trusted nodes. Let G' be a graph that is obtained from G by adding additional nodes V' and additional edges E' that each have at least one endpoint in V'—in such a way that every node in V' is separated from T by a vertex cut of size at most k. Then, applying Π_k^* to $G' = (V \cup V', E \cup E')$ and T results in the same nodes being deemed legitimate as applying Π_k^* to G and T.*

Theorem 2. *Π_k^* is k-robust (and hence, by Theorem 1, guarantees false-name-proofness for coalitions of size at most k). Moreover, under Π_k^*, a coalition S_j's actions also do not affect which of its own legitimate nodes $V_{S_j}^t$ are deemed legitimate. Finally, Π_k^* is guaranteed to label every illegitimate node as suspect.*

[4] The different orders of course correspond to different *procedures* for computing which nodes are deemed legitimate, but we will show that as a *function* that determines which nodes are finally deemed legitimate, they are all the same.

Optimality. We now show that Π_k^* is the best possible suspicion policy in the sense that any other policy satisfying the desirable properties in Theorem 2 must label more nodes as suspect.

Theorem 3. *Let Π' be a suspicion policy that (1) is k-robust, (2) is such that a coalition S_j's actions also do not affect which of its own legitimate nodes $V_{S_j}^t$ are deemed legitimate, and (3) is guaranteed to label every illegitimate node as suspect. Then, if Π_k^* labels a node as suspect, then so must Π'.*

Polynomial-time Algorithm for Determining Whether a Node is Suspect. In this subsection, we give a polynomial-time algorithm for determining whether nodes are deemed legitimate or suspect according to Π_k^*. The key step is to find an algorithm for figuring out which nodes are separated from the trusted nodes by a vertex cut of size at most k; then we can simply iterate this in order to execute Π_k^* (and by Lemma 2 we do not need to be careful about the order in which we eliminate nodes). It turns out that by Lemma 1, we can do this by solving a sequence of maximum flow problem instances.

Theorem 4. *Given $G = (V, E)$ and $T \subseteq V$, we can determine in polynomial time which nodes are not separated from T by a vertex cut of size at most k. As the number of iterations of Π_k^* is bounded by $|V|$, we can run Π_k^* in polynomial time.*

5 Choosing Nodes to Verify (Endogenous Trust)

Our methodology requires some nodes to be trusted. So far, we have considered settings where some nodes are trusted for exogenous reasons (for example, the organizer's own friends may be the only trusted nodes). However, we can also endogenize which nodes are trusted, by assuming that the organizer can invest some effort in *verifying* some of the identities to establish their legitimacy (for example, by asking these identities for information that identifies them in the real world). This is an approach that has been considered before in the context of false-name-proofness [4], but that prior work paid no regard to social network structure. The social network structure can drastically reduce the amount of verification required, because, as we have seen earlier in this paper, once we have some nodes that are trusted, we can infer that others are legitimate.

There are (at least) two approaches to consider here: verify enough nodes so that no suspect nodes remain at all (and try to minimize the number of verified nodes under this constraint), or try to maximize the number of nodes deemed legitimate, given a budget of verifications (say, at most b verifications). In this paper, we focus on the former.

Technically, a verification policy consists of a contingency plan, where the next node to verify depends on the results of earlier verifications of nodes (which can either fail or succeed). If a node fails the verification, that node is classified as illegitimate, and the verification continues. The verification continues until no nodes remain suspect (other than ones that failed the verification step)—that is, until no unverified nodes are separated by a vertex cut of size at most k from the nodes that were successfully verified. (This vertex cut can include successfully verified nodes. We note that in this context there is no longer a reason to *iteratively* remove nodes in the procedure that computes the trust policy (Π_k^*): because our goal is for *all* remaining nodes to be deemed legitimate, we simply need to check whether any nodes are removed in the first iteration.)

Optimally Deciding Which Nodes to Verify. We now turn to the following optimization problem: how do we minimize the number of nodes that we verify before reaching the point where all the remaining nodes are deemed legitimate? To answer this question, we first note that, since there will be no incentive to create illegitimate nodes, we can assume that all nodes will in fact be legitimate. (This does not mean that we can afford to not do the verification, because if we did not, then there would be incentives to create illegitimate nodes again.) Hence, the problem becomes to find a minimum-size subset of nodes so that no other node is separated from these nodes by a vertex cut of size at most k (which may include nodes in this subset)—or, equivalently, by Lemma 1, to find a minimum-size subset of nodes so that every other node is connected by $k + 1$ vertex-disjoint paths to (distinct nodes in) this subset.

This problem is a special case of the *source location* problem. A polynomial-time algorithm for this problem is given in a paper by Nagamochi et al. [12]. They show that the problem has a matroidal property, as follows. Instead of thinking about minimizing the number of verified nodes, we can think about maximizing the number of unverified nodes. Say a subset $U \subseteq V$ is *feasible* if, for every $v \in U$, there exist $k + 1$ vertex-disjoint (apart from v) paths to (distinct) nodes in $V \setminus U$.

Theorem 5 ([12]). *The feasible sets satisfy the independence axioms of a matroid.*

Finding an independent set of maximum size in a matroid is easy: start with an empty set, and attempt to include the elements one at a time, being careful not to violate the independence property. In the context of trying to find a minimum-size set of nodes to verify, this corresponds to starting with the set of *all* nodes, and attempting to *ex*clude the nodes one at a time, being careful that it will still result in all the excluded nodes being deemed legitimate. To check the latter, we only need to consider the current node:

Lemma 4. *Suppose $S \subseteq V$ is such that from every $u \in V - S$, there exist $k+1$ vertex-disjoint paths to (distinct nodes in) S, and suppose that for some v, $S - \{v\}$ does not have this property. Then, there do not exist $k+1$ vertex-disjoint paths from v to (distinct nodes in) $S - \{v\}$.*

This results in the following simple polynomial-time algorithm Φ_k for finding a minimum-size set of nodes to verify.

Definition 8. *Φ_k takes as input a graph $G = (V, E)$ and proceeds as follows to determine the nodes S to verify:*

1. *Initialize $S \leftarrow V$.*
2. *For each node $v \in S$: if there are $k + 1$ vertex-disjoint paths from $S - \{v\}$ to v, then remove v from S.*
3. *Return S.*

6 Conclusions and Future Research

From the above, it becomes clear that false-name-proofness, while achievable in social networking settings, does not come for free: we either cannot let all agents participate, or we must spend significant effort verifying identities. How severe these downsides are depends on the exact structure of the social network. If we have a sufficiently densely connected social network, then almost everyone can participate even when there are relatively few trusted identities, or, alternatively, we only need to verify a small number of identities to let everyone participate. But, is this likely to be the case in realistic social networks? The full version of our paper has some simulation results. Future research may also be devoted to considering some changes in the basic model and their effect on our results. What happens if agents can decide to drop edges (that is, not declare friendships) for strategic reasons? What happens if agents can get other agents to link to their fake identities at a cost? Results here may be reminiscent of those obtained in existing models where additional identifiers can be obtained at a cost [14]. What happens when we can only verify a limited number of nodes and try to maximize the number of nodes deemed legitimate?

Acknowledgements. Conitzer and Letchford were supported by NSF under award numbers IIS-0812113 and CAREER 0953756, and by an Alfred P. Sloan Research Fellowship. Munagala was supported by an Alfred P. Sloan Research Fellowship, and by NSF via CAREER award CCF-0745761 and grants CCF-1008065 and CNS-0540347. Wagman was supported by an IIT Stuart School of Business Research Grant.

References

1. Albert, R., Barabási, A.–L.: Statistical mechanics of complex networks (2001)
2. Andersen, R., Borgs, C., Chayes, J., Hopcroft, J., Mirrokni, V., Teng, S.: Local computation of PageRank contributions. In: WAW (2007)
3. Andersen, R., Borgs, C., Chayes, J.T., Hopcroft, J.E., Jain, K., Mirrokni, V.S., Teng, S.-H.: Robust PageRank and locally computable spam detection features. In: AIRWeb, pp. 69–76 (2008)
4. Conitzer, V.: Limited verification of identities to induce false-name-proofness. In: TARK, Brussels, Belgium, pp. 102–111 (2007)
5. Conitzer, V.: Anonymity-proof voting rules. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 295–306. Springer, Heidelberg (2008)
6. Gyöngyi, Z., Berkhin, P., Garcia-Molina, H., Pedersen, J.: Link spam detection based on mass estimation. In: Proceedings of VLDB. ACM, New York (2006)
7. Gyöngyi, Z., Garcia-Molina, H., Pedersen, J.: Combating web spam with TrustRank. In: Proceedings of VLDB, pp. 576–587. Morgan Kaufmann, San Francisco (2004)
8. Iwasaki, A., Conitzer, V., Omori, Y., Sakurai, Y., Todo, T., Guo, M., Yokoo, M.: Worst-case efficiency ratio in false-name-proof combinatorial auction mechanisms. In: Proceedings of AAMAS, Toronto, Canada, pp. 633–640 (2010)
9. Kleinberg, J.M.: Navigation in a small world. Nature (2000)
10. Luce, D.R., Raiffa, H.: Games and decisions: introduction and critical survey, New York (1957)
11. Menger, K.: Zur allgemeinen Kurventheorie. Fund. Math. 10, 96–115 (1927)
12. Nagamochi, H., Ishii, T., Ito, H.: Minimum cost source location problem with vertex-connectivity requirements in digraphs. Information Processing Letters 80(6), 287–293 (2001)
13. Raj, R., Krishnan, V.: Web spam detection with anti-trust rank. In: AIRWeb, pp. 381–389 (2006)

14. Wagman, L., Conitzer, V.: Optimal false-name-proof voting rules with costly voting. In: Proceedings of AAAI, Chicago, IL, USA, pp. 190–195 (2008)
15. Yokoo, M., Sakurai, Y., Matsubara, S.: Robust combinatorial auction protocol against false-name bids. Artificial Intelligence 130(2), 167–181 (2001)
16. Yokoo, M., Sakurai, Y., Matsubara, S.: The effect of false-name bids in combinatorial auctions: New fraud in Internet auctions. GEB 46(1), 174–188 (2004)
17. Yu, H., Gibbons, P.B., Kaminsky, M., Xiao, F.: SybilLimit: A near-optimal social network defense against sybil attacks. ToN 18(3), 885–898 (2010)
18. Yu, H., Kaminsky, M., Gibbons, P.B., Flaxman, A.: SybilGuard: Defending against sybil attacks via social networks. ToN 16(3), 576–589 (2008)
19. Zuckerberg, M.: Voting begins on governing the Facebook site (2009), http://blog.facebook.com/blog.php?post=76815337130

Stackelberg Strategies for Network Design Games*

Angelo Fanelli[1], Michele Flammini[2], and Luca Moscardelli[3]

[1] Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
angelo.fanelli@ntu.edu.sg
[2] Department of Computer Science, University of L'Aquila, Italy
flammini@di.univaq.it
[3] Department of Science, University of Chieti-Pescara, Italy
moscardelli@sci.unich.it

Abstract. We consider the Network Design game introduced by Anshelevich et al. [1] in which n source-destination pairs must be connected by n respective players equally sharing the cost of the used links. By considering the classical SUM social function corresponding to the total network cost, it is well known that the price of anarchy for this class of games may be as large as n. One approach for reducing this bound is that of resorting on the Stackelberg model in which for a subset of $\lfloor \alpha n \rfloor$ coordinated players, with $0 \leq \alpha \leq 1$, communication paths inducing better equilibria are fixed. In this paper we show the effectiveness of Stackelberg strategies by providing optimal and nearly optimal bounds on the performance achievable by such strategies. In particular, differently from previous works, we are also able to provide Stackelberg strategies computable in polynomial time and lowering the price of anarchy from n to $2\left(\frac{1}{\alpha}+1\right)$. Most of the results are extended to the social function MAX, in which the maximum player cost is considered.

1 Introduction

Congestion games [15] are a well established approach to model resource sharing among selfish players. In such games a set of resources is available to a set of n players. Each player comes along with a set of strategies, each corresponding to the selection of a subset of the resources. A state of the game is any combination of strategies for the players. The cost incurred by a player in a given state is defined as the sum of the costs associated with each selected resource, which depends on the number of players choosing that resource. The total cost of a state denotes its quality from a global perspective, which is typically defined as the sum of the players' costs or the maximum among the players' costs.

* This research was partially supported by the grant NRF-RF2009-08 "Algorithmic aspects of coalitional games" and by the PRIN 2008 research project COGENT (COmputational and GamE-theoretic aspects of uncoordinated NeTworks), funded by the Italian Ministry of University and Research.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 222–233, 2010.

Rosenthal [15] has shown that the natural decentralized mechanism known as Nash dynamics, in which at each step some player performs an improvement step switching his strategy to a better alternative, is guaranteed to converge to a pure *Nash equilibrium* [14], i.e., a fixed point of such dynamic in which no player can perform an improvement step. The Nash equilibrium may not necessarily minimize the total cost. The main tool for quantifying the quality of equilibria and thus the performance degradation due to the players' selfish behavior is the *price of anarchy* (PoA), introduced by Koutsoupias and Papadimitriou [12], which is formally defined as the worst-case ratio of the total cost of a Nash equilibrium to the optimal total cost.

Network Design games with fair cost allocation, introduced by Anshelevich et al. [1], are one of the most interesting subclass of congestion games. In the sequel we will refer to this class as Network Design games. In a Network Design game we are given an undirected graph with non-negative costs on the edges and, for each player, a source and a destination node. The goal of each player is to choose a path connecting his source and destination node. Thus the edges of the graph corresponds to the resources of the game and the strategy set of each player is given by the set of paths connecting the source and destination node associated to the player. The cost of each edge e is shared equally by the set of all players whose selected paths contain e. It is well known that the price of anarchy for this game may be as large as the number of players even for a simple game with two parallel edges.

A few natural approaches for reducing the price of anarchy in non-cooperative games have been investigated. An interesting one is the *Stackelberg model* [11], which consists in assuming that a central authority exploits a small fraction of coordinated players for improving the quality of the Nash equilibrium reached by the remaining selfish players. The central authority selects a fraction of players, called *coordinated players*, and assigns them to appropriately selected strategies. The algorithm adopted by the authority in selecting the coordinated players and assigning them to strategies is called *Stackelberg strategy*. Given the strategy for the coordinated players, each of the remaining players, called the *selfish players*, selects his strategy selfishly trying to minimize his cost. The behavior of selfish players leads to a *(Stackelberg) Nash equilibrium* in which none of the selfish players can improve his cost. The goal is to determine an effective Stackelberg strategy which improves the price of anarchy of the game.

Related Work. Network Design with fair cost allocation has been introduced by Anshelevich et al. [1]. In this seminal paper the authors raised the problem of the bad performance, in terms of price of anarchy, of the game due to the selfish behavior of the players. Motivated by this issue, they started to explore the middle ground between centrally enforced solutions and completely unregulated anarchy by proposing the notion of price of stability, that is the ratio of the cost of the cheapest Nash equilibrium to the cost of the optimal solution.

Korilis et al. [11] have been the first to consider the use of the Stackelberg model as a mean of improving the performance of a system. Subsequently, Roughgarden [16] considered the problem of improving the price of anarchy of non-cooperative games by means of Stackelberg strategies. Stackelberg strategies have been investigated in the context of congestion games with non-decreasing latency functions. In particular, all previous research focused on the non-atomic setting (e.g., [2,9,10,13,16,17,18]) and just recently Fotakis [7] considered atomic congestion games.

To the best of our knowledge, no work has investigated the effectiveness of Stackelberg strategies for congestion games with decreasing delay functions and in particular for Network Design games. Despite of that, several works dealt with the problem of improving the price of anarchy of Network Design games. In particular, Chen et al. [6] studied the problem of designing a different mechanism for sharing the cost of each edge to optimize the equilibrium behavior. Chekuri et al. [5] observed that the price of anarchy strongly depends on the initial state from which the players start to play. In particular they proved that the price of anarchy strongly decreases by considering only dynamics starting from "empty" states, that is a state in which no player has selected any strategy. All the results in [5] have been subsequently improved by Charikar et al. [4].

Our Contribution. In this paper we investigate Stackelberg strategies for Network Design games. To the best of our knowledge, this is the first work on Stackelberg strategy for congestion games with decreasing latency functions. In particular, we show the effectiveness of Stackelberg strategies in reducing the price of anarchy by providing optimal and nearly optimal bounds on the performance achievable for the two main social cost functions, i.e., the sum of all the players' costs (Sum) and the maximum players cost (Max). More precisely, given a subset of $\lfloor \alpha n \rfloor$ coordinated players with $0 < \alpha \leq 1$, in the case of a single source node, the price of anarchy becomes $\frac{1}{\alpha} + \frac{1}{2}$ for Sum and $\frac{2}{\alpha}$ for Max. Moreover, in the general multiple sources case, it is at most $\frac{1}{\alpha} + 1$ for Sum, that is only a subtle additive constant apart from the lower bound induced by single source, and $\frac{4}{\alpha}$ for Max.

Differently from previous works [7], we finally address the question of the selection of good Stackelberg strategies in polynomial running time. Namely, given ρ-approximation algorithms for the minimization of the two social functions, we show that in the single source case it is possible to determine in polynomial time communication paths for the coordinated players inducing a price of anarchy at most $\rho\left(\frac{1}{\alpha} + \frac{1}{2}\right)$ for Sum and $\frac{2\rho}{\alpha}$ for Max, and in the general case at most $\rho\left(\frac{1}{\alpha} + 1\right)$ for Sum and $\frac{4\rho}{\alpha}$ for Max. While to the best of our knowledge the existence of good approximation algorithms for the Max function is still an open question, for Sum this gives a polynomial time selection inducing price of anarchy at most $1.39\left(\frac{1}{\alpha} + \frac{1}{2}\right)$ for the single source case and $2\left(\frac{1}{\alpha} + 1\right)$ in the general case, by exploiting the Steiner tree and Steiner forest approximation results in [3] and [8], respectively.

The paper is structured as follows. In the next section we define the model and introduce some useful definitions. In Section 3 we show that there exist Stackelberg strategies dramatically lowering the price of anarchy from n to a value proportional to $\frac{1}{\alpha}$, both under the Sum and the Max social function. Section 4 is devoted to provide efficient, i.e., polynomial, strategies having such properties. Due to space limitations, some proofs are omitted and will appear in the full version of the paper.

2 Model and Definitions

A *Network Design game* is defined by a tuple $\mathcal{G} = \Big(N, G = (V, E), (w_e)_{e \in E}, \big((r_i, t_i) \in V^2\big)_{i \in N}\Big)$, where N is the set of players, G is an undirected graph having for each edge $e \in E$ a non-negative cost w_e and each player $i \in N$ has a pair of nodes $(r_i, t_i) \in V^2$ that he wants to connect, r_i and t_i being the source and the destination node, respectively. Notice that if r_i is the same node for every players, we are in the special case of *Single Source* Network Design game. Let Σ_i denote the strategy set of player i, with any strategy $s_i \in \Sigma_i$ of i consisting of a path connecting r_i and t_i. Let $S = (s_1, s_2, \ldots, s_n) \in \times_{i \in N} \Sigma_i$ be the strategy profile (state) in which player i chooses his strategy $s_i \in \Sigma_i$. We denote by $G(S)$ the subgraph of G composed by all edges used by all players in state S, i.e., $G(S) = \bigcup_{i \in N} s_i$. Given a strategy profile $S = (s_1, \ldots, s_n)$ and an edge $e \in E$, let $n_e(S)$ be the number of players using e in S, i.e., $n_e(S) = |\{i \in N | e \in s_i\}|$. We assume that all players using an edge equally share its cost, i.e., for each edge e and each player i using e in state S, the cost charged to player i for e is $c_i^e(S) = \frac{w_e}{n_e(S)}$. The total cost incurred by player i in S is defined as the sum of the shared costs of all edges used by i, i.e., $c_i(S) = \sum_{e \in s_i} c_i^e(S) = \sum_{e \in s_i} \frac{w_e}{n_e(S)}$. The social cost of a strategy profile S can be defined either as the sum of all the players' costs, i.e., $\textsc{Sum}(S) = \sum_{i \in N} c_i(S)$, or as the maximum among the players' costs, i.e., $\textsc{Max}(S) = \max_{i \in N} c_i(S)$. Obviously $\textsc{Sum}(S) = \sum_{e \in G(S)} w_e$, that is the cost of all the edges used by the players in S. An optimal strategy profile is one with minimum social cost, that we denote by $\textsc{Opt}_{\textsc{Sum}}$ and $\textsc{Opt}_{\textsc{Max}}$ with respect to the cost functions $\textsc{Sum}$ and $\textsc{Max}$ respectively. Obviously, if S^* is a strategy profile minimizing the function $\textsc{Sum}$, then $G(S^*)$ denotes an optimal Steiner forest, that is a forest with minimum cost connecting all nodes $\{r_i, t_i\}_{i \in N}$. Finally, there always exists a strategy profile S^* which minimizes $\textsc{Max}$ such that $G(S^*)$ is a Steiner forest; in fact, it is easy to check that, given an optimal strategy profile S'^* such that $G(S'^*)$ contains one or more cycles, they can be eliminated by obtaining a new strategy profile S^* with equal social cost such that $G(S^*)$ is a Steiner forest.

A *Stackelberg strategy* is an algorithm performed by a centralized authority that selects a subset $M \subseteq N$ of m players, called *coordinated players*, and assign them to determined strategies. We denote by $Stack(i)$ the strategy assigned to player $i \in M$. Thus, given an instance of the game, the output of a Stackelberg strategy is a pair $(M \subseteq \mathrm{N}, (Stack(i))_{i \in M})$. We assume that α is the fraction of coordinated players, that is $m = \lfloor \alpha n \rfloor$ with $\alpha \in (0, 1]$. The subset of players $N \setminus M$ is the set of the *selfish players*. Each player in $N \setminus M$ acts selfishly and aims at choosing the strategy lowering his cost, given the strategy choices of other players. Given a strategy profile S and a strategy $s_i' \in \Sigma_i$, let $(S \oplus s_i') = (s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n)$, i.e., the strategy profile obtained from S if player i changes his strategy from s_i to s_i'. A state S is a *Stackelberg Nash equilibrium* if for every player $i \in N \setminus M$ and all strategy $s_i' \in \Sigma_i$, $c_i(S) \leq c_i(S \oplus s_i')$, i.e., no player in $N \setminus M$ can improve his individual cost by unilaterally changing his strategy. The *price of anarchy* (PoA) is the ratio $\textsc{Sum}(S)/\textsc{Opt}_{\textsc{Sum}}$ (respectively $\textsc{Max}(S)/\textsc{Opt}_{\textsc{Max}}$), where S is the Stackelberg Nash equilibrium of maximum cost with respect to the considered function. The goal is to design Stackelberg strategies able to lower the price of anarchy.

3 Existence of "Good" Strategies

In this section, we prove the existence of "good" Stackelberg strategies, i.e., strategies lowering the price of anarchy from n to a value proportional to $\frac{1}{\alpha}$, for Network Design games.

We first show a lower bound to the performance of any Stackelberg strategy controlling at most $\lfloor \alpha n \rfloor$ players.

Theorem 1. *For any $\epsilon > 0$ and arbitrarily small values of α, there exists a (Single Source) Network Design game for which no Stackelberg strategy inducing a price of anarchy lower than $\frac{1}{\alpha} + \frac{1}{2} - \epsilon$ and $\frac{2}{\alpha} - \epsilon$, under social functions* Sum *and* Max *respectively, exists.*

A natural class of strategies that we define is the (α, β, S)-*deterministic scale* (DS) class with $\alpha, \beta \in (0, 1]$ and $S \in \times_{i \in N} \Sigma_i$, in which given a configuration S the goal is to control at most $\lfloor \alpha n \rfloor = |M|$ players such that, for every resource $e \in E$, at least $\lfloor \beta n_e(S) \rfloor$ players in M use e.

Lemma 1. *The* PoA *induced by any strategy in the (α, β, S)-DS class is at most $\rho\left(\frac{1}{\beta} + \frac{1}{2}\right)$ and $\frac{2\rho}{\beta}$ under social functions* Sum *and* Max*, respectively, where ρ is equal to $\frac{\textsc{Sum}(S)}{\textsc{Opt}_{\textsc{Sum}}}$ for* Sum *and to $\frac{\textsc{Max}(S)}{\textsc{Opt}_{\textsc{Max}}}$ for* Max.

We first show that there exists a strategy in the DS class for the Single-Source Network Design game.

Algorithm 1. DS for Single Source Network Design games

procedure DS-SS(state $S = (s_1, s_2, \ldots, s_n)$ such that $G(S)$ is a tree)
  $M \leftarrow \emptyset$
  Let $T = G(S)$ be the tree induced by S
  Visit the edges of T in reverse order with respect to the Breadth First Search
  for every visited edge e **do**
    Let $n_e'(S) = |\{i \in M | e \in s_i\}|$
    Let Q be a subset of $\lfloor \alpha n_e(S) \rfloor - n_e'(S)$ players in $N \setminus M$ using edge e
    $M \leftarrow M \cup Q$
  end for
  For all $i \in M$, $Stack(i) = s_i$
end procedure

Theorem 2. *For the Single Source Network Design game, the strategy defined by Algorithm 1 on input S belongs to the (α, α, S)-DS class.*

The following corollary is an immediate consequence of Lemma 1 and Theorem 2.

Corollary 1. *In the Single Source Network Design game, let S^* denote the optimal state with respect to function* SUM, *the strategy defined by Algorithm 1 on input S^* is optimal, i.e., it induces games with* PoA *at most $\frac{1}{\alpha} + \frac{1}{2}$ under the social function* SUM.

Corollary 2. *In the Single Source Network Design game, let S^* denote the optimal state with respect to function* MAX, *the strategy defined by Algorithm 1 on input S^* is optimal, i.e., it induces games with* PoA *at most $\frac{2}{\alpha}$, under the social function* MAX.

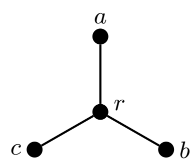


Fig. 1. The subgraph of G induced by an optimal solution.

Unfortunately, a strategy belonging to the (α, α, S)-DS class cannot be obtained for general (not only single source) Network Design games. In fact, by Figure 1 we show that it is not always possible to control at least $\alpha n_e(S)$ players for each resource e by controlling at most αn players in the whole instance. In Figure 1 we consider the subgraph $G(S)$ induced by a state S in a game with 3 players having requests (a, b),(b, c) and (c, a), respectively. Every edge is used by 2 players, thus if we consider $\alpha = \frac{1}{2}$ we must select a subset of the players such that every edge is covered by at least one of them. In order to do that, it is easy to see that we must pick at least 2 out of the 3 players, that is more than half of the total number of players.

In order to obtain an optimal deterministic strategy for the general case under the SUM social function, let us introduce another strategy, the (α, S)-*Probabilistic Scale* (PS) strategy, in which the αn players to be controlled are uniformly randomly selected and the strategy they use in the initial solution S is chosen for them.

Theorem 3. *In the Network Design game the (α, S)-PS strategy induces games with expected* PoA *at most $\rho\left(\frac{1}{\alpha} + 1\right)$ under the* SUM *social function, where $\rho = \frac{\text{SUM}(S)}{\text{OPT}_{\text{SUM}}}$.*

Proof. Let $\hat{S}$ be any Nash equilibrium for the game. We are interested in upper bounding the expected cost in $\hat{S}$ of every player $i \in N \setminus M$. In fact, players in M pay at most the cost of the initial solution S, i.e., $\sum_{i \in M} c_i(\hat{S}) \leq \text{SUM}(S)$.

For the sake of simplicity, we will consider all the already defined quantities (e.g. $c_i(S)$, $c_i^e(S)$, $\textsc{Sum}(S)$, etc.) also as random variables; it will be clear from the notation when they denote a random variable because they will be always used with the $\mathbb{E}[\cdot]$ expectation operator.

We first upper bound $\mathbb{E}[c_i^e(\hat{S})]$ for any $i \in N \setminus M$. To this aim, we introduce the random variable a_i^e indicating how much player i would pay on edge e if only coordinated players (and himself) use such an edge. Clearly, $\mathbb{E}[c_i^e(\hat{S})] \leq \mathbb{E}[a_i^e]$. Since $Pr\left(a_i^e = \frac{w_e}{x+1}\right)$ induces an hypergeometric probability distribution, $Pr\left(a_i^e = \frac{w_e}{x+1}\right) = \frac{\binom{n_e(S)}{x}\binom{n-n_e(S)}{\lfloor \alpha n \rfloor - x}}{\binom{n}{\lfloor \alpha n \rfloor}}$ and we obtain

$$\begin{aligned}
\mathbb{E}[c_i^e(\hat{S})] &\leq w_e \sum_{x=0}^{\lfloor \alpha n \rfloor} \frac{Pr\left(a_i^e = \frac{w_e}{x+1}\right)}{x+1} = \frac{w_e}{\binom{n}{\lfloor \alpha n \rfloor}} \sum_{x=0}^{\lfloor \alpha n \rfloor} \frac{\binom{n_e(S)}{x}\binom{n-n_e(S)}{\lfloor \alpha n \rfloor - x}}{x+1} \\
&= \frac{w_e}{\binom{n}{\lfloor \alpha n \rfloor}(n_e(S)+1)} \sum_{x=0}^{\lfloor \alpha n \rfloor} \binom{n_e(S)+1}{x+1}\binom{n-n_e(S)}{\lfloor \alpha n \rfloor - x} \\
&= \frac{w_e}{\binom{n}{\lfloor \alpha n \rfloor}(n_e(S)+1)} \sum_{y=1}^{\lfloor \alpha n \rfloor + 1} \binom{n_e(S)+1}{y}\binom{n-n_e(S)}{\lfloor \alpha n \rfloor + 1 - y} \\
&\leq \frac{w_e\binom{n+1}{\lfloor \alpha n \rfloor + 1}}{\binom{n}{\lfloor \alpha n \rfloor}(n_e(S)+1)} = \frac{w_e(n+1)}{(\lfloor \alpha n \rfloor + 1)(n_e(S)+1)} \\
&\leq \frac{w_e n}{\alpha n n_e(S)} = \frac{w_e}{\alpha n_e(S)}.
\end{aligned}$$

By summing over all players $i \in N$, since $\sum_{i \in M} c_i(\hat{S}) \leq C(S)$, we obtain

$$\begin{aligned}
\mathbb{E}[\textsc{Sum}(\hat{S})] &= \sum_{i \in M} \mathbb{E}[c_i(\hat{S})] + \sum_{i \in N \setminus M} \mathbb{E}[c_i(\hat{S})] \leq \textsc{Sum}(S) + \sum_{i \in N \setminus M} \sum_{e \in s_i} \mathbb{E}[c_i^e(\hat{S})] \\
&\leq \textsc{Sum}(S) + \sum_{i \in N \setminus M} \sum_{e \in s_i} \frac{w_e}{\alpha n_e(S)} = \textsc{Sum}(S) + \sum_{i \in N \setminus M} \sum_{e \in s_i} \frac{c_i^e(S) n_e(S)}{\alpha n_e(S)} \\
&= \textsc{Sum}(S) + \frac{1}{\alpha} \sum_{i \in N \setminus M} \sum_{e \in s_i} c_i^e(S) = \textsc{Sum}(S) + \frac{1}{\alpha} \sum_{i \in N \setminus M} c_i(S) \\
&\leq \textsc{Sum}(S) + \frac{1}{\alpha}\textsc{Sum}(S) = \left(1 + \frac{1}{\alpha}\right) \textsc{Sum}(S).
\end{aligned}$$

The claim follows by recalling that $\textsc{Sum}(S) \leq \rho \textsc{Opt}_{\textsc{Sum}}$. □

Let S^* be the optimal state with respect to function $\textsc{Sum}$, by considering the (α, S^*)-PS strategy, the following corollary directly follows.

Corollary 3. *In the Network Design game there exists a probabilistic strategy controlling αn players and inducing games with expected* PoA *at most* $\frac{1}{\alpha} + 1$ *under the* $\textsc{Sum}$ *social function.*

Moreover, since the expected PoA is at most $\frac{1}{\alpha}+1$, there must exist a deterministic strategy with PoA greater than the expected one, and the following corollary holds.

Corollary 4. *In the Network Design game there exists an almost optimal deterministic strategy controlling αn players and inducing games with* PoA *at most $\frac{1}{\alpha}+1$ under the* Sum *social function.*

Let us remark some interesting points. Clearly, the almost optimal deterministic strategy of Corollary 4 also holds for the particular case of Single-Source Network Design game, but unfortunately it holds only for the Sum social function. In fact, it heavily exploit the *linearity of expectation* property of random variables. Moreover, strategies in the (α, α, S)-DS class ensure a stronger property: every selfish player i pays at equilibrium at most $\frac{1}{\alpha}c_i(S)$, and any coordinated player i at most $\frac{2}{\alpha}c_i(S)$. Therefore, such strategies also work for the Max social function, and they ensure a sort of *fairness* between the players. Moreover, as it will be discussed in Section 4, the computational cost of Algorithm 1 is much lower than the one of the strategy described in the proof of Theorem 5.

Therefore, in the following we focus on the existence of a (sub-optimal) Stackelberg strategy for the general Network Design game, by requiring such a strategy being "fair" and also inducing a constant PoA under the Max social function.

In order to describe the desired strategy, we need an additional definition. Given a strategy profile $S = (s_1, \ldots, s_n)$ such that $G(S)$ is a tree, we define the split instance of the considered game as the same instance in which we have a new *split player set* N'. In particular, consider the tree $G(S)$ rooted at a generic node u; in order to build the player set N', we split each player (r_i, t_i), $i = 1, \ldots, n$ in at most two players (r_i, v_i) and (t_i, v_i), where v_i is the common ancestor in tree $G(S)$ of r_i and t_i (notice that if $v_i \equiv r_i$ or $v_i \equiv t_i$ player i is not split). Therefore, $|N| \leq |N'| \leq 2|N|$, and the *split player strategy profile* S' corresponding to S is built by associating to each player (r_i, v_i) (respectively (t_i, v_i)) in N' the strategy corresponding to the unique path connecting r_i and v_i (respectively t_i and v_i) in tree $G(S)$.

For the sake of simplicity, we will assume $\frac{2}{\alpha}$ being an integer. A more involved version of the Algorithm works for general values of α; all the details will be given in the full version of the paper.

Theorem 4. *For the Network Design game, the strategy defined by Algorithm 2 on input S belongs to the $(\alpha, \frac{\alpha}{2}, S)$-DS class.*

The following corollary is an immediate consequence of Lemma 1 and Theorem 4.

Corollary 5. *In the Network Design game, let S^* denote the optimal state with respect to function* Sum *(respectively* Max*); the strategy defined by Algorithm 2 on input S^* induces games with* PoA *at most $\frac{2}{\alpha}+\frac{1}{2}$ (respectively $\frac{4}{\alpha}$) under the social function* Sum *(respectively* Max*).*

Algorithm 2. DS for Network Design games

procedure DS(state $S = (s_1, s_2, \ldots, s_n)$ such that $G(S)$ is a tree)
  $M' \leftarrow \emptyset$
  $R \leftarrow \emptyset$
  $j \leftarrow 0$
  Let $T = G(S)$ be the tree induced by S rooted at a generic node u.
  Let N' and S' be the *split player set* and the *split player strategy profile* with respect to the tree T rooted at u, respectively. We will denote a generic player in N' as x.
  Visit the edges of T in reverse order with respect to the Breadth First Search
  for every visited edge e **do**
    Let $n_e(S') = |\{x \in N' | e \in s'_x\}|$
    Let $n'_e(S') = |\{x \in R | e \in s'_x\}|$
    while $n_e(S') - n'_e(S) \geq \frac{2}{\alpha}$ **do**
      $j \leftarrow j + 1$
      Let Q_j be a subset of $\frac{2}{\alpha}$ players in $N' \setminus R$ using edge e.
      Let $x \in Q_j$ be a player whose strategy in S' has the endpoint closest to u as close as possible to u.
      $R \leftarrow R \cup Q_j$
      $M' \leftarrow M' \cup \{x\}$
    end while
  end for
  Put in M all the players $i \in N$ such that $(r_i, v_i) \in M'$ or $(t_i, v_i) \in M'$ (or both)
  For all $i \in M$, $Stack(i) = s_i$
end procedure

4 Efficient Strategies

In this section we focus on the efficient determination of good Stackelberg strategies for Network Design games under the SUM social function.

Most of the results of Section 3 can be exploited in order to obtain efficient Stackelberg strategies that, although being sub-optimal, are able to lower the price of anarchy to a value proportional to $\frac{1}{\alpha}$.

The basic idea is that of considering a ρ-approximation $\bar{S}$ instead of an optimal solution S^*. Such an approximated solution is known to be efficiently computable in the case of the SUM social function, in which the social optimization problem is basically a minimum Steiner Forest problem and a constant approximation is easily obtainable.

In fact, given an edge-weighted graph $G = (V, E)$ and a set of couple of nodes $\{s_i, t_i\}$, $s_i, t_i \in V$, the Steiner Forest Problem consists in finding a minimum-cost forest such that all the couple of nodes of the instance are connected. In [8] a 2-approximation algorithm for such a problem has been provided, running in time $O(|V|^2 \ln |V|)$. Let $\bar{S}$ denote such a 2-approximated solution.

We remark that similar results can be obtained also for the MAX social function, as soon as the important open question of finding an algorithm efficiently approximating an optimal solution will be solved.

By Lemma 1 and Theorem 2, since, as it is easy to verify, Algorithm 1 has a worst case time complexity $O(n|E|)$, the following proposition holds.

Proposition 1. *In the Single Source Network Design game the strategy defined by Algorithm 1 on input $\hat{S}$ induces games with* PoA *at most* $\frac{2}{\alpha}+1$ *under the* Sum *social function. The time complexity of determining such a strategy is* $O(n|E| + |V|^2 \ln |V|)$.

Notice that by exploiting the 1.39-approximation algorithm for the Steiner Tree problem by Byrka et al. [3], the PoA can be lowered to $1.39\left(\frac{1}{\alpha}+\frac{1}{2}\right)$, but the running time of such an approximation algorithm fast increases as the guaranteed ratio approaches 1.39.

Now we turn our attention to the (α, S)-PS probabilistic strategy. The following proposition is an immediate consequence of Theorem 3.

Proposition 2. *In the Network Design game the* $(\alpha, \bar{S})$*-PS is a probabilistic polynomial strategy controlling* $\lfloor \alpha n \rfloor$ *players and inducing games with expected* PoA *at most* $\frac{2}{\alpha}+2$ *under the* Sum *social function. The time complexity of determining such a strategy is* $O(|V|^2 \ln |V|)$.

Starting from the (α, S)-PS strategy and applying standard *derandomization* arguments (in particular the Method of Conditional Probabilities, as Algorithm 3 does), it is possible to obtain a new deterministic strategy for Network Design games, realizing the properties claimed in Corollary 4.

We perform the derandomization with respect to a random variable, $\widehat{\text{Sum}}$, being an upper bound to the cost of any Nash equilibrium. In particular, such an upper bound is given by the costs a_i^e, $e \in s_i$, that the player i would experience on the resources he selects in S, assuming that the cost of such resources are shared only with the coordinated players: $\widehat{\text{Sum}} = \sum_{i \in N} \sum_{e \in s_i} a_i^e$.

Theorem 5. *For Network Design games, the strategy defined by Algorithm 3 on input S is a deterministic strategy controlling* $\lfloor \alpha n \rfloor$ *players and inducing games with* PoA *at most* $\rho\left(\frac{1}{\alpha}+1\right)$ *under the* Sum *social function, where* $\rho = \frac{\text{Sum}(S)}{\text{Opt}_{\text{Sum}}}$.

Algorithm 3. Derandomization

procedure Derand(state $S = (s_1, s_2, \dots, s_n)$)
  Let $q(d_1, \dots, d_i)$ be the conditional expectation $\mathbb{E}[\widehat{\text{Sum}} | (j \in M \Leftrightarrow d_j = 1) \wedge Stack(j) = s_j, 1 \le j \le i]$
  for $i \leftarrow 1, n$ **do**
    if $q(d_1, \dots, d_{i-1}, 0) < q(d_1, \dots, d_{i-1}, 1)$ **then**
      $d_i \leftarrow 0$
    else
      $d_i \leftarrow 1$
    end if
  end for
  Put in M all and only the players i for which $d_i = 1$
  For all $i \in M$, $Stack(i) = s_i$
end procedure

Proof. By the proof of Theorem 3 we know that the expected value of $\widehat{\text{SUM}}$ is at most $\left(\frac{1}{\alpha}+1\right)\text{SUM}(S)$. Therefore, there must exist a deterministic choice of $d_1,\ldots,d_n$ inducing a worst case equilibrium $\hat{S}$ such that $\text{SUM}(\hat{S}) \leq \widehat{\text{SUM}} \leq \rho\left(\frac{1}{\alpha}+1\right)\text{OPT}_{\text{SUM}}$.

It is easy to check that at any iteration of the *for* block, d_i is always chosen in such a way that the expected value of $\widehat{\text{SUM}}$ is at most $\left(\frac{1}{\alpha}+1\right)\text{SUM}(S)$. Notice that this is always possible because for any $i=1,\ldots,n$, starting from a situation in which the expected value of $\widehat{\text{SUM}}$ conditioned to the choice of $d_1,\ldots,d_{i-1}$ is at most x, there must exist a choice for d_i ($d_i=0$ or $d_i=1$) for which the expected value of $\widehat{\text{SUM}}$, conditioned also by the new choice of d_i, is at most x. □

Notice that the time complexity of Algorithm 3 depends on the computational cost of the conditional expectations q. In order to compute such expectations, techniques similar to the one exploited in the proof of Theorem 3 can be used, i.e., letting $\gamma \equiv (i \in M \Leftrightarrow d_i = 1) \wedge \mathit{Stack}(i) = s_i$, $D = |i \in N | d_i = 1|$ and $D_e = |i \in N | d_i = 1 \wedge e \in \mathit{Stack}(i)|$,

$$\begin{aligned}\mathbb{E}[\widehat{\text{SUM}}|\gamma] &= \sum_{i\in N}\sum_{e\in s_i}\mathbb{E}[a_i^e|\gamma]\\ &= \sum_{i\in M}\sum_{e\in s_i}\sum_{x=0}^{\lfloor\alpha n\rfloor - D}\frac{w_e}{D_e+x}Pr\left(a_i^e = \frac{w_e}{D_e+x}|\gamma\right)\\ &\quad + \sum_{i\in N\setminus M}\sum_{e\in s_i}\sum_{x=0}^{\lfloor\alpha n\rfloor - D}\frac{w_e}{D_e+x+1}Pr\left(a_i^e = \frac{w_e}{D_e+x+1}|\gamma\right),\end{aligned}$$

where for any $i \in M$ and $j \in N\setminus M$, $Pr\left(a_i^e = \frac{w_e}{D_e+x}|\gamma\right) = Pr\left(a_j^e = \frac{w_e}{D_e+x+1}|\gamma\right) = \frac{\binom{n_e(S)-D_e}{x}\binom{n-D-n_e(S)+D_e}{\lfloor\alpha n\rfloor - D - x}}{\binom{n-D}{\lfloor\alpha n\rfloor - D}}$.

It is easy to verify that the computation of $\mathbb{E}[a_i^e|\gamma]$ can be performed in time $O(n^2)$. Since the number of edges used in S is $O(|E|)$ and Algorithm 3 has n iterations, its worst case time complexity is $O(n^3|E|)$.

Therefore, by running Algorithm 3 on input $\bar{S}$, we obtain the following result.

Proposition 3. *In the Network Design game the Strategy induced by Algorithm 3 on input $\hat{S}$ is a deterministic polynomial strategy controlling $\lfloor\alpha n\rfloor$ players and inducing games with* PoA *at most $\frac{2}{\alpha}+2$ under the* SUM *social function. The time complexity of determining such a strategy is $O(n^3|E|+|V|^2\ln|V|)$.*

Finally, by exploiting Algorithm 2, it is possible to reduce the time complexity of the Stackelberg strategy by paying a factor almost equal to 2 in the PoA. In fact, by Lemma 1 and Theorem 4, since, at it is easy to verify, Algorithm 2 has a worst case time complexity $O(n|E|)$, the following proposition holds.

Proposition 4. *In the Network Design game the strategy defined by Algorithm 2 on input $\hat{S}$ induces games with* PoA *at most $\frac{4}{\alpha}+1$ under the* SUM *social function. The time complexity of determining such a strategy is $O(n|E|+|V|^2\ln|V|)$.*

References

1. Anshelevich, E., Dasgupta, A., Kleinberg, J.M., Tardos, É., Wexler, T., Roughgarden, T.: The price of stability for network design with fair cost allocation. SIAM J. Comput. 38(4), 1602–1623 (2008)
2. Bonifaci, V., Harks, T., Schäfer, G.: Stackelberg routing in arbitrary networks. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 239–250. Springer, Heidelberg (2008)
3. Byrka, J., Grandoni, F., Rothvoß, T., Sanità, L.: An improved lp-based approximation for steiner tree. In: STOC, pp. 583–592 (2010)
4. Charikar, M., Karloff, H.J., Mathieu, C., Naor, J., Saks, M.E.: Online multicast with egalitarian cost sharing. In: SPAA, pp. 70–76 (2008)
5. Chekuri, C., Chuzhoy, J., Lewin-Eytan, L., Naor, J., Orda, A.: Non-cooperative multicast and facility location games. In: ACM Conference on Electronic Commerce, pp. 72–81 (2006)
6. Chen, H.-L., Roughgarden, T., Valiant, G.: Designing networks with good equilibria. In: SODA, pp. 854–863 (2008)
7. Fotakis, D.: Stackelberg strategies for atomic congestion games. In: Arge, L., Hoffmann, M., Welzl, E. (eds.) ESA 2007. LNCS, vol. 4698, pp. 299–310. Springer, Heidelberg (2007)
8. Goemans, M.X., Williamson, D.P.: A general approximation technique for constrained forest problems. SIAM J. Comput. 24(2), 296–317 (1995)
9. Kaporis, A.C., Spirakis, P.G.: The price of optimum in stackelberg games on arbitrary single commodity networks and latency functions. Theor. Comput. Sci. 410(8-10), 745–755 (2009)
10. Karakostas, G., Kolliopoulos, S.G.: Stackelberg strategies for selfish routing in general multicommodity networks. Algorithmica 53(1), 132–153 (2009)
11. Korilis, Y.A., Lazar, A.A., Orda, A.: Achieving network optima using stackelberg routing strategies. IEEE/ACM Trans. Netw. 5(1), 161–173 (1997)
12. Koutsoupias, E., Papadimitriou, C.H.: Worst-case equilibria. In: Meinel, C., Tison, S. (eds.) STACS 1999. LNCS, vol. 1563, pp. 404–413. Springer, Heidelberg (1999)
13. Anil Kumar, V.S., Marathe, M.V.: Improved results for stackelberg scheduling strategies. In: Widmayer, P., Triguero, F., Morales, R., Hennessy, M., Eidenbenz, S., Conejo, R. (eds.) ICALP 2002. LNCS, vol. 2380, pp. 776–787. Springer, Heidelberg (2002)
14. Nash, J.F.: Equilibrium points in n-person games. Proceedings of the National Academy of Sciences 36, 48–49 (1950)
15. Rosenthal, R.W.: A class of games possessing pure-strategy nash equilibria. International Journal of Game Theory 2, 65–67 (1973)
16. Roughgarden, T.: Stackelberg scheduling strategies. SIAM J. Comput. 33(2), 332–350 (2004)
17. Sharma, Y., Williamson, D.P.: Stackelberg thresholds in network routing games or the value of altruism. In: ACM Conference on Electronic Commerce, pp. 93–102 (2007)
18. Swamy, C.: The effectiveness of stackelberg strategies and tolls for network congestion games. In: SODA, pp. 1133–1142 (2007)

Winner-Imposing Strategyproof Mechanisms for Multiple Facility Location Games*

Dimitris Fotakis and Christos Tzamos

School of Electrical and Computer Engineering,
National Technical University of Athens, 157 80 Athens, Greece
fotakis@cs.ntua.gr, ctzamos@gmail.com

Abstract. We study Facility Location games, where a number of facilities are placed in a metric space based on locations reported by strategic agents. A mechanism maps the agents' locations to a set of facilities. The agents seek to minimize their connection cost, namely the distance of their true location to the nearest facility, and may misreport their location. We are interested in mechanisms that are *strategyproof*, i.e., ensure that no agent can benefit from misreporting her location, do not resort to monetary transfers, and approximate the optimal social cost. We focus on the closely related problems of *k-Facility Location* and *Facility Location* with a uniform facility opening cost, and mostly study *winner-imposing* mechanisms, which allocate facilities to the agents and require that each agent allocated a facility should connect to it. We show that the winner-imposing version of the Proportional Mechanism (Lu *et al.*, EC '10) is stategyproof and $4k$-approximate for the k-Facility Location game. For the Facility Location game, we show that the winner-imposing version of the randomized algorithm of (Meyerson, FOCS '01), which has an approximation ratio of 8, is strategyproof. Furthermore, we present a deterministic non-imposing group strategyproof $O(\log n)$-approximate mechanism for the Facility Location game on the line.

1 Introduction

We consider *Facility Location games*, where a number of facilities are placed in a metric space based on the preferences of strategic agents. Such problems are motivated by natural scenarios in social choice, where the government plans to build a number of public facilities in an area. The choice of the locations is based on the preferences of local people, or *agents*. So each agent reports her ideal location, and the government applies a *mechanism* mapping the agents' preferences to a set of facility locations. The government's objective is to minimize the *social cost*, namely the total distance of the agents' locations to the nearest facility plus the *construction cost*, in case where the number of facilities is not fixed and may depend on the agents' preferences. On the other hand, the agents seek to minimize their *connection cost*, namely the distance of their ideal location to the nearest facility. In fact, an agent may report a false preference in an attempt of manipulating the mechanism. Therefore, the mechanism should be *strategyproof*, i.e., should ensure that no agent can benefit from misreporting her

* Research partially supported by an NTUA Basic Research Grant (PEBE 2009).

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 234–245, 2010.

location, or even *group strategyproof*, i.e., should ensure that for any group of agents misreporting their locations, at least one of them does not benefit. At the same time, the mechanism should achieve a reasonable approximation to the optimal social cost.

In this work, we consider two closely related facility location problems, and present computationally efficient strategyproof approximate mechanisms for both. In the k-*Facility Location* game, we place k facilities in a metric space so as to minimize the agents' total connection cost. In the *Facility Location* game, there is a uniform facility opening cost, instead of a fixed number of facilities, and we place a number of facilities in a metric space so as to minimize the sum of the agents' total connection cost and the total facility opening cost. This problem is motivated by natural scenarios where the social planner is willing to trade off the agents' connection cost against its own construction cost, so that a socially more desirable solution is achieved, and has been widely used as a natural relaxation of the k-Facility Location problem (see e.g. [6,14]).

Related Work. The problems of Facility Location and k-Facility Location, a.k.a. k-Median, are classical and have received considerable attention in Operations Research (see e.g. [14]), Approximation and Online Algorithms (see e.g. [6,21,8,13,2]), Social Choice (see e.g. [16,5,22,15,20,4,9]), and recently, Algorithmic Mechanism Design (see e.g. [19,1,11,10,18]). The related work in Social Choice mostly focuses on locating a single facility on the real line, where the agents' preferences are single-peaked. A classical result due to Moulin [16], Barberà and Jackson [5], and Sprumont [22] characterizes the class of generalized median voter schemes as the only strategyproof mechanisms when agents have single-peaked preferences on the line (see also [3,23] and [17, Chapter 10]). Schummer and Vohra [20] extended this result to tree metrics, where the class of extended median voter schemes are the only strategyproof mechanisms. On the negative side, Schummer and Vohra proved that for non-tree metrics, only dictatorial rules can be both strategyproof and onto. The problem of designing mechanisms with desirable properties for multiple facility location games has also been considered (see e.g. [15,4,9]). This line of work however does not address the issue of designing strategyproof mechanisms that approximate the optimal social cost.

Our work fits in the framework of *approximate mechanism design without money*, recently initiated by Procaccia and Tennenholtz [19]. They suggested that for optimization problems, such as 2-Facility Location on the line and 1-Facility Location on non-tree metrics, where computing the optimal solution is not strategyproof, approximation can circumvent impossibility results and yield strategyproof mechanisms that do not resort to monetary transfers. Procaccia and Tennenholtz [19] applied this approach to several location problems on the real line, and obtained strategyproof approximate mechanisms and lower bounds on the best approximation ratio achievable by a strategyproof mechanism. For the 2-Facility Location game on the line, they presented a deterministic $(n-1)$-approximate mechanism, where n is the number of agents, proved a lower bound of $3/2$ on the approximation ratio of any deterministic strategyproof mechanism, and conjectured that the lower bound for deterministic mechanisms is $\Omega(n)$.

Subsequently, Lu, Wang, and Zhou [11] improved the lower bound for deterministic mechanisms to 2, established a lower bound of 1.045 for randomized mechanisms, and presented a simple randomized $n/2$-approximate mechanism. For locating two facilities on the line, Lu, Sun, Wang, and Zhu [10] improved the lower bound for deterministic mechanisms to $(n-1)/2$, thus settling the conjecture of [19]. Moreover, they presented a deterministic $(n-1)$-approximate mechanism for locating two facilities on the circle, and proved that a natural randomized mechanism, the *Proportional Mechanism*, is strategyproof and achieves an approximation ratio of 4 for 2-Facility Location on any metric space. Unfortunately, Lu *et al.* observed that the Proportional Mechanism is not strategyproof for more than two facilities. For 1-Facility Location, Alon, Feldman, Procaccia, and Tennenholtz [1] gave an almost complete characterization of the approximation ratios achievable by randomized and deterministic strategyproof mechanisms.

Following a more general agenda, McSherry and Talwar [12] suggested the use of differentially private algorithms as almost-strategyproof approximate mechanisms. Any agent has a limited influence on the outcome of differentially private algorithm, and thus a limited incentive to lie. McSherry and Talwar presented a general (randomized exponential-time) differentially private mechanism that approximates the optimal social cost within an additive logarithmic term. Subsequently, Gupta *et al.* [7] presented computationally efficient differentially private algorithms for several combinatorial optimization problems, including (k-)Facility Location.

Building on [12], Nissim, Smorodinsky, and Tennenholtz [18] developed the only known general technique for the design of strategyproof approximate mechanisms without money. Nissim *et al.* consider *imposing mechanisms*, namely mechanisms able to restrict how agents exploit their outcome. Restricting the set of allowable post-actions for the agents, the mechanism can penalize liars. For Facility Location games in particular, an imposing mechanism requires that an agent should connect to the facility nearest to her reported location, thus increasing her connection cost if she lies. Despite being stronger, imposing mechanisms do not circumvent the lower bounds of [11]. Nissim *et al.* combined the differentially private mechanism of [12] with an imposing mechanism that penalizes lying agents, and obtained a general imposing strategyproof mechanism. As a by-product, Nissim *et al.* obtained a randomized imposing mechanism for k-Facility Location with a running time exponential in k. The mechanism approximates the optimal average connection cost, namely the optimal connection cost divided by n, within an additive term of roughly $1/n^{1/3}$. Even though the error term is diminishing as n grows, it may happen that the optimal average cost decreases much faster. In fact, for the class of instances in [11, Theorem 3], the optimal average cost is $1/n$ and the mechanism's error is at least $1/n^{1/3}$. Thus, the additive approximation guarantee of [18] does not imply any constant approximation ratio for k-Facility Location.

Contribution. Our work is motivated by the absence of any positive results on the approximability of multiple facility location games by non-imposing mechanisms, and by the recent striking result of [18] on their approximability by imposing mechanisms. In fact, the only work prior to ours that addresses approximate mechanism design for location problems with more than two facilities is [18]. Throughout this work, we restrict our attention to computationally efficient strategyproof mechanisms without money[1] and to the standard multiplicative notion of approximation. We suggest two orthogonal

[1] We consider the problems of Facility Location and k-Facility Location, with k being part of the input, which are $\mathcal{NP}$-hard. Thus one cannot directly apply VCG payments (see e.g. [17, Chapter 9]) and obtain a computationally efficient strategyproof mechanism that minimizes the social cost.

ways of relaxing approximate mechanism design for the k-Facility Location game, and show that both lead to strong positive results.

We mostly consider a natural class of imposing mechanisms, which we call *winner-imposing* mechanisms. Such a mechanism operates by allocating facilities to the agents. If an agent is allocated a facility, the facility is placed to her reported location, and the agent should connect to it. Agents not allocated a facility connect to the facility closest to their ideal location. Thus a winner-imposing mechanism penalizes a lying agent only if she succeeds in manipulating the mechanism. Moreover, the "penalty" a lying agent receives equals the distance of her ideal location to her misreported location.

In contrast to the observation of [10] that the Proportional Mechanism is not strategyproof for more than two facilities, we prove that its winner-imposing version is strategyproof for any number of facilities (cf. Lemma 1). Establishing that its approximation ratio is at most $4k$ (cf. Lemma 2), we obtain a randomized winner-imposing strategyproof $4k$-approximate mechanism for k-Facility Location, for any k.

Next we consider the Lagrangian relaxation of the k-Facility Location game, namely the Facility Location game with a uniform facility opening cost, instead of a hard constraint on the number of facilities. In fact, considering the Facility Location problem as a relaxation of k-Facility Location, a.k.a. k-Median, has been a standard and quite successful approach in the fields of Operations Research (see e.g. [14]) and Approximation Algorithms (see e.g. [6,8]).

For the Facility Location game, we first show that the winner-imposing version of Meyerson's randomized algorithm for Facility Location [13] is strategyproof (cf. Theorem 2). Combining this with [13, Theorem 2.1], we obtain a randomized winner-imposing strategyproof 8-approximate mechanism for the Facility Location game.

Moreover, we present a deterministic non-imposing mechanism for the Facility Location game on the line. The mechanism is based on a hierarchical partitioning of the line, and is motivated by the online algorithm for Facility Location on the plane by Anagnostopoulos, Bent, Upfal, and van Hentenryck [2]. We prove that the mechanism is group strategyproof (cf. Lemma 4) and $O(\log n)$-approximate (cf. Lemma 5). Notably, its approximation ratio is exponentially better than the lower bound of [10, Theorem 3.7] on the best ratio achievable by deterministic strategyproof mechanisms for the 2-Facility Location game on the line. Thus, our results demonstrate that the Facility Location game allows for some significantly (even exponentially) better approximation guarantees (by non-imposing strategyproof mechanisms) than the k-Facility Location game, and may suggest a potential connection between approximate mechanism design without money and online optimization.

We also consider (randomized) oblivious winner-imposing mechanisms, and derive a natural condition for them to be strategyproof. A mechanism is oblivious if conditional on the event that an agent is not allocated a facility, her presence has no impact on the mechanism's outcome. The Proportional Mechanism and Meyerson's algorithm are oblivious. We show that an oblivious winner-imposing mechanism for the (k-)Facility Location game on a continuous metric space is strategyproof iff it is *locally strategyproof*, i.e., no agent can benefit by reporting a location arbitrarily close to her true location (cf. Lemma 3). On the other hand, we note that local strategyproofness does not imply strategyproofness for the non-imposing version of Meyerson's algorithm.

2 Model, Definitions, and Notation

For an integer $m \geq 1$, we let $[m] = \{1, \dots, m\}$. For an event E in a sample space, we let $\mathbb{P}\mathrm{r}[E]$ be the probability of E happening. For a random variable X, we let $\mathbb{E}[X]$ be the *expectation* of X.

We assume an underlying *metric space* (M, d), where $d : M \times M \mapsto \mathbb{R}$ is the distance function, which is non-negative, symmetric, and satisfies the triangle inequality. For $x \in M$ and a non-empty $M' \subseteq M$, we let $d(x, M') = \inf\{d(x, y) : y \in M'\}$. For a location $x \in M$ and a positive real r, we let $\mathrm{Ball}(x, r) = \{y \in M : d(x, y) \leq r\}$. A metric space (M, d) is continuous if for any $x, y \in M$ with $d(x, y) \leq 2r$, there is a $z \in \mathrm{Ball}(x, r) \cap \mathrm{Ball}(y, r)$ such that $d(x, y) = d(x, z) + d(z, y)$.

For a tuple $\boldsymbol{x} = (x_1, \dots, x_n)$, we let $\boldsymbol{x}_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$ be the tuple without x_i. For a non-empty $S \subset [n]$, we let $\boldsymbol{x}_S = (x_i)_{i \in S}$ and $\boldsymbol{x}_{-S} = (x_i)_{i \in [n] \setminus S}$. We write $\boldsymbol{x} = (x_i, \boldsymbol{x}_{-i})$ and $\boldsymbol{x} = (\boldsymbol{x}_S, \boldsymbol{x}_{-S})$.

Mechanisms. Let $N = \{1, \dots, n\}$ be a set of agents. Each agent $i \in N$ has a location $x_i \in M$, which is i's private information. Next we refer to $\boldsymbol{x} = (x_1, \dots, x_n)$ as the *location profile*. A *deterministic mechanism* F maps a location profile $\boldsymbol{x}$ to a tuple of non-empty sets $(C, C^1, \dots C^n)$, where $C \subseteq M$ is the facility set of F and each $C^i \subseteq C$ contains the facilities where agent i should connect. We write $F(\boldsymbol{x})$ to denote the facility set of F and $F^i(\boldsymbol{x})$ to denote the facility subset of each agent i. For the k-Facility Location game, $|F(\boldsymbol{x})| = k$, while for the Facility Location game, $|F(\boldsymbol{x})|$ can be any positive number. A *randomized mechanism* is a probability distribution over deterministic mechanisms.

A mechanism F is *non-imposing* if for all location profiles $\boldsymbol{x}$ and all agents i, $F^i(\boldsymbol{x}) = F(\boldsymbol{x})$, and *imposing* otherwise. We only consider imposing mechanisms where each agent i can connect to the facility in $F(\boldsymbol{x})$ closest to her reported location, namely where $\{z \in F(\boldsymbol{x}) : d(x_i, z) = d(x_i, F(\boldsymbol{x}))\} \subseteq F^i(\boldsymbol{x})$ for all i. A mechanism F is said to allocate facilities to the agents[2] if $F(\boldsymbol{x}) \subseteq \{x_1, \dots, x_n\}$. A mechanism F that allocates facilities to the agents is *winner-imposing* if for every agent i, $F^i(\boldsymbol{x}) = \{x_i\}$ if $x_i \in F(\boldsymbol{x})$, and $F^i(\boldsymbol{x}) = F(\boldsymbol{x})$ otherwise. For a winner-imposing mechanism F and some location profile $\boldsymbol{x}$, we write either that F allocates a facility to agent i or that F places a facility at x_i to denote that F adds x_i in its facility set $F(\boldsymbol{x})$. Moreover, we write that F connects agent i to the facility at x_i to denote that $F^i(\boldsymbol{x}) = \{x_i\}$, as a result of $x_i \in F(\boldsymbol{x})$.

Individual and Social Cost. Given a deterministic mechanism F and a location profile $\boldsymbol{x}$, the cost of agent i is $\mathrm{cost}[x_i, F(\boldsymbol{x})] = d(x_i, F^i(\boldsymbol{x}))$. If F is a randomized mechanism, the expected cost of agent i is $\mathrm{cost}[x_i, F(\boldsymbol{x})] = \mathbb{E}_{C^i \sim F^i(\boldsymbol{x})}[d(x_i, C^i)]$.

The *social cost* for the *k-Facility Location* game of a deterministic mechanism F for a location profile $\boldsymbol{x}$ is $\mathrm{SC}_k[F(\boldsymbol{x})] = \sum_{i=1}^n d(x_i, F(\boldsymbol{x}))$, subject to the constraint that $|F(\boldsymbol{x})| = k$. For the *Facility Location* game, there is a uniform facility opening

[2] To simplify and unify the presentation, we implicitly assume here that all locations x_i are distinct. This assumption does not affect the generality of our model and our results, and can be removed by letting the mechanism map each location profile to a tuple $(C, C^1, \dots C^n)$, with $C \subseteq N$ and $C^i \subseteq C$, $C^i \neq \emptyset$, and place a facility at x_i for each $i \in C$.

cost $f > 0$, and the social cost of a deterministic mechanism F for a location profile $\boldsymbol{x}$ is $\mathrm{SC}[F(\boldsymbol{x})] = f\,|F(\boldsymbol{x})| + \sum_{i=1}^{n} d(x_i, F(\boldsymbol{x}))$. Scaling the distances appropriately, we assume that the facility opening cost is equal to 1. The expected social cost of a randomized mechanism F for a location profile $\boldsymbol{x}$ is defined by taking the expectation of $\mathrm{SC}_k[F(\boldsymbol{x})]$ (resp. $\mathrm{SC}[F(\boldsymbol{x})]$) over the distribution of $F(\boldsymbol{x})$.

A (randomized) mechanism F achieves an *approximation ratio* of $\rho \geq 1$, if for all location profiles $\boldsymbol{x}$, the (resp. expected) social cost of $F(\boldsymbol{x})$ is at most ρ times the optimal social cost for $\boldsymbol{x}$.

Strategyproofness and Group Strategyproofness. A mechanism F is *strategyproof* if for any location profile $\boldsymbol{x}$, any agent i, and any location y, $\mathrm{cost}[x_i, F(\boldsymbol{x})] \leq \mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})]$. A mechanism F is *group strategyproof* if for any location profile $\boldsymbol{x}$, any non-empty set of agents S, and any location profile $\boldsymbol{y}_S$ for them, there exists some agent $i \in S$ such that $\mathrm{cost}[x_i, F(\boldsymbol{x})] \leq \mathrm{cost}[x_i, F(\boldsymbol{y}_S, \boldsymbol{x}_{-S})]$.

3 The Winner-Imposing Proportional Mechanism

We consider the winner-imposing version of the Proportional Mechanism [10] for the k-Facility Location game. Given a location profile $\boldsymbol{x} = (x_i)_{i \in N}$, the Winner-Imposing Proportional Mechanism, or WIProp in short, works in k rounds, fixing the location of one facility in each round. For each $\ell = 1, \ldots, k$, let C_ℓ be the set of the first ℓ facilities of WIProp. Initially, $C_0 = \emptyset$. WIProp proceeds as follows:

1st Round: WIProp selects i_1 uniformly at random from N, places the first facility at x_{i_1}, connects agent i_1 to it, and lets $C_1 = \{x_{i_1}\}$.

ℓ-th Round, $\ell = 2, \ldots, k$: WIProp selects $i_\ell \in N$ with probability $\frac{d(x_{i_\ell}, C_{\ell-1})}{\sum_{i \in N} d(x_i, C_{\ell-1})}$, places the ℓ-th facility at x_{i_ℓ}, connects agent i_ℓ to it, and lets $C_\ell = C_{\ell-1} \cup \{x_{i_\ell}\}$.

The output of the mechanism is C_k, and every agent not allocated a facility is connected to the facility in C_k closest to her true location. The proof of the following theorem follows from Lemma 1 and Lemma 2 below.

Theorem 1. WIProp *is a strategyproof* $4k$*-approximation mechanism for the* k*-Facility Location game on any metric space.*

Strategyproofness. Even though the non-imposing version of the Proportional Mechanism is not strategyproof for $k \geq 3$ [10], WIProp is strategyproof for any k.

Lemma 1. *For any* $k \geq 1$*,* WIProp *is a strategyproof mechanism for the* k*-Facility Location game.*

Proof. For each $\ell = 0, 1, \ldots, k$, we let $\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i}) | C_\ell]$ be the expected connection cost of an agent i at the end of WIProp, given that i reports location y and that the facility set of WIProp at the end of round ℓ is C_ℓ. For $\ell = k$, $\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i}) | C_k] = d(x_i, C_k)$. For each $\ell = 1, \ldots, k-1$, with probability proportional to $d(y, C_\ell)$ the next facility of WIProp is placed at i's reported location, in which case i is connected to y and incurs a connection cost of $d(x_i, y)$, while for each agent $j \neq i$, with probability proportional to $d(x_j, C_\ell)$ the next facility of WIProp is placed at x_j, in which case the expected connection cost of i is $\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})|C_\ell \cup \{x_j\}]$. Therefore:

$$\mathrm{cost}[x_i,\, F(y, \boldsymbol{x}_{-i})|C_\ell] = \\ = \frac{d(x_i, y)\, d(y, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)\, \mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})|C_\ell \cup \{x_j\}]}{d(y, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)} \quad (1)$$

Similarly, for $\ell = 0$, the expected connection cost of agent i is:

$$\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})] = \frac{d(x_i, y) + \sum_{j \neq i} \mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})|\{x_j\}]}{n} \quad (2)$$

By induction on ℓ, we show that for any y, any $\ell = 0, 1, \ldots, k$, and any C_ℓ,

$$\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})|C_\ell] \geq \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell] \quad (3)$$

Thus agent i has no incentive to misreport her location, which implies the lemma.

For the basis, we observe that (3) holds for $\ell = k$. Indeed, if i's location is not in C_k, her connection cost is $d(x_i, C_k)$ and does not depend on her reported location y, while if i's location is in C_k her connection cost is $d(x_i, y) \geq d(x_i, x_i)$. We inductively assume that (3) holds for $\ell + 1$ and any facility set $C_{\ell+1}$, and show that (3) holds for ℓ and any facility set C_ℓ. If $\ell \geq 1$, we use (1) and obtain that:

$$\mathrm{cost}[x_i,\, F(y, \boldsymbol{x}_{-i})|C_\ell] \geq \\ \geq \frac{d(x_i, y)\, d(y, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)\, \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell \cup \{x_j\}]}{d(y, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)} \\ = \frac{d(x_i, y)\, d(y, C_\ell) + \left(d(x_i, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)\right) \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell]}{d(y, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)} \quad (4)$$

The inequality follows from (1) and the induction hypothesis. For the equality, we apply (1) with $y = x_i$. If $d(x_i, C_\ell) \geq d(y, C_\ell)$, (4) implies that $\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})|C_\ell] \geq \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell]$. Otherwise, we continue from (4) and obtain that:

$$\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})|C_\ell] \geq \frac{d(x_i, y) + d(x_i, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)}{d(y, C_\ell) + \sum_{j \neq i} d(x_j, C_\ell)} \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell] \\ \geq \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell]$$

For the first inequality, we use that $d(y, C_\ell) > d(x_i, C_\ell) \geq \mathrm{cost}[x_i, F(\boldsymbol{x})|C_\ell]$. For the second inequality, we use that $d(x_i, y) + d(x_i, C_\ell) \geq d(y, C_\ell)$.

If $\ell = 0$, using (2) and the induction hypothesis, we obtain that:

$$\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})] \geq \frac{1}{n} \sum_{j \neq i} \mathrm{cost}[x_i, F(\boldsymbol{x})|\{x_j\}] = \mathrm{cost}[x_i, F(\boldsymbol{x})] \quad (5)$$

Thus we have established (3) for any location y, any $\ell = 0, 1, \ldots, k$, and any C_ℓ. □

Approximation Ratio. To establish the approximation ratio, we extend the ideas of [10, Theorem 4.2] to the case where $k \geq 3$.

Lemma 2. *For any $k \geq 1$,* WIProp *achieves an approximation ratio of at most $4k$ for the k-Facility Location game.*

4 A Randomized Mechanism for Facility Location

Next we consider the winner-imposing version of Meyerson's randomized algorithm for Facility Location [13], and show that it is strategyproof. Meyerson's algorithm, or OFL in short, processes the agents one-by-one in a random order, and places a facility at the location of each agent with probability equal to her distance to the nearest facility divided by the facility opening cost (which we assume to be 1). For simplicity, we assume that the agents are indexed according to the random permutation chosen by OFL. Also we let C_i denote the facility set of OFL just after agent i is processed.

Formally, given the locations $\boldsymbol{x} = (x_i)_{i\in N}$ of a randomly permuted set of agents, the (winner-imposing) OFL mechanism first places a facility at x_1, connects agent 1 to it, and lets $C_1 = \{x_1\}$. Then, for each $i = 2, \ldots, n$, with probability $d(x_i, C_{i-1})$, OFL opens a facility at x_i, connects agent i to it, and lets $C_i = C_{i-1} \cup \{x_i\}$. Otherwise, OFL lets $C_i = C_{i-1}$. The output of the mechanism is C_n, and every agent not allocated a facility is connected to the facility in C_n closest to her true location.

Theorem 2. *The winner-imposing version of* OFL *is a strategyproof* 8*-approximation mechanism for the Facility Location game on any metric space.*

Proof. The approximation ratio follows from [13, Theorem 2.1]. Next, we show that the winner-imposing version of OFL is strategyproof for any permutation of agents.

Let i be any agent, and let x_i be i's true location. If $i = 1$ or $d(x_i, C_{i-1}) \geq 1$, OFL places a facility at x_i with certainty, so i has no incentive to lie about her location. So we restrict our attention to the case where $d(x_i, C_{i-1}) < 1$.

Let $\mathrm{cost}[x_i, F(y, x_{i+1}, \ldots, x_n)|C]$ be the expected connection cost of agent i at the end of OFL, given that i reports location y, and that just before i's location is processed, the set of facilities is C. Similarly, let $\mathrm{cost}[x_i, F(x_{i+1}, \ldots, x_n)|C]$ be the expected connection cost of agent i at the end of OFL, given that just after i's location is processed, the set of facilities is C. To establish the strategyproofness of OFL, we have to show that for any agent i located at x_i, for any location y, and for any C_{i-1},

$$\mathrm{cost}[x_i, F(x_i, x_{i+1}, \ldots, x_n)|C_{i-1}] \leq \mathrm{cost}[x_i, F(y, x_{i+1}, \ldots, x_n)|C_{i-1}] \quad (6)$$

Calculating i's expected connection cost for x_i and y, we obtain that (6) holds iff

$$(d(y, C_{i-1}) - d(x_i, C_{i-1}))\,\mathrm{cost}[x_i, F(x_{i+1}, \ldots, x_n)|C_{i-1}] \leq d(x_i, y)\, d(y, C_{i-1})$$

If $d(y, C_{i-1}) \leq d(x_i, C_{i-1})$, (6) holds because the lhs of the inequality above becomes non-positive. Otherwise, (6) holds because $d(y, C_{i-1}) - d(x_i, C_{i-1}) \leq d(x_i, y)$ and $\mathrm{cost}[x_i, F(x_{i+1}, \ldots, x_n)|C_{i-1}] \leq d(x_i, C_{i-1}) < d(y, C_{i-1})$. □

Remark. The argument above fails to establish that the non-imposing version of OFL is strategyproof. This is demonstrated by a simple instance with n agents on the real line. The first agent is located at $-1/2$, the second at 0, the third at $1/2 - \varepsilon$, for some small $\varepsilon > 0$, and the remaining $n - 3$ agents are located at 0. For appropriately chosen n and ε and the particular permutation, the second agent can improve her expected connection cost in the non-imposing version of OFL by reporting $1/2$. On the other hand, no agent has an incentive to lie if the expectation of their connection cost is also taken over all random agents' permutations. Thus, our example does not exclude the possibility that the non-imposing version of OFL is strategyproof for the Facility Location game. □

5 Oblivious Winner-Imposing Mechanisms

Next we consider the class of oblivious winner-imposing mechanisms for (k-)Facility Location, and show that they are strategyproof iff they are locally strategyproof.

A randomized mechanism F that allocates facilities to the agents is *oblivious* if for any location profile $\boldsymbol{x} = (x_i)_{i \in N}$, any agent i, and any location y (y may be x_i),

$$\mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i}) | i \notin F(y, \boldsymbol{x}_{-i})] = \overline{d}(x_i, F(\boldsymbol{x}_{-i}))\,,$$

where $\overline{d}(x_i, F(\boldsymbol{x}_{-i}))$ is the expected distance of x_i to the nearest facility in $F(\boldsymbol{x}_{-i})$, i.e., F's outcome on the locations of all agents other than i. Namely, F is oblivious if conditional on the event that an agent i is not allocated a facility, her presence has no impact on F's outcome. WIProp and OFL are oblivious mechanisms.

A mechanism F is *locally strategyproof* for the (k-)Facility Location game if there exists an $r > 0$, such that for any location profile $\boldsymbol{x} = (x_i)_{i \in N}$, any agent i, and any $y \in \mathrm{Ball}(x_i, r)$, $\mathrm{cost}[x_i, F(\boldsymbol{x})] \leq \mathrm{cost}[x_i, F(y, \boldsymbol{x}_{-i})]$.

Lemma 3. *Let F be an oblivious winner-imposing mechanism for (k-)Facility Location on a continuous metric space. Then F is locally strategyproof iff it is strategyproof.*

Proof. Clearly, any strategyproof mechanism is locally strategyproof. For the other direction, let $\boldsymbol{x} = (x_i)_{i \in N}$ be any location profile. For any agent i and any location x, we let $p(x) = \mathbb{P}\mathrm{r}[i \in F(x, \boldsymbol{x}_{-i})]$ be the probability that i is allocated a facility by F if she reports location x. Similarly to the proof of Theorem 2, we observe that F is strategyproof iff for any agent i with true location x_i and any location y,

$$p(y)\,(\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(x_i, y)) \leq p(x_i)\,\overline{d}(x_i, F(\boldsymbol{x}_{-i})) \tag{7}$$

We show that if F is locally strategyproof for some $r > 0$, (7) holds for any location y.

Let i be any agent. If $r \geq \overline{d}(x_i, F(\boldsymbol{x}_{-i}))$, (7) holds for any location y, since any $y \notin \mathrm{Ball}(x_i, r)$ makes its lhs non-positive. Otherwise, we show that (7) holds for any location $y \in \mathrm{Ball}(x_i, 2r)$. Let $y \in \mathrm{Ball}(x_i, 2r) \setminus \mathrm{Ball}(x_i, r)$. Since the metric space is continuous, there is a $z \in \mathrm{Ball}(x_i, r) \cap \mathrm{Ball}(y, r)$ with $d(x_i, y) = d(x_i, z) + d(z, y)$. If $d(x_i, y) \geq \overline{d}(x_i, F(\boldsymbol{x}_{-i}))$, (7) holds because its lhs is non-positive. Otherwise,

$$\begin{aligned} p(y) \leq\ & p(z)\,\frac{\overline{d}(z, F(\boldsymbol{x}_{-i}))}{\overline{d}(z, F(\boldsymbol{x}_{-i})) - d(z, y)} \leq p(z)\,\frac{\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(x_i, z)}{\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(z, y) - d(x_i, z)} \\ =\ & p(z)\,\frac{\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(x_i, z)}{\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(x_i, y)} \leq p(x_i)\,\frac{\overline{d}(x_i, F(\boldsymbol{x}_{-i}))}{\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(x_i, y)} \end{aligned}$$

For the first inequality, we use that F is locally strategyproof for r, and apply (7) for locations z, y. For the first two inequalities, since $\overline{d}(x_i, F(\boldsymbol{x}_{-i})) > d(x_i, y)$, we have that $\overline{d}(z, F(\boldsymbol{x}_{-i})) - d(z, y) > 0$, that $\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(x_i, z) > 0$, and that $\overline{d}(x_i, F(\boldsymbol{x}_{-i})) - d(z, y) - d(x_i, z) > 0$. For the last inequality, we use that F is locally strategyproof for r, and apply (7) for locations x_i, z. □

Remark. OFL is locally strategyproof for any permutation of agents and r equal to the minimum distance separating two different locations. On the other hand, we presented an instance where for certain permutations, the non-imposing version of OFL allows an agent to improve her expected cost by misreporting her location. Thus local strategyproofness does not imply strategyproofness for non-imposing OFL. □

6 A Deterministic Mechanism for Facility Location on the Line

We present a deterministic non-imposing group strategyproof $O(\log n)$-approximate mechanism for Facility Location on the real line. To simplify the presentation, we assume that the facility opening cost is 1, and that the agents are located in $\mathbb{R}_+ = [0, \infty)$.

The *Line Partitioning* mechanism, or LPart in short, is motivated by the online algorithm of [2] for Facility Location on the plane. LPart assumes a hierarchical partitioning of $[0, \infty)$ with at most $1 + \log_2 n$ levels. The partitioning at level 0 consists of intervals of length 1. Namely, for $p = 0, 1, \ldots$, the p-th level-0 interval is $[p, p+1)$. Each level-ℓ interval $[p\,2^{-\ell}, (p+1)2^{-\ell})$, $\ell = 0, 1, \ldots, \lfloor \log_2 n \rfloor - 1$, is partitioned into two disjoint level-$(\ell+1)$ intervals of length $2^{-(\ell+1)}$, namely $[p\,2^{-\ell}, p\,2^{-\ell} + 2^{-(\ell+1)})$ and $[p\,2^{-\ell} + 2^{-(\ell+1)}, (p+1)\,2^{-\ell})$. A level-0 interval is active if it includes the (reported) location of at least one agent. A level-ℓ interval, $\ell \geq 1$, is *active* if it includes the locations of at least $2^{\ell+1}$ agents, and *inactive* otherwise. Intuitively, an interval is active if it includes so many agents that the optimal solution opens a facility nearby.

LPart opens three facilities, two at the endpoints and one at the midpoint, of each level-0 active interval, and one facility at the midpoint of each level-ℓ active interval, for each $\ell \geq 1$. In particular, for each level-0 active interval $[p, p+1)$, LPart opens three facilities at p, at $p+\frac{1}{2}$, and at $p+1$. For each $\ell \geq 1$ and each level-ℓ active interval $[p\,2^{-\ell}, (p+1)2^{-\ell})$, LPart opens a facility at $p\,2^{-\ell} + 2^{-(\ell+1)}$. LPart is non-imposing, so each agent is connected to the open facility closest to her true location.

Theorem 3. LPart *is a group strategyproof* $O(\log n)$*-approximate mechanism for the Facility Location game on the real line.*

Proof. We start with some observations regarding the structure of the solution produced by LPart. We observe that if an interval q is active, any interval containing q is active, while if an interval q is inactive, any interval included in q is inactive as well. Moreover, all level-$\lfloor \log_2 n \rfloor$ intervals are inactive, since each of them contains less than $2^{\lfloor \log_2 n \rfloor + 1}$ agents. So each agent is included in at least one active and at least one inactive interval. In the following, each agent i is associated with the maximal (i.e., that of the smallest level) inactive interval, denoted q_i, that contains her true location. The maximal inactive intervals q_i, q_j of two agents i, j either coincide with each other or are disjoint.

A simple induction shows that each active interval q has three open facilities, two at its endpoints and one at its midpoint. Moreover, if an active level-ℓ interval contains an inactive level-$(\ell+1)$ subinterval q', q' has two open facilities at its endpoints. Therefore, the connection cost of each agent i is equal to the distance of her true location to the nearest endpoint of her maximal inactive interval q_i. Furthermore, i's connection cost is at least as large as the distance of her true location to the nearest endpoint of any inactive interval containing her true location.

Group Strategyproofness. The above properties of LPart immediately imply that:

Lemma 4. LPart *is group strategyproof.*

Proof. Let $S \subseteq N$, $S \neq \emptyset$, be any coalition of agents who misreport their locations so as to improve their connection cost, and let $\boldsymbol{x}_S = (x_i)_{i \in S}$ and $\boldsymbol{y}_S = (y_i)_{i \in S}$ be the profiles with their true and their misreported locations respectively. If for some agent i,

i's maximal inactive interval q_i contains the same number of agents in $\mathrm{LPart}(\boldsymbol{x}_S, \boldsymbol{x}_{-S})$ and in $\mathrm{LPart}(\boldsymbol{y}_S, \boldsymbol{x}_{-S})$, q_i is inactive in $\mathrm{LPart}(\boldsymbol{y}_S, \boldsymbol{x}_{-S})$ as well, and i's connection cost does not improve. On the other hand, if q_i contains more agents in $\mathrm{LPart}(\boldsymbol{y}_S, \boldsymbol{x}_{-S})$ than in $\mathrm{LPart}(\boldsymbol{x}_S, \boldsymbol{x}_{-S})$, there are some agents in S whose maximal inactive interval is disjoint to q_i in $\mathrm{LPart}(\boldsymbol{x}_S, \boldsymbol{x}_{-S})$ and is included in q_i in $\mathrm{LPart}(\boldsymbol{y}_S, \boldsymbol{x}_{-S})$. Therefore, there is some agent $j \in S$ whose maximal inactive interval q_j contains less agents in $\mathrm{LPart}(\boldsymbol{y}_S, \boldsymbol{x}_{-S})$ than in $\mathrm{LPart}(\boldsymbol{x}_S, \boldsymbol{x}_{-S})$. Thus q_j is inactive in $\mathrm{LPart}(\boldsymbol{y}_S, \boldsymbol{x}_{-S})$ as well, and j's connection cost does not improve due to the deviation of S. □

Approximation Ratio. We proceed along the lines of [2, Theorem 1]. We first show that the optimal solution has a facility close to each active interval.

Proposition 1. *Let $q = [p\,2^{-\ell}, (p+1)2^{-\ell})$ be an active level-ℓ interval, for some $\ell \geq 0$. Then, the optimal solution has a facility in $[(p-1)2^{-\ell}, (p+2)2^{-\ell})$.*

Proof. Let $q_l = [(p-1)2^{-\ell}, p\,2^{-\ell})$ be the interval next to q on the left, let $q_r = [(p+1)2^{-\ell}, (p+2)2^{-\ell})$ be the interval next to q on the right, and let n_q be the number of agents in q. For sake of contradiction, we assume that the optimal solution does not have a facility in $q_l \cup q \cup q_r$. Then the connection cost of the agents in q is greater than $n_q 2^{-\ell}$. If $\ell = 0$, placing an optimal facility at the location of some agent in q costs 1 and decreases the connection cost of the agents in q to at most $n_q - 1$. If $\ell \geq 1$, placing an optimal facility at the midpoint of q decreases the connection cost of the agents in q to at most $n_q 2^{-(\ell+1)}$. Since q is active and $n_q \geq 2^{\ell+1}$ ($n_q \geq 1$ for $\ell = 0$), the total cost in the later case is less than the connection cost of the agents in q to a facility outside $q_l \cup q \cup q_r$, a contradiction. □

Lemma 5. LPart *has an approximation ratio of* $O(\log n)$.

Proof. Let k be the number of facilities in the optimal solution. By Proposition 1, there are at most 3 active intervals per optimal facility at each level $\ell = 0, 1, \ldots, \lfloor \log n \rfloor - 1$. The total facility cost for the three (neighboring) active level-0 intervals is 7, and the facility cost for each active level-ℓ interval, $\ell \geq 1$, is 1. Therefore, the number of active intervals is at most $3k \log_2 n$, and the total facility cost of LPart is at most $4k + 3k \log_2 n$.

To bound the connection cost of LPart, we consider the set of maximal inactive intervals that include the location of at least one agent (i.e., they are non-empty). This accounts for the connection cost of all agents, since each agent i is associated with her maximal inactive interval q_i. Each maximal inactive interval q at level ℓ, $\ell \geq 1$, contains less than $2^{\ell+1}$ agents and has two facilities at its endpoints. Thus the total connection cost for the agents in q is at most $2^{\ell+1}2^{-\ell}/2 = 1$. Furthermore, q is included in some active level-($\ell - 1$) interval. Thus, the total number of non-empty maximal inactive intervals, and thus the total connection cost of LPart, is at most $6k \log_2 n$. Overall, the total cost of LPart is at most $4k + 9k \log_2 n$, i.e. $O(\log_2 n)$ times the optimal cost. □

Acknowledgements: We wish to thank Angelina Vidali for helpful discussions and for bringing [18] to our attention.

References

1. Alon, N., Feldman, M., Procaccia, A.D., Tennenholtz, M.: Strategyproof approximation of the minimax on networks. Mathematics of Operations Research (to appear, 2010)
2. Anagnostopoulos, A., Bent, R., Upfal, E., van Hentenryck, P.: A simple and deterministic competitive algorithm for online facility location. Information and Computation 194, 175–202 (2004)
3. Barberà, S.: An introduction to strategyproof social choice functions. Social Choice and Welfare 18, 619–653 (2001)
4. Barberà, S., Beviá, C.: Locating public libraries by majority: Stability, consistency and group formation. Games and Economic Behaviour 56, 185–200 (2006)
5. Barberà, S., Jackson, M.: A characterization of strategy-proof social choice functions for economies with pure public goods. Journal of Economic Theory 61, 262–289 (1994)
6. Guha, S.: Approximation Algorithms for Facility Location Problems. PhD thesis, Stanford University (2000)
7. Gupta, A., Ligett, K., McSherry, F., Roth, A., Talwar, K.: Differentially private combinatorial optimization. In: Proc. of the 21st ACM-SIAM Symposium on Discrete Algorithms, SODA 2010 (2010)
8. Jain, K., Vazirani, V.: Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and Lagrangian relaxation. Journal of the ACM 48(2), 274–296 (2001)
9. Ju, B.-G.: Efficiency and consistency for locating multiple public facilities. Journal of Economic Theory 138, 165–183 (2008)
10. Lu, P., Sun, X., Wang, Y., Zhu, Z.A.: Asymptotically optimal strategy-proof mechanisms for two-facility games. In: Proc. of the 11th ACM Conference on Electronic Commerce (EC 2010), pp. 315–324 (2010)
11. Lu, P., Wang, Y., Zhou, Y.: Tighter bounds for facility games. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 137–148. Springer, Heidelberg (2009)
12. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: Proc. of the 48th IEEE Symposium on Foundations of Computer Science (FOCS 2007), pp. 94–103 (2007)
13. Meyerson, A.: Online facility location. In: Proc. of the 42nd IEEE Symposium on Foundations of Computer Science (FOCS 2001), pp. 426–431 (2001)
14. Mirchandani, P.B., Francis, R.L. (eds.): Discrete Location Theory. Wiley, Chichester (1990)
15. Miyagawa, E.: Locating libraries on a street. Social Choice and Welfare 18, 527–541 (2001)
16. Moulin, H.: On strategy-proofness and single-peakedness. Public Choice 35, 437–455 (1980)
17. Nisan, N., Roughgarden, T., Tardos, É., Vazirani, V.: Algorithmic Game Theory. Cambridge University Press, Cambridge (2007)
18. Nissim, K., Smorodinsky, R., Tennenholtz, M.: Approximately optimal mechanism design via differential privacy. CoRR abs/1004.2888 (2010)
19. Procaccia, A.D., Tennenholtz, M.: Approximate mechanism design without money. In: Proc. of the 10th ACM Conference on Electronic Commerce (EC 2009), pp. 177–186 (2009)
20. Schummer, J., Vohra, R.V.: Strategyproof location on a network. Journal of Economic Theory 104, 405–428 (2002)
21. Shmoys, D.: Approximation algorithms for facility location problems. In: Jansen, K., Khuller, S. (eds.) APPROX 2000. LNCS, vol. 1913, pp. 27–33. Springer, Heidelberg (2000)
22. Sprumont, Y.: The division problem with single-peaked preferences: A characterization of the uniform allocation rule. Econometrica 49, 509–519 (1991)
23. Sprumont, Y.: Strategyproof collective choice in economic and political environments. The Canadian Journal of Economics 28(1), 68–108 (1995)

Constrained Non-monotone Submodular Maximization: Offline and Secretary Algorithms

Anupam Gupta[1], Aaron Roth[2], Grant Schoenebeck[3], and Kunal Talwar[4]

[1] Carnegie Mellon University, Pittsburgh, PA
anupamg@cs.cmu.edu
[2] Microsoft Research New England, Cambridge, MA
alroth@cs.cmu.edu
[3] Princeton University, Princeton, NJ
gschoene@princeton.edu
[4] Microsoft Research, Mountain View, CA
kunal@microsoft.com

Abstract. Constrained submodular maximization problems have long been studied, most recently in the context of auctions and computational advertising, with near-optimal results known under a variety of constraints when the submodular function is *monotone*. In this paper, we give constant approximation algorithms for the *non-monotone* case that work for *p-independence systems* (which generalize constraints given by the intersection of p matroids that had been studied previously), where the running time is $\text{poly}(n, p)$. Our algorithms and analyses are simple, and essentially reduce non-monotone maximization to multiple runs of the greedy algorithm previously used in the monotone case.

We extend these ideas to give a simple greedy-based constant factor algorithms for non-monotone submodular maximization subject to a *knapsack constraint*, and for *(online) secretary setting* (where elements arrive one at a time in random order and the algorithm must make irrevocable decisions) subject to uniform matroid or a partition matroid constraint. Finally, we give an $O(\log k)$ approximation in the secretary setting subject to a general matroid constraint of rank k.

1 Introduction

We present algorithms for maximizing (not necessarily monotone) non-negative submodular functions satisfying $f(\emptyset) = 0$ under a variety of constraints considered earlier in the literature. Lee et al. [28,29] gave the first algorithms for these problems via local-search algorithms: in this paper, we consider greedy approaches that have been successful for *monotone* submodular maximization, and show how these algorithms can be adapted very simply to non-monotone maximization as well. Using this idea, we show the following results:

– We give an $O(p)$-approximation for maximizing submodular functions subject to a p-independence system. This extends the result of Lee et al. [28,29] which applied to constraints given by the intersection of p matroids, where p was a constant. (Intersections of p matroids give p-indep. systems, but the converse is not true.) Our

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 246–257, 2010.

greedy-based algorithm has a run-time polynomial in p, and hence gives the first polynomial-time algorithms for non-constant values of p.

- We give a constant-factor approximation for maximizing submodular functions subject to a knapsack constraint. This greedy-based algorithm gives an alternate approach to solve this problem; Lee et al. [28] gave LP-rounding-based algorithms that achieved a $(5 + \epsilon)$-approximation algorithm for constraints given by the intersection of p knapsack constraints, where p is a constant.

Armed with simpler greedy algorithms for nonmonotone submodular maximization, we are able to perform constrained nonmonotone submodular maximization in several special cases in the secretary setting as well: when items arrive online in random order, and the algorithm must make irrevocable decisions as they arrive.

- We give an $O(1)$-approximation for maximizing submodular functions subject to a cardinality constraint and subject to a partition matroid. (Using a reduction of [4], the latter implies $O(1)$-approximations to e.g., graphical matroids.) Our secretary algorithms are simple and efficient.
- We give an $O(\log k)$-approximation for maximizing submodular functions subject to an arbitrary rank k matroid constraint. This matches the known bound for the *matroid secretary problem*, in which the function to be maximized is simply linear.

No prior results were known for submodular maximization in the secretary setting, even for *monotone* submodular maximization; there is some independent work, see §1.3 for details.

Compared to previous offline results, we trade off small constant factors in our approximation ratios of our algorithms for exponential improvements in run time: maximizing nonmonotone submodular functions subject to (constant) $p \geq 2$ matroid constraints currently has a $(\frac{p^2}{p-1} + \epsilon)$ approximation due to a paper of Lee, Sviridenko and Vondrák [29], using an algorithm with run-time exponential in p. For $p = 1$ the best result is a 3.23-approximation by Vondrák [34]. In contrast, our algorithms have run time only linear in p, but our approximation factors are worse by constant factors for the small values of p where previous results exist. We have not tried to optimize our constants, but it seems likely that matching, or improving on the previous results for constant p will need more than just choosing the parameters carefully. We leave such improvements as an open problem.

1.1 Submodular Maximization and Secretary Problems in an Economic Context

Submodular maximization and secretary problems have both been widely studied in their economic contexts. The problem of selecting a subset of people in a social network to maximize their influence in a viral marketing campaign can be modeled as a constrained submodular maximization problem [21,30]. When costs are introduced, the influence minus the cost gives us *non-monotone* submodular maximization problems; prior to this work, *online* algorithms for non-monotone submodular maximization problems were not known. Asadpour et al. studied the problem of adaptive stochastic (monotone) submodular maximization with applications to budgeting and sensor placement [2], and Agrawal et al. showed that the *correlation gap* of submodular functions was bounded by a constant using an elegant cost-sharing argument, and related this result to social welfare maximizing auctions [1]. Finally, secretary problems, in which elements arriving in random order must be selected so as to maximize some constrained objective function have well-known connections to online auctions [23,5,3,18]. Our simpler *offline* algorithms allow us to generalize these results to give the first secretary algorithms capable of handling a non-monotone submodular objective function.

1.2 Our Main Ideas

At a high level, the simple yet crucial observation for the offline results is this: many of the previous algorithms and proofs for constrained monotone submodular maximization can be adapted to show that the set S produced by them satisfies $f(S) \geq \beta f(S \cup C^*)$, for some $0 < \beta \leq 1$, and C^* being an optimal solution. In the monotone case, the right hand side is at least $f(C^*) = \mathbf{OPT}$ and we are done. In the non-monotone case, we cannot do this. However, we observe that if $f(S \cap C^*)$ is a reasonable fraction of $\mathbf{OPT}$, then (approximately) finding the most valuable set within S would give us a large value—and since we work with constraints that are downwards closed, finding such a set is just *unconstrained* maximization on $f(\cdot)$ restricted to S, for which Feige et al. [13] give good algorithms! On the other hand, if $f(S \cap C^*) \leq \epsilon\mathbf{OPT}$ and $f(S)$ is also too small, then one can show that deleting the elements in S and running the procedure again to find another set $S' \subseteq \Omega \setminus S$ with $f(S') \geq \beta f(S' \cap (C^* \setminus S))$ would guarantee a good solution! Details for the specific problems appear in the following sections; we first consider the simplest cardinality constraint case in Section 2 to illustrate the general idea, and then give more general results in Sections 3.1 and 3.2.

For the secretary case where the elements arrive in random order, algorithms were not known for the monotone case either—the main complication being that we cannot run a greedy algorithm (since the elements are arriving randomly), and moreover the value of an incoming element depends on the previously chosen set of elements. Furthermore, to extend the results to the non-monotone case, one needs to avoid the local-search algorithms (which, in fact, motivated the above results), since these algorithms necessarily implement multiple passes over the input, while the secretary model only allows a single pass over it. The details on all these are given in Section 4.

Due to space, this manuscript is missing many of the details and even some of the exposition. We would recommend the reader to the full version of this paper.

1.3 Related Work

Monotone Submodular Maximization. The (offline) monotone submodular optimization problem has been long studied: Fisher, Nemhauser, and Wolsey [31,15] showed that the greedy and local-search algorithms give a $(\mathrm{e}/\mathrm{e}-1)$-approximation with cardinality constraints, and a $(p+1)$-approximation under p matroid constraints. In another line of work, [20,25,19] showed that the greedy algorithm is a p-approximation for maximizing a *modular* (i.e., additive) function subject to a p-independence system. This proof extends to show a $(p+1)$-approximation for monotone submodular functions under the same constraints (see, e.g., [8]). A long standing open problem was to improve on these results; nothing better than a 2-approximation was known even for monotone maximization subject to a single partition matroid constraint. Calinescu et al. [7] showed how to maximize monotone submodular functions representable as weighted matroid rank functions subject to any matroid with an approximation ratio of $(\mathrm{e}/\mathrm{e}-1)$, and soon thereafter, Vondrák extended this result to *all* submodular functions [33]; these highly influential results appear jointly in [8]. Subsequently, Lee et al. [29] give algorithms that beat the $(p+1)$-bound for p matroid constraints with $p \geq 2$ to get a $(\frac{p^2}{p-1}+\epsilon)$-approximation.

Knapsack constraints. Sviridenko [32] extended results of Wolsey [35] and Khuller et al. [22] to show that a greedy-like algorithm with partial enumeration gives an $(\mathrm{e}/\mathrm{e}-1)$-approximation to monotone submodular maximization subject to a knapsack constraint. Kulik et al. [27] showed that one could get essentially the same approximation subject to a constant number of knapsack constraints. Lee et al. [28] give a 5-approximation for the same problem in the non-monotone case.

Mixed Matroid-Knapsack Constraints. Chekuri et al. [10] give strong concentration results for dependent randomized rounding with many applications; one of these applications is a $((\mathrm{e}/\mathrm{e}-1)-\epsilon)$-approximation for monotone maximization with respect to a matroid and any constant number of knapsack constraints. [17, Section F.1] extends ideas from [9] to give polynomial-time algorithms with respect to non-monotone submodular maximization with respect to a p-system and q knapsacks: these algorithms achieve an $p+q+O(1)$-approximation for constant q (since the running time is $n^{\mathrm{poly}(q)}$), or a $(p+2)(q+1)$-approximation for arbitrary q; at a high level, their idea is to "emulate" a knapsack constraint by a polynomial number of partition matroid constraints.

Non-Monotone Submodular Maximization. In the non-monotone case, even the unconstrained problem is NP-hard (it captures max-cut). Feige, Mirrokni and Vondrák [13] first gave constant-factor approximations for this problem. Lee et al. [28] gave the first approximation algorithms for constrained non-monotone maximization (subject to p matroid constraints, or p knapsack constraints); the approximation factors were improved by Lee et al. [29]. The algorithms in the previous two papers are based on local-search with p-swaps and would take $n^{\Theta(p)}$ time. Recent work by Vondrák [34] gives much further insight into the approximability of submodular maximization problems.

Secretary Problems. The original secretary problem seeks to maximize the probability of picking the element in a collection having the highest value, given that the elements are examined in random order [12,16,14]. The problem was used to model item-pricing problems by Hajiaghayi et al. [18]. Kleinberg [23] showed that the problem of maximizing a *modular* function subject to a cardinality constraint in the secretary setting admits a $(1+\frac{\Theta(1)}{\sqrt{k}})$-approximation, where k is the cardinality. (We show that maximizing a *submodular* function subject to a cardinality constraint cannot be approximated to better than some universal constant, independent of the value of k.) Babaioff et al. [5] wanted to maximize modular functions subject to matroid constraints, again in a secretary-setting, and gave constant-factor approximations for some special matroids, and an $O(\log k)$ approximation for general matroids having rank k. This line of research has seen several developments recently [3,11,26,4].

Independent Work on Submodular Secretaries. Concurrently and independently of our work, Bobby Kleinberg has given an algorithm similar to that in §4.1 for monotone secretary submodular maximization under a cardinality constraint [24]. Again independently, Bateni et al. consider the problem of non-monotone submodular maximization in the secretary setting [6]; they give a different $O(1)$-approximation subject to a cardinality constraint, an $O(L \log^2 k)$-approximation subject to L matroid constraints, and an $O(L)$-approximation subject to L knapsack constraints in the secretary setting. While we do not consider multiple constraints, it is easy to extend our results to obtain $O(L \log k)$ and $O(L)$ respectively using standard techniques.

1.4 Preliminaries

Given a set S and an element e, we use $S+e$ to denote $S \cup \{e\}$. A function $f : 2^{\Omega} \to \mathbb{R}_+$ is *submodular* if for all $S, T \subseteq \Omega$, $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$. Equivalently, f is submodular if it has *decreasing marginal utility*: i.e., for all $S \subseteq T \subseteq \Omega$, and for all $e \in \Omega$, $f(S+e) - f(S) \geq f(T+e) - f(T)$. Also, f is called *monotone* if $f(S) \leq f(T)$ for $S \subseteq T$. Given f and $S \subseteq \Omega$, define $f_S : 2^{\Omega} \to \mathbb{R}$ as $f_S(A) := f(S \cup A) - f(S)$. The following facts are standard.

Proposition 1. *If f is submodular with $f(\emptyset) = 0$, then*

- *for any S, f_S is submodular with $f_S(\emptyset) = 0$, and*
- *f is also* subadditive*; i.e., for disjoint sets A, B, we have $f(A)+f(B) \geq f(A \cup B)$.*

Matroids. A *matroid* is a pair $\mathcal{M} = (\Omega, \mathcal{I} \subseteq 2^{\Omega})$, where $\mathcal{I}$ contains $\emptyset$, if $A \in \mathcal{I}$ and $B \subseteq A$ then $B \in \mathcal{I}$, and for every $A, B \in \mathcal{I}$ with $|A| < |B|$, there exists $e \in B \setminus A$ such that $A + e \in \mathcal{I}$. The sets in $\mathcal{I}$ are called *independent*, and the *rank* of a matroid is the size of any maximal independent set (base) in $\mathcal{M}$. In a *uniform* matroid, $\mathcal{I}$ contains all subsets of size at most k. A *partition* matroid, we have groups $g_1, g_2, \ldots, g_k \subseteq \Omega$ with $g_i \cap g_j = \emptyset$ and $\cup_j g_j = \Omega$; the independent sets are $S \subseteq \Omega$ such that $|S \cap g_i| \leq 1$.

Unconstrained (Non-Monotone) Submodular Maximization. We use $\mathsf{FMV}_{\alpha}(S)$ to denote an approximation algorithm given by Feige, Mirrokni, and Vondrák [13] for unconstrained submodular maximization in the non-monotone setting: it returns a set $T \subseteq S$ such that $f(T) \geq \frac{1}{\alpha} \max_{T' \subseteq S} f(T')$. In fact, Feige et al. present many such algorithms, the best approximation ratio among these is $\alpha = 2.5$ via a local-search algorithm, the easiest is a 4-approximation that just returns a uniformly random subset of S.

2 Submodular Maximization Subject to a Cardinality Constraint

We first give an offline algorithm for submodular maximization subject to a cardinality constraint: this illustrates our simple approach, upon which we build in the following sections. Formally, given a subset $X \subseteq \Omega$ and a non-negative submodular function f that is potentially non-monotone, but has $f(\emptyset) = 0$. We want to approximate $\max_{S \subseteq X : |S| \leq k} f(S)$. The greedy algorithm starts with $S \leftarrow \emptyset$, and repeatedly picks an element e with maximum marginal value $f_S(e)$ until it has k elements.

Lemma 1. *For any set $|C| \leq k$, the greedy algorithm returns a set S that satisfies $f(S) \geq \frac{1}{2} f(S \cup C)$.*

Proof. Suppose not. Then $f_S(C) = f(S \cup C) - f(S) > f(S)$, and hence there is at least one element $e \in C \setminus S$ that has $f_S(\{e\}) > \frac{f(S)}{|C \setminus S|} > \frac{f(S)}{k}$. Since we ran the greedy algorithm, at each step this element e would have been a contender to be added, and by submodularity, e's marginal value would have been only higher then. Hence the elements actually added in each of the k steps would have had marginal value more than e's marginal value at that time, which is more than $f(S)/k$. This implies that $f(S) > k \cdot f(S)/k$, a contradiction.

This theorem is existentially tight: observe that if the function f is just the cardinality function $f(S) = |S|$, and if S and C happen to be disjoint, then $f(S) = \frac{1}{2} f(S \cup C)$.

Lemma 2 (Special Case of Claim 2.7 in [28]). *Given sets $C, S_1 \subseteq U$, let $C' = C \setminus S_1$, and $S_2 \subseteq U \setminus S_1$. Then $f(S_1 \cup C) + f(S_1 \cap C) + f(S_2 \cup C') \geq f(C)$.*

Proof. By submodularity, it follows that $f(S_1 \cup C) + f(S_2 \cup C') \geq f(S_1 \cup S_2 \cup C) + f(C')$. Again using submodularity, we get $f(C') + f(S_1 \cap C) \geq f(C) + f(\emptyset)$. Putting these together and using non-negativity of $f(\cdot)$, the lemma follows.

We now give our algorithm Submod-Max-Cardinality (Figure 1) for submodular maximization: it has the same multi-pass structure as that of Lee et al., but uses the greedy analysis above instead of a local-search algorithm.

let $X_1 \leftarrow X$
for $i = 1$ to 2 **do**
  let $S_i \leftarrow \text{Greedy}(X_i)$
  let $S'_i \leftarrow \mathsf{FMV}_\alpha(S_i)$
  let $X_{i+1} \leftarrow X_i \setminus S_i$.
end for
return best of S_1, S'_1, S_2.

Fig. 1. Submod-Max-Cardinality(X, k, f)

Theorem 1. *The algorithm Submod-Max-Cardinality is a $(4 + \alpha)$-approximation.*

Proof. Let C^* be the optimal solution with $f(C^*) = \mathbf{OPT}$. We know that $f(S_1) \geq \frac{1}{2} f(S_1 \cup C^*)$. Also, if $f(S_1 \cap C^*)$ is at least $\epsilon\,\mathbf{OPT}$, then we know that the α-approximate algorithm FMV_α gives us a value of at least $(\epsilon/\alpha)\mathbf{OPT}$. Else,

$$f(S_1) \geq \tfrac{1}{2} f(S_1 \cup C^*) \geq \tfrac{1}{2} f(S_1 \cup C^*) + \tfrac{1}{2} f(S_1 \cap C^*) - \epsilon\,\mathbf{OPT}/2 \qquad (1)$$

Similarly, we get that $f(S_2) \geq \frac{1}{2} f(S_2 \cup (C^* \setminus S_1))$. Adding this to (1), we get

$$\begin{aligned} 2\max(f(S_1), f(S_2)) &\geq f(S_1) + f(S_2) \\ &\geq \tfrac{1}{2}\big(f(S_1 \cup C^*) + f(S_1 \cap C^*) + f(S_2 \cup (C^* \setminus S_1))\big) - \epsilon\mathbf{OPT}/2 \qquad (2) \\ &\geq \tfrac{1}{2} f(C^*) - \epsilon\mathbf{OPT}/2 \qquad (3) \\ &\geq \tfrac{1}{2}(1 - \epsilon)\,\mathbf{OPT}. \end{aligned}$$

where we used Lemma 2 to get from (2) to (3). Hence $\max\{f(S_1), f(S_2)\} \geq \frac{1-\epsilon}{4}\,\mathbf{OPT}$. The approximation factor now is $\max\{\alpha/\epsilon, 4/(1-\epsilon)\}$. Setting $\epsilon = \frac{\alpha}{\alpha+4}$, we get a $(4 + \alpha)$-approximation, as claimed.

Using the known value of $\alpha = 2.5$ from Feige et al. [13], we get a 6.5-approximation for submodular maximization under cardinality constraints. While this is weaker than the 3.23-approximation of Vondrák [34], or even the 4-approximation we could get from Lee et al. [28] for this special case, the algorithm is faster, and the idea behind the improvement works in several other contexts, as we show in the following sections.

3 Fast Algorithms for p-Systems and Knapsacks

In this section, we show our greedy-style algorithms which achieve an $O(p)$-approximation for submodular maximization over p-systems, and a constant-factor approximation for submodular maximization over a knapsack. Due to space constraints, many proofs are deferred to the full version of this paper.

3.1 Submodular Maximization for Independence Systems

Let Ω be a universe of elements and consider a collection $\mathcal{I} \subseteq 2^{\Omega}$ of subsets of Ω. $(\Omega, \mathcal{I})$ is called an *independence system* if (a) $\emptyset \in \mathcal{I}$, and (b) if $X \in \mathcal{I}$ and $Y \subseteq X$, then $Y \in \mathcal{I}$ as well. The subsets in $\mathcal{I}$ are called *independent*; for any set S of elements, an inclusion-wise maximal independent set T of S is called a *basis* of S. For brevity, we say that T is a basis, if it is a basis of Ω.

Definition 1. *Given an independence system $(\Omega, \mathcal{I})$ and a subset $S \subseteq \Omega$. The* rank *$r(S)$ is defined as the cardinality of the* largest *basis of S, and the* lower rank *$\rho(S)$ is the cardinality of the* smallest *basis of S. The independence system is called a p-independence system (or a p-system) if $\max_{S \subseteq \Omega} \frac{r(S)}{\rho(S)} \leq p$.*

See, e.g., [8] for a discussion of independence systems and their relationship to other families of constraints; it is useful to recall that intersections of p matroids form a p-independent system.

The Algorithm for p-Independence Systems. Suppose we are given an independence system $(\Omega, \mathcal{I})$, a subset $X \subseteq \Omega$ and a non-negative submodular function f that is potentially non-monotone, but has $f(\emptyset) = 0$. We want to find (or at least approximate) $\max_{S \subseteq X : S \in \mathcal{I}} f(S)$. The greedy algorithm for this problem is what you would expect: start with the set $S = \emptyset$, and at each step pick an element $e \in X \setminus S$ that maximizes $f_S(e)$ and ensures that $S + e$ is also independent. If no such element exists, the algorithm terminates, else we set $S \leftarrow S + e$, and repeat. (Ideally, we would also check to see if $f_S(e) \leq 0$, and terminate at the first time this happens; we don't do that, and instead we add elements even when the marginal gain is negative until we cannot add any more elements without violating independence.) The proof of the following lemma appears in the full version of the paper, and closely follows that for the monotone case from [8].

Lemma 3. *For a p-independence system, if S is the independent set returned by the greedy algorithm, then for any independent set C, $f(S) \geq \frac{1}{p+1} f(C \cup S)$.*

The algorithm Submod-Max-p-Systems (Figure 2) for maximizing a non-monotone submodular function f with $f(\emptyset) = 0$ over a p-independence system now immediately suggests itself. The theorem is proven in the full version of the paper.

Theorem 2. *The algorithm Submod-Max-p-System is a $(1 + \alpha)(p + 2 + 1/p)$-approximation for maximizing a non-monotone submodular function over a p-independence system, where α is the approximation guarantee for unconstrained (non-monotone) submodular maximization.*

1: $X_1 \leftarrow X$
2: **for** $i = 1$ to $p + 1$ **do**
3: $\quad S_i \leftarrow \mathsf{Greedy}(X_i, \mathcal{I}, f)$
4: $\quad S'_i \leftarrow \mathsf{FMV}_\alpha(S_i)$
5: $\quad X_{i+1} \leftarrow X_i \setminus S_i$
6: **end for**
7: **return** $S \leftarrow$ best among $\{S_i\}_{i=1}^{p+1} \cup \{S'_i\}_{i=1}^{p+1}$.

Fig. 2. Submod-Max-p-System$(X, \mathcal{I}, f)$

Note that even using $\alpha = 1$, our approximation factors differ from the ratios in Lee et al. [28,29] by a small constant factor. However, the proof here is somewhat simpler and also works seamlessly for all p-independence systems instead of just intersections of matroids. Moreover our running time is only linear in the number of matroids, instead of being exponential as in the local-search: previously, no polynomial time algorithms were known for this problem if p was super-constant. Note that running the algorithm just twice instead of $p+1$ times reduces the run-time further; we can then use Lemma 2 instead of the full power of [28, Claim 2.7], and hence the constants are slightly worse.

3.2 Submodular Maximization Over Knapsacks

The paper of Sviridenko [32] gives a greedy algorithm with partial enumeration that achieves a $\frac{e}{e-1}$-approximation for *monotone* submodular maximization with respect to a knapsack constraint. In particular, each element $e \in X$ has a size c_e, and we are given a bound B: the goal is to maximize $f(S)$ over subsets $S \subseteq X$ such that $\sum_{e \in S} c_e \leq B$. His algorithm is the following—for each possible subset $S_0 \subseteq X$ of at most three elements, start with S_0 and iteratively include the element which maximizes the gain in the function value per unit size, and the resulting set still fits in the knapsack. (If none of the remaining elements gives a positive gain, or fit in the knapsack, stop.) Finally, from among these $O(|X|^3)$ solutions, choose the best one—Sviridenko shows that in the monotone submodular case, this is an $\frac{e}{e-1}$-approximation algorithm. One can modify Sviridenko's algorithm and proof to show the following result for non-monotone submodular functions. (The details are in the full paper).

Theorem 3. *There is a polynomial-time algorithm that given the above input, outputs a polynomial sized collection of sets such that for any valid solution C, the collection contains a set S satisfying $f(S) \geq \frac{1}{2} f(S \cup C)$.*

Note that the tight example for cardinality constraints shows that we cannot hope to do better than a factor of $1/2$. Now using an argument very similar to that in Theorem 1 gives us the following result for non-monotone submodular maximization with respect to a knapsack constraint.

Theorem 4. *There is an* $(4+\alpha)$*-approximation for the problem of maximizing a submodular function with respect a knapsack constraint, where* α *is the approximation guarantee for unconstrained (non-monotone) submodular maximization.*

4 Constrained Submodular Maximization in the Secretary Setting

We will give algorithms for submodular maximization in the secretary setting: first subject to a cardinality constraint, then with respect to a partition matroid, and finally an algorithm for general matroids. Much of this section has been omitted for space restrictions, and the reader is referred to the full version of the paper. The main algorithmic concerns tackled in this section when developing secretary algorithms are: (a) previous algorithms for non-monotone maximization required local-search, which seems difficult in an online secretary setting, so we developed greedy-style algorithms; (b) we need multiple passes for non-monotone optimization, and while that can be achieved using randomization and running algorithms in parallel, these parallel runs of the algorithms may have correlations that we need to control (or better still, avoid); and of course (c) the marginal value function changes over the course of the algorithm's execution as we pick more elements—in the case of partition matroids, e.g., this ever-changing function creates several complications.

We also show an information theoretic lower bound: no secretary algorithm can approximately maximize a submodular function subject to a cardinality constraint k to a factor better than some universal constant greater than 1, independent of k (This is ignoring computational constraints, and so the computational inapproximability of offline submodular maximization does not apply). This is in contrast to the additive secretary problem, for which Kleinberg gives a secretary algorithm achieving a $\frac{1}{1-5/\sqrt{k}}$-approximation [23]. This lower bound is found in the full version of the paper. (For a discussion about independent work on submodular secretary problems, see §1.3.)

4.1 Subject to a Cardinality Constraint

The offline algorithm presented in Section 2 builds three potential solutions and chooses the best amongst them. We now want to build just one solution in an *online* fashion, so that elements arrive in random order, and when an element is added to the solution, it is never discarded subsequently. We first give an online algorithm that is given the optimal value $\mathbf{OPT}$ as input but where the elements can come in *worst-case* order (we call this an "online algorithm with advice"). Using sampling ideas we can estimate $\mathbf{OPT}$, and hence use this advice-taking online algorithm in the secretary model where elements arrive in random order.

To get the advice-taking online algorithm, we make two changes. First, we do not use the greedy algorithm which selects elements of highest marginal utility, but instead use a *threshold algorithm*, which selects any element that has marginal utility above a certain threshold. Second, we will change Step 4 of Algorithm Submod-Max-Cardinality to use FMV_4, which simply selects a random subset of the elements to get a 4-approximation to the unconstrained submodular maximization problem [13]. The *Threshold Algorithm* with inputs (τ, k) simply selects each element as it appears if it has marginal utility at least τ, up to a maximum of k elements.

Lemma 4 (Threshold Algorithm). *Let C^* satisfy $f(C^*) = \mathbf{OPT}$. The threshold algorithm on inputs (τ, k) returns a set S that either has k elements and hence a value of at least τk, or a set S with value $f(S) \geq f(S \cup C^*) - |C^*|\tau$.*

Theorem 5. *If we change Algorithm Submod-Max-Cardinality from §2 to use the threshold algorithm with threshold $\tau = \frac{\mathbf{OPT}}{7k}$ in Step 3, and to use the random sampling algorithm FMV_4 in Step 4, and return a (uniformly) random one of S_1, S_1', S_2 in Step 7, the expected value of the returned set is at least $\mathbf{OPT}/21$.*

Observation 6. *Given the value of $\mathbf{OPT}$, the algorithm of Theorem 5 can be implemented in an online fashion where we (irrevocably) pick at most k elements.*

Observation 7. *In both the algorithms of Theorems 1 and 5, if we use some value $Z \leq \mathbf{OPT}$ instead of $\mathbf{OPT}$, the returned set has value at least $Z/(4+\alpha)$, and expected value at least $Z/21$, respectively.*

Finally, it will be convenient to recall Dynkin's algorithm: given a stream of n numbers randomly ordered, it samples the first $1/e$ fraction of the numbers and picks the next element that is larger than all elements in the sample.

The Secretary Algorithm for the Cardinality Case For a constrained submodular optimization, if we are given *(a)* a ρ_{off}-approximate offline algorithm, and also *(b)* a ρ_{on}-approximate online advice-taking algorithm that works given an estimate of $\mathbf{OPT}$, we can now get an algorithm in the secretary model thus: we use the offline algorithm to estimate $\mathbf{OPT}$ on the first half of the elements, and then run the advice-taking online algorithm with that estimate. The formal algorithm appears in Figure 3. Because of space constraints, we have deferred the proof of the following theorem to the full paper.

```
Let Solution ← ∅.
Flip a fair coin
if heads then
    Solution ← most valuable item using Dynkin's-Algo
else
    Let m ∈ B(n, 1/2) be a draw from the binomial distribution
    A_1 ← ρ_off-approximate offline algorithm on the first m elements.
    A_2 ← ρ_on-approximate advice-taking online algorithm with
            f(A_1) as the guess for OPT.
    Return A_2
end if
```

Fig. 3. Algorithm SubmodularSecretaries

Theorem 8. *The above algorithm is an $O(1)$-approximation algorithm for the cardinality-constrained submodular maximization problem in the secretary setting.*

4.2 Subject to a Partition Matroid Constraint

The full version of the paper additionally exhibits and proves a constant factor approximation for online setting with a partition matroid constraint.

4.3 Subject to a General Matroid Constraint

The full version of the paper additionally exhibits and proves a $\log(k)$ factor approximation for on-line setting with a general matroid constraint of rank k.

Acknowledgments. We thank C. Chekuri, V. Nagarajan, M.I. Sviridenko, J. Vondrák, and especially R.D. Kleinberg for valuable comments, suggestions, and conversations. Thanks to C. Chekuri also for pointing out an error in an earlier manuscript, and to M.T. Hajiaghayi for informing us of the results in [6].

References

1. Agrawal, S., Ding, Y., Saberi, A., Ye, Y.: Correlation robust stochastic optimization. CoRR, abs/0902.1792 (2009)
2. Asadpour, A., Nazerzadeh, H., Saberi, A.: Stochastic submodular maximization. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 477–489. Springer, Heidelberg (2008)
3. Babaioff, M., Immorlica, N., Kempe, D., Kleinberg, R.: A knapsack secretary problem with applications. In: Charikar, M., Jansen, K., Reingold, O., Rolim, J.D.P. (eds.) RANDOM 2007 and APPROX 2007. LNCS, vol. 4627, pp. 16–28. Springer, Heidelberg (2007)
4. Babaioff, M., Dinitz, M., Gupta, A., Immorlica, N., Talwar, K.: Secretary problems: weights and discounts. In: 19th Proceedings of ACM-SIAM Symposium on Discrete Algorithms, pp. 1245–1254 (2009)
5. Babaioff, M., Immorlica, N., Kleinberg, R.: Matroids, secretary problems, and online mechanisms. In: SODA 2007, pp. 434–443 (2007)
6. Bateni, M., Hajiaghayi, M., Zadimoghaddam, M.: Submodular secretary problem and extensions (2010) (manuscript), http://hdl.handle.net/1721.1/51336
7. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a submodular set function subject to a matroid constraint (extended abstract). In: Proceedings, MPS Conference on Integer Programming and Combinatorial Optimization, pp. 182–196 (2007)
8. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a monotone submodular function subject to a matroid constraint. To appear in SICOMP (2009)
9. Chekuri, C., Khanna, S.: A polynomial time approximation scheme for the multiple knapsack problem. SIAM J. Comput. (electronic) 35(3), 713–728 (2005)
10. Chekuri, C., Vondrák, J., Zenklusen, R.: Randomized pipage rounding for matroid polytopes and applications. CoRR, abs/0909.4348 (2009); To appear in FOCS 2010
11. Dimitrov, N.B., Plaxton, C.G.: Competitive weighted matching in transversal matroids. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part I. LNCS, vol. 5125, pp. 397–408. Springer, Heidelberg (2008)
12. Dynkin, E.B.: Optimal choice of the stopping moment of a Markov process. Dokl. Akad. Nauk SSSR 150, 238–240 (1963)
13. Feige, U., Mirrokni, V., Vondrak, J.: Maximizing non-monotone submodular functions. In: Proceedings of 48th Annual IEEE Symposium on Foundations of Computer Science, FOCS (2007)
14. Ferguson, T.S.: Who solved the secretary problem? Statistical Science 4, 282–296 (1989)
15. Fisher, M.L., Nemhauser, G.L., Wolsey, L.A.: An analysis of approximations for maximizing submodular set functions. II. Math. Programming Stud. (8), 73–87 (1978); Polyhedral combinatorics
16. Freeman, P.R.: The secretary problem and its extensions: a review. Internat. Statist. Rev. 51(2), 189–206 (1983)

17. Gupta, A., Nagarajan, V., Ravi, R.: Thresholded covering algorithms for robust and max-min optimization. CoRR, abs/0912.1045 (2009); To appear in ICALP 2010
18. Hajiaghayi, M.T., Kleinberg, R., Parkes, D.C.: Adaptive limited-supply online auctions. In: EC 2004: Proceedings of the 5th ACM Conference on Electronic Commerce, pp. 71–80. ACM, New York (2004)
19. Hausmann, D., Korte, B., Jenkyns, T.A.: Worst case analysis of greedy type algorithms for independence systems. Math. Programming Stud. (12), 120–131 (1980); Combinatorial optimization
20. Jenkyns, T.A.: The efficacy of the "greedy" algorithm. In: Proceedings of the Seventh Southeastern Conference on Combinatorics, Graph Theory, and Computing (Louisiana State Univ., Baton Rouge, La., 1976), pp. 341–350. Congressus Numerantium, No. XVII, Winnipeg, Man, Utilitas Math. (1976)
21. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146. ACM, New York (2003)
22. Khuller, S., Moss, A., Naor, J.: The budgeted maximum coverage problem. Inform. Process. Lett. 70(1), 39–45 (1999)
23. Kleinberg, R.: A multiple-choice secretary algorithm with applications to online auctions. In: 16th SODA, pp. 630–631. ACM, New York (2005)
24. Kleinberg, R.D.: A secretary problem with submodular payoff function (2009) (manuscript)
25. Korte, B., Hausmann, D.: An analysis of the greedy heuristic for independence systems. Ann. Discrete Math. 2, 65–74 (1978); Algorithmic aspects of combinatorics Conf., Vancouver Island, B.C. (1976)
26. Korula, N., Pál, M.: Algorithms for secretary problems on graphs and hypergraphs, pp. 508–520 (2009)
27. Kulik, A., Shachnai, H., Tamir, T.: Maximizing submodular set functions subject to multiple linear constraints. In: SODA 2009: Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 545–554 (2009)
28. Lee, J., Mirrokni, V.S., Nagarajan, V., Sviridenko, M.: Maximizing nonmonotone submodular functions under matroid or knapsack constraints. SIAM J. Discrete Math. 23(4), 2053–2078 (2009/2010); Preliminary versions in STOC 2009 and arXiv:0902.0353
29. Lee, J., Sviridenko, M., Vondrák, J.: Submodular maximization over multiple matroids via generalized exchange properties. In: Proceedings, International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, pp. 244–257 (2009)
30. Mossel, E., Roch, S.: On the submodularity of influence in social networks. In: Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, p. 134. ACM, New York (2007)
31. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions. I. Math. Programming 14(3), 265–294 (1978)
32. Sviridenko, M.: A note on maximizing a submodular set function subject to a knapsack constraint. Oper. Res. Lett. 32(1), 41–43 (2004)
33. Vondrák, J.: Optimal approximation for the submodular welfare problem in the value oracle model. In: Proceedings, ACM Symposium on Theory of Computing, pp. 67–74 (2008)
34. Vondrák, J.: Symmetry and approximability of submodular maximization problems. In: Proceedings, IEEE Symposium on Foundations of Computer Science (2009) (page to appear)
35. Wolsey, L.A.: Maximising real-valued submodular functions: primal and dual heuristics for location problems. Math. Oper. Res. 7(3), 410–425 (1982)

Strategic Cooperation in Cost Sharing Games

Martin Hoefer*

Dept. of Computer Science, RWTH Aachen University, Germany
mhoefer@cs.rwth-aachen.de

Abstract. In this paper we consider a large variety of strategic cost sharing games with so-called *arbitrary sharing* based on various combinatorial optimization problems, such as vertex and set cover, facility location, and network design problems. We concentrate on the existence and computational complexity of strong equilibria, in which no coalition can decrease the cost of every member.

Our main result reveals a connection between strong equilibrium in strategic games and the core in traditional coalitional cost sharing games studied in economics. For set cover and facility location games this results in a tight characterization of the existence of strong equilibrium using the integrality gap of suitable linear programming formulations. In addition, we are able to show that in general the strong price of anarchy is always 1, whereas the price of anarchy is known to be $\Theta(n)$ for Nash equilibria. Finally, we indicate that the LP-approach can also be used to compute near-optimal and near-stable approximate strong equilibria.

1 Introduction

How can a set of self-interested actors share the cost of a joint investment in a fair and stable way? This fundamental question has motivated a large amount of research in economics in the last decades. More recently, this question has been studied in computer science to understand the development of the Internet and questions arising in e-commerce.A classic framework to study cost sharing problems without centralized control are *cost sharing games*, in which cost can be specified as an abstract parameter for each player and/or each coalition. Relevant to real-world optimization problems are cost sharing games, where the cost is tied to the investment into specific resources. There is a set of players, and each coalition of players has an associated cost value coming from an optimal solution to an optimization problem for the coalition. For example, consider a multicast network design game, in which players strive to establish connections to a common source vertex s. This scenario can be formulated as a MST game, in which each vertex $v \neq s$ is a player, and the edges have costs. The cost of a coalition in this game is the cost of the MST for the set $C \cup s$. In the literature many interesting cost sharing games have been studied, e.g., based on MST and Steiner tree [8, 18, 26, 19], covering and packing problems [13], or facility location [15].

Coalitional cost sharing games are usually *transferable utility (TU) games*, which allows for the largest level of generality in the bargaining and coalition formation process. The foremost concept of stability and fairness in TU cost sharing games is the

* Supported by DFG through UMIC Research Centre at RWTH Aachen University and grant Ho 3831/3-1.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 258–269, 2010.

core. The core is a set of imputations, i.e., of distributions of the cost for the complete player set to the players. To be in the core an imputation has to fulfill the additional property that no coalition of players in sum pays more than its associated cost value. A problem is that the cost shares represent a strong abstraction. The game does not take into account who pays how much for which resource. This, however, is crucial when studying the incentives of players in large unregulated settings such as the Internet. The need to understand cost sharing on a strategic level prompted computer scientists to study strategic cost sharing scenarios. On the one hand, there is recent work on *designing* strategic cost sharing games to obtain favorable Nash equilibrium properties [12]. Here a central authority dictates cost shares for each player. This is close to cost sharing mechanisms, which have received a lot of attention [22,23]. Designing cost shares, e.g., using Shapley value cost sharing [4,11,14], can yield favorable conditions for existence and cost of Nash equilibria. In contrast, such a model is unsuitable when there is very little control over players and their bargaining options. A model that allows for general cost sharing between players is sometimes referred to as *arbitrary* cost sharing, and it has been studied, e.g., in [5,3,10,20,14]. In these games the strategy of a player is a payment function that specifies his exact contribution to the cost of each resource. The outcomes of such strategic cost sharing games based on combinatorial optimization problems will be the subject of this paper.

The most prominent stability concept in strategic games is the Nash equilibrium (NE). While a NE (in mixed strategies) always exists, a drawback is that it is only resilient to unilateral deviations. In many reasonable scenarios agents might be able to coordinate their actions, and to address this issue we consider the *strong equilibrium (SE)* in this paper. A SE [7] is a state, from which no coalition (of arbitrary size) has a deviation that lowers the cost of *every* member of the coalition. This resilience to coalitional deviations is highly attractive, but SE might not exist. This may be the reason they have not received an equivalent amount of interest despite their attractive properties. We partly circumvent this problem by studying approximate SE that are guaranteed to exist. However, a deeper treatment of these aspects is mostly left for future work.

Our main interest is to characterize the existence, social cost, and computational complexity of SE in strategic cost sharing games based on combinatorial optimization problems. A set of simple but striking observations reveals that a SE in a strategic cost sharing game can always be turned into a core imputation of the corresponding coalitional game. Hence, a SE is a *strategic refinement* of a core solution, and existence of a SE implies non-emptiness of the core. It also implies that the strong price of anarchy (SPoA) [1] is 1. In addition, SE are equivalent to seemingly stronger coalitional equilibrium concepts in these games. This motivates us to furhter explore the connection between core and SE and to examine machinery developed for core solutions to obtain SE, as well.

Contribution and Outline. In Section 2 we consider games based on vertex and set cover, facility location, MST and Steiner tree problems, for which we show an equivalence result. Whenever the core in the coalitional game is nonempty, there is a SE for the strategic game. Our main proof technique relies on LP duality and allows to tightly characterize the existence and cost of SE in all these games. As a byproduct, this yields simple proofs of all known results for SE in strategic cost sharing games with arbitrary sharing, which were previously shown [14] via complicated combinatorial arguments.

The equivalence between SE and core solutions is an interesting and notable fact. However, for non-transferable utility (NTU) games and appropriate extensions to strategic games a similar equivalence is obvious. Thus, it may be more surprising that the relation between SE and core solutions in cost sharing games can be more complicated. In Section 3 we explore equivalence without relying on linear programming. In some cases like Terminal Backup Games [3] we can resort to combinatorial arguments. For other interesting games such as connection games or network cutting games the core might be non-empty but a SE is absent. A similar result is established in Section 4 even for simple vertex cover games when we allow resources to be purchased fractionally or in mulitple units. Characterizing SE in these games remains as an intriguing open problem. We observe in Section 5 that linear programming can be used to obtain approximate (α, β)-SE in vertex and set cover, as well as factility location games. Finally, we conclude in Section 6 with some interesting questions for further research. Due to spacial reasons parts of the paper are omitted. A full version of this paper is available online [21].

Our main conceptual contribution is to reveal a non-trivial and close relation between coalitional and strategic games defined on the same instance of the optimization problem. We believe that this connection between traditional coalitional games from economics and strategic cost sharing games from computer science should stimulate further research on cost sharing with rational agents.

Preliminaries. We consider classes of cost sharing games based on combinatorial optimization problems. In each of these games there is a set R of resources. Resource $r \in R$ can be *bought* if the associated cost $c(r) \geq 0$ is paid for. For $R' \subseteq R$ let $c(R') = \sum_{r \in R'} c(r)$. We assume that there is set of players K. Each player $k \in K$ strives to satisfy a certain constraint on the bought resources. For example, in the case of the *set cover problem* the player set is the element set $K = E$. The resources are sets $R = \mathcal{S} \subseteq 2^E$ over E. The constraint of player e states that there must be at least one bought set S with $e \in S$. In a similar way we can base our construction on various cost minimization problems like facility location or network design. We will describe them in more detail in the corresponding sections. However, a common assumption in our problems is a free disposal property, i.e., if for a set of bought resources all player constraints are satisfied, then a superset of bought resources can never make a player constraint become violated.

For a given set of players, resources, and constraints we define two games - a *coalitional* and a *strategic* cost sharing game. The *coalitional game* $\Delta = (K, c)$ is given by the set of players K and a cost function $c : 2^K \to \mathbb{R}_0^+$ that specifies a cost value for every subset of players. For a coalition $C \subseteq K$, the cost is $c(C) = \sum_{r \in R(C)^*} c(r)$ for an *optimum solution* $R(C)^* \subseteq R$ for C. In particular, $R(C)^*$ is a minimum cost set of resources that must be bought to satisfy all constraints of players in C. For example, in a set cover game $R(C)^*$ is the minimum cost set cover for the elements in C. We denote the special case $R^* = R(K)^*$ as the *social optimum*.

The goal in a coalitional game is to find a cost sharing of $c(K)$ for the so-called grand coalition K. A vector of cost shares $\gamma_1, \ldots, \gamma_k$ is called an *imputation* if $\sum_{i \in K} \gamma_i = c(K)$. The game Δ is a *transferable utility (TU)* game, i.e., we are free to choose $0 \leq \gamma_i \leq c(K)$. The central concept of stability and fairness in coalitional games is the *core*. The core is the set of imputations γ, for which $c(C) \geq \sum_{i \in C} \gamma_i$. Intuitively, when sharing the cost according to a member of the core, no subset of players has an incentive to deviate from the grand coalition and make a separate investment - depending on the underlying optimization problem, e.g., purchase different sets or construct an independent network.

The *strategic game* $\Gamma = (K, (S_i)_{i \in K}, (c_i)_{i \in K})$ is specified by strategies and individual cost for each player. The *strategy space* S_i of player $i \in K$ consists of all functions $s_i : R \to \mathbb{R}_0^+$. Strategy s_i allows him to specify for each resource $r \in R$ how much he is willing to contribute to r. A resource r is *bought* if $\sum_{i \in K} s_i(r) \geq c(r)$. A vector of strategies s is a *state* of the game. For a state s we define $|s_i| = \sum_{r \in R} s_i(r)$ and the *individual cost* of player i as $c_i(s) = |s_i|$ if the bought resources satisfy his constraint. Otherwise, $c_i(s) = \infty$ or a different value that is prohibitively large.

In this paper we consider coalitional incentives and resort to strong equilibria [7]. A state s has a *violating coalition* $C \subseteq K$ if there are strategies $s'_C = (s'_i)_{i \in C}$ such that $c_i(s'_C, s_{-C}) < c_i(s)$ for each $i \in C$. A violating coalition has a deviation, in which all players in C strictly pay less. A *strong equilibrium (SE)* is a state s that has no violating coalition. Note that in a SE a set of resources is bought such that all player constraints are satisfied. Each resource r is either paid for exactly or not contributed to at all. Thus, a SE represents a cost sharing of a feasible solution for the grand coalition. In addition, we briefly consider the concept of a (α, β)-SE. These are strategy profiles, which constitute an approximate solution concept. In a (α, β)-SE no coalition of players can reduce the cost of every member by strictly more than a factor of α, and the cost of the bought solution represents a β-approximation to $c(K)$.

For a strategic game Γ, it is a simple to observe that in every SE a social optimum R^* is bought. Otherwise, a suitable player set could deviate to buy R^*, thereby reducing the contribution of each player proportionally and strictly. A similar trick shows that SE are resilient to stronger improving moves, in which only one player strictly improves but no player gets worse, or even against moves where only the sum of all player costs decreases. More importantly, SE can be turned into core solutions for the corresponding coalitional game Δ. While the total payment in both cases is $c(R^*)$, optimal deviations for a coalition C are cheaper in Γ than in Δ, because in Δ we assume that all players outside C stop contributing. These insights are formally proved in the full version [21].

2 Strong Equilibria Using Linear Programming

Vertex and Set Cover Games. The insights in the last section imply that non-emptiness of the core is necessary for existence of SE. In the following we consider various classes of games, in which it is also sufficient. In these cases the SE is a strategic refinement of the core, as it allows to specify a strategic allocation of payments to resources. We can relate SE existence to the core via linear programming duality. For simplicity we outline the general argument in the setting of set cover games. In a set cover game, we are given a set of players as elements E and a set system $\mathcal{S} \subseteq 2^E$, where each $S \in \mathcal{S}$ has a cost $c(S) \geq 0$. The constraint of player e is that at least one set S with $e \in S$ must be bought.

Theorem 1. *If a set cover game Δ has a non-empty core, then the strategic game Γ has a SE.*

Proof. We consider the integer programming formulation of set cover. In particular, we consider the following linear relaxation, which employs $x_S \geq 0$ instead of $x_S \in \{0,1\}$ and thus allows sets to be included fractionally in the solution. We also consider the corresponding LP dual.

$$\begin{array}{llll} \text{Min} & \sum_{S \in \mathcal{S}} x_S c(S) & & \\ \text{s.t.} & \sum_{S: e \in S} x_S \geq 1 & \forall\, e \in E & \\ & x_S \geq 0 & \forall\, S \in \mathcal{S}. & \end{array} \qquad \begin{array}{llll} \text{Max} & \sum_{e \in E} \gamma_e & & \\ \text{s.t.} & \sum_{e \in S} \gamma_e \leq c(S) & \forall\, S \in \mathcal{S} & \\ & \gamma_e \geq 0 & \forall\, e \in E. & \end{array}$$

It has been shown by Deng et al. [13] that the core of Δ is non-empty if and only if the integrality gap of the underlying set cover problem is 1, i.e., if the LP has an integral optimal solution. This is a prerequisite for existence of a SE in Γ. We strengthen this result by showing that core solutions can also be turned into SE, i.e., an integral optimum is also sufficient.

For the above programs consider the optimum primal solution x^* and the optimum dual solution γ^*, where x^* is integral and defines a feasible cover. Both x^* for the primal and γ^* for the dual yield the same objective value. Now assign each player e to pay $s_e(S) = \gamma_e^* x_S^*$. The theorem follows if every set in the cover is purchased exactly and no coalition C can reduce their total payments $\sum_{e \in C} |s_e|$. The first condition is clearly necessary for a SE, the second one implies that no coalition can be sum violating (and thus violating). We first show that the sets are exactly paid for. If $x_S^* > 0$, then due to complementary slackness the inequality $\sum_{e \in E} \gamma_e^* \leq c(S)$ is tight, hence by this assignment all the purchased sets get exactly paid for.

We now show that no coaltion can reduce the total payments. Suppose a coalition C is violating. We consider an adjusted game derived by iteratively removing elements and payments of other players $e \notin C$. Upon removing an element e, we remove its contribution from the costs of sets S including e. This yields the cost function $c_C(S)$ with $c_C(S) = c(S) - \sum_{e \notin C, e \in S} \gamma_e^* x_S^*$. It captures the reduced problem of finding a minimum cost cover for coalition C with costs adjuted by the payments of other players $e \notin C$. Note that for this reduced problem the solution x^* is still feasible. By obtaining the dual we can set the covering requirement to 0 for every removed element $e \notin C$. Then γ^* still represents a feasible solution to the LP-dual of the reduced problem. It yields the same objective value as x^* for the primal. By strong duality both x^* and γ^* must be optimal solutions to the reduced primal and dual problems. This proves that the total payments of C are optimal. Hence, C cannot be sum violating and not violating, a contradiction. This proves that s is a SE. □

For the special case of vertex cover games we can use results from [13] to efficiently compute SE. In particular, a game allows a core solution (and thus a SE) if and only if a maximum matching in the graph has the same size as the minimum vertex cover. This condition can be checked in polynomial time by computing corresponding vertex covers and matchings [13, Theorem 7 and Corollary 7]. Hence, we can check in polynomial time whether a SE exists. If it exists, we can use the computed vertex cover as primal solution for our LP and compute cost shares for a SE with the corresponding dual solution.

In addition, we can check in polynomial time whether a given strategy profile is a SE. If the state is a SE, it must exactly pay for a vertex cover of the instance. This yields a primal solution for the LP. In addition, the accumulated cost shares of players must yield a corresponding dual solution. Finally, both primal and dual solutions must generate the same value of the objective function. This is a sufficient and necessary condition for being a SE, which can be checked in polynomial time.

Corollary 1. *In a vertex cover game Γ we can decide in polynomial time if a SE exists. If it exists, we can compute a SE in polynomial time. Given a state s for Γ we can verify in polynomial time if it is a SE.*

Another interesting case are edge cover games. Here players are the vertices of a graph and resources are the edges. Each vertex wants to ensure that at least one incident edge is bought. Using the characterization of the non-emptiness of the core in [13, Theorem 8 and Corollary 8] we can obtain similar results for this game as well.

Corollary 2. *In an edge cover game Γ we can decide in polynomial time if a SE exists. If it exists, we can compute a SE in polynomial time. Given a state s for an edge cover game Γ we can verify in polynomial time if it is a SE.*

Facility Location Games. Another class of games that can be handled via similar arguments are facility location games. We outline the arguments on the simple class of *uncapacitated facility location games* (UFL games) and show below how to extend this approach to a more general class of games considered in [15, 10]. In a *UFL problem* there is a set T of terminals and a set F of facilities. We set $n_t = |T|$ and $n_f = |F|$. Each facility $f \in F$ has an opening cost $c(f) \geq 0$, for each terminal $t \in T$ and each facility $f \in F$ there is a connection cost $c(t, f) \geq 0$. The goal is to open a subset of facilities and buy a set of connections of minimum total cost, such that each terminal is connected to an opened facility. In the *UFL game* each player owns a terminal, i.e., $K = T$. The constraint of player t is satisfied if there is a bought connection (t, f) to some opened facility f. We can formalize the UFL problem by the standard integer program (see [21]).

Theorem 2. *If a UFL game Δ has a non-empty core, then the strategic game Γ has a SE.*

This result can be combined with insights from [15] to characterize computational properties of SE. In particular, we can decide in polynomial time if a given strategy profile for Γ is a SE. We first check if the payments of players are made only to their own connection and opening costs. Then we accumulate contributions to cost shares and check if this yields a core solution - i.e., if the primal solution (given by the purchased solution to the facility location problem) and the dual solution (given by the cost shares) correspond to each other and yield the same optimal value for primal and dual LPs.

Corollary 3. *Given a strategy vector for a UFL game Γ we can verify in polynomial time if it is a SE.*

This implies that the problem of computing a SE is in NP. In fact, in [15] it is shown for a class of UFL games that deciding the existence of a core solution is NP-complete.

Corollary 4. *It is* NP*-complete to decide if a given UFL game Γ has a SE.*

In the full version [21] we show how to extend these results to connection-restricted facility location games (CRFL games) from [15], in which access to a facility f can be obtained only by certain allowed coalitions $\mathcal{A}_f \subseteq 2^T$.

Connection Games. We can also use LPs to formulate network design games in directed and undirected graphs. Perhaps the most frequently studied variant is a *connection game* [5], in which there is a graph $G = (V, E)$, resources are the edges with costs $c(e) \geq 0$. Each player $k \in K$ has a source-sink pair (s_k, t_k) in G. A player is satisfied if there is a path of bought edges connecting his pair. The game is based on the Steiner Network problem [16]. A variant based on Steiner Tree is called *single-source* game, where every player has the same source s. We characterize existence of SE based on a Flow-LP previously studied, e.g., in [26].

Theorem 3. *If the Flow-LP has an integral optimum solution, then the strategic connection game Γ has a strong equilibrium.*

The Flow-LP for single source games on directed series-parallel graphs has integrality gap 1 (a proof can be derived from [25]). Solving this LP then allows us to obtain one of the main results from [14] in a simple and compact way.

Theorem 4. *[14] Every single source connection game Γ on a directed series-parallel graph has a SE that can be computed in polynomial time.*

In addition, consider MST games, i.e., single source games with every vertex of G being a sink node for at least one player. For directed graphs, a SE can be computed from dual solutions of the LP [26]. In particular, a simple rule due to Bird [8] (i.e., each player k pays exactly for the unique arc of the tree leaving sink t_k) is a SE, even for the undirected MST game. In general connection games, however, we can interpret UFL games as single source connection games on directed graphs. Hence, deciding the existence of SE is generally NP-hard.

Theorem 5. *Every MST game Γ has a SE that can be computed in polynomial time.*

Corollary 5. *It is* NP*-hard to decide if a given single source connection game Γ on a directed graph has a SE.*

3 Strong Equilibria beyond Linear Programming

Connection Games. For set cover and facility location games the integrality gap condition provides a complete characterization of games Δ having core solutions. We obtain a complete characterization also for the existence of SE in strategic games Γ. For network design games like the connection game, the LP argument is sufficient to show non-emptiness of the core but not necessary. A tight characterization of games with

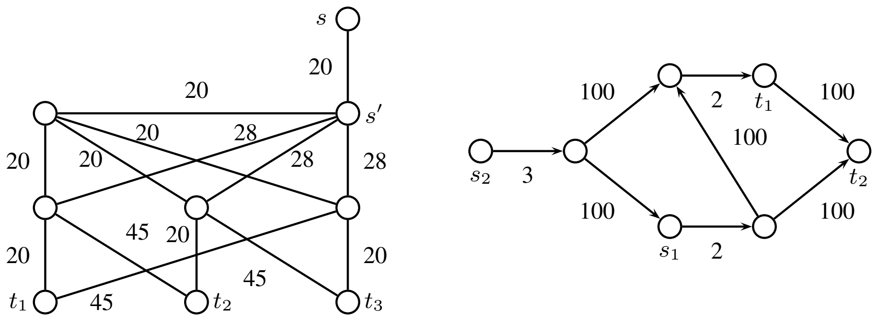


Fig. 1. Left: A single source connection game with 3 players, a non-empty core, but without a SE. R^* is an MST of G and consists of all edges of cost 20. Right: A multicut game on a directed graph with 2 players and a non-empty core. The game has no NE.

non-empty core has not been obtained so far. For strategic games and SE a single source connection game without SE is given in [14], but coaltional game in their example has an empty core. Coalitional connection games with an empty core (and thus without SE) have already been presented in [19]. We here show that even a spanning property of the optimum solution R^* is not sufficient to guarantee SE existence or to obtain SE from core solutions. The relation between core and SE here is not as robust as for games considered previously.

Lemma 1. *There are corresponding strategic and coalitional single source connection games Γ and Δ such that R^* is a MST of G and Δ has a core solution but Γ has no SE.*

Proof. Our example game is shown in Fig. 1. It is based on a game presented in [19], which consisted only of the three lower layers up to node s'. It was shown that this game has an empty core, but R^* passes through all vertices of G. This also implies that there can be no SE.

To obtain our game in Fig. 1, we added the new source s and an edge of cost 20 to the old source s'. Then the constraints for the contributions of the coalitions allow a feasible cost sharing by assigning each player a share of $160/3 \approx 53.33$. This removes the incentives to deviate on a global scale, which is sufficient for non-emptiness of the core. On a local scale, however, the instable structure up to s' is still intact. The additional contributions towards (s', s) do not change the strategic incentives within the lower parts of the graph. It can be verified that in this game no SE exists. This proves the lemma. □

Terminal Backup Games. In terminal backup games [3, 6] there is a graph $G = (V, E)$, each player is a vertex ($K \subset V$), and resources are the edges with costs $c(e) \geq 0$. Each player strives to be connected to at least $d - 1$ other player vertices, for $d \geq 2$. The terminal backup problem can be solved in polynomial time for $d = 2$ [6]. Here we show that every core solution can be turned into a SE for these games. In addition, we show how to decide if a game has a SE and how to obtain SE in polynomial time if they exist.

Theorem 6. *A terminal backup game Δ with $d = 2$ has a non-empty core if and only if the strategic game Γ has a SE. In the case of $d = 2$ there is a polynomial time algorithm to determine Γ has a SE and to compute one in polynomial time if it exists.*

For $d \geq 4$, a core solution cannot be turned into a SE. In fact, an example game can be derived directly from the single source connection game in Fig. 1 above. We simply replace the source s by a clique of 4 or more terminals and 0-cost edges.

Lemma 2. *For any $d \geq 4$ there is a coalitional terminal backup game Δ with a core solution and a corresponding strategic game Γ without a strong equilibrium.*

Network Cutting Games. In this section we briefly treat network cutting games, in which there is a graph $G = (V, E)$ and each player i strives to disconnect $S_i \subset V$ from $T_i \subset V$. Each edge $e \in E$ has a cost $c(e) > 0$ for disconnection. This approach yields coalitional and strategic games based on a variety of minimum-cut problems like s-t-cut, multicut, mulitway cut, etc. It is introduced and studied with respect to NE in the special cases of mulitway cut and multicut in [2]. More formally, for each player i denote by $\mathcal{P}_i$ the set of all paths in G from a node in S_i to a node in T_i. When we introduce a variable x_e for each edge $e \in E$, then for each path $P \in \mathcal{P}_k$ player k has the constraint $\sum_{e \in P} x_e \geq 1$. The resulting integer program is a special case of the set cover integer program presented above. This implies that if the integrality gap is 1, we have existence of core solutions and SE – e.g., for directed and undirected graphs and single-source games with $S_i = \{s\}$ for every $i \in K$.

Theorem 7. *If the Covering-LP has an integral optimum solution, then the strategic network cutting game Γ has a SE.*

There is a subtle twist to this result. While in the set cover game every element (i.e., every path) is a player, in the cutting game players strive to cover multiple elements (i.e., cut multiple paths). By clustering elements we simply reduce the granularity of possible coalitions to those, which can be obtained by the union of sets $\mathcal{P}_i$. Thereby, we enlarge the sets of games that allow a SE and a core solution.

Proposition 1. *There are network cutting games Γ with SE, for which the underlying network cutting problem has an integrality gap of more than 1.*

A similar observation can be made for multiway cut games, in which geometric LP relaxations [9] have an integrality gap of more than 1. In fact, it can be observed that the arguments from [2] for existence of optimal NE can be extended in a straightforward way to show that every mutiway cut game on an undirected graph admits a SE. In general network cutting games, however, the set of strategic games with a SE is not equivalent to the set of cooperative games with a non-empty core.

Lemma 3. *There are corresponding coalitional and strategic network cutting games Δ and Γ such that Δ has a core solution but Γ has no SE.*

Proof. For undirected graphs we consider two players and a star graph. We set $S_1 = \{s_1\}$, $S_2 = \{s_2\}$, $T_1 = \{t_1\}$ and $T_2 = \{s_1, t_1\}$. The edge costs to the center node u are $c(s_1, u) = c(t_1, u) = 2$ and $c(s_2, u) = 3$. The set of core solutions is $\gamma_1 = 2 - \epsilon$ and $\gamma_2 = 2 + \epsilon$ for $0 \leq \epsilon \leq 1$. Note that the unique optimum solution is to cut (s_1, u) and (t_1, u). In such a solution, however, if $|s_1| > 0$, player 1 can unilaterally improve by removing the larger of his payments. Player 2 does not pay for both edges, because paying only for (s_1, u) is cheaper.

For directed graphs we can even leave $T_2 = \{t_2\}$ as a singleton. We transform the graph to the one shown in Fig. 1. A similar argument shows non-existence of SE. □

This implies that relaxing the assumption that *every* element or terminal is a player in a set cover or facility location game harms the equivalence between core and SE. On another note, the proof shows absence of NE in general strategic network cutting games on undirected games. For directed graphs the absence of NE holds true even for minimum multicut games, in which S_i and T_i are singleton sets for all players $i \in K$.

4 Fractional and Non-binary Resources

Apart from equivalence of core and SE, another issue is to see when we can derive SE from core solutions, which is possible with an integrality gap of 1 in all games described above. All LPs studied here consist of linear constraints of the following type. One type is $\sum_i x_i \geq 1$, i.e., a simple covering constraint with 0/1 coefficients. The other type is $y_i - \sum_j x_{ij} \geq 0$, i.e., a coordination constraint that requires a resource to become bought when at least one player uses it. This second type allows to treat facility location and network design games. What happens if we slightly generalize these constraints?

Let us first consider dropping the integrality requirement. It is simple to show, for instance, that vertex cover games always allow a core solution if vertices can be bought *fractionally*. Does a SE also exist for strategic games in these cases? The obvious adjustment in the strategic game is to define the bought fraction proportional to the total payment. In a state s of the strategic *fractional vertex cover game* a vertex v is bought to the degree $x_v = \sum_{k \in K} s_k(e)/c(v)$. For a player k corresponding to edge $e = (u, v)$ the individual cost is $|s_k|$ if $x_u + x_v \geq 1$ and prohibitively large otherwise.

A second, closely related variant is to increase the covering requirements and allow multiple integer units of a resouce to be bought. Here the constraints become $\sum_i x_i \geq b$, where $b > 0$ and $x_i \in \mathbb{N}$. Again, the total payments of the players determine the number of units bought of a resource. We term these games *non-binary vertex cover games*. More formally, in a state s we have $x_u = \lfloor \sum_{k \in K} s_k(u) \rfloor$. Player k corresponding to edge (u, v) has a required coverage of $b_k \in \mathbb{N}$ and individual cost $c_k(s) = |s_k|$ if $x_u + x_v \geq b_k$ and prohibitively large otherwise.

For both classes SE being core solutions and an SPoA of 1 continue to hold. In contrast, we show that there might be no SE – although non-emptiness of the core can be established via the same linear programming machinery that was used before.

Theorem 8. *There are corresponding strategic and coalitional fractional or non-binary vertex cover games Δ and Γ such that Δ has a core solution but Γ has no SE.*

Proof. For both variants we use a triangle, vertex costs $c(u) = 3$, $c(v) = 5$, and $c(w) = 7$, and players 1 to 3 corresponding to edges (u, w), (u, v) and (v, w), respectively.

In the fractional game the unique optimum solution to the underlying vertex cover problem is $x^*_u = x^*_v = x^*_w = 1/2$, and the unique core solution is $\gamma_1 = 2.5$, $\gamma_2 = 0.5$ and $\gamma_3 = 4.5$. Note that x^* has to be purchased in every SE, and obviously $s_2(w) = 0$. If $s_1(w) > 0$, player 1 can deviate unilaterally and achieve the amount $s_1(w)/7$ of coverage by contribution to u with less payments. The same holds for player 2 and vertex v. This proves that there is no SE.

For the non-binary version, we set all covering requirements to $b_1 = b_2 = b_3 = 4$. Then the unique optimum x^* to the underlying vertex cover problem and the unique core payments γ are the same as before scaled by factor 4. Observe that we have an integrality gap of 1 in this game. The core solution is unique, so we know that in every SE $|s_1| = 10$ and $|s_2| = 2$. This implies $4 \leq s_1(w) \leq 6$. By removing this payment from w, player 1 reduces the number of units bought of w by exactly 1. However, he can obtain an additional unit of u at a cost of 3. This proves the theorem. □

5 Approximate Equilibria

A disadvantage of the concept of SE is that they might not exist in a game. However, our LP approach proves to be applicable even for obtaining approximate SE. With primal-dual algorithms we can compute (α, β)-SE with small (constant) ratios in polynomial time for vertex cover, set cover, and facility location games.

Theorem 9. *There are efficient primal-dual algorithms to compute* $(2, 2)$*-SE for vertex cover,* (f, f)*-SE for set cover (where* f *is the maximum frequency of any element in the sets), and* $(3, 3)$*-SE for metric UFL games in polynomial time.*

6 Open Problems

We believe that the linkage between core and SE could be present in other cost sharing games, which go beyond the classes of games treated in this paper. Exploring these classes of games is an interesting avenue for further research. More concretely, our games have LP formulations within Owens linear production model [24]. Non-emptiness of the core, however, can also be shown within a more general class of problems termed generalized linear production model in [17]. It has a non-additive structure, and it encompasses for instance the cut-based LP-formulation for Steiner Network problems. It is a fascinating open problem to see if this framework can also be used to derive exact and approximate SE in strategic cost sharing games.

Acknowledgement. The author would like to thank Elliot Anshelevich and Bugra Caskurlu for valuable discussions and feedback about the results in this paper.

References

1. Andelman, N., Feldman, M., Mansour, Y.: Strong price of anarchy. Games Econom. Behav. 65(2), 289–317 (2009)
2. Anshelevich, E., Cascurlu, B., Hate, A.: Strategic multiway cut and multicut games. In: Proc. 8th Intl. Workshop Approximation and Online Algorithms, WAOA (to appear, 2010)

3. Anshelevich, E., Caskurlu, B.: Exact and approximate equilibria for optimal group network formation. In: Fiat, A., Sanders, P. (eds.) ESA 2009. LNCS, vol. 5757, pp. 239–250. Springer, Heidelberg (2009)
4. Anshelevich, E., Dasgupta, A., Kleinberg, J., Roughgarden, T., Tardos, É., Wexler, T.: The price of stability for network design with fair cost allocation. SIAM J. Comput. 38(4), 1602–1623 (2008)
5. Anshelevich, E., Dasgupta, A., Tardos, É., Wexler, T.: Near-optimal network design with selfish agents. Theory of Computing 4, 77–109 (2008)
6. Anshelevich, E., Karagiozova, A.: Terminal backup, 3D matching, and covering cubic graphs. In: Proc. 39th Symp. Theory of Computing (STOC), pp. 391–400 (2007)
7. Aumann, R.: Acceptable points in general cooperative n-person games. In: Contributions to the Theory of Games IV. Annals of Mathematics Study, vol. 40, pp. 287–324. Princeton University Press, Princeton (1959)
8. Bird, C.: On cost allocation for a spanning tree: A game theoretic approach. Networks 6, 335–350 (1976)
9. Calinescu, G., Karloff, H., Rabani, Y.: An improved approximation algorithm for multiway cut. J. Comput. Syst. Sci. 60(3), 564–574 (2000)
10. Cardinal, J., Hoefer, M.: Non-cooperative facility location and covering games. Theoret. Comput. Sci. 411(16–18), 1855–1876 (2010)
11. Chekuri, C., Chuzhoy, J., Lewin-Eytan, L., Naor, J., Orda, A.: Non-cooperative multicast and facility location games. IEEE J. Sel. Area Comm. 25(6), 1193–1206 (2007)
12. Chen, H.-L., Roughgarden, T., Valiant, G.: Designing network protocols for good equilibria. SIAM J. Comput. 39(5), 1799–1832 (2010)
13. Deng, X., Ibaraki, T., Nagamochi, H.: Algorithmic aspects of the core of combinatorial optimization games. Math. Oper. Res. 24(3), 751–766 (1999)
14. Epstein, A., Feldman, M., Mansour, Y.: Strong equilibrium in cost sharing connection games. Games Econom. Behav. 67(1), 51–68 (2009)
15. Goemans, M., Skutella, M.: Cooperative facility location games. J. Algorithms 50(2), 194–214 (2004)
16. Goemans, M., Williamson, D.: A general approximation technique for constrained forest problems. SIAM J. Comput. 24(2), 296–317 (1995)
17. Granot, D.: A generalized linear production model: A unifying model. Math. Prog. 34, 212–222 (1986)
18. Granot, D., Huberman, G.: On minimum cost spanning tree games. Math. Prog. 21, 1–18 (1981)
19. Granot, D., Maschler, M.: Spanning network games. Intl. J. Game Theory 27, 467–500 (1998)
20. Hoefer, M.: Non-cooperative tree creation. Algorithmica 53(1), 104–131 (2009)
21. Hoefer, M.: Strategic cooperation in cost sharing games. arXiv 1003.3131 (March 2010)
22. Immorlica, N., Mahdian, M., Mirrokni, V.: Limitations of cross-monotonic cost sharing schemes. ACM Trans. Algorithms 4(2) (2008); Special Issue SODA 2005
23. Könemann, J., Leonardi, S., Schäfer, G., van Zwam, S.: A group-strategyproof cost sharing mechanism for the Steiner forest game. SIAM J. Comput. 37(5), 1319–1341 (2008)
24. Owen, G.: On the core of linear production games. Math. Prog. 9, 358–370 (1975)
25. Prodon, A., Libeling, T., Gröflin, H.: Steiner's problem on two-trees. Technical report, Départment de Mathemátiques, EPF Lausanne (1985); Working paper RO 850315
26. Tamir, A.: On the core of network synthesis games. Math. Prog. 50, 123–135 (1991)

Coalition Formation and Price of Anarchy in Cournot Oligopolies

Nicole Immorlica[1], Evangelos Markakis[2,⋆], and Georgios Piliouras[3,⋆⋆]

[1] Northwestern University, Dept. of EECS, Evanston, IL, USA
nicimm@gmail.com
[2] Athens Univ. of Economics and Business, Dept of Informatics, Athens, Greece
markakis@gmail.com
[3] Georgia Institute of Technology, Dept. of EE, Atlanta, GA, USA
John Hopkins University, Dept. of Economics, Baltimore, MD, USA
georgios.piliouras@gmail.com

Abstract. Non-cooperative game theory purports that economic agents behave with little regard towards the negative externalities they impose on each other. Such behaviors generally lead to inefficient outcomes where the social welfare is bounded away from its optimal value. However, in practice, self-interested individuals explore the possibility of circumventing such negative externalities by forming coalitions. What sort of coalitions should we expect to arise? How do they affect the social welfare?

We study these questions in the setting of Cournot markets, one of the most prevalent models of firm competition. Our model of coalition formation has two dynamic aspects. First, agents choose strategically how to update the current coalition partition. Furthermore, coalitions compete repeatedly between themselves trying to minimize their long-term regret. We prove tight bounds on the social welfare, which are significantly higher than that of the Nash equilibria of the original game. Furthermore, this improvement in performance is robust across different supply-demand curves and depends only on the size of the market.

1 Introduction

It is a basic tenet of algorithmic game theory that agents act selfishly in the pursuit of their own interests. Borrowing from economics, the literature purports that these agents will take actions that lead to a Nash equilibrium (or a related solution concept). Hence the actions could be potentially far from the social optimum. For example, in a road network, each driver, observing traffic patterns, selects the route which minimizes his own delay. The resulting total delay can be much greater than that of the optimal flow.

⋆ Partially supported by the EU FP7 Network of Excellence Euro-NF.
⋆⋆ Supported by NSF grants CCF-0325453, AF-0910940, AFOSR grant FA9550-09-1-0420 and ONR grant N00014–09-1-0751.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 270–281, 2010.

In a seminal paper in 1999, Koutsoupias and Papadimitriou [16] initiated the investigation of the so-called *price of anarchy* which measures the ratio of the social value in the worst-case equilibrium to the optimal social value. Recent years have seen a profusion of results exploring the price of anarchy of various non-cooperative games. The traffic example mentioned above, known as *selfish routing* in the literature, has a bounded price of anarchy of 4/3 for linear latency functions [21]. This can be viewed as a positive result. However, many settings have a drastically large price of anarchy, e.g., Cournot oligopoly games, which model competition between firms, have a linear price of anarchy for certain production functions [15].

The pursuit of self-interest however may very well encourage cooperation between agents. Such cooperation will almost certainly alter the set of stable outcomes. In an attempt to understand the implications of these issues on the price of anarchy, recent papers have studied the quality of outcomes which are stable against either all possible coalitions [2] or against arbitrary but exogenously defined coalition structures[14],[10] (e.g., the worst possible partitioning of the agents). The effect of coalition formation on social welfare has been shown to be extremely unpredictable ranging anywhere from significant improvements [2], to slight changes [10], all the way to vast degradation [14].

Tackling the issue of cooperation is pivotal in making accurate predictions about the quality of stable outcomes, especially in settings where coalitions are likely to arise. However, the theoretical models that have been introduced so far, focus mostly on the extreme cases, where either any coalition is enforceable or an arbitrary, static coalition structure is exogenously defined. Here, we introduce a model that allows for *strategically evolving* coalition structures and we examine how endogenously formed coalitions affect the quality of stable outcomes.

We focus on the setting of oligopolistic (Cournot) markets, where coalitions are known to arise in practice and we define a coalition formation game on top of the market that captures the dynamic evolution of cooperation. In our coalition formation game actions correspond to changes in the current coalition structure, hence the strategy space of the game evolves over time. Specifically, a new coalition can be created by a merger between two or more existing coalitions. An existing coalition can also be destroyed due to a deviation of a subset of its current players who decide either to form a coalition by themselves or join an existing coalition. For a new coalition to be formed, it must be the case that its creation benefits all its members.

Given a current coalition structure, we treat each coalition as a super-player who, as in [14], acts on behalf of its members and tries to maximize its aggregate utility. Any such game between the super-players (coalitions) has Nash equilibria and in the case of Cournot oligopolies we show that the utilities of the super-players at Nash equilibria are unique. This defines the value of a coalition given the current partition, which is reminiscent of the approach in [20]. Finally we divide this utility equally among the members of a coalition, since in symmetric Cournot games all players have equal production costs.

Given the rules of the game described above, we are interested in stable coalition configurations, i.e., partitions where no profitable deviating actions exist with regard to the allowed actions we have defined. We analyze the social welfare of the worst such stable partition and compare it to the cost of the optimum and refer to this ratio as the price of anarchy of our coalition formation game. We find that the price of anarchy of our coalition formation game for Cournot oligopolies is $\Theta(n^{2/5})$, where n is the number of firms that participate in the market, implying a significant improvement of the actual price of anarchy of Cournot oligopolies which is $\Theta(n)$.

The value assignment to coalitions, as described in the previous paragraphs relies on the assumption that if a coalition structure is stable and hence not transient, then the super-players coalitions will reach a Nash equilibrium. We show that we can weaken this assumption considerably. Specifically, we can show that if the coalitions participate in the Cournot oligopoly repeatedly in a fashion that minimizes their long term regret then the average utility of the super-players (coalitions) will converge to their levels at Nash equilibria. Regret compares the average utility of a player to that of the best fixed constant action with hindsight. Having no-regret means that no deviating action would significantly improve the firm's utility. Several learning algorithms are known to provide such guarantees ([3,23] and references therein). More importantly, the assumption is not tied to any specific algorithmic procedure, but instead captures successful long-term behavior. Finally, since the setting of oligopolies markets is in its nature repeated, this observation significantly strengthens the justification of our model.

Paper Structure. Section 2 offers the definition of Cournot oligopolies and a detailed exposition of our coalition formation model. In Section 3 we prove tight bounds for the price of anarchy of the Cournot coalition formation game. Finally, Section 4 extends our analysis to the case of no-regret behavior.

1.1 Related Work

Quantifying the inefficiency of outcomes when coalitions are allowed to form has been the subject of much recent work. In [14], the authors initiate the study of the *price of collusion*, which is a measure of the inefficiency of the worst possible partition of the set of players. In [10], both the quality and tractability of stable outcomes is examined in atomic congestion games with coalitions. The models above do not raise any strategic issues in the formation of the coalitions, which are essentially exogenously enforced upon the game. In contrast, we focus on a strategic setting, where we study only stable partitions, i.e., partitions where agents have no incentive to deviate, as we define in Section 2.

Other notions of inefficiency have also been analyzed. In [2], the authors analyze the *price of strong anarchy*, i.e., the inefficiency of Nash equilibria which are resilient to deviations by coalitions[1]. In [5], a different measure is introduced, namely the *price of democracy.* This notion captures the inefficiency of a given

[1] Unfortunately, for several classes of games including Cournot markets, strong Nash equilibria do not exist.

coalition formation process (e.g. a bargaining process) with respect to a cooperative game. The authors study this notion in the context of weighted voting games for certain intuitive bargaining processes. Hence the inefficiency is measured with regard to the arising partitions in the subgame perfect equilibria of the corresponding bargaining game.

Regarding Cournot games, it has been long known that the loss of efficiency at Nash equilibria can be quite high. Earlier studies focused on empirical analysis [12] whereas more recently, price of anarchy bounds have been obtained in [11,15]. Collusion and cartel enforcement in Cournot games have been studied experimentally, see e.g., [22]. Mechanism design aspects of collusion have also been explored, see [7]. For more on Cournot games and their variants, we refer the reader to [17].

Dynamic coalition formation has been studied extensively both in the economics as well as in the computer science literature. We refer the reader to [8,4,13,19] and [1][Section 5.1] as well as the numerous references therein. The main goals of these works have been to provide appropriate game theoretic solution concepts (both from a cooperative and noncooperative point of view) and to design intuitive procedures that converge experimentally or theoretically to such solution concepts.

Conceptually, the closest example to our approach that we know of, is the work of Ray and Vohra in [20]. The authors propose a solution concept ("binding agreement") that allows for the formation of coalition structures and examine the inefficiency of stable partitions. Unlike in our work, their deviations can only make the existing coalition structure finer- never coarser. In the case of symmetric Cournot games, it is shown that there always exists a stable partition with social welfare $O(\sqrt{n})$ worse than the optimal. However, the social welfare of the worst stable partition is always at least as bad as that of the worst Nash. In follow-up work [19], Ray analyzes a class of bargaining processes which assumes players with infinite foresight and shows that in symmetric Cournot games the only coalition structure that is stable for all of them, has social welfare $\Theta(\sqrt{n})$ worse than the optimal.

Finally, there has been some recent work on the behavior of no-regret algorithms in Cournot oligopolies. In [9], [18] several convergence results are shown for different classes of Cournot oligopolies. To our knowledge our paper is the first to consider the behavior of *coalitions* which are behaving in a no-regret fashion in any kind of setting.

2 The Model

We will demonstrate the main point of our work in the context of Cournot games. The definitions presented in this Section can be easily generalized and applied to other contexts but we postpone a more general treatment for an extended version.

Cournot games describe a fundamental model of competition between firms. They were introduced by Cournot in his much celebrated work [6]. In Cournot games, firms control their production levels and by doing so influence the market prices. In the simplest Cournot model all the firms produce the same good; the demand for this product is linear in the total production (i.e. the price decreases linearly with total production); the unit cost of production is fixed and equal across all firms. The revenue of a firm is the product of the firm's part of the market production times the price. Finally, the utility of a firm is equal to its revenue minus its total production cost. Overproducing leads to low prices, while at the same time an overly cautious production rate leads to a small market share and reduced revenue. The balancing act between these two competing tendencies is known to give rise to a unique Nash equilibrium. More formally:

Definition 1. *A linear and symmetric Cournot oligopoly is a noncooperative game between a set $N = \{1, 2, ..., n\}$ of players (firms), all capable of producing the same product. The strategy space of each firm is $\mathbb{R}_+$, corresponding to the quantity of the product that the firm decides to produce. Given a profile of strategies, $q = (q_1, ..., q_n)$, the utility of firm i is $u_i(q) = q_i p(q) - cq_i$, where $p(q)$ is the price of the product, determined by $p(q) = \max\{0, a - b \sum_i q_i\}$, for some parameters a, b, and c is a production cost, with $a > c$.*

Proposition 1 ([6]). *In the unique Nash equilibrium of a Cournot oligopoly with n players, the production level is the same for all players and equal to $q_i = q^* = \frac{(a-c)}{b(n+1)}$. The utility of each player is equal to $u_i = \frac{(a-c)^2}{b(n+1)^2}$ and the social welfare is equal to $\frac{(a-c)^2 n}{b(n+1)^2}$.*

2.1 Cournot Games with a Fixed Partitioning of the Players

Suppose now that the players are given the opportunity to form coalitions and sign agreements with other firms, as a means of reducing competition and improving on their welfare. Given a partition of the players into coalitions, we can think of the new situation as a super-game whose super-players are the coalitions themselves. The strategy for a coalition, or super-player, is now a vector assigning a strategy to each of its members. The payoff to the super-player is the aggregate payoff its members would achieve with their assigned strategies in the original game. This definition can be used to model coalitions in general games as in [10,14].

Definition 2. *Let $\mathcal{G}$ be a game of n players, with A_j being the set of available actions and $u_j^{\mathcal{G}}(a_1, \ldots, a_n)$ the utility function for each player j. Given a partitioning $\Pi = (S_1, ..., S_k)$ of the players, then the corresponding super-game consists of the following:*

- *k super-players*
- *The strategy set for super-player S_i is the set of vectors $\overrightarrow{a}_{S_i} \in \prod_{j \in S_i} A_j$.*
- *The utility of super-player S_i is $u_{S_i}(\overrightarrow{a}_{S_1}, \ldots, \overrightarrow{a}_{S_k}) = \sum_{j \in S_i} u_j^{\mathcal{G}}(a_1, \ldots, a_n)$ where a_j is the strategy assigned to player j by his coalition S_i in the coalition's strategy $\overrightarrow{a}_{S_i}$.*

It is straightforward to check that for Cournot games, the super-game with k super-players is essentially equivalent to a Cournot game with k players[2].

Lemma 1. *Consider a Cournot oligopoly super-game for a fixed partitioning* $\Pi = (S_1, ..., S_k)$ *of players. The players' utilities and the social welfare in this game under any strategy profile* $\overrightarrow{q}_{S_1}, \ldots, \overrightarrow{q}_{S_k}$ *(where* $\overrightarrow{q}_{S_i} \in \mathcal{R}_+^{|S_i|}$*) are equal to the corresponding utilities and social welfare of a linear and symmetric Cournot game with* k *players where each player* i *produces the aggregate production* $\sum_{j \in S_i} (\overrightarrow{q}_{S_i})_j$ *of the corresponding coalition* S_i*. Furthermore, a strategy profile for the super-game with the fixed partitioning is a Nash equilibrium if and only if the* k*-tuple of the aggregate levels of productions for each coalition is the unique Nash equilibrium for the Cournot game on* k *players (without coalitions).*

Lemma 1 allows us to use theorems regarding Cournot games to study the Nash equilibria and welfare of Cournot games with coalitions. Specifically, it implies that the social welfare is the same in all Nash equilibria of the Cournot game with a fixed partitioning. Hence we can define the price of anarchy as the ratio of this social welfare over the optimal social welfare, which is realized when all agents unite into a single coalition. By combining proposition 1 with lemma 1 we derive:

Lemma 2. *The price of anarchy of a Cournot oligopoly with a fixed partition* $\Pi = (S_1, ..., S_k)$ *is* $\frac{(k+1)^2}{4k}$.

As a consequence, the price of anarchy in the original noncooperative Cournot oligopoly with n players is very high, namely linear in the number of players, as has been observed previously [15].

Corollary 1. *The price of anarchy in the original Cournot game with* n *players, where no coalitions are allowed to form is* $\Theta(n)$.

2.2 Cournot Coalition Formation Games

Next, we move away from the fixed coalition structure assumption and instead we will allow the players to dynamically form coalitions. We will call this game the *Cournot coalition formation game*. Given some initial partition, players or sets of players can consider deviations according to the rules that we define below. As we have seem by Lemma 1, for any resulting partition, say with k coalitions, the utility of each coalition is unique in all Nash equilibria of the Cournot game with fixed coalitions, and equal to the utility of a player in the unique Nash equilibrium of a symmetric Cournot game with k players. In the coalition formation game, each of the n players, when evaluating a possible action of hers, estimates her resulting utility to be equal to her equiproportional share of the Nash equilibrium utility of the coalition to which she belongs, given the resulting coalition structure. More formally:

[2] Henceforth, when it is clear from the context, we will use game instead of super-game and player instead of super-player.

Definition 3. *We define a coalition formation game on top of a symmetric Cournot game to consist of the following:*

- *n players and a current partitioning of them into k coalitions $\Pi = (S_1, ..., S_k)$*
- *Given the current partition Π, the allowed moves (deviations) that players can use along with the consequences for the coalition left behind (i.e., the non-deviators) are as follows:*

  Type 1: *A subset S'_i of a current coalition S_i decides to deviate and form a new coalition. The rest of the members, if any, of the original coalition (i.e. S_i/S'_i) dissolve into singletons.*[3]

  Type 2: *A strict subset S'_i of a current coalition S_i decides to leave its current coalition S_i and join another coalition of Π, say S_j. The rest of the members of the original coalition (i.e. S_i/S'_i) dissolve into singletons.*

  Type 3: *A set of coalitions of Π decide to unite and form a coalition. The rest of the coalitions remain as they were.*
- *Given a partition Π and a player i in coalition S_j of Π, denote by $u_{S_j}(\Pi)$, the uniquely defined Nash utility of coalition S_j in the symmetric Cournot game with fixed coalition structure Π. The utility of player i in this case, is defined to be equal to $u_{S_j}(\Pi)/|S_j|$.*

In terms of our assumptions about allowable actions, unlike the work of [20], we allow both the creation as well as the destruction of coalitions. Furthermore, we assume that in some of the deviating actions (Type 1 and 2), the leftover coalition from where the deviation emerged, dissolves into singletons. This is reminiscent of past approaches[7,13]. Essentially, our assumption encodes that non-deviators will react cautiously.

We will be interested in analyzing the price of anarchy for partitions in which no player or set of players has an incentive to change the current coalition structure. In order to characterize stable coalitions, we need to define when a deviation is successful. A deviation is successful if and only if the utility of all the players that induce this deviation strictly increases as a result. More formally:

Definition 4. *A deviation is successful iff all the players that facilitate the deviation strictly increase their payoff by doing so. Specifically, a deviation of*

- Type 1 *is successful iff all the players in S'_i increase their payoffs.*
- Type 2 *is successful iff all the deviating players in S'_i as well as all the members of the coalition S_j who accept them increase their payoffs.*
- Type 3 *is successful iff all the members of all the merging coalitions increase their payoffs.*

Definition 5. *A partition Π is stable if there exists no successful deviation of any type.*

[3] This type of actions also includes the non-action option (i.e. the coalition structure remains unaltered), when $S'_i = S_i$.

In the usual manner of the "price of anarchy" literature, we are interested in bounding the ratio of the social welfare of the worst stable outcome (i.e. coalition partition) divided by the optimal social welfare. In our setting, the stable outcomes do not correspond exactly to Nash, since we allow bilateral moves (e.g. type 3). Nevertheless, we will still use the term price of anarchy to refer to this ratio, since it characterizes the loss in performance due to the lack of a centralized authority that could enforce the optimal (grand) coalition.

Definition 6. *Given a Cournot coalition formation game, we define the price of anarchy as the ratio of the social welfare that is achieved at the worst stable partition divided by the optimal social welfare.*

3 The Main Result

The starting point of our work is the observation of Corollary 1 that without coalition formation the price of anarchy is $\Theta(n)$. Hence our goal is to understand the quality of the worst stable partition structure and compare it to the optimal. The optimal partition structure is trivially the one where all players have united in a single coalition, as there is no competition in such a setting. Our main result is that the price of anarchy is significantly reduced when coalition formation is allowed. Formally:

Theorem 1. *The price of anarchy of the coalition formation game is* $\Theta(n^{2/5})$.

3.1 The Proof of the Upper Bound

We begin by proving that the price of anarchy is $O(n^{2/5})$. We will first establish this upper bound on a restricted version of our model. In particular, we restrict each type of the allowed deviations of Definition 3 as follows:
Type 1: A member of a coalition of Π, decides to form a singleton coalition on his own. The coalition from which the player left dissolves into singleton players.
Type 2: A member of a coalition of Π decides to leave its current coalition S_i (where $|S_i| \geq 2$), and join another coalition of Π, say S_j. The rest of coalition S_i dissolves into singleton players.
Type 3: A set of singleton players of Π decide to unite and form a coalition.

We will refer to this game as the *restricted coalition formation game*. Once we establish the upper bound in the restricted model, it is trivial to extend it to the general model since the set of stable partitions only gets smaller in the general model. To analyze the price of anarchy, we will derive a characterization of the stable partitions. Throughout the analysis we will normally denote the cardinality of a coalition S_i by $s_i = |S_i|$. The first Lemma below says that for coalitions of size at least 2, its members need only consider Type 1 deviations.

Lemma 3. *Consider a partition* $\Pi = (S_1, ..., S_k)$, *with* $k \geq 2$. *For a player that belongs to a coalition of* Π *of size at least* 2, *the most profitable deviation (though not necessarily a successful one) is the deviation where the player forms a singleton coalition on his own.*

Proof. Consider a coalition S_i of Π of size s_i. Suppose $s_i \geq 2$ and consider a player $j \in S_i$. The available deviations for j are either to form a coalition on his own or to join an existing coalition. In the former case, the coalition S_i will dissolve and the total number of coalitions in the new game will be $k + s_i - 1$. Hence the payoff of j will be $u = \frac{(a-c)^2}{b(k+s_i)^2}$. On the other hand, if j goes to an existing coalition, then S_i again dissolves but the total number of coalitions is now $k + s_i - 2$. Since j will be in a coalition with at least 2 members, the payoff to j will be at most: $u' \leq \frac{(a-c)^2}{2b(k+s_i-1)^2}$.

We wish to have $u \geq u'$. It suffices to show that $(k + s_i)^2 \leq 2(k + s_i - 1)^2$, which is equivalent to $(k + s_i)^2 - 4(k + s_i) + 2 \geq 0$. For this it suffices to show that $(k + s_i) \geq 2 + \sqrt{2}$. But we have assumed that $k \geq 2$ and that $s_i \geq 2$, hence the proof is complete. □

The next lemma is based on Lemma 3 and characterizes coalitions of size at least 2, for which there are no successful deviations for its members.

Lemma 4. *Consider a partition $\Pi = (S_1, ..., S_k)$, with $k \geq 2$. For a coalition S_i with $s_i \geq 2$, there is no successful deviation for its members iff $s_i \geq k^2$.*

Proof. Consider a coalition of partition Π, say S_i with $s_i \geq 2$. The payoff that a player in S_i now receives is $u = \frac{(a-c)^2}{s_i b(k+1)^2}$. By Lemma 3 the most profitable deviation for any player of S_i is to form a singleton coalition, in which case he would receive a payoff of $u = \frac{(a-c)^2}{b(k+s_i)^2}$. In order that no player has an incentive to deviate, we need that $(k + s_i)^2 \geq s_i(k + 1)^2$, which is equivalent to $s_i \geq k^2$. □

We now deal with deviations of players that form singleton coalitions in a partition Π. By definition, we only need to consider Type 3 deviations for singleton players.

Lemma 5. *Consider a partition $\Pi = (S_1, ..., S_k)$, with $k \geq 2$. Suppose that Π contains k_1 singleton coalitions with $k_1 \geq 2$, and k_2 non-singleton ones ($k_1 + k_2 = k$). The merge of the k_1 singletons is not a successful deviation iff $k_1 \leq (k_2 + 1)^2$.*

Proof. The k_1 singletons receive in Π a payoff of $(a - c)^2/(b(k_1 + k_2 + 1)^2)$. After the merge, their payoff will be $(a - c)^2/(k_1 b(k_2 + 2)^2)$. Hence, the merge will not be successful, iff $(a - c)^2/(b(k_1 + k_2 + 1)^2) \geq (a - c)^2/(k_1 b(k_2 + 2)^2)$. By manipulation of terms this is shown to equivalent to $(k_2 + 1)^2 \geq k_1$. □

Finally we show that for ensuring stability there is no need to consider any other Type 3 deviation of smaller coalitions.

Lemma 6. *Consider a partition $\Pi = (S_1, ..., S_k)$, with $k \geq 2$ and suppose that it contains k_1 singleton coalitions with $k_1 \geq 2$, and k_2 non-singleton ones. There is a successful Type 3 deviation iff the merge of all k_1 singletons is a successful deviation.*

Proof. One direction is trivial, namely if the merge of all k_1 singletons is a successful deviation. For the reverse direction, suppose there is a successful type

3 deviation which is not the merge of all the k_1 singletons. Let m be the number of players who merge and suppose $2 \leq m < k_1$. By arguing as in Lemma 5, we get that in order for the deviation to be successful, it should hold that $(k_1+k_2+1)^2 > m(k_1+k_2-m+2)^2$. Let $\lambda = k_2+1$ and $\theta = \lambda + k_1 - m$. Restating the condition in terms of λ and θ we get $(k_1+\lambda)^2 = (\theta+m)^2 > m(\theta+1)^2$, which via a rearranging of terms can be shown to be equivalent to $m > \theta^2$.

However, we have that $k_1 > m > (\lambda+k_1-m)^2 > \lambda^2 = (k_2+1)^2$. By Lemma 5, this means that the merge of all k_1 singletons is also a successful deviation. □

All the above can be summarized as follows:

Corollary 2. *Consider a partition Π. For $n \leq 2$, Π is stable iff it is the grand coalition. For $n \geq 3$, suppose $\Pi = (S_1, ..., S_k)$ with k_1 singleton coalitions and k_2 non-singleton ones. Then Π is stable iff it is either the grand coalition or the following hold:*

- $k_1 \leq (k_2+1)^2$.
- *For every non-singleton coalition S_i, $s_i \geq k^2$.*

Having acquired a characterization of the stable partitions, we can now analyze the (pure) price of anarchy of the restricted coalition formation game on top of a symmetric Cournot oligopoly. We omit the proof due to lack of space.

Theorem 2. *The price of anarchy of the restricted coalition formation game under symmetric Cournot oligopoly is $O(n^{2/5})$, where n is the total number of players.*

Finally, we come back to the original coalition formation game of Definition 3. Since in that game we have only enlarged the set of possible deviations with regard to the restricted coalition formation game, the set of stable partitions can only decrease. As a result, the price of anarchy for the original game is also $O(n^{2/5})$. This completes the proof for the upper bound of Theorem 1.

3.2 The Construction of the Lower Bound

The lower bound is obtained by the construction in the following lemma whose proof appears in the extended version of our paper:

Lemma 7. *For any N, let n be the number: $n = \lceil 4N^{4/5} \rceil \lfloor N^{1/5} \rfloor + \lfloor N^{2/5} \rfloor$. Consider a game on n players and a partition of the n players consisting of $k_1 = \lfloor N^{2/5} \rfloor$ singletons and $k_2 = \lfloor N^{1/5} \rfloor$ coalitions of size $s = \lceil 4N^{4/5} \rceil$ each. This coalition structure is stable for the Cournot coalition formation game.*

Since the total number of coalitions in the construction is $k = k_1 + k_2 \geq N^{2/5} = \Omega(n^{2/5})$, by Lemma 2, we obtain the desired lower bound.

Theorem 3. *For any number of players n, there exist stable partitions with cost $\Omega(n^{\frac{2}{5}})$ the cost of the optimal partition.*

4 Coalition Formation under No-Regret

So far, given a partition Π, we assign to each coalition S_i value equal to its uniquely defined utility at the Nash equilibria of the Cournot game with a fixed coalition partition Π. The reasoning behind this is that if the coalition partition Π is not transient, then the players/coalitions will hopefully reach a Nash equilibrium and hence their uniquely defined utility at it, is a good estimator of how much they value their current coalition partition.

Here, we argue that we can significantly weaken the assumption that the players/coalitions will reach an equilibrium. In fact, we will establish that if the coalitions participate in the Cournot oligopoly repeatedly in a fashion that minimizes their long term regret then their average utility will converge to their levels at Nash equilibria. The regret of an online learning algorithm[4] is defined as the maximum over all input instances of the expected difference in payoff between the algorithms actions and the best action. If this difference is guaranteed to grow sublinearly with time, we say it is a no-regret learning algorithm [3,23].

This notion captures successful long-term behavior and can be achieved in practice by several natural learning algorithms [3,23]. Putting all these together, we have that the values assigned to coalitions by our model, are in excellent agreement with the average utilities they would actually receive by participating repeatedly (and successfully) in the market.

Theorem 4. *Consider a Cournot oligopoly game with a fixed partitioning of the n players in k coalitions $\Pi = (S_1, ..., S_k)$. If all k (super-)players employ no-regret strategies, then their average utilities converge to their Nash levels.*

5 Discussion and Future Work

We have introduced a model of coalition formation and have identified tight bounds on the inefficiency of stable coalitions in oligopolistic markets. Our approach combines elements of cooperative game theory (e.g. payoff distribution amongst the members of the coalition) and noncooperative game theory (e.g. coalitions compete against each other). Such balancing acts of cooperation-competition are common in real-life economic settings and we believe that this work opens up a promising avenue for future research. The natural next step would be to examine the sensitivity of our results to changes in the underlying coalition formation process, as well as extensions to different classes of games.

References

1. Airiau, S.: Lecture Notes in Coalitional Games. In: 11th European Agent Systems Summer School, EASSS 2009 (2009), `http://staff.science.uva.nl/~stephane/Teaching/EASSS09/notes-coopGT-easss09.pdf`

[4] An online learning algorithm is an algorithm for choosing a sequence of elements of some fixed set of actions, in response to an observed sequence of cost functions mapping actions to real numbers. The t-th action chosen by the algorithm may depend on the first $t - 1$ observations but not on any later observations.

2. Andelman, N., Feldman, M., Mansour, Y.: Strong price of anarchy. In: ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 189–198 (2007)
3. Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning and Games. Cambridge University Press, Cambridge (2006)
4. Chalkiadakis, G.: A Bayesian Approach to Multiagent Reinforcement Learning and Coalition Formation under Uncertainty. PhD thesis, Department of Computer Science, University of Toronto, Canada (2007)
5. Chalkiadakis, G., Elkind, E., Polukarov, M., Jennings, N.R.: The price of democracy in coalition formation. In: International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 401–408 (2009)
6. Cournot, A.A.: Recherches sur les Principes Mathématiques de la Théorie des Richesses. Hatchette, Paris (1838); English translation: Researches into the Mathematical Principles of the Theory of Wealth. Macmillan, New York (1897)
7. Cramton, P., Palfrey, T.: Cartel enforcement with uncertainty about costs. International Economic Review 31(1), 17–47 (1990)
8. Dieckmann, T., Schwalbe, U.: Dynamic Coalition Formation and the Core (1998); Economics Department Working Paper Series, Department of Economics, National University of Ireland - Maynooth
9. Even-Dar, E., Mansour, Y., Nadav, U.: On the convergence of regret minimization dynamics in concave games. In: ACM Symposium on Theory of Computing, STOC (2009)
10. Fotakis, D., Kontogiannis, S., Spirakis, P.: Atomic congestion games among coalitions. ACM Trans. Algorithms 4(4), 1–27 (2008)
11. Guo, X., Yang, H.: The price of anarchy of cournot oligopoly. In: Deng, X., Ye, Y. (eds.) WINE 2005. LNCS, vol. 3828, pp. 246–257. Springer, Heidelberg (2005)
12. Harberger, A.: Monopoly and resource allocation. Amer. Econ. Review 44, 77–87 (1954)
13. Hart, S., Kurz, M.: Endogenous formation of coalitions. Econometrica 51, 1047–1064 (1983)
14. Hayrapetyan, A., Tardos, É., Wexler, T.: The effect of collusion in congestion games. In: ACM Symposium on Theory of Computing (STOC), pp. 89–98 (2006)
15. Kluberg, J., Perakis, G.: Generalized quantity competition for multiple products and loss of efficiency. In: 46th Annual Allerton Conference on Communication, Control, and Computing, pp. 930–936 (2008)
16. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. Computer Science Review 3(2), 65–69 (2009)
17. Mas-Colell, A., Green, J., Whinston, M.D.: Microeconomic Theory. Oxford University Press, Oxford (1995)
18. Nadav, U., Piliouras, G.: No regret learning in oligopolies: Cournot vs bertrand. In: 3rd International Symposium on Algorithmic Game Theory, SAGT (2010)
19. Ray, D.: A Game-Theoretic Perspective on Coalition Formation. Oxford Univ. Press, Oxford (2007)
20. Ray, D., Vohra, R.: Equilibrium binding agreements. Journal of Economic Theory 73, 30–78 (1997)
21. Roughgarden, T., Tardos, E.: How bad is selfish routing? J. ACM 49(2), 236–259 (2002)
22. Suetens, S., Potters, J.: Bertrand colludes more than Cournot. Experimental Economics 10, 71–77 (2007)
23. Young, H.: Strategic Learning and Its Limits. Oxford University Press, Oxford (2004)

An FPTAS for Bargaining Networks with Unequal Bargaining Powers

Yashodhan Kanoria⋆

Department of Electrical Engineering, Stanford University
ykanoria@stanford.edu

Abstract. Bargaining networks model social or economic situations in which agents seek to form the most lucrative partnership with another agent from among several alternatives. There has been a flurry of recent research studying Nash bargaining solutions (also called 'balanced outcomes') in bargaining networks, so that we now know when such solutions exist, and that they can be computed efficiently, even by market agents behaving in a natural manner.

In this work we study a generalization of Nash bargaining, that models the possibility of unequal 'bargaining powers'. This generalization was introduced in [12], where it was shown that the corresponding 'unequal division' (UD) solutions exist if and only if Nash bargaining solutions exist, and also that a certain local dynamics converges to UD solutions when they exist. However, the convergence time for that dynamics was exponential in network size for the unequal division case. Other approaches, such as the one of Kleinberg and Tardos, do not generalize to the unsymmetrical case. Thus, the question of computational tractability of UD solutions has remained open.

In this paper, we provide an FPTAS for the computation of UD solutions, when such solutions exist. On a graph $G = (V, E)$ with weights (i.e. pairwise profit opportunities) uniformly bounded above by 1, our FPTAS finds an ϵ-UD solution in time polynomial in the input and $1/\epsilon$. We also provide a fast *local* algorithm for finding ϵ-UD solution.

1 Introduction

Bargaining networks serve as a model for various social or economic interactions where agents seek to form pairs for mutual benefit (e.g. [7,21,14]). Situations which can be modeled as such include a housing market with buyers and sellers, a job market with job seekers and employers, or individuals seeking to form relationships and pair up. Bargaining networks are also referred to in the literature as 'assignment markets' [16] or 'exchange networks' [19,13].

A bargaining network is an undirected graph, with weights on the edges representing potential profits if the corresponding pair of agents 'trade' with each

⋆ Part of this work was done while the author was visiting Microsoft Research New England. The author is supported by a 3Com Corporation Stanford Graduate Fellowship.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 282–293, 2010.

other (see Section 1.1 for formal definitions). Profit from a trade is split between the participating agents as per a mutual agreement. Agents are constrained on the number of trades they can participate in. A natural postulate in this setting is that an outcome should be *stable*, i.e. no pair of agents should be able to do better by each abandoning a current partner and trading with each other instead. The solution concept of 'balanced outcomes' [16,7,13] postulates further that each pair of agents that trade must play the pairwise Nash bargaining solution, given the behavior of the rest of the network. Thus, the 'edge surplus' (cf. Eq. (1)), or the excess over the sum of 'best alternatives' for each of the two parties, is postulated to be split equally. This is called the *balance* condition.

However, it is natural to expect that such symmetry is rare in practice, and that some players tend to have greater 'bargaining power' than others. Such bargaining power can arise due to a variety of reasons. For example, a more patient player has more bargaining power, all else being equal. This phenomenon is well known in the Rubinstein game [17] where nodes alternately make offers to each other until an offer is accepted – the node with less time discounting earns more in the subgame perfect Nash equilibrium.

Empirical findings confirm that there is a lack of symmetry. A recent experimental study of bargaining networks [6] found that individual differences played a part in determining outcomes, including the observation that patience correlated positively with earnings. A previous study even estimates and 'corrects' for the effects of particular subject pairs to better uncover network structure effects [19]. This leads us to ask if the concept of 'balanced outcomes' can be suitably generalized to account for such asymmetry. It turns out that there is, in fact a simple generalization to the unsymmetrical case. Our previous work [12] introduced the generalized concept of unsymmetrical 'unequal division' (UD) solutions, and also characterized the existence of such solutions.

Somewhat surprisingly, the various algorithms devised to compute solutions in the symmetric setting fail to generalize to the unequal division setting (see also Section 1.2). For example, the algorithm of Kleinberg and Tardos [13] proceeds via a sequence of linear programs that maximize the minimum 'slack'. This does not seem to have a simple generalization to the asymmetric case. Thus, the question of computational tractability of solutions for the unsymmetrical case in bargaining networks has been open. We address this question.

Besides computational tractability, another important question is "Can a market find the solution concept on its own?" In this context, one looks for simple, local mechanisms that converge to a solution concept. Azar et al [1] and our recent work [12] do this for the bargaining networks problem. However, a crucial issue (see Section 4 of this paper) leads to a worst case exponential time to convergence in the unsymmetrical case for the algorithms proposed in [1,12]. In this paper we resolve this issue, providing a new efficient local algorithm for the unsymmetrical case.

Contributions. This work makes the following contributions in the context of bargaining networks:

– We establish computational tractability for bargaining networks with unequal bargaining powers by providing the first FPTAS for the corresponding 'unequal division' solutions.
– We provide a simple local algorithm and show that it converges fast to approximate unequal division solutions. Specifically, it is a two phase algorithm: (i) The first phase consists of finding the maximum weight matching and a stable allocation using belief propagation [4]. (ii) The second phase consists of unsymmetrical edge balancing of the allocation, converging to an approximate solution in polynomial time.

We note that the local algorithm we provide is similar to the one given by Azar et al [1] for the symmetric case. However, critical differences in both the design and the analysis of the algorithm enable us to overcome limitations of their approach.

1.1 Model

A bargaining network consists of an undirected graph $G = (V, E)$ with positive weights on the edges, denoted by $(w_e, e \in E) \in (0, W]^{|E|}$ (where $W > 0$ denotes an arbitrary bound on weights). Edges represent potential 'trades', and weights are the corresponding 'profits'. Players are constrained on the number of trades they are allowed to participate in. For simplicity, we will work with the *one exchange rule*, i.e. each player is allowed to participate in at most one trade. All our results easily generalize to the case of arbitrary integral constraints on number of trades for each player.

If a pair of players trade with each other, the profit must be divided between them. Thus, a *trade outcome* or just an *outcome* consists of a matching M between players, and an *allocation* $\underline{\gamma} \in \mathbb{R}_+^{|V|}$ such that $\gamma_i + \gamma_j = w_{ij}$ for each pair $(i, j) \in M$, and for each node $k \in V$ that is unmatched under M, $\gamma_k = 0$.

Given a trade outcome $(\underline{\gamma}, M)$, we define implicit *offers* on all edges not in M. Let $(x)_+ \equiv \max(x, 0)$. For any $(i, j) \in E \backslash M$, node i offers node j an amount $(w_{ij} - \gamma_i)_+$, the idea being that i should be willing to switch partners if she can earn even slightly more. Thus, each node has a set of well defined 'alternatives' to its current partner in M. A natural postulate is that an outcome should be *stable*, i.e. for each node i, γ_i should be no smaller than the best alternative of node i (if i is unmatched under M, she should receive no non-zero offers). The stability condition can be concisely written as $\gamma_i + \gamma_j \geq w_{ij}$ for all $(i, j) \in E \backslash M$.

Let ∂i denote the set of neighbors of node i in G. For each edge $(ij) \in M$, we define the 'edge surplus' as the excess of w_{ij} over the sum of best alternatives, i.e.,

$$\mathcal{S}\text{urp}_{ij}(\underline{\gamma}) = w_{ij} - \max_{k \in \partial i \backslash j} (w_{ik} - \gamma_k)_+ - \max_{l \in \partial j \backslash i} (w_{jl} - \gamma_l)_+ . \quad (1)$$

Now each node in the network may have an inherent 'bargaining power', such that $\mathcal{S}\text{urp}_{ij}$ should be split in a manner determined by the bargaining powers of i and j. We adopt a general model where the surplus is postulated to be split as per a fraction $r_{ij} \in (0,1)$ for each matched edge $(ij) \in M$. We call this *correct division*. Each r_{ij} can be an arbitrary number in the interval $(0,1)$, independently for all edges.

Definition 1 (Problem instance). *A* problem instance I *consists of an undirected graph* $G = (V, E)$, *with positive weights* $(w_e)_{e \in E}$ *and split fractions* $(r_{ij})_{(ij)\in E} \in (0,1)^{|E|}$. *An arbitrary direction is chosen on each edge for purposes of specifying the split fraction. If* r_{ij} *is specified, then it is implicit that* $r_{ji} = 1 - r_{ij}$.

Definition 2 (Correct division). *An outcome* $(\underline{\gamma}, M)$ *is said to satisfy* correct division *if, for all* $(ij) \in M$,

$$\gamma_i = \max_{k \in \partial i \backslash j} (w_{ik} - \gamma_k)_+ + r_{ij} \mathcal{S}\text{urp}_{ij} \,, \tag{2}$$

where $\mathcal{S}\text{urp}_{ij}$ *is defined by Eq.* (1).

Note that it follows from Eq. (2) and Eq. (1) that

$$\gamma_j = w_{ij} - \gamma_i = \max_{l \in \partial j \backslash i} (w_{jl} - \gamma_l)_+ + r_{ji} \mathcal{S}\text{urp}_{ij} \,.$$

Definition 3 (UD solution). *An outcome* $(\underline{\gamma}, M)$ *is said to be an* unequal division (UD) solution *if it is* stable *and satisfies* correct division *(cf. Definition 2).*

1.2 Related Work

We present here a short review of relevant related work.

Recall the linear programming relaxation of the maximum weight matching problem

$$\begin{aligned} &\text{maximize} \quad \sum_{(i,j)\in E} w_{ij} x_{ij} \,, \\ &\text{subject to} \quad \sum_{j \in \partial i} x_{ij} \le 1 \quad \forall i \in V, \qquad x_{ij} \ge 0 \quad \forall (i,j) \in E \,. \end{aligned} \tag{3}$$

The dual problem to (3) is:

$$\begin{aligned} &\text{minimize} \quad \sum_{i \in V} y_i \,, \\ &\text{subject to} \quad y_i + y_j \ge w_{ij} \quad \forall (i,j) \in E, \qquad y_i \ge 0 \quad \forall i \in V \end{aligned} \tag{4}$$

Sotomayor [20] characterized the existence of stable outcomes in exchange networks.

Lemma 4 ([20,13]). *Stable outcomes exist if and only if the LP (3) has an integral optimum. Further, if* $(\underline{\gamma}, M)$ *is a stable outcome, then* $\underline{\gamma}$ *is an optimum solution of the dual LP (4) and M is a maximum weight matching. Conversely, if the LP (3) has an integral optimum, then for any maximum weight matching* M^* *and any optimum* $\underline{y}^*$ *of the dual LP (4),* $(\underline{y}^*, M^*)$ *is a stable outcome.*

The above lemma follows from the stability condition $\gamma_i + \gamma_j \geq w_{ij}$ for all $(ij) \in M$. It implies, in particular, that all instances on bipartite graphs possess stable outcomes.

There have been several recent works on the symmetrical 'balanced outcome' solution concept (corresponding to $r_{ij} = 1/2$ for all $(ij) \in E$), following a paper by Kleinberg and Tardos [13,1,3,12].

Though our previous work [12] focuses on the symmetrical case, it also introduces unequal division solutions. Further, it shows that unequal division solutions exist if and only if Nash bargaining solutions exist.

Theorem 5 ([12]). *A problem instance admits a UD solution if and only if it admits a stable outcome (which occurs iff the LP (3) has an integral optimum).*

This generalizes a result of Kleinberg and Tardos on existence of balanced outcomes [13].

[12] also shows that a certain local dynamics converges to UD solutions, when such solutions exist. However, the bound on time to convergence is exponential in the network size (in contrast to the symmetrical case), and this bound turns out to be tight in worst case (see Section 4). Here, we resolve this issue, providing a new FPTAS for computing approximate UD solutions.

Relationship to Cooperative games. Rochford [16] and recent work by Bateni et al [3] show that the bargaining network setting can be viewed as a cooperative game, making this problem susceptible to a large body of literature. This literature defines various solution concepts such as nucleolus, kernel and prekernel, and also describes algorithms to compute these solutions for various classes of games.[1] It is noteworthy that the solution concepts studied are typically symmetric in the players. Whereas such solution concepts may form a reasonable predictive framework in the absence of player specific information, we also want to ask "Can the players *find* an appropriate 'solution' when there is asymmetry?" To this end, we would like to establish computational tractability in the asymmetric case.

However, a little investigation reveals that the approaches devised to compute various (symmetric) solution concepts rely heavily on the symmetry in their respective definitions. For instance, the polynomial time algorithm in Faigle et al [8] for finding a point in the least core intersection prekernel uses two components –a transfer scheme and a linear programming based update– neither of which work in the unsymmetrical case.

[1] [3] shows that stable, balanced outcomes in bargaining networks correspond to the core intersection prekernel.

The situation is similar with regard to iterative schemes that converge to a solution concept. For the general cooperative game problem, Maschler proposed a simple transfer scheme to approximate points in the prekernel. A version of this scheme was shown to converge by Stearns, and a simpler proof of convergence was provided by Faigle et al [8], in the general cooperative game setting. However, both proofs suffer from two drawbacks: (a) they depend on the symmetry of the solution concepts, (b) the bound on convergence time (if any) is exponential in network size. Essentially the same transfer scheme was used in Azar et al [1] for bargaining networks (see [3] for the connection), and the proof of convergence suffered from the same drawbacks.

The current work addresses computational tractability for the asymmetric case in the bargaining network setting, where an appropriate asymmetric solution concept can be readily defined.

1.3 Outline of the Paper

We present our FPTAS in Section 2, along with a proof that it returns an ϵ-UD solution in polynomial time. We present a fast local algorithm for this problem in subsection 2.1. Each of the algorithms involve an iterative 'rebalancing' phase. Section 3 contains proofs of some key Lemmas used. In Section 4, we demonstrate the importance of ensuring that we stay within the subset of *stable* allocations in our iterative updates. This insight is critically used in our construction of an FPTAS.

2 Main Results

First we define an approximate version of correct division, asking that Eq. (2) be satisfied to within an additive ϵ, for all matched edges.

Definition 6 (ϵ-Correct division). *An outcome* $(\underline{\gamma}, M)$ *is said to satisfy* ϵ-correct division *if, for all* $(ij) \in M$*,*

$$|\gamma_i - \max_{k \in \partial i \setminus j}(w_{ik} - \gamma_k)_+ - r_{ij}\mathcal{S}\text{urp}_{ij}(\underline{\gamma})| \leq \epsilon\,, \tag{5}$$

where $\mathcal{S}\text{urp}_{ij}(\cdot)$ *is defined by Eq.* (1)*.*

We define approximate UD solutions as follows:

Definition 7 (ϵ-UD solution). *An outcome* $(\underline{\gamma}, M)$ *is an* ϵ-UD solution *for* $\epsilon \geq 0$ *if it is* stable *and it satisfies* ϵ-correct division *(cf. Definition 6).*

It follows from Lemma 4 that ϵ-UD solutions exist iff the LP (3) admits an integral optimum. This is the same as the requirement for existence of UD solutions (see Theorem 5). Our main result is the following:

Theorem 8. *There is an algorithm that is polynomial in the input and* $1/\epsilon$*, such that for any problem instance with weights uniformly bounded by* 1*, i.e.,* $(w_e, e \in E) \in (0, 1]^{|E|}$*:*

– *If the instance admits a UD solution, the algorithm finds an ϵ-UD solution.*
– *If the instance does not admit a UD solution, the algorithm returns the message* UNSTABLE.

Our approach to finding an ϵ-UD solution consists of two main steps:

1. Find a maximum weight matching M^* and a dual optimum $\underline{\gamma}$ (solution to the dual LP (4)) . Thus, form a stable outcome $(\underline{\gamma}, M^*)$. Else certify that the instance has no UD solution.
2. Iteratively update the allocation $\underline{\gamma}$ without changing the matching. Updates are local, and are designed to converge fast to an allocation satisfying the ϵ-correct division solution *while maintaining stability*. Thus, we arrive at an ϵ-UD solution.

As mentioned earlier, this is similar to the approach of [1]. The crucial differences (enabling our results) are: (i) we stay within the space of stable outcomes, (ii) our analysis of convergence.

First let us focus on obtaining an FPTAS using the steps above. Later we describe how to make the algorithm local.

Step 1 can be carried out by finding a maximum weight matching M^* (see, e.g., [9]) and also solving the the dual linear program (4). For the dual LP, let v be the optimum value and let $\underline{\gamma}$ be an optimum solution. We now use Lemma 4. If the weight of M^* is smaller than v, we return UNSTABLE, since we know that no stable outcome exists, hence no UD solution (or ϵ-UD solution) exists. Else, $(\underline{\gamma}, M^*)$ is a stable outcome. This completes step 1! The computational effort involved is polynomial in the input size. All unmatched nodes have earnings of 0.

In step 2, we fix the matching M^*, and rebalance the matched edges iteratively. It turns out to be crucial that our iterative updates preserve stability. Section 4 demonstrates that the rebalancing procedure can take an exponentially large time to reach an approximate UD solution if stability is not preserved.

We now motivate the rebalancing procedure briefly, before we give a detailed description and state results. Imagine an edge $(i, j) \in M^*$. Since we start with a stable outcome, the edge weight w_{ij} is at least the sum of the best alternatives, i.e. $\mathcal{S}\text{urp}_{ij} \geq 0$. Suppose we change the division of w_{ij} into γ_i', γ_j' so that the $\mathcal{S}\text{urp}_{ij}$ is divided as per the prescribed split fraction r_{ij}. Earnings of all other nodes are left unchanged. Since $r_{ij} \in (0, 1)$, γ_i' is at least as large as the best alternative of i, as was the case for γ_i. This leads to $\gamma_i' + \gamma_k \geq w_{ik}$ for all $k \in \partial i \backslash j$. A similar argument holds for node j. In short, *stability is preserved*!

It turns out that the analysis of convergence is simpler if we analyze synchronous updates, as opposed to asynchronous updates as described above. Moreover, we find that simply choosing an appropriate 'damping factor' allows us to ensure that stability is preserved even with synchronous updates. We use a powerful technique introduced in our recent work [13] to prove convergence.

Table 1 shows the algorithm EDGE REBALANCING we use to complete step 2. Note that each iteration of the loop can requires $O(|E|)$ simple operations.

Table 1. Local algorithm that converts stable outcome to ϵ-UD solution

EDGE REBALANCING (`Instance` I, `Stable outcome` $(\underline{\gamma}, M)$, `Damping factor` κ, `Error target` ϵ)

1: `Check` $\kappa \in (0, 1/2]$, $\epsilon > 0$, $(\underline{\gamma}, M)$ `is stable outcome`
2: `If (Check fails) Return` ERROR
3: $\underline{\gamma}^0 \leftarrow \underline{\gamma}$
4: $t \leftarrow 0$
5: `Do`
6:     `ForEach` $(i, j) \in M$
7:         $\gamma_i^{\mathrm{reb}} \leftarrow \max_{k \in \partial i \backslash j}(w_{ik} - \gamma_k^t)_+ + r_{ij}\mathcal{S}\mathrm{urp}_{ij}(\underline{\gamma}^t)$
8:         $\gamma_j^{\mathrm{reb}} \leftarrow \max_{l \in \partial j \backslash i}(w_{jl} - \gamma_l^t)_+ + r_{ji}\mathcal{S}\mathrm{urp}_{ij}(\underline{\gamma}^t)$
9:     `End ForEach`
10:     `ForEach` $i \in V$ `that is unmatched under` M
11:         $\gamma_i^{\mathrm{reb}} \leftarrow 0$
12:     `End ForEach`
13:     `If` $\left(\|\underline{\gamma}^{\mathrm{reb}} - \underline{\gamma}^t\|_\infty \leq \epsilon\right)$ `Break Do`
14:     $\underline{\gamma}^{t+1} = \kappa\underline{\gamma}^{\mathrm{reb}} + (1 - \kappa)\underline{\gamma}^t$
15:     $t \leftarrow t + 1$
16: `End Do`
17: `Return` $(\underline{\gamma}^t, M)$

Correctness of EDGE REBALANCING
It is easy to check that $(\underline{\gamma}^{\mathrm{reb}}, M)$ and $(\underline{\gamma}^t, M)$ are valid outcomes for all t. We show that $\underline{\gamma}^t$ computed by EDGE REBALANCING is, in fact, a stable allocation (proof in Section 3):

Lemma 9. *If* EDGE REBALANCING *is given a valid input satisfying the 'Check' on line 1, then* (γ^t, M) *is a stable outcome for all* $t \geq 0$.

Convergence of EDGE REBALANCING
Note that the termination condition $\|\underline{\gamma}^{\mathrm{reb}} - \underline{\gamma}^t\|_\infty \leq \epsilon$ on Line 13 is equivalent to ϵ-correct division. We show that the rebalancing algorithm terminates fast (sketch of proof in Section 3):

Lemma 10. *For any instance with weights bounded by* 1*, i.e.* $(w_e, e \in E) \in (0, 1]^{|E|}$*, if* EDGE REBALANCING *is given a valid input, it terminates in* T *iterations, where*

$$T \leq \left\lceil \frac{1}{\pi\kappa(1 - \kappa)\epsilon^2} \right\rceil . \tag{6}$$

and returns an outcome satisfying ϵ*-correct division (cf. Definition 6). Here* $\pi =$ 3.14159 ...

Using Lemmas 9 and Lemmas 10, we immediately obtain our main result, Theorem 8.

Proof (of Theorem 8). We showed that step 1 can be completed in polynomial time. If the instance has no UD solutions then the algorithm returns UNSTABLE. Else we obtain a stable outcome and proceed to step 2.

Step 2 is performed using EDGE REBALANCING. The input is the instance, the stable outcome obtained from step 1, $\kappa = 1/2$ (for example) and the target error value $\epsilon > 0$. Lemmas 9 and 10 show that EDGE REBALANCING terminates after at most $\lceil 1/(\pi\kappa(1-\kappa)\epsilon^2)\rceil$ iterations, returning a outcome that is stable and satisfies ϵ-correct division, i.e. an ϵ-UD solution. Moreover, each iteration requires $O(|E|)$ simple operations. Hence, step 2 is completed in $O(|E|/\epsilon^2)$ simple operations.

The total number of operations required by the entire algorithm is thus polynomial in the input and in $(1/\epsilon)$.

2.1 A Fast Local Algorithm

Our algorithm EDGE REBALANCING for step 2 is local/distributed, with each matched edge in the graph being updated according to the same, time invariant rule. This rule is a simple function of the edge parameters (weight, split fraction), and the current earnings of nodes in the 1-hop neighborhood. It is also worth mention that since stability is preserved, no player ever has incentive to change her partner. Thus, EDGE REBALANCING constitutes a plausible model for behavior of market participants, after they have attained a stable outcome.

Step 1 can also be accomplished by a fast local algorithm, when the LP (3) has a unique optimum (this condition is generic, see the discussion in [11]). The local algorithm we use is belief propagation for maximum weight matching [4,5,18]. This algorithm yields both the maximum weight matching M and a stable allocation. We omit a detailed discussion in the interest of space. The reader is encouraged to look at the full version of the paper [11].

We mention here that [11] includes a proof of the following:

Claim 11. *Given a maximum weight matching M^* for an instance possessing a UD solution, an ϵ-UD solution can be constructed by a local algorithm with computational effort* $\text{poly}(|V|, 1/\epsilon)$.

Thus, we show a local, polynomial time 'reduction' from the problem of finding an ϵ-UD solution to the sub-problem of finding a maximum weight matching.

3 Proofs of Lemmas 9 and 10

Proof (of Lemma 9). We prove this lemma by induction on time t. Clearly (γ^0, M) is a stable outcome, since the input is valid. Suppose (γ^t, M) is a stable outcome.

Consider any $(i,j) \in M$. It is easy to verify that $\gamma_i^{\text{reb}} + \gamma_j^{\text{reb}} = w_{ij}$, for $\underline{\gamma}^{\text{reb}}$ computed from $\underline{\gamma}^t$ in Lines 6-9 of EDGE REBALANCING. Also, we know that $\gamma_i^t + \gamma_j^t = w_{ij}$. It follows that $\gamma_i^{t+1} + \gamma_j^{t+1} = w_{ij}$ as needed. For $i \in V$ unmatched under M, $\gamma_i^t = 0$ by hypothesis and as per Lines 10-12, $\gamma_i^{\text{reb}} = 0 \Rightarrow \gamma_i^{t+1} = 0$ as needed.

Consider any $(i,k) \in E \backslash M$. We know that $\gamma_i^t + \gamma_k^t \geq w_{ik}$. We want to show the corresponding inequality at time $t+1$. Define $\sigma_{ik}^t \equiv \gamma_i^t + \gamma_k^t - w_{ik} \geq 0$.

Claim. $\gamma_i^{\text{reb}} \geq \gamma_i^t - \sigma_{ik}^t$.

If we prove the claim, it follows that a similar inequality holds for γ_k^{reb}, and hence $\gamma_i^{\text{reb}} + \gamma_k^{\text{reb}} \geq \gamma_i^t + \gamma_k^t - 2\sigma_{ik} = w_{ik} - \sigma_{ik}^t$. It then follows from the definition in Line 14 that $\gamma_i^{t+1} + \gamma_k^{t+1} \geq w_{ik}$, for any $\kappa \in (0, 1/2]$. This will complete our proof that $(\underline{\gamma}^{t+1}, M)$ is a stable outcome.

Let us now prove the claim. Suppose i is matched under M. Using the definition in Line 7 (Line 8 contains a symmetrical definition), $\gamma_i^{\text{reb}} \geq \max_{k' \in \partial i \backslash j}(w_{ik'} - \gamma_{k'}^t)_+$ since $\mathcal{S}\text{urp}_{ik}(\underline{\gamma}^t) \geq 0$. Hence,

$$\gamma_i^{\text{reb}} \geq (w_{ik} - \gamma_k^t)_+ \geq (w_{ik} - \gamma_k^t) = \gamma_i^t - \sigma_{ik}^t\,,$$

as needed. If i is not matched under M, then $\gamma_i^t = \gamma_i^{\text{reb}} = 0$, so the claim follows from $\sigma_{ik}^t \geq 0$.

Proof (of Lemma 10, sketch only). This result is proved using the powerful technique introduced in our recent work [12]. The iterative updates of Edge Rebalancing can be written as

$$\underline{\gamma}^{t+1} = \kappa \mathsf{T} \underline{\gamma}^t + (1-\kappa)\underline{\gamma}^t\,, \tag{7}$$

where T is a self mapping of the (convex) set of allocations corresponding to matching M, $\mathcal{A}_M \subseteq [0,1]^{|E|}$. The 'edge balancing' operator T essentially corresponds to Lines 6-9 in Table 1. It is fairly straightforward to show that T is non-expansive with respect to sup norm. The main theorem in [2] then tells us that

$$\|\mathsf{T}\underline{\gamma}^t - \underline{\gamma}^t\|_\infty \leq \frac{1}{\sqrt{\pi\kappa(1-\kappa)t}}\,. \tag{8}$$

The result follows.

For the full proof, see [11].

Remark 11. It we remove the termination condition on Line 12 of Edge Rebalancing (and iterate forever), [10, Corollary 1] tells us that we always converge to $\underline{\gamma}^*$ such that $\mathsf{T}\underline{\gamma}^* = \underline{\gamma}^*$, i.e., we reach an exact UD solution. (Note that Lemma 9 gives stability of the iterates, and stability of the limit point $\underline{\gamma}^*$ follows.) As a corollary, we recover Theorem 5 on existence of UD solutions.

4 Stability Is Critical

This section demonstrates that our approach of starting with a stable allocation, and ensuring that stability is preserved, plays a critical role in our construction of an FPTAS using iterative edge rebalancing.

It turns out that (unstable) approximate fixed points of the edge balancing operator T (cf. Eq. (7)) *do not* correspond to approximate UD solutions in general.[2] Let $n \equiv |V|$. We show the following in the full version [11, Appendix C] via a constructive proof: There is a sequence of instances $(I_n, n \geq 8)$, such that for each instance in the sequence the following holds. (a) The instance admits a UD solution. (b) There is an outcome $(\underline{\gamma}, M^*)$ on a maximum weight matching M^* such that:

1. The outcome satisfies ϵ-correct division for $\epsilon = 2^{-cn}$.
2. (Stability violation) There is a 'bad' edge $(i,j) \notin M^*$ such that $\gamma_i + \gamma_j \leq w_{ij} - 1$,

where $c > 0$ is a constant. Split fractions are bounded within $[r, 1-r]$ for arbitrary desired $r \in (0, 1/2)$ (c depends on r). Also, the edge weights are uniformly bounded by a constant $W(r)$.

We now describe the implications of such a construction. Suppose we perform edge balancing on the example outcome as per Eq. (7), starting with $\underline{\gamma}^0 \equiv \underline{\gamma}$. We know that $\|\mathsf{T}\underline{\gamma}^0 - \underline{\gamma}^0\|_\infty \leq \epsilon$, since $\underline{\gamma}^0$ satisfies ϵ-correct division. Define $\mathsf{T}_\kappa \equiv \kappa\mathsf{T} + (1-\kappa)\mathsf{I}$, where I is the identity operator. Eq. (7) simply corresponds to iterating with T_κ, i.e. $\underline{\gamma}^t = \mathsf{T}_\kappa^t \underline{\gamma}^0$. Clearly, $\|\mathsf{T}_\kappa \underline{\gamma}^0 - \underline{\gamma}^0\|_\infty \leq \epsilon$. Also, it follows from non-expansivity of T that T_κ is non-expansive in sup norm. As a consequence $\|\mathsf{T}_\kappa \underline{\gamma}^t - \underline{\gamma}^t\|_\infty \leq \epsilon$ for all $t \geq 0$. Thus, successive iterates differ by at most ϵ in sup norm, meaning that no coordinate changes by more than ϵ per iteration. Suppose we want to reach a configuration that satisfies both (1/2)-stability ($\gamma_k + \gamma_l \geq w_{kl} - 1/2$ for each $(k,l) \in E$) and (1/2)-correct division. One of γ_i and γ_j must change by at least 1/4 for the 'bad' edge (i,j) to satisfy (1/2)-stability, i.e., $\gamma_i + \gamma_j \geq w_{ij} - 1/2$. But this will take at least $1/(4\epsilon) = 2^{\Omega(n)}$ iterations!

Thus, *it can take exponential time to reach an approximate UD solution if we do not stay within the space of stable outcomes while rebalancing.*

Remark 12. Essentially the same construction and reasoning shows that the dynamics of [12] can take exponential time to reach an ϵ-UD solution.

5 Further Directions

It remains open whether there is a polynomial algorithm that finds an exact UD solution. Second, it would be interesting to understand the structure of unequal division solutions, and how it differs from the structure of balanced outcomes. Third, it would be interesting to identify other classes of games where solution concepts that are not symmetrical in the players can be naturally defined and studied. Finally, though we have found a fast local algorithm for finding ϵ-UD solutions, it does not constitute a natural description of market behavior of the type proposed in [12].

Acknowledgements. The author would like to thank Andrea Montanari, Mohsen Bayati, R. Ravi and Mohammad Hossein Bateni for helpful discussions.

[2] This is in stark contrast to the situation for the balanced case (cf. [12, Theorem 4]).

References

1. Azar, Y., Birnbaum, B., Celis, L.E., Devanur, N.R., Peres, Y.: Convergence of Local Dynamics to Balanced Outcomes in Exchange Networks. In: 50th IEEE Symp. Foundations of Computer Science, Atlanta (2009)
2. Baillon, J., Bruck, R.E.: The rate of asymptotic regularity is $O(1/\sqrt{n})$. In: Kartsatos, A.G. (ed.) Theory and applications of nonlinear operators of accretive and monotone type. Lecture Notes in Pure and Appl. Math, vol. 178, pp. 51–81. Marcel Dekker, Inc., New York (1996)
3. Bateni, M., Hajiaghayi, M., Immorlica, N., Mahini, H.: The cooperative game theory foundations of network bargaining games. In: Intl. Colloquium on Automata, Languages and Programming (2010)
4. Bayati, M., Shah, D., Sharma, M.: Max-Product for Maximum Weight Matching: Convergence, Correctness, and LP Duality. IEEE Trans. Inform. Theory 54, 1241–1251 (2008)
5. Bayati, M., Borgs, C., Chayes, J., Zecchina, R.: On the exactness of the cavity method for Weighted b-Matchings on Arbitrary Graphs and its Relation to Linear Programs. arXiv:0807.3159 (2007)
6. Chakraborty, T., Judd, S., Kearns, M., Tan, J.: A Behavioral Study of Bargaining in Social Networks. In: Proc. 11th ACM Conf. Electronic Commerce (2010)
7. Cook, K.S., Yamagishi, T.: Power exchange in networks: A power-dependence formulation. Social Networks 14, 245–265 (1992)
8. Faigle, U., Kern, W., Kuipers, J.: On the computation of the nucleolus of a cooperative game. Intl. Journal of Game Theory 30, 79–98 (2001)
9. Gabow, H.N., Tarjan, R.E.: Faster scaling algorithms for general graph-matching problems. J. ACM 38(4), 815–853 (1991)
10. Ishikawa, S.: Fixed points and iteration of a nonexpansive mapping in a Banach space. Proc. American Mathematical Society 59(1) (1976)
11. Kanoria, Y.: An FPTAS for Bargaining Networks with Unequal Bargaining Powers. Full version of this paper: arXiv:1008.0212 (2010)
12. Kanoria, Y., Bayati, M., Borgs, C., Chayes, J., Montanari, A.: Fast Convergence of Natural Bargaining Dynamics in Exchange Networks. arXiv:1004.2079 (April 2010); To appear in Proc. ACM-SIAM Symp. Discrete Algorithms (2011)
13. Kleinberg, J., Tardos, E.: Balanced outcomes in social exchange networks. In: Proc. 40th ACM Symposium on Theory of Computing (2008)
14. Lucas, J.W., Younts, C.W., Lovaglia, M.J., Markovsky, B.: Lines of power in exchange networks. Social Forces 80, 185–214 (2001)
15. Nash, J.: The bargaining problem. Econometrica 18, 155–162 (1950)
16. Rochford, S.C.: Symmetric pairwise-bargained allocations in an assignment market. J. Economic Theory 34, 262–281 (1984)
17. Rubinstein, A.: Perfect equilibrium in a bargaining model. Econometrica 50, 97–109 (1982)
18. Sanghavi, S., Malioutov, D., Willsky, A.: Linear Programming Analysis of Loopy Belief Propagation for Weighted Matching. In: Neural Information Processing Systems, NIPS (2007)
19. Skvoretz, J., Willer, D.: Exclusion and power: A test of four theories of power in exchange networks. American Sociological Review 58, 801–818 (1993)
20. Sotomayor, M.: On The Core Of The One-Sided Assignment Game (2005), http://www.usp.br/feaecon/media/fck/File/one_sided_assignment_game.pdf
21. Willer, D. (ed.): Network Exchange Theory. Praeger (1999)

Value of Learning in Sponsored Search Auctions

Sai-Ming Li, Mohammad Mahdian, and R. Preston McAfee

Yahoo! Inc.
Santa Clara, CA, USA

Abstract. The standard business model in the sponsored search marketplace is to sell click-throughs to the advertisers. This involves running an auction that allocates advertisement opportunities based on the value the advertiser is willing to pay per click, times the click-through rate of the advertiser. The click-through rate of an advertiser is the probability that if their ad is shown, it would be clicked on by the user. This quantity is unknown in advance, and is learned using historical click data about the advertiser. In this paper, we first show that in an auction that does not explore enough to discover the click-through rate of the ads, an advertiser has an incentive to increase their bid by an amount that we call *value of learning*. This means that in sponsored search auctions, exploration is necessary not only to improve the efficiency (a subject which has been studied in the machine learning literature), but also to improve the incentive properties of the mechanism. Secondly, we show through an intuitive theoretical argument as well as extensive simulations that a mechanism that sorts ads based on their expected value per impression *plus* their value of learning, increases the revenue *even in the short term*.

1 Introduction

Online advertising provides the major revenue source for most online services today. The most common standard in the online advertising marketplace is Pay-Per-Click, which means that the publisher sells "click-throughs" to the advertisers. An advertiser is charged only when a user clicks on their ad. The allocation and pricing of such ads are often done through an auction: each advertiser specifies the maximum they are willing to pay for a click-through, and the auction mechanism decides which ad(s) should be shown and how much each of them should pay in the event of a click. The most prominent example of online ad auctions is sponsored search auctions, which allocate the ad space on the side of search results pages of major search engines.

The efficient allocation of ad space in a pay-per-click system is based on the expected value from each impression of the ad. This expected value is the product of the advertiser's value for each click and the probability that if the ad is shown, it will be clicked on. Estimating the latter parameter, called the *Click-Through Rate* (CTR), is a central piece of an ad allocation engine.

The problem of efficiently allocating the ad space and simultaneously estimating the CTR for future is essentially a form of the multi-armed bandits problem [4]. In this problem, the task is to strike a balance between *exploring*,

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 294–305, 2010.

i.e., showing an ad to get a better estimate of its CTR, and *exploiting*, i.e., showing ads that have the best performance, according to our current estimates of the CTRs. There are several papers that give explore-exploit algorithms for this problem from a machine learning perspective [1,6,7,14,15,16,11]. The goal of this paper is not to give yet another explore-exploit algorithm for sponsored search (even though our analysis involves designing an algorithm for a simple setting). Instead, we seek to make two points in this paper: First, even a second-price auction, which is incentive compatible in most settings, fails to be incentive compatible when the mechanism does not perform exploration. Specifically, in such a mechanism, an advertiser has an incentive to increase their bid by some amount, which we call their *value of learning*. This means that performing exploration improves not only the efficiency of the mechanism, but also its incentive properties. Furthermore, this suggests an exploration-exploitation mechanism that is quite natural from an economic standpoint: sort the ads based on their expected value per impression *plus* their value of learning. Multi-armed bandits algorithms based on Upper Confidence Bounds [3,4] can be considered in this vein.

Second, despite the intuition that "exploration has some short-term cost", we show that incorporating value of learning in the auction mechanism (the way described above) can lead to a higher revenue *even in the short term.* In other words, in a mechanism that performs exploration by incorporating value of learning, the cost of learning is paid by the advertisers, and not by the seller. This is based on the intuition that value of learning gives higher boost to advertisers in lower slots, thereby helping to level the playing field among advertisers competing for the same ad space and increasing the competition. We show this through a non-rigorous theoretical argument (as making the statement rigorous requires arguing about a complex Bayesian model), as well as extensive simulations using real advertisers' data.

Previous Work. In addition to the vast literature on the explore-exploit algroithms for various forms of the multi-armed bandit problem [1,6,7,14,15,16,11], the paper by Goel and Munagala [10] is related to our work. They attack the problem of uncertainty about click-through rates using a different approach, by allowing the advertiser to make a per-impression as well as a per-click bid.

2 Model and Notations

We consider a setting where n advertisers (or bidders) are competing to be placed in one of the m slots, numbered 1 through m. Advertiser i has a value v_i and bids b_i for a click. We assume a separable model for click-through rates, i.e., there is a value γ_j associated with each slot $j = 1, \ldots, m$ (called the position bias of slot j) and a value λ_i for each advertiser (called the clickability of this advertiser), such that if the ad of advertiser i is displayed in position j, it will be clicked on with probability $\gamma_j \lambda_i$.[1] We assume that the slots are numbered

[1] This is assumed to be independent of other ads placed on the page; for models that do not make this assumption see [9,2,12].

in decreasing order of their position bias, i.e., $\gamma_1 \geq \gamma_2 \geq \ldots \geq \gamma_m$. The exact clickability of advertisers are not known, and the system tries to estimate these quantities based on the past performance of the ad. We denote the estimate of the clickability of advertiser i by $\hat{\lambda}_i$. Note that this value can change as time progresses.

The most common mechanism for allocating the ad slots to the advertisers is the so called *generalized second price auction (GSP)* [8,17]. In this mechanism, the advertisers are ordered in their decreasing order of their $\hat{\lambda}_i b_i$, and slots 1 through m are allocated to the top m advertisers in this order (or are left empty if the number of advertisers is less than m). The amount the i'th advertiser is charged in the event of a click is the minimum this advertiser could bid and still win the same slot. This means that if advertiser i is allocated slot i, the price per click for this advertiser is $p_i := \hat{\lambda}_{i+1} b_{i+1} / \hat{\lambda}_i$. As shown in [8,17], bidding truthfully is not an equilibrium in the GSP mechanism, i.e., the advertisers have incentive to submit bids other than their true value per click, but it has full-information equilibria which coincide with the outcome of the Vickrey-Clark-Groves (VCG) mechanism, which is a well-knwon incentive compatible mechanism. In the case that there is only one slot ($m = 1$), the GSP mechanism is the same as the second-price auction, which is incentive compatible. In the next section, we will focus on this case to separate out the strategic issues of the GSP mechanism from the incentive issues resulting from the uncertainty in click-through rates.

3 Incentives in Auctions without Exploration

In this section, we look at the second price auction described in the previous section from the perspective of one advertiser, and show that in a repeated second-price auction without exploration, when there is uncertainty about the clickability of the advertiser, it is no longer in the advertiser's best interest to bid her value per click. Specifically, the advertiser has the incentive to increase her bid in order to induce the mechanism to explore her. This is done through a simple model defined below.

We assume that the advertiser i faces a price per impression distribution $\mathcal{D}_p$, i.e., the highest bid times click-through rate among other advertisers is distributed according to $\mathcal{D}_p$. We make the simplifying assumption that this distribution does not change over time and is independent in each time step. The advertiser has a value per click v_i, and a clickability λ_i, which is distributed according to a prior $\mathcal{D}_\lambda$. Neither the advertiser nor the auctioneer knows the value of λ_i. Instead, an unbiased estimate $\hat{\lambda}_i$ of this value is calculated using Bayesian updating given the current history, i.e., the estimate $\hat{\lambda}_i$ at any point in time is equal to the expected value of λ_i given the prior $\mathcal{D}_\lambda$ and the observed click/no-click history. In each time step, the advertiser decides how much to bid (the bid b_i can change as time progresses); then a price p is picked according to $\mathcal{D}_p$, and if $\hat{\lambda}_i \geq p$, i's ad is displayed and i is charged $p/\hat{\lambda}_i$ in the event of a click. Both advertiser i and the auctioneer will observe whether or not the ad is clicked on. We assume an infinite time horizon (i.e., an infinite sequence of auctions) and a discount factor of $\delta < 1$.

In the above model, if there is no uncertainty about the clickability λ_i (i.e., if $\mathcal{D}_\lambda$ has a singleton support), the optimal strategy for advertiser i to bid $b_i = v_i$ in every round. In the rest of this section, we show that this is not the case in general where there is uncertainty about λ_i. To demonstrate this point, we calculate the advertiser's optimal strategy as a recurrence and prove that it is non-negative in general. We will also give a lower bound for the advertiser's optimal bid in the case of uniform distributions.

At any point, the state can be described by two numbers (k, N), indicating a state where the ad of the advertiser has been shown N times and out of these impressions, k of them have lead to clicks. Based on the prior $\mathcal{D}_\lambda$, the posterior distribution of the clickability at this state can be computed. Let $\hat{\lambda}_{k,N}$ denote the expected value of this posterior distribution. Let $U(k, N)$ denote the optimal utility of an infinite sequence of auctions, starting from this posterior distribution on λ. We obtain a recurrence relation for U as follows: let b denote the bid of the advertiser in the first round. If $p < \hat{\lambda}_{k,N} b$, then the advertiser wins and has to pay $p/\hat{\lambda}_{k,N}$ in the event of a click. By the definition of $\hat{\lambda}$, this means that the advertiser pays p per impression in expectation. Therefore, the total utility of the advertiser in this round can be written as:

$$\Pr[p < \hat{\lambda}_{k,N} b](v\hat{\lambda}_{k,N} - E[p | p < \hat{\lambda}_{k,N} b]).$$

We denote the above value by $g(\hat{\lambda}_{k,N}, b)$. Denoting the pdf and the cdf of $\mathcal{D}_p$ by $f(.)$ and $F(.)$ respectively, the above expression can be written as:

$$g(\hat{\lambda}_{k,N}, b) = \int_0^{\hat{\lambda}_{k,N} b} (v\hat{\lambda}_{k,N} - p) f(p) dp. \tag{1}$$

If the ad is shown (which happens with probability $F(\hat{\lambda}_{k,N} b)$), it is either clicked on (with probability $\hat{\lambda}_{k,N}$), or not (with probability $1 - \hat{\lambda}_{k,N}$); leading us to one of the states $(k+1, N+1)$ or $(k, N+1)$. Therefore, the overall utility of the advertiser can be written as:

$$\begin{aligned} U(k, N) = \max_b \Bigg\{ & g(\hat{\lambda}_{k,N}, b) \\ & + \delta F(\hat{\lambda}_{k,N} b) \Big(\hat{\lambda}_{k,N} U(k+1, N+1) + (1 - \hat{\lambda}_{k,N}) U(k, N+1) \Big) \\ & + \delta (1 - F(\hat{\lambda}_{k,N} b)) U(k, N) \Bigg\}. \end{aligned} \tag{2}$$

This implies:

$$U(k, N) = \frac{1}{1-\delta} \max_b \Bigg\{ g(\hat{\lambda}_{k,N}, b) + \delta F(\hat{\lambda}_{k,N} b) \mathit{\Delta}(k, N) \Bigg\}, \tag{3}$$

where

$$\Delta(k,N) := \hat{\lambda}_{k,N} U(k+1,N+1) + (1-\hat{\lambda}_{k,N}) U(k,N+1) - U(k,N). \tag{4}$$

Intuitively, $\Delta(k,N)$ indicates the advertiser's value for the information she obtains by observing the outcome of one additional impression. We take the derivative of the expression in Equation (3) with respect to $z = \hat{\lambda}_{k,n} b$ to compute the optimal bid. Using (1), this derivative can be written as:

$$\frac{\partial(g(\hat{\lambda}_{k,N}, b) + \delta F(\hat{\lambda}_{k,N} b)\Delta(k,N))}{\partial z} = (v\hat{\lambda}_{k,N} - z) f(z) + \delta f(z) \Delta(k,N),$$
$$= f(z)(v\hat{\lambda}_{k,N} + \delta\Delta(k,N) - z).$$

Given that $f(z)$ is non-negative, the root of the linear term in the paranthesis satisfies the second-order condition and is therefore a maximizer of the function. Thus, the optimal bid of the advertiser can be written as:

$$b^* = v + \frac{\delta}{\hat{\lambda}_{k,N}} \Delta(k,N). \tag{5}$$

This shows that the optimal bid of the advertiser is not the true value per click v, but the value per click plus some additional term. This additional term is proportional to the information value of one additional impression, and can be expressed with a recurrence relation. In general, this recurrence is hard to solve explicitly. However, here we prove that the optimal bid of the advertiser is always greater than or equal to her value per click. Later, we will give a lower bound on the optimal bid in the special case of uniform distributions.

Theorem 1. *In the above model of repeated auctions, the optimal bid of the advertiser in every state (k,N) is at least v.*

Proof. By Equation (5), we need to prove that $\Delta(k,N) \geq 0$. In other words, we need to show that the expected optimal revenue starting from the state (k,N) (which we call scenario 1) is less than the optimal revenue when we first start from (k,N), observe the outcome of one impression, and then proceed (we call this scenario 2).[2] We prove this inequality by analyzing the strategy for scenario 2 that simulates the optimal strategy of scenario 1. This gives a lower bound on the optimal strategy in scenario 2.

To simulate the optimal strategy of scenario 1 in scenario 2, in each step we take the optimal bid b of scenario 1, and submit a bid in scenario 2 that leads to the same expected bid per impression. For example, in the first step (i.e.,

[2] Note that this statement is not trivial, since the additional information (the outcome of one impression) is observed by both the advertiser and the auctioneer. While the additional information enables the advertiser to make more informed decisions to improve her utility, it also enables the auctioneer to allocate and price future impressions more accurately. It is not clear a priori whether the latter effect helps or hurts the advertiser.

when we are in state (k, N) in scenario 1), the corresponding bid in scenario 2 is either $b\hat{\lambda}_{k,N}/\hat{\lambda}_{k+1,N+1}$ or $b\hat{\lambda}_{k,N}/\hat{\lambda}_{k,N+1}$, depending on whether the state is $(k+1, N+1)$ or $(k, N+1)$. The expected utility of the advertiser in scenario 2 in this step can be written as

$$\hat{\lambda}_{k,N} g(\hat{\lambda}_{k+1,N+1}, b\hat{\lambda}_{k,N}/\hat{\lambda}_{k+1,N+1}) + (1 - \hat{\lambda}_{k,N}) g(\hat{\lambda}_{k,N+1}, b\hat{\lambda}_{k,N}/\hat{\lambda}_{k+1,N+1})$$

Using (1), this can be written as:

$$\hat{\lambda}_{k,N} \int_0^{\hat{\lambda}_{k,N} b} (v\hat{\lambda}_{k+1,N+1} - p) f(p) dp + (1 - \hat{\lambda}_{k,N}) \int_0^{\hat{\lambda}_{k,N} b} (v\hat{\lambda}_{k,N+1} - p) f(p) dp$$
$$= \int_0^{\hat{\lambda}_{k,N} b} \big(v(\hat{\lambda}_{k,N}\hat{\lambda}_{k+1,N+1} + (1 - \hat{\lambda}_{k,N})\hat{\lambda}_{k,N+1}) - p\big) f(p) dp$$

Using the definition $\hat{\lambda}_{k,N}$ as the posterior probability of getting a click conditioned on having had k clicks out of the first N impressions, it is easy to show that

$$\hat{\lambda}_{k,N}\hat{\lambda}_{k+1,N+1} + (1 - \hat{\lambda}_{k,N})\hat{\lambda}_{k,N+1} = \hat{\lambda}_{k,N}. \tag{6}$$

Therefore, the expected utility in the first step in scenario 2 is equal to

$$\int_0^{\hat{\lambda}_{k,N} b} (v\hat{\lambda}_{k,N} - p) f(p) dp = g(\hat{\lambda}_{k,N}, b),$$

which is the same as the expected utility in the first step in scenario 1. Similarly, in any step the simulated strategy in scenario 2 obtains the same expected payoff as in scenario 1. Thus, $\Delta(k, N) \geq 0$.

The above theorem only shows that the optimal bid of the advertiser is never smaller than her true value. To show that this bid is sometimes strictly larger than the value, we focus on the case of uniform distributions: We assume a uniform prior $\mathcal{D}_\lambda = U[0, 1]$ on the clickability and a uniform price distribution $\mathcal{D}_p = U[0, 1]$. Straightforward calculations using the Bayes rule and the prior $\mathcal{D}_\lambda$ shows that the posterior probability density for the clickability λ in a state (k, N) is

$$(n + 1)\binom{n}{k}\lambda^k (1 - \lambda)^{N-k}.$$

The expected value of λ given this posterior is $\hat{\lambda}_{k,N} = \frac{k+1}{N+2}$, and the function $g(.)$ from (1) can be written as $g(\hat{\lambda}_{k,N}, b) = \hat{\lambda}_{k,N}^2 b(v - \frac{b}{2})$.

Theorem 2. *In the above model of repeated auction, the optimal bid of the advertiser in every state (k, N) is strictly larger than v. More specifically, we have $\Delta(k, N) = \Omega(N^{-2})$.*

Proof (Proof Sketch). As in the proof of Theorem 1, we need to bound the difference between the optimal expected utility of scenarios 1 and 2. Again, we do this by taking the optimal strategy in scenario 1, and simulating it in scenario 2. Unlike the proof of Theorem 1, we simulate a strategy that submits a bid of b in scenario 1 by submitting the same bid in scenario 2. First, notice that with this strategy, the probability of winning the first auction in scenario 2 can be written as

$$\hat{\lambda}_{k,N}F(b\hat{\lambda}_{k+1,N+1})+(1-\hat{\lambda}_{k,N})F(b\hat{\lambda}_{k,N+1}) = (\hat{\lambda}_{k,N}\hat{\lambda}_{k+1,N+1}+(1-\hat{\lambda}_{k,N})\hat{\lambda}_{k,N+1})b$$

Using (6), the above probability is equal to $\hat{\lambda}_{k,N}b$, which is the same as the probability of winning in scenario 1. This ensures that the simulated strategy in scenario 2 has the same branching probabilities as the optimal strategy in scenario 1. Next, we need to bound the difference between the expected utility of one auction in the two scenarios. Here we only do this for the first auction. The inequality for the other auctions can be proved similarly. The difference between the expected utilities of the advertiser in the first auction in the two scenarios can be written as:

$$\begin{aligned}
&\hat{\lambda}_{k,N}g(\hat{\lambda}_{k+1,N+1},b) + (1-\hat{\lambda}_{k,N})g(\hat{\lambda}_{k,N+1},b) - g(\hat{\lambda}_{k,N},b)\\
&= b(v-\frac{b}{2})\left(\hat{\lambda}_{k,N}\hat{\lambda}^2_{k+1,N+1} + (1-\hat{\lambda}_{k,N})\hat{\lambda}^2_{k,N+1} - \hat{\lambda}^2_{k,N}\right)\\
&= b(v-\frac{b}{2})\left((\frac{k+1}{N+2})(\frac{k+2}{N+3})^2 + (1-\frac{k+1}{N+2})(\frac{k+1}{N+3})^2 - (\frac{k+1}{N+2})^2\right)\\
&= b(v-\frac{b}{2})\frac{(k+1)(N-k+1)}{(N+2)^2(N+3)^2} = \Omega(N^{-2}).
\end{aligned}$$

4 Value of Learning

Given the result in the previous section, we can define the *value of learning* of an advertiser as the difference between the optimal bid of the advertiser and her value-per-click. More formally, the value of learning is the difference between the Gittins index in the Markov Decision Process (MDP) defined based on the auction. If we could compute these indices, we could simply design an alternative auction mechanism that allocates according to these indices, thereby achieving the optimal MDP solution and eliminating the incentive to overbid. Unfortunately, Gittins indices are quite hard to compute.

As a practical alternative, we can use proxies for the value of learning that are easy to compute. Perhaps the simplest method for doing this is to take the value of learning of an advertiser to be proportional to the variance of our estimate of the clickability of this advertiser. This has the advantage that it can be easily computed, and gives a boost to ads that we currently do not have an accurate estimate of its clickability.

The strongest theoretical evidence that taking the value of learning of an ad to be proportional to the variance of its clickability estimate and then sorting the ads based on their expected value per impression plus their value of learning leads to close-to-optimal outcomes comes from the literature on the multi-armed bandits problem. Multi-armed bandits algorithms based Upper Confidence Bounds are shown to achieve asymptotically optimal regrets [3,4]. These algorithms in each iteration pick the arm that has the maximum expected value plus an additional factor that is close to the variance of the performance of the arm so far. The literature on multi-armed bandits is a vast literature and we do not intend to add yet another algorithm to this literature. Instead, we describe a practical method for incorporating the value of learning in sponsored search auctions, and analyze its revenue and efficiency impacts through simulations with real advertisers' bid and click-through rate data.

A practical value-of-learning mechanism. Recall that in sponsored search, a sequence of m slots need to be allocated to the advertisers. The position bias of slot j is denoted by γ_j. At any point in time, the history for each ad consists of the number of times this advertiser is shown in each slot, and the number of such instances that have lead to clicks. We can compute the *cumulative expected clicks* ec_i of advertiser i as the sum of the position biases of the positions this ad is shown so far. This is essentially the number of clicks we would expect this ad to receive, if it had a clickability of 1. Our estimate of the clickability is then

$$\hat{\lambda}_i = \frac{c_i}{ec_i}, \tag{7}$$

where c_i is the total number of clicks advertiser i has received. It is not hard to show that in a reasonable Bayesian setting (e.g., uniform priors), the variance of this estimate is of the order of $\sqrt{\frac{\hat{\lambda}_i}{ec_i}}$. Therefore, we define the value of learning for this advertiser as $\hat{\theta}_i b_i$, where

$$\hat{\theta}_i = C\sqrt{\frac{\hat{\lambda}_i}{ec_i}} \tag{8}$$

for a constant C. We will change the value of C in our simulations to study the effects of increaseing the value of learning on the efficiency of and revenue of the auctions. The mechanism computes a score s_i for each advertiser as follows:

$$s_i = b_i(\hat{\lambda}_i + \hat{\theta}_i). \tag{9}$$

It then sorts the advertisers in decreasing order of their scores, allocates the i'th position to the i'th advertiser in this order, and in the event of a click, charges this advertiser an amount equal to

$$p_i = \frac{b_{i+1}(\hat{\lambda}_{i+1} + \hat{\theta}_{i+1})}{(\hat{\lambda}_i + \hat{\theta}_i)} \tag{10}$$

Note that this value is never greater than the bid of the advertiser.

5 Revenue of Auctions with Value of Learning

Intuitively, one might think that exploration in repeated sponsored search auctions is a costly activity that is done in order to achieve a better outcome in the long run. In fact, many of the exploration-exploitation algorithms based on the ϵ-greedy algorithm for the multi-armed bandits problem give out exploration impressions to the advertisers for free [7]. However, we will show experimentally in the next section that the mechanism in the previous section can lead to a higher revenue *even in the short term.* In this section, we explain the theoretical intuition behind this result.

In auction theory [13], it is known that giving an advantage to weaker bidders (e.g., minority-owned firms participating in spectrum auctions [5]) can increase the revenue by leveling the playing field between competing bidders. Here, also, the value of learning added to each advertiser's bid is inversely proportional to the square root of the number of times this ad has been clicked on. This means that an ad that is typically in a lower position has a higher value of learning, and this can increase the price that the advertisers in higher positions pay. A formal proof of this fact in the model with repeated auctions is out of reach, as it would require analyzing optimal strategies in a Bayesian multi-player version of the model studied in Section 3. Instead, we ignore incentives resulting from learning by studying a one-shot auction, and then prove that the GSP-like mechanism that allocates slots to bidders in decreasing order of $(\lambda_i + \theta_i)v_i$ has a minimal envy-free equilibrium similar to the VCG-equivalent equilibria of [8,17]. Furthermore, the revenue of this equilibrium at the λ_i, θ_i values computed in the previous section is typically larger than the similar revenue when θ_i's are zero. The result, whose proof is omitted here, can be stated as follows.

Theorem 3. *Consider a multi-slot auction between n bidders. Assume that the i'th bidder has a value of $v_i\hat{\lambda}_i\gamma_j$ for being placed in slot j. The mechanism $\mathcal{M}_\theta$ sorts the advertisers based on their $(\hat{\lambda}_i + \theta_i)b_i$, allocates the i'th slot to the i'th advertiser in this order (which we call advertiser i), and charges her $\frac{(\hat{\lambda}_{i+1}+\theta_{i+1})b_{i+1}\hat{\lambda}_i\gamma_i}{\hat{\lambda}_i+\theta_i}$ in expectation. This mechanism has a minimal envy-free equilibrium whose revenue is denoted by $R(\theta)$. Furthermore, let $\hat{\lambda}_i$ and $\hat{\theta}_i$ be the values calculated in (7) and (8) and assume that the ordering of the values $(\hat{\lambda}_i+\hat{\theta}_i)v_i$ is the same as the ordering of the values $\hat{\lambda}_i v_i$ and that the historical number of clicks c_i of a bidder in a higher slot is higher. Then we have $R(\hat{\theta}) > R(0)$.*

The main assumption of this theorem (apart from restricting equilibrium analysis to a 1-shot game) is that the ordering of the advertisers in decreasing order of $(\hat{\lambda}_i + \hat{\theta}_i)v_i$ is the same as their ordering in decreasing order of $\hat{\lambda}_i v_i$, and their ordering in decreasing order of c_i. Since θ_i's are typically small and higher slots get more clicks, this assumption is often true, except for rare cases where the mechanism reverses the ordering to do some exploration. The above theorem guarantees that in *normal* cases, the mechanism with value of learning has a higher revenue. Intuitively, this revenue increase can more than make up for the occasional revenue loss due to exploration. This is why in the simulations in the next section we will see that incorporating value of learning leads to a considerable increase in revenue, averaged over thousands of auctions.

6 Simulation Results

In this section we provide simulation results to illustrate the performance of incorporating value of learning in the auction mechanism when applied in a popular search engine like Yahoo! Search. We collect a representative sample of sponsored search results from the Yahoo! search log. For each search sample, we collect the position bias for each position due to the specific page layout used. For each ad, we also collect the bid and its estimated clickability at the time of sampling.

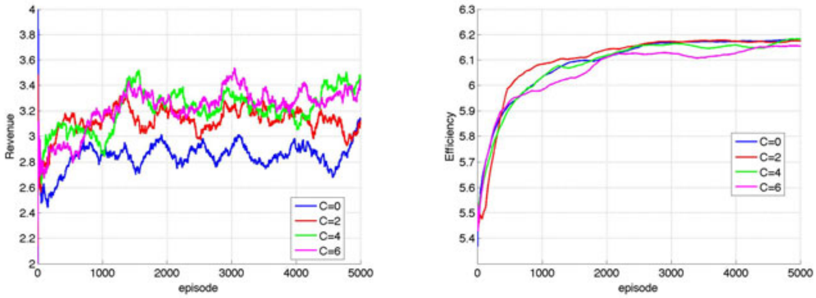
For the purpose of conducting the simulation study, we assume that the ads in each search in the dataset are unique, *i.e.*, the same ad cannot appear across multiple sample searches, hence its clickability estimate only depend on its own history. We also assume that the page layout remains the same, *i.e.*, the same position bias as in the search log will be used to simulate click event and efficiency. We use the estimated clickability of each ad at the time of sampling as their true clickability. The simulation is initialized by simulating a small number of impressions using the assumed true clickability of each ad, and the position effect is based on the one at position one. Then the initial clickability estimate of each ad is computed based on the simulated clicks during those impressions.

After the initialization stage, we simulated the sample searches for 5,000 episodes. Each episode involves simulating all sample searches once. For each sample search s, the value of learning term $\hat{\theta}_{s,i}$ for ad i was determined based on the current clickability estimate $\hat{\lambda}_{s,i}$ and cumulative expected clicks as in (8). The price and rank of each ad was determined by the GSP algorithm using the ranking score (9) and pricing equation (10). The number of clicks for each ad was simulated using the probability of click $\lambda_{s,i}\gamma_{s,j}$, where j is the slot occupied by ad i. Then we updated the clickability estimate for all ads after every simulated search according to (7). After each episode, we computed the total revenue and the total efficiency across the sample searches based on the simulated clicks, the PPC of the ads that were clicked, and their bids. Specifically, the revenue R and efficiency E at each episode is defined as

$$R = \sum_{s,i} p_{s,i} c_{s,i}, \qquad E = \sum_{s,i} b_{s,i} \lambda_{s,i} \gamma_{s,j},$$

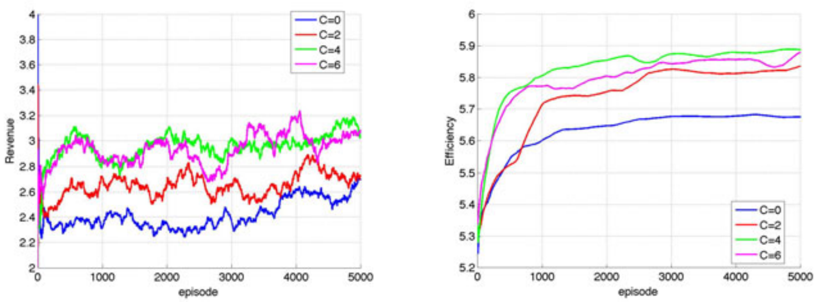
where $p_{s,i}$, $c_{s,i}$, $b_{s,i}$, $\lambda_{s,i}$, denote the price per click, number of clicks, bid, and clickability of ad i in search s respectively, and $\gamma_{s,j}$ denote the position effect of the slot (j) occupied by ad i in search s. Note that in computing the efficiency, we made the simplifying assumption that the bid $b_{s,i}$ does not change over time, and it is the same as the value per click for the advertiser. Nevertheless, we believe that E serves as a good proxy for the true efficiency of the algorithm under investigation.

We simulated the auction and click behavior for a range of C to illustrate the effect of imposing different degree of learning in the mechanism. The case $C = 0$



(a) revenue (b) efficiency

Fig. 1. Moving average of revenue and efficiency for different setting of C



(a) revenue (b) efficiency

Fig. 2. Moving average of revenue and efficiency for different setting of C when $m = 5$

corresponds to the case when there is no value of learning included in the auction. The higher C is, the more impact value of learning has on price and ranking, and hence revenue and efficiency. Figure 1 (a) shows the moving average of the total revenue generated over the duration of the simulation, and figure 1 (b) shows the moving average of the total efficiency. The moving average window used in the graphs has width 400. As can be seen in the figures, the revenue is consistently higher when value of learning is used in the auction. Furthermore, efficiency is higher in the transient when the appropriate value of learning ($C = 2$) is used. It should be noticed that when C is large ($C = 6$), both the transient and final efficiency can suffer as too much exploration is being done.

We also simulated the case when not all ads in each sample search are shown in every auction by reducing the number of slots that can be shown in each auction m to five (in Yahoo! search this can be as high as twelve). In other words, when the number of ads available is more than five, the algorithm is forced to select only five ads to show, with the rest not getting any exposure at all. As can be seen in Figure 2, the power of incorporating value of learning in the auction is more evident in this case. The efficiency of the auction with C other than zero is much higher than when C is zero. This is because the set of ads that are shown are fixed very early as other ads with high clickability are never given a chance to prove themselves. The revenue is also higher when C is non-zero, due to the price effect of value of learning as well as improved efficiency.

References

1. Agarwal, D., Chen, B., Elango, P.: Explore/exploit schemes for web content optimization. In: ICDM (2009)
2. Aggarwal, G., Feldman, J., Muthukrishnan, S., Pal, M.: Sponsored Search Auctions with Markovian Users. In: Workshop on Ad Auctions (2008)
3. Audibert, J.Y., Munos, R., Szepesvári, C.: Exploration–Exploitation Tradeoff Using Variance Estimates in Multi-armed Bandits. Theoretical Computer Science 410(19), 1876–1902 (2009)
4. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time Analysis of the Multiarmed Bandit Problem. Machine Learning 47(2-3), 235–256 (2002)
5. Ayres, I., Cramton, P.: Deficit reduction through diversity: How affirmative action at the fcc increased auction competition. Stanford Law Review 48(4), 761–815 (1996)
6. Babaioff, M., Sharma, Y., Slivkins, A.: Characterizing Truthful Multi-Armed Bandit Mechanisms. In: EC, pp. 79–88 (2009)
7. Devanur, N.R., Kakade, S.M.: The Price of Truthfulness for Pay-Per-Click Auctions. In: EC, pp. 99–106 (2009)
8. Edelman, B., Ostrovsky, M., Schwarz, M.: Internet Advertising and the Generalized Second-price Auction: Selling Billions of Dollars Worth of Keywords. American Economic Review 97(1), 242–259 (2007)
9. Ghosh, A., Mahdian, M.: Externalities in Online Advertising. In: WWW, pp. 161–168 (2008)
10. Goel, A., Munagala, K.: Hybrid keyword search auctions. In: Proceedings of the 18th World Wide Web Conference (2009)
11. Kale, S., Mahdian, M., Punera, K., Sarlos, T., Sharma, A.: Position-aware multi-armed bandits. Working Paper (2010)
12. Kempe, D., Mahdian, M.: A Cascade Model for Externalities in Sponsored Search. In: Workshop on Internet Ad Auctions (2008)
13. Krishna, V.: Auction Theory. Academic Press, London (2002)
14. Langford, J., Li, L., Vorobeychik, Y., Wortman, J.: Maintaining Equilibria during Exploration in Sponsored Search Auctions. In: Deng, X., Graham, F.C. (eds.) WINE 2007. LNCS, vol. 4858, p. 119. Springer, Heidelberg (2007)
15. Pandey, S., Olston, C.: Handling Advertisements of Unknown Quality in Search Advertising. NIPS 19, 1065 (2007)
16. Das Sarma, A., Gujar, S., Narahari, Y.: Multi-Armed Bandit Mechanisms for Multi-Slot Sponsored Search Auctions. Arxiv preprint arXiv:1001.1414 (2010)
17. Varian, H.R.: Position Auctions. International Journal of Industrial Organization 25(6), 1163–1178 (2007)

Exploiting Myopic Learning

Mohamed Mostagir

Social and Information Sciences Laboratory
California Institute of Technology
mosta@caltech.edu

Abstract. We show how a principal can exploit myopic social learning in a population of agents in order to implement social or selfish outcomes that would not be possible under the traditional fully-rational agent model. Learning in our model takes a simple form of imitation, or replicator dynamics; a class of learning dynamics that often leads the population to converge to a Nash equilibrium of the underlying game. We show that, for a large class of games, the principal can always obtain strictly better outcomes than the corresponding Nash solution and explicitly specify how such outcomes can be implemented. The methods applied are general enough to accommodate many scenarios, and powerful enough to generate predictions that allude to some empirically-observed behavior.

1 Introduction and Related Work

The assumptions imposed on the traditional rational agent can be too restrictive, requiring instantaneous reaction to changes in the environment, perfect look-ahead and planning skills, and unlimited computational resources. In reality, even if individuals are interested in maximizing their own welfare, they may be unable to do so because of a myriad of reasons. For example, it maybe the case that finding an optimal course of action is computationally difficult or even infeasible. It can also be that agents utilize a decision making process that is different from what the traditional model dictates. For instance, they may partially or wholly base their decisions on the actions of other agents rather than carefully charting out their own course. In this paper, we deal with the following question: if we relax some of the assumptions about rationality and consider agents that do not act in full compliance with the traditional agent model, can we leverage the resulting framework to implement better outcomes, either for society or for the principal designing the system?

This is a question of mechanism design, of course. Some of the concerns above have been and continue to be addressed by algorithmic mechanism design; a subfield of mechanism design that concerns itself with computability issues [11], but it is only recently that behavioral aspects have been taken into consideration in mechanism design. This is perhaps a little surprising, given the advanced state of behavioral and experimental game theory, two of the field's basic building blocks. One possible reason for this lag in development is the many ways in

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 306–318, 2010.

which behavior can deviate from the classical agent model, making it difficult to develop an all-encompassing behavioral framework. In this paper, we take a small step in this direction by utilizing a simple form of social learning dynamics to set up a model that allows a system designer (henceforth referred to as the principal) to manipulate social learning to his advantage.

The social learning model we employ in this paper is that of replicator dynamics [3]. This class of learning dynamics was developed in an attempt to understand how a population arrives at a steady state of a dynamical system, and was further pursued in economics as an explanation to how agents arrive at a Nash equilibrium. Under this model, an infinite pool of agents plays a game repeatedly. After each round of the game, agents are paired together randomly to compare and contrast payoffs. If agent i is paired with agent j and agent j has obtained a better payoff than i in the last round of the game, then i considers switching to j's strategy in the next round with a probability that is proportional to the difference in payoffs between the two. This way the proportion of strategies that are performing better than average grows in the population as the share of poorly-performing strategies shrink, and more often than not these dynamics lead to a Nash equilibrium of the underlying game[1]. What makes replicator dynamics particularly appealing is that it is perhaps the most rudimentary form of learning dynamic that nicely straddles the line between behavioral and rational models. On one hand, agents are updating their strategies in a myopic fashion based on simple comparisons with how their peers are doing, but on the other hand this seemingly simple behavior can and does lead to fully rational equilibrium outcomes. The canonical selfish-routing model is one example amongst many where agents converge to a Nash equilibrium by following a replicator dynamic [6]. Another nice behavioral aspect captured by the model is the tendency of human decision makers to fall into habit, as a result of the aversion to try new strategies if one is unaware of others for whom these strategies have performed well. Even in the case of meeting others with more successful strategies, the switching is only probabilistic, underlying the fact that switching to a new strategy is not always costless.

The central idea developed in this paper revolves around the indirect influence that a principal can exert on agents' decisions via exploiting the learning dynamic discussed above. We will focus on games where the principal and the population's interests are diametrically opposed, though the methods readily extend to other settings as we discuss in Section 5. We will give a formal definition of the class of games we consider in Section 2.1, but an informal description follows. There is an infinite population where each member has the choice of one out of two pure actions. For simplicity, we can think about these actions as whether to cheat or to be honest. There is a multitude of examples that fall under this setting: agents can decide whether to cheat on their taxes or not, whether to break the speed limit, put low effort into their work, etc. The principal's action against each member of the population is either to audit the agent, at a cost, or to ignore

[1] For example, replicator behavior leads to equilibrium in Prisoner's dilemma, Battle of the sexes, and a large variety of coordination and routing games (see [7]).

and run the risk of incurring a higher cost if the agent is cheating. Agents are interested in maximizing their payoffs, while the principal tries to minimize his cost. The game is repeated indefinitely. Because the population is infinite, the principal's move in each round consists of choosing a fraction of the population to audit. Under the traditional rationality assumptions this game has a unique Nash equilibrium where the agents cheat with some fixed probability and the principal audits the same fraction of the population in each round. Ideally, the principal would like to do little auditing *and* have the population stick to his desired outcome of as little cheating as is reasonable within the payoff structure of the game

The primary contribution of this paper is twofold. On the conceptual front, we argue that imperfect decision making –in its various formats– can in some cases be considered a resource that the system planner should utilize. The second contribution is methodical, where we take the main idea and build a framework that implements it in the context of naive social learning. While the abstract idea behind our framework is simple, the implications can sometimes be quite surprising. One counter-intuitive outcome of the model is that the principal's optimal strategy always makes things temporarily worse for everybody, including possibly the principal himself, in order to achieve better outcomes later. Moreover, as we discuss later in the paper, the application of the model to some real-life problems result in findings that correspond to empirically-observed behavior. This suggests that the approach proposed in this paper not only provides a normative prescription for optimizing systems with a social learning component, but also describes how some existing systems actually operate.

There has been a lot of recent work on social learning and when it can lead a society of agents to converge to the true value of an underlying state of the world, the so-called 'wisdom of the crowds' effect (for example, [2], [1]). While it would be interesting to investigate whether this kind of learning is susceptible to manipulation by a principal, it is outside our scope of interest since we explicitly focus on agents in a behavioral setting, unlike the fully-rational Bayesian agents employed in the work above. Manipulating Bayesian agents, albeit outside of a social learning setting, has been the recent focus of some work [9]. Other recent work that aims to explore the boundaries of mechanism design under behavioral assumptions is auction design for *level-k* bidders [4]. In this paper, the authors show that under such an experimentally-plausible model, it is possible to obtain revenues that are higher than those generated by Myerson's optimal auction [10].

Finally, repeated games and reputation building is a topic with an extensive body of work in the economics literature. The main results in this area are folk theorems that show what outcomes can be obtained if a game is repeated indefinitely. The traditional approach to proving such results relies on retaliation and punishment among players, a method that fails in a setting with a large population, since the identity of a deviator cannot be detected [8]. Indeed, for the class of games we consider here, the unique equilibrium of the repeated game is the same as the one-shot version and no better outcomes can be implemented under the rational model.

2 Cheat-Audit Games

In this section we consider a class of 2×2 games that encompasses a large number of scenarios. We call this class of games Cheat-Audit games. In these games, as mentioned in Section 1, a very large population of agents plays an infinitely repeated game against a principal. In each round of the game an agent has one of two choices, a 'safe' choice with low payoffs, and a risky choice with a higher payoff. For example, in a tax-auditing situation the safe choice would be to report honestly, whereas cheating is a choice that can provide a higher payoff if the agent is not audited by the principal. The principal on the other hand faces a choice between a costly and a costless action when it comes to dealing with each agent. In the context of the preceding examples, a costly action for the tax scenario would be to audit an agent, and a costless action would be to ignore that agent. Of course, it might be the case that auditing leads to catching a cheating agent, in which case the principal obtains a higher payoff than if he had chosen the costless action. By the same token, not auditing an honest agent is a better action for the principal, since auditing an honest agent expends auditing resources with no useful returns to the principal.

	A	I
C	$0,c_1$	v_3,c_3
H	v_1,c_2	$v_2,0$

Fig. 1. The Cheat-Audit Game

2.1 The Game

To formalize the preceding discussion, the payoffs of the game are as shown in Figure 1, with the principal being the column player. Each agent is considered a row player and has the row player's payoffs. The actions available to an agent is to either be honest (action H) or cheat (action C). The principal either audits (action A) or ignores (action I) each agent. An agent's payoffs satisfy $0 < v_1 \leq v_2 < v_3$. To conserve notation, we will assume that $v_1 = v_2$, so that an agent is indifferent to auditing as long as he is honest. An agent is interested in *maximizing* his payoff, while the principal is interested in *minimizing* his cost, where the costs satisfy $0 < c_1 < c_2 < c_3$. There is thus an implicit constraint on the principal's resources, since auditing with no gain (outcome (H, A)) is more costly than auditing a guilty agent (outcome (C, A)). The principal's preferred outcome is (H, I), where no auditing cost is incurred and no crime is committed, and the payoff to this outcome is normalized to zero. Similarly, an agent's least preferred outcome is (C, A), and is also normalized to zero. Notice that the principal's least preferred outcome, (C, I), is also the agent's most preferred one. Because of the large population assumption, the principal's action consists of choosing a fraction $0 \leq \alpha \leq 1$ of the population to which he will apply action A. We will call this fraction the *audit rate*. The upper bound on α does not have to be equal to 1, but can instead be set to $\bar{\alpha}$ to indicate that it is not possible to audit the whole population.

This diametric opposition of the principal and agents' interests suggests that the game has no pure strategy equilibria, as indeed can be checked from the figure and the relationship between the various payoffs. In fact, similar to a game of matching pennies, the single stage game as well as its repeated version possess only a unique mixed equilibrium. Let the audit rate and the fraction of C players in the fully rational setting be given by α_{Nash} and x_{Nash}, respectively. It is straightforward to verify that

$$\begin{aligned} \alpha_{Nash} &= \frac{v_3 - v_2}{v_3}; \\ x_{Nash} &= \frac{c_2}{c_3 + c_2 - c_1} \end{aligned} \tag{1}$$

As mentioned, we consider this game in an infinitely-repeated setting. Each moment in time, the game in Figure 1 is played. We will let the state of the system at time t be the fraction of the population taking action C at that time, and we will denote this fraction by $x(t)$. The principal's choice of audit rate at time t is denoted by $\alpha(t)$. Given a state $x(t)$, audit rate $\alpha(t)$, and denoting the payoff to the principal at time t by $g(t)$, we can write

$$\begin{aligned} g(x(t), \alpha(t)) &= c_1\alpha(t)x(t) + c_2\alpha(t)(1 - x(t)) + c_3(1 - \alpha(t))x(t) \\ &= (c_1 - c_2 - c_3)\alpha(t)x(t) + c_2\alpha(t) + c_3x(t) \end{aligned} \tag{2}$$

where the terms in (2) correspond to the costs discussed above. The first term is the cost associated with catching offending agents, the second term represents the cost of auditing honest agents, and the last term is the cost of ignoring agents who were in fact playing action C.

2.2 Learning Dynamics

The learning dynamics work as follows. After each round of the game, members of the population are randomly matched to compare and contrast strategies and payoffs. Under our model, there are only two possible scenarios that can lead to switching strategies: an agent who obtained the outcome (C, A) considers changing his strategy if he meets an agent who played H. Similarly, an agent who played H considers changing his strategy to C if he meets an agent who obtained the outcome (C, I). The probabilities with which these changes in strategy occur depend on the differences in payoffs between agents, as well as a transmission factor $k > 0$. We will think of k as a 'speed of transmission': the willingness of an agent to change their strategy when faced with a potentially better one. Without loss of generality, we will assume that an agent obtaining payoff u switches to the strategy of an agent getting payoff v with probability $\max\{0, \frac{v-u}{v}\}$. From Figure 1, the probability of switching in the first scenario is simply $\min\{k\frac{v_1-0}{v_1} = k, 1\}$. The probability of switching in the second scenario is given by $\min\{k\frac{v_3-v_1}{v_3}, 1\}$. It is important to stress that the way these probabilities are defined does not affect any structural results we obtain. Any scheme where the switching probabilities are proportional to the payoff differences essentially leads to the same results. We will make the derivations less cumbersome and more general by assuming that the switch in scenario one happens with probability p and in scenario 2 with probability q. We can later substitute for p and q with whatever values that are appropriate for the application under consideration. Utilizing this notation, the fraction of switchers from C to H at any moment t is equal to the fraction of C players who were audited, $\alpha(t)x(t)$, multiplied by the probability of meeting an H player, which is $1 - x(t)$, times the probability of switching p. Likewise, the fraction of switchers from H to C is equal to the fraction of H players, $1 - x(t)$, who meet C players that were not audited, which is $x(t)(1-\alpha(t))$, multiplied by the probability q. We can then write the dynamics of the system as a function of $x(t)$ and $\alpha(t)$

$$\begin{aligned}\dot{x}(t) = f(x(t), \alpha(t)) &= q(1 - \alpha(t))x(t)(1 - x(t)) - p\alpha(t)x(t)(1 - x(t)) \\ &= x(t)(1 - x(t))(q - \alpha(t)(q + p)) \qquad (3)\end{aligned}$$

2.3 Objective

The principal's problem is now the following. Given the different values in Figure 1, the parameters of the problem, and the learning dynamics, the principal is interested in minimizing his long-run discounted cost subject to those dynamics. This long-run cost is the sum of all costs accrued from playing the game over time. Recall that the payoff at time t is given by (2). The problem can then be written as

$$\begin{aligned}&\min_{\alpha(t)} \int_0^\infty e^{-rt}((c_1 - c_2 - c_3)\alpha(t)x(t) + c_2\alpha(t) + c_3x(t))dt \qquad (4)\\ &\text{s.t.} \qquad \dot{x}(t) = x(t)(1 - x(t))(q - \alpha(t)(q + p)) \\ &\qquad\qquad 0 \le \alpha(t) \le 1\end{aligned}$$

where $0 \le r < 1$ is a discount factor. Thus the principal's problem involves finding the function $\alpha^*(t)$ that solves (4). Like any dynamic problem, the difficulty facing the principal is that current decisions affect not only the immediate cost but also future costs through the dependence of the rate of change of $x(t)$ on $\alpha(t)$.

3 Optimal Policy

3.1 Single Round

Before delving into finding the optimal solution to (4), let us first develop an intuition by considering the solution if the game is played only once. The stage game cost described by (2) can be factored and rewritten as

$$g(x, \alpha) = c_3 x + \alpha(c_2 + (c_1 - c_2 - c_3)x)$$

and is obviously a linear function in α. This implies that depending on the value of x, α takes the values of either 0 or 1 in the optimal solution. Specifically, the optimal solution to the single period problem is given by

$$\alpha^* = \begin{cases} 0, \, x < \frac{c_2}{c_2+c_3-c_1}; \\ 1, \, x \geq \frac{c_2}{c_2+c_3-c_1}. \end{cases} \tag{5}$$

which is well defined because of the relationship stipulated on the costs. Thus, *assuming that x is known*, the optimal solution to a single period problem takes the form of a threshold rule: if the fraction of C players is low enough, it does not pay to audit anybody since the cost of auditing honest agents outweighs the gains from catching C players. Conversely, when the concentration of C players is over a certain level, then it is always better to audit indiscriminately since the costs incurred in auditing H players are more than made up for by catching every single C player in the population. It is easy to see that the optimal cost $g^*(x)$ is a concave function of x:

$$g^*(x) = \begin{cases} c_3 x, & x < \frac{c_2}{c_2+c_3-c_1}; \\ c_2 + (c_1 - c_2)x, & x \geq \frac{c_2}{c_2+c_3-c_1}. \end{cases} \tag{6}$$

We will see that a part of the single period solution, where a crackdown occurs if the fraction of C players is above a certain threshold and nothing is done otherwise, is somewhat retained in the solution to the general problem. The nature of the optimal cost implies that, from a strictly policing viewpoint, the principal may prefer a higher ratio of cheaters in the population to a lower one, since it increases the rate of successful audits and incurs a lower overall cost than scenarios where resources are expended without additional benefit.

3.2 General Policy

We will derive the optimal policy for (4) by formulating the Hamiltonian function and using the Euler-Lagrange equation. We assume that the principal knows $x(0)$, the initial state of the system. This is without loss of generality, since if that was not the case then the large population assumption together with the law of large numbers and the fact that state transitions happen with probability 1 ensure that the principal can initially determine the state of the system by auditing a random sample of the population. The current value Hamiltonian function for the problem maps triplets $(x, \alpha, \lambda) \in [0,1] \times [0,1] \times R$ to real numbers and is given by

$$\begin{aligned} H(x, \alpha, \lambda) &= g(x, \alpha) + \lambda f(x, \alpha) \\ &= c_3 x + \alpha(c_2 + (c_1 - c_2 - c_3)x) + \lambda x(1-x)(q - \alpha(q+p)) \\ &= c_3 x + \lambda q x(1-x) + \alpha(c_2 + (c_1 - c_2 - c_3)x - \lambda(p+q)x(1-x)) \end{aligned} \quad (7)$$

where λ is a co-state variable that one can think of as a price attached to the change induced in x through the decision α. Of course, like the state x and the control α, λ itself is also a function of time, but the power of the Hamiltonian approach is that it reduces the general problem to an essentially single period one. The following lemma utilizes the Hamiltonian to provide necessary (but not sufficient) conditions on the optimal control trajectories.

Lemma 1. *The optimal control for Problem (4) is a bang-bang solution.*

Proof. A bang-bang solution implies that $\alpha(t)$ takes on extremal values in its domain until the solution trajectory reaches a final state. Let us denote by $\alpha^*(t)$ and $x^*(t)$ the optimal control and state trajectories. By the Minimum Principle, it must hold at each moment in time that

$$\begin{aligned} \alpha^*(t) &= \arg\min_{0 \le \alpha \le 1} \; H(x^*(t), \alpha, \lambda(t)) \\ &= \arg\min_{0 \le \alpha \le 1} \; c_3 x + \lambda q x(1-x) + \alpha(c_2 + (c_1 - c_2 - c_3)x - \lambda(p+q)x(1-x)) \end{aligned}$$

Similar to the single period problem, the Hamiltonian is a linear function in α. Minimizing the Hamiltonian w.r.t α, we find that the optimal control trajectory, $\alpha^*(t)$ satisfies

$$\alpha^*(t) = \begin{cases} 0, & \lambda(t) < \frac{c_2 + (c_1 - c_2 - c_3)x(t)}{(p+q)x(t)(1-x(t))}; \\ 1, & \lambda(t) > \frac{c_2 + (c_1 - c_2 - c_3)x(t)}{(p+q)x(t)(1-x(t))}; \\ [0,1], & \lambda(t) = \frac{c_2 + (c_1 - c_2 - c_3)x(t)}{(p+q)x(t)(1-x(t))}. \end{cases} \quad (8)$$

Thus α assumes values at the boundary except when $\lambda(t) = \frac{c_2 + (c_1 - c_2 - c_3)x(t)}{(p+q)x(t)(1-x(t))}$, in which case α disappears from the Hamiltonian and can be set to any value in its domain. However, as we will see shortly, on the optimal control and state trajectories this case cannot happen except for precisely a single pair (α^*, x^*).

Lemma 1 implies that, except for the third case where the co-state variable is exactly equal to the R.H.S, the optimal control either audits the whole population or does nothing. This provides some information about the structure of the optimal policy, but not enough to completely characterize it. To do this, let us formulate (4) as a calculus of variations problem. From (3), we have

$$\alpha(t) = \frac{1}{p+q}\left(p - \frac{\dot{x}(t)}{x(t)(1-x(t))}\right)$$

Substituting this into the objective, the problem becomes

$$\min_{x(t)} \int_0^\infty e^{-rt} g\left(x(t), \frac{1}{p+q}\left(p - \frac{\dot{x}(t)}{x(t)(1-x(t))}\right)\right) dt$$
$$= \min_{x(t)} \int_0^\infty e^{-rt}\left(c_3 x(t) + \frac{(c_2 + (c_1 - c_2 - c_3)x(t))\left(p - \frac{\dot{x}(t)}{(1-x(t))x(t)}\right)}{p+q}\right) dt \quad (9)$$

The solution to (9) provides a necessary condition on the optimal state trajectory. Specifically, the following lemma tells us that there is a constant for which the integral in (9) is stationary.

Lemma 2. *Let $x^*(t)$ be the minimizer to (9), then $x^*(t) = C$, where C is a constant that depends on the parameters of the problem.*

Proof. See Appendix.

We now fully characterize the optimal policy.

Theorem 1. *There is a value $\bar{x}$ such that the optimal policy audits everybody whenever $x(t) > \bar{x}$ and does nothing when $x(t) < \bar{x}$. If $x(t) = \bar{x}$ then the optimal policy sets $\alpha^*(t) = \frac{q}{p+q}$ and the system stays in this state indefinitely.*

Proof. We will show that the policy in the statement of the theorem is optimal by showing that an optimal policy exists and that only the policy given in the theorem satisfies the necessary conditions for an optimum. That an optimal policy exists follows from the boundedness of the cost per stage and the continuity of both g and f in the compact sets $x(t)$ and $\alpha(t)$. The presence of the discount factor ensures that the value of the optimal solution is $< \infty$.

From Lemma 2, we know that a necessary condition for the optimal path $x^*(t)$ to minimize (9) (and consequently, (4)), is that $x^*(t)$ is a constant, which we will denote by $\bar{x}$ (where $\bar{x}$ is as given in the proof of Lemma 2). This implies that as soon as $x^*(t) = \bar{x}$ there should be no further changes in the system, so that $\dot{x}^*(t)$ is equal to zero. Given the system dynamics in (3), this occurs if

$$f(x^*(t), \alpha^*(t)) = 0$$
$$x^*(t)(1 - x^*(t))(q - \alpha^*(t)(q+p)) = 0$$

For any nontrivial specification of the problem, $\bar{x}$ is neither equal to zero or one, and hence the only solution to the above equation is $\alpha^*(t) = \frac{q}{p+q}$. From (8), we have to have $\lambda(t) = \frac{c_2 + (c_1 - c_2 - c_3)\bar{x}}{(p+q)\bar{x}(1-\bar{x})}$. The R.H.S of this is a constant, and hence $\dot{\lambda}(t) = 0$ and the system remains in the state $(\bar{x}, \frac{q}{q+p})$ forever.

Now consider any trajectory that sets $\alpha(t) \neq 1$ when $x^*(t) > \bar{x}$. By Lemma 1, if $x^*(t) \neq \bar{x}$ and $\alpha(t) \neq 1$ then $\alpha(t) = 0$[2] , in which case $\dot{x}(t) > 0$ and $x(t)$ increases. Let $x(t_1) > \bar{x}$ and $\alpha(t_1) = 0$, then for $t_2 > t_1$, $x(t_2) > x(t_1)$, i.e. the

[2] The Minimum Principle posits the following condition on $\dot{\lambda}(t)$; $\dot{\lambda}(t) = -\frac{\partial H(x^*(t), \alpha^*(t), \lambda(t))}{\partial x}$, so that the third case in (8) cannot hold unless $x(t)$ is a constant.

system moves farther from $\bar{x}$. However, because of Lemma 2, we know that the system should eventually move *towards* $\bar{x}$. Since the system is continuous, the trajectory going from $x(t_2)$ to $\bar{x}$ has to pass through $x(t_1)$ again, at which point the system returns to the same state it was in at time t_1, but with the additional cost accrued between times t_1 and t_2 added to the total cost, indicating that such a scenario cannot be optimal, and that it would have been cheaper to set $\alpha(t_1) = 1$. The reverse argument applies in the case of $x(t) < \bar{x}$.

Thus the optimal policy drives the fraction $x(t)$ to its steady state value as quickly as possible, by not doing anything when $x(t) < \bar{x}$ or by cracking down on the population when $x(t) > \bar{x}$. Once the steady state is reached, the system stays there forever through fixing the audit rate at the value given in the statement of the theorem.

4 Discussion

4.1 Comparison with Nash Equilibrium

It is natural to ask how the behavioral solution for the class of games we considered fares in comparison to the fully rational Nash equilibrium outcome. We have already discussed in Section 2.1 that the (fully rational) repeated game possesses a unique equilibrium, given by (1). This equilibrium is also a *center* of the repeated behavioral game. This means that, under the replicator assumption, the principal has a strategy such that if the game is played long enough, the fraction with which each action is played is the same as the corresponding fraction in the Nash equilibrium [7], i.e. the principal can implement the Nash outcome in the behavioral setting, if he so desires. However, the optimal solution that we obtained in this paper is not the Nash equilibrium, indicating that the Nash solution is dominated by the policy in Theorem 1. Furthermore, as soon as the game reaches steady state, the optimal policy does *less* auditing than the Nash solution. Let us denote the audit rate in the behavioral setting by α_B. From Theorem 1, α_B is given by $\frac{q}{p+q}$. Replacing p and q by the values from Section 2.2, we have $p = k$ and $q = k\frac{v_3 - v_1}{v_3}$, and hence

$$\alpha_B = \frac{q}{p+q} = \frac{\frac{v_3 - v_1}{v_3}}{1 + \frac{v_3 - v_1}{v_3}} \tag{10}$$

which is always *strictly less* than the Nash audit rate in (1). Because of this, the Nash solution never coincides with the policy in Theorem 1, so that the optimal solution always gives a strictly better outcome for the principal while at the same time reducing the amount of auditing required. It is worth noting that the speed of transmission k has no effect on α_B.

4.2 An Empirical Example

We have analyzed our model in a continuous time framework. In reality however, many of the games that fit the model take place in discrete time, or the resources required by the optimal solution can be infeasible to implement forever. In both of these scenarios, the level of $x(t)$ inadvertently increases above $\bar{x}$, and hence the optimal solution cracks down on the population by setting α^* to its maximum possible value, in an attempt to bring $x(t)$ back to $\bar{x}$. Because of the discreteness, the crackdowns always bring the value of $x(t)$ below $\bar{x}$, hence leading to a short period of low activity on the principal's part. The whole cycle is then repeated as $x(t)$ increases again. These periodic crackdowns are widely observed in many situations. In a recent paper [5], the authors empirically observe crackdowns by the police on speeders in Belgium. The paper mentions the periodicity of such crackdowns, but does not provide an explaination for such behavior. It is also mentioned that crackdowns are planned as early as a month in advance. Both of these observations are explained by our model. The recurrence of the crackdowns takes place as the police tries to bring the fraction of speeders to an optimal level, and since the evolution of the population of speeders can be determined from the current state and future controls of the system, the time at which such a crackdown would be necessary can be determined in advance as well.

5 Conclusion

We have shown how a principal can exploit myopic social learning to his advantage for a wide class of games where the interests of the population and the principal are directly opposed. In addition to the class of games we presented, the application domain of the methods we employed in this paper is vast. The basic idea is to indirectly influence decisions in the population through manipulating the payoffs associated with certain actions. Naturally, since the modified payoffs are not part of the initial system, such a manipulation comes at a personal and/or a social cost. In our example the principal had to expend an initial cost by either over-auditing or by letting the guilty population go unpunished. At the same time, there is a social cost to increased auditing that comes from the disutility honest agents obtain from being audited (the case where $v_2 > v_3$ in Figure 1). This initial phase however, is justified by later gains: since the population's reaction time to changes in the system is not instantaneous, the principal can revert back to the original game while the population *still plays as if they are in the modified game*. During this time, the principal enjoys a period of improved system performance. Generally, the solution either takes the form of a policy like the one we saw in this paper, where an initial period of extreme (in)activity is followed by a steady state, or it can be more cyclical in nature, with a cycle consisting of a phase that creates, via population learning, a certain impression about the environment followed by a phase where that impression is exploited. One obvious application is advertising. In this scenario, periods of heavy (and costly) advertising are followed by periods of relatively little advertising activity. During these latter periods, the effects from the initial advertising campaign continues to reverberate through the population, essentially providing free advertising until the effect dies down, at which time the advertiser starts the cycle again. A very different example is traffic regulation through periodic closures of specific roads. Such closures force drivers to change their driving habits.

Later, when these roads are re-opened, drivers take a while to adjust back to the initial equilibrium, as can be seen in [6]. Depending on the system's parameters, this lag in adjustment can provide the population with an average decrease in travel latency[3]. Applying the same approach of exploiting behavioral trends to other behavioral models would be an interesting next step in this line of research, with an eventual goal of cataloging the benefits that a principal or a society can obtain (or lose) as the level of sophistication of the population increases.

References

1. Acemoglu, D., Bimpikis, K., Ozdaglar, A.: Communication Dynamics in Endogenous Social Networks. Working Paper (2010)
2. Acemoglu, D., Dahleh, M., Lobel, I., Ozdaglar, A.E.: Bayesian learning in social networks. NBER Working Paper (2008)
3. Borgers, T., Sarin, R.: Learning through reinforcement and replicator dynamics. Journal of Economic Theory 77(1), 1–14 (1997)
4. Crawford, V.P., Kugler, T., Neeman, Z., Pauzner, A.: Behaviorally Optimal Auction Design: Examples and Observations. Journal of the European Economic Association 7(2-3), 377–387 (2009)
5. Eeckhout, J., Persico, N., Todd, P.: A Theory of Optimal Random Crackdowns. American Economic Review (2010)
6. Fischer, S., Vöcking, B.: On the evolution of selfish routing. In: Albers, S., Radzik, T. (eds.) ESA 2004. LNCS, vol. 3221, pp. 323–334. Springer, Heidelberg (2004)
7. Fudenberg, D., Levine, D.K.: The theory of learning in games. The MIT Press, Cambridge (1998)
8. Fudenberg, D., Maskin, E.: The folk theorem in repeated games with discounting or with incomplete information. Econometrica: Journal of the Econometric Society 54(3), 533–554 (1986)
9. Kamenica, E., Gentzkow, M.: Bayesian persuasion. NBER Working Paper (2009)
10. Myerson, R.B.: Optimal auction design. Mathematics of operations research 6(1), 58 (1981)
11. Nisan, N., Ronen, A.: Algorithmic mechanism design. Games and Economic Behavior 35(1-2), 166–196 (2001)

[3] For example, in Pigou's network with latencies 1 on the top link and x on the bottom link, a traffic authority can change the latency on the bottom link by slightly *increasing* it above 1 (say, by closing down a few lanes), so that the population migrates upwards towards the now-faster link. The latency is then restored back to its original value x, and the population starts migrating downwards. During that second phase, the traffic is temporarily more balanced than the Nash equilibrium, where everyone uses the bottom link all the time. The process is then repeated. The traffic authority chooses the exact latencies for the bottom link as well as the duration with which they remain in effect in order to obtain a net-gain in the average latency in the system.

Appendix

A Proof of Lemma 2

Proof. Denoting the function inside the integral in (9) by $L(t, x, \dot{x})$, the Euler-Lagrange equation gives another necessary condition that the optimal $x^*(t)$, if it exists, satisfies. Writing down the equation, we have

$$\begin{aligned}
0 &= \frac{\partial L}{\partial x} - \frac{\partial}{\partial t}\frac{\partial L}{\partial \dot{x}} \\
&= e^{-rt}\left(c_3 + \frac{1}{p+q}(c_2 + (c_1 - c_2 - c_3)x(t))\left(\frac{\dot{x}(t)}{(1-x(t))x(t)^2} - \frac{\dot{x}(t)}{(1-x(t))^2 x(t)}\right)\right) \\
&+ e^{-rt}\left(\frac{(c_1 - c_2 - c_3)\left(p - \frac{\dot{x}(t)}{(1-x(t))x(t)}\right)}{p+q}\right) \\
&- e^{-rt}\frac{\left(r(-1+x(t))x(t)(-c_2 + (-c_1 + c_2 + c_3)x(t)) + \left(c_2 - 2c_2 x(t) + (-c_1 + c_2 + c_3)x(t)^2\right)\dot{x}(t)\right)}{((p+q)(-1+x(t))^2 x(t)^2)}
\end{aligned}$$

After some algebra and simplifying the above, we get

$$\frac{e^{-rt}\left((c_2 r - (c_1 + c_2)(p - r) + c_3(q + r))x(t) + ((c_1 - c_2)p + c_3 q)x(t)^2\right)}{(p+q)(x(t)-1)x(t)} = 0$$

which is a quadratic function in $x(t)$. Solving that equation and enforcing the constraint that $0 \leq x(t) \leq 1$, we obtain the solution

$$x^*(t) = \frac{(c_2 - c_1)p - c_3 q + (c_1 - c_2 - c_3)r + \sqrt{4c_2((c_2 - c_1)p - c_3 q)r + ((c_1 - c_2)p + c_3 q + (c_1 r - c_2 - c_3)r)^2}}{2((c_2 - c_1)p - c_3 q)}$$

which is time-independent and only depends on the parameters of the problem.

The difference between $x^*(t)$ and x_{Nash} depends on the parameters. For example, if we set all the parameters to 1 and compare the resulting steady state optimum with the Nash equilibrium in (1). We get

$$x^*(t) = \frac{c_2}{c_3 + \sqrt{c_3^2 + c_2^2 - c_2(c_1 + c_3)}}$$

which is always less than x_{Nash}.

The Limits of Smoothness: A Primal-Dual Framework for Price of Anarchy Bounds

Uri Nadav[1,*] and Tim Roughgarden[2,**]

[1] Department of Computer Science, Stanford University, 470 Gates Building, 353 Serra Mall, Stanford, CA 94305
urinadav@cs.stanford.edu
[2] Department of Computer Science, Stanford University, 462 Gates Building, 353 Serra Mall, Stanford, CA 94305
tim@cs.stanford.edu

Abstract. We show a formal duality between certain equilibrium concepts, including the correlated and coarse correlated equilibrium, and analysis frameworks for proving bounds on the price of anarchy for such concepts. Our first application of this duality is a characterization of the set of distributions over game outcomes to which "smoothness bounds" always apply. This set is a natural and strict generalization of the coarse correlated equilibria of the game. Second, we derive a refined definition of smoothness that is specifically tailored for coarse correlated equilibria and can be used to give improved POA bounds for such equilibria.

1 Introduction

A rigorous way to argue that a system with self-interested participants has good performance is to prove that every "plausible outcome" of the system has objective function value close to that of an optimal outcome. For example, one could model a system as a one-shot game, identify "plausible outcomes" with the pure-strategy Nash equilibria (PNE) — outcomes in which each player deterministically picks one strategy so that it has no incentive to unilaterally deviate from it — and prove a relative approximation bound for the PNE of the game.

Such *price of anarchy (POA)* bounds become increasingly robust and compelling as one increases the set of "plausible outcomes". For example, a POA bound that applies only to the pure-strategy Nash equilibria of a game presumes that the system reaches such a state. This can be a bold assumption, for example in contexts where it is computationally difficult to compute a PNE (see e.g. [7]). A POA bound that applies more generally to "easily learned" outcomes, such as the correlated equilibria [1] or coarse correlated equilibria [8] of a game, presumes

* Supported by the ONR Young Investigator and PECASE Awards of the second author.
** Supported in part by NSF CAREER Award CCF-0448664, an ONR Young Investigator Award, an ONR PECASE Award, and an AFOSR MURI grant.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 319–326, 2010.

far less about the game's participants [2,3]. Of course, worst-case approximation bounds typically degrade as the assumptions about play are weakened — for example, the expected performance of the worst coarse correlated equilibrium of a game is typically worse than that of the worst pure-strategy Nash equilibrium.

This paper shows a precise duality between certain equilibrium concepts, including correlated and coarse correlated equilibria, and analysis frameworks for proving POA bounds for such concepts. This duality makes formal the intuitive trade-off between the plausibility of the rationality assumptions imposed on the game participants and the quality of the corresponding worst-case approximation bound. We offer two applications.

1. Roughgarden [11] showed that every POA bound proved using a "smoothness argument" (see Definition 1) — the most frequently employed method for establishing POA bounds (e.g. [5,6,9,10,12]) — applies automatically to (at least) all CCE of the game. A basic problem is to characterize the distributions over outcomes to which smoothness bounds always apply. We solve this problem (Theorem 1) and show that the answer is a generalization of CCE in which the *average* regret of players is non-positive, as opposed to the CCE condition that *every* player has non-positive regret (see Definition 2).
2. Applying the duality result in the opposite direction yields analysis frameworks that are guaranteed to be tight for the corresponding equilibrium concepts. We illustrate this idea with the set of CCE, where the corresponding multi-parameter analysis framework refines the simpler two-parameter smoothness paradigm in [11]. This more flexible analysis framework is, by definition, specifically tailored for CCE and can be used to give improved POA bounds for such equilibria.

2 The Primal-Dual Framework

Section 2.1 reviews standard definitions of cost-minimization games, equilibrium concepts, and the price of anarchy. Section 2.2 presents our first contribution and shows that, for every equilibrium concept that can be expressed as the probability distributions over outcomes that are solutions to a set of homogeneous inequalities, there is a corresponding analysis framework that is guaranteed to prove tight bounds on the price of anarchy for that concept. Our second contribution, described in Section 2.3, is an application of this framework: POA bounds proved using the "smoothness paradigm" introduced in [11] apply precisely to a generalization of coarse correlated equilibria that we call "average coarse correlated equilibria". Section 2.4 demonstrates how a sharper analysis method tailored specifically for coarse correlated equilibria, which follows directly from our primal-dual framework, can be used to prove bounds superior to those that follow from the standard smoothness paradigm.

2.1 Preliminaries

Cost-minimization games. We denote a cost-minimization game by a tuple $\Gamma = (N, \{S_i\}_{i \in N}, \{C_i\}_{i \in N})$, where $N = \{1, \ldots, n\}$ is the set of n players, S_i is the set of actions of player i, and $c_i : S \mapsto \mathbb{R}^{++}$ is player's i positive cost function, where $S = S_1 \times S_2 \times \cdots \times S_n$ is the joint action set.[1] We use $\Delta(S)$ to denote the set of probability distributions over S and $\mathbf{s}_{-i}$ to denote the strategies played in $\mathbf{s}$ by the players other than i.

Equilibrium concepts and the price of anarchy. In this paper, we consider equilibrium concepts that can be described as subsets of $\Delta(S)$. In particular, recall that a *correlated equilibrium (CE)* is a joint probability distribution σ over outcomes of Γ with the property that $\mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s})|\mathbf{s}_i] \le \mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s}'_i, \mathbf{s}_{-i})|\mathbf{s}_i]$ for every i and $\mathbf{s}_i, \mathbf{s}'_i \in S_i$. Thus a distribution σ over outcomes is a CE if the following holds for a random sample $\mathbf{s} \sim \sigma$: for each player i and "recommended strategy" s_i, the player minimizes its expected cost, conditioned on the recommendation s_i and assuming that other players play according to $\mathbf{s}_{-i}$, by playing s_i. CE are also the limits of sequences of repeated play in which each player has vanishing per-step *swap* or *internal regret* (see [4]). The mixed Nash equilibria of a game are precisely the CE that are also product distributions.

A *coarse correlated equilibrium (CCE)* is a joint probability distribution σ over outcomes of Γ with the property that $\mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s})] \le \mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s}'_i, \mathbf{s}_{-i})]$ for every i and $\mathbf{s}'_i \in S_i$. These equilibrium constraints consider only player deviations that are independent of the recommendation s_i, so every CE is also a CCE (and, generally, the converse fails). CCE are also the limits of sequences of repeated play in which each player has vanishing per-step *external regret* (see [4]).

We assume that the objective function is to minimize the total cost $C(\mathbf{s}) = \sum_{i\in N} C_i(\mathbf{s})$, and use $\mathbf{s}^*$ to denote an optimal outcome. The *price of anarchy (POA)* of a game for an equilibrium concept EQ $\subseteq \Delta(S)$ is the ratio between the expected total cost of the worst (i.e., highest-cost) equilibrium $\sigma \in$ EQ and the social cost of $\mathbf{s}^*$.

2.2 A Primal-Dual Framework for POA Bounds

This section describes our primal-dual framework, which formalizes a duality between equilibrium concepts that can be represented as solutions of homogeneous inequalities and analysis methods that are necessary and sufficient to prove tight bounds on the POA for such concepts.

Fix a game Γ, and an equilibrium concept EQ that can be written as EQ $= \{\sigma \in \Delta(S) : A\sigma \le 0\}$, where $A \in \mathbb{R}^{|S|\times m}$ is a matrix that can depend on players' cost functions in Γ. For example, the equilibrium concepts CE and and CCE can be described in this way:

Example 1 (Correlated Equilibria). We can express the CE of a cost-minimization game as the probability distributions over outcomes that satisfy

$$\mathrm{CE} = \left\{\sigma : \sum_{\mathbf{s}:\mathbf{s}_i = s_i} \sigma_{\mathbf{s}}(C(s'_i, \mathbf{s}_{-i}) - C(\mathbf{s})) \le 0, \text{for every } i \in N, \text{ and } s_i, s'_i \in S_i, \sigma_{\mathbf{s}} \ge 0\right\}.$$

[1] Our results can be reworked without difficulty for payoff-maximization games.

Example 2 (Coarse Correlated Equilibrium). We can express the CCE of a cost-minimization game as the probability distributions that satisfy

$$\text{CCE}=\left\{\sigma : \sum_{\mathbf{s}} \sigma_{\mathbf{s}}(C(s_i', \mathbf{s}_{-i}) - C(\mathbf{s})) \leq 0, \text{ for every } i \in N, \text{ and } s_i' \in S_i, \sigma_{\mathbf{s}} \geq 0\right\}.$$

A third example will arise naturally in Section 2.3.

We now develop our simple primal-dual framework. We can formally write the POA of a game Γ and an equilibrium concept EQ as

$$\text{POA}_{\text{EQ}}(\Gamma) = \sup_{\sigma\in\text{EQ}} \left\{\frac{\mathbf{E}_{\mathbf{s}\sim\sigma}[\text{C}(\mathbf{s})]}{\text{C}(\mathbf{s}^*)}\right\}.$$

After scaling by $C(\mathbf{s}^*)$, this maximization problem can be expressed as the solution of the following linear program:

$$\begin{array}{lll} \text{PRIMAL-EQ} : & \text{Maximize} & \sum_{\mathbf{s}\in S} \sigma_{\mathbf{s}} C(\mathbf{s}) \\ & \text{subject to} & \sum_{\mathbf{s}\in S} \sigma_{\mathbf{s}} C(\mathbf{s}^*) = 1 \\ & & A\sigma \leq 0, \sigma_{\mathbf{s}} \geq 0 \end{array}$$

The dual problem of PRIMAL-EQ is

$$\begin{array}{lll} \text{DUAL-EQ} : & \text{Minimize} & p \\ & \text{subject to} & C(\mathbf{s}^*)p \cdot \mathbf{1^n} + zA^T \geq \mathbf{0}, \\ & & z \geq 0, p \geq 0, \qquad\qquad p \in \mathbb{R}, z \in \mathbb{R}^m \end{array}$$

where $\mathbf{1^n}$ is the n dimensional vector with all entries 1, and m is the number of inequalities in A.

We say that a game is *p-bounded* for the equilibrium concept EQ if there exists a vector $z \in \mathbb{R}^m$ such that the pair (p, z) is feasible for DUAL-EQ, or simply p-bounded when the equilibrium concept is clear. We refer to z as a *dual certificate* for Γ and EQ.

Strong linear programming duality immediately implies the following.

Proposition 1. *For every cost-minimization game Γ and equilibrium concept* EQ *representable as the solution of homogeneous inequalities,* $\text{POA}_{\text{EQ}}(\Gamma) \leq \text{p}$ *if and only if Γ is p-bounded for* EQ.

The following example instantiates Proposition 1 for correlated equilibria. The next two sections provide further examples.

Example 3 (Primal-Dual Framework for Correlated Equilibria). For a cost-minimization game Γ, the quantity $\text{POA}_{\text{CE}}(\Gamma)$ is, by definition, the optimal solution to the problem PRIMAL-CE:

$$\begin{array}{lll} \text{PRIMAL-CE} : & \text{Maximize} & \sum_{\mathbf{s}\in S} \sigma_{\mathbf{s}} C(\mathbf{s})/C(\mathbf{s}^*) \\ & \text{subject to} & \sum_{\mathbf{s}\in S} \sigma_{\mathbf{s}} = 1 \\ & & \sum_{\mathbf{s}:\mathbf{s}_i=a} \sigma_{\mathbf{s}}(C(b, \mathbf{s}_{-i}) - C(\mathbf{s})) \leq 0, \text{ for every } i \in N, \\ & & \qquad\qquad \text{and } a, b \in S_i \\ & & \sigma_{\mathbf{s}} \geq 0. \end{array}$$

The corresponding DUAL-CE problem is then

$$\begin{aligned} \text{DUAL-CE : Minimize} \quad & p \\ \text{subject to} \quad & pC(\mathbf{s}^*) + \textstyle\sum_i \sum_{b \in S_i} z^i_{\mathbf{s}_i, b} \left(C_i(\mathbf{s}) - C_i(b, \mathbf{s}_{-i})\right) \geq C(\mathbf{s}), \\ & \qquad\qquad\qquad\qquad\qquad\qquad \text{for all } \mathbf{s} \in S \\ & z \geq 0, p \geq 0. \end{aligned}$$

Hence, to prove an upper bound of p on the POA for correlated equilibrium, it suffices to show that the game is p-bounded for CE — that is, to find a dual certificate $z = \{z^i_{a,b}\}_{i \in N, a,b \in S_i}$ so that (p, z) is feasible for DUAL-CE.

2.3 The Limits of (λ, μ)-Smoothness

Roughgarden [11] defined a smooth game as follows.

Definition 1 (Smooth Games). *A cost-minimization game with minimum-cost outcome* $\mathbf{s}^*$ *is* (λ, μ)-smooth *if*

$$\sum_{i=1}^{k} C_i(\mathbf{s}^*_i, \mathbf{s}_{-i}) \leq \lambda \cdot C(\mathbf{s}^*) + \mu \cdot C(\mathbf{s}) \tag{1}$$

for every outcome $\mathbf{s}$*.*

One of the main results in [11] is that $\text{POA}_{\text{CCE}}(\Gamma) \leq \lambda/(1-\mu)$ whenever Γ is (λ, μ)-smooth.[2] In addition, many known POA bounds — often stated only for pure or mixed Nash equilibria — are or can be recast as smoothness bounds (see [11]), and thus these bounds "extend automatically" to the more general concept of CCE.

This section addresses the basic question of characterizing the distributions over outcomes to which a (λ, μ)-smoothness bound applies. The answer, which we derive via the primal-dual framework in the previous section, turns out to be a strict generalization of CCE that we call an *average coarse correlated equilibrium, with respect to* $\mathbf{s}^*$ (ACCE*).

Definition 2 (ACCE*)**.** *For a fixed game and an outcome* $r \in S$

$$\text{ACCE}^{\text{r}} = \{\sigma \in \Delta(S) : E_{\mathbf{s} \sim \sigma}[C(\mathbf{s})] \leq E_{\mathbf{s} \sim \sigma}[\sum_i C_i(r_i, \mathbf{s}_{-i})]\}.$$

When r *is the minimum-cost outcome* $\mathbf{s}^*$*, we abbreviate* $\text{ACCE}^{\mathbf{s}^*}$ *by* ACCE**.*

Conceptually, there are two differences between a CCE and an ACCE*. In a CCE, the expected cost incurred by a player is at most that of unconditionally deviating to an any fixed action — i.e., every player has non-positive "regret". ACCE* is a more permissive equilibrium concept. First, we measure the regret of

[2] In [11] the definition of (λ, μ)-smoothness requires that inequality (1) holds for every pair $\mathbf{s}, \mathbf{s}^*$ outcomes. The weaker requirement stated here still translates, via the same proofs, to an upper bound on the POA for CCE.

a player i by comparing its expected cost only to that incurred under a deviation to s_i^*, rather than to an arbitrary (or best) strategy. Second, in an ACCE*, some players i can have negative regret with respect to s_i^* as long as the *average* (over players) such regret is non-positive. Unsurprisingly, many games have ACCE* that are not CCE; the proof of Proposition 2 provides one concrete example.

The next theorem shows that every (λ, μ)-smoothness argument bounds the worst-case expected cost of *precisely* the set of ACCE*. This characterization has both positive and negative implications. First, even the *ACCE** distributions of a (λ, μ)-smooth game have good expected cost (and not only the CCE, as proved in [11]). Second, conversely, the worst-case *ACCE** constrains the best-possible upper bound that can be proved via a (λ, μ)-smoothness argument.

Theorem 1 (Duality Between (λ, μ)-Smoothness and $ACCE^*$). *For every cost-minimization game Γ, the best smoothness upper bound for Γ equals its POA for the equilibrium concept* ACCE*:

$$\inf\left\{\frac{\lambda}{1-\mu} : (\lambda, \mu) \text{ s.t. the game } \Gamma \text{ is } (\lambda, \mu)\text{-smooth}\right\} = \mathrm{POA}_{\mathrm{ACCE}^*}(\Gamma).$$

Proof. We prove that the (λ, μ)-smoothness requirements are equivalent to the constraints of the DUAL problem for the equilibrium concept ACCE*. We consider the linear fractional problem for obtaining the best (i.e., least) upper bound using (λ, μ)-smoothness:

$$\begin{aligned}
\text{(LFP)} : \text{Minimize} \quad & \tfrac{\lambda}{1-\mu} \\
\text{subject to} \quad & \textstyle\sum_{i \in N} C_i(\mathbf{s}_i^*, \mathbf{s}_{-i}) \le \lambda C(\mathbf{s}^*) + \mu C(\mathbf{s}), \ \text{ for all } \mathbf{s} \in S \\
& \mu < 1.
\end{aligned}$$

By rearranging terms in the first inequality of problem (LFP) and dividing through by $1 - \mu > 0$ we obtain

$$\begin{aligned}
\text{(LFP2)} : \text{Minimize} \quad & \tfrac{\lambda}{1-\mu} \\
\text{subject to} \quad & \tfrac{\lambda}{1-\mu} C(\mathbf{s}^*) + \tfrac{1}{1-\mu}\left(C(\mathbf{s}) - \textstyle\sum_{i \in N} C_i(\mathbf{s}_i^*, \mathbf{s}_{-i})\right) \ge C(\mathbf{s}) \\
& \qquad \text{for all } \mathbf{s} \in S \\
& \mu < 1.
\end{aligned}$$

Now, re-writing (LFP2) with a change of variables $p = \frac{\lambda}{1-\mu}$, and $z = \frac{1}{1-\mu}$ gives the following linear program (LP):

$$\begin{aligned}
\text{(LP)} : \text{Minimize} \ & p \\
\text{subject to} \ & pC(\mathbf{s}^*) + z\left(C(\mathbf{s}) - \textstyle\sum_i C_i(\mathbf{s}_i^*, \mathbf{s}_{-i})\right) \ge C(\mathbf{s}), \ \text{ for all } \mathbf{s} \in S \\
& z > 0.
\end{aligned}$$

The dual problem of (LP) is:

$$\begin{aligned}
\text{(D)}: \text{Maximize} \ & \textstyle\sum_{\mathbf{s} \in S} \sigma_{\mathbf{s}} C(\mathbf{s}) \\
\text{subject to} \ & \textstyle\sum_{\mathbf{s} \in S} \sigma_{\mathbf{s}} C(\mathbf{s}^*) \le 1 \\
& \textstyle\sum_{\mathbf{s}} \sigma(\mathbf{s}) \left(\sum_i C(\mathbf{s}_i^*, \mathbf{s}_{-i}) - C(\mathbf{s})\right) \ge 0 \\
& \sigma_{\mathbf{s}} \ge 0, \ \text{ for all } \mathbf{s} \in S.
\end{aligned}$$

We can replace the first inequality in (D) with an equality since the social cost function is positive by assumption. Then, after scaling by $C(\mathbf{s}^*)$ we get an equivalent linear program that corresponds to the POA for ACCE*:

$$\begin{aligned}
(\text{PRIMAL-ACCE}^*): \text{Maximize } & \textstyle\sum_{\mathbf{s}\in S} \sigma_{\mathbf{s}} \frac{C(\mathbf{s})}{C(\mathbf{s}^*)} \\
\text{subject to } & \textstyle\sum_{\mathbf{s}\in S} \sigma_{\mathbf{s}} = 1 \\
& \textstyle\sum_{\mathbf{s}} \sigma(\mathbf{s}) \left(\sum_i C(\mathbf{s}_i^*, \mathbf{s}_{-i}) - C(\mathbf{s})\right) \geq 0 \\
& \sigma_{\mathbf{s}} \geq 0, \text{ for all } \mathbf{s} \in S.
\end{aligned}$$

2.4 Better Dual Certificates Give Better POA Upper Bounds

Theorem 1 shows that the smoothness analysis framework in [11] corresponds precisely to worst-case upper bounds on the set of ACCE*. In this section we assume that the goal is to prove upper bounds on the quantity $\text{POA}_{\text{CCE}}(\Gamma)$, and view the fact that (λ, μ)-smoothness bounds the expected cost of a strictly larger set of outcome distributions as an "accident". Motivated by this perspective, this section uses the primal-dual framework of Section 2.2 to derive a condition tailored for CCE that is sharper than (λ, μ)-smoothness and that can be used to prove better upper bounds on $\text{POA}_{\text{CCE}}(\Gamma)$.

Let CCE* denote the equilibrium concept where each player's expected cost is at most that of deviating to its action in $\mathbf{s}^*$, i.e., $\sigma \in \text{CCE}^*$, if and only if

$$\mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s})] \leq \mathbf{E}_{\mathbf{s}\sim\sigma}[C_i(\mathbf{s}_i^*, \mathbf{s}_{-i})]$$

for every $i \in N$. Obviously, $\text{CCE} \subseteq \text{CCE}^* \subseteq \text{ACCE}^*$.

Proposition 1 shows that every equilibrium concept that is the solution to homogeneous inequalities, such as CCE*, has a corresponding tight analysis framework. To bound the POA for CCE*, we only need to find a suitable dual certificate. The DUAL-CCE* problem is

$$\begin{aligned}
(\text{DUAL-CCE}^*): \text{Minimize } & p \\
\text{subject to } & pC(\mathbf{s}^*) + \textstyle\sum_{i\in N} z_i \left(C_i(\mathbf{s}) - C_i(\mathbf{s}_i^*, \mathbf{s}_{-i})\right) \geq C(\mathbf{s}), \\
& \qquad \text{for all } \mathbf{s} \in S \\
& z_i \geq 0, \text{ for all } i \in N.
\end{aligned}$$

Thus, a dual certificate for CCE* is an n-dimensional vector z such that $pC(\mathbf{s}^*) + \sum_{i\in N} z_i \left(C_i(\mathbf{s}) - C_i(\mathbf{s}_i^*, \mathbf{s}_{-i})\right) \geq C(\mathbf{s}),$ for all $\mathbf{s} \in S$. This is evidently more flexible than the single-parameter dual certificate one is forced to use for ACCE*. Can this flexibility lead to better worst-case upper bounds? The following proposition gives an affirmative answer.

Proposition 2. *There is a game Γ such that* $\text{POA}_{\text{CCE}}(\Gamma) < \text{POA}_{\text{ACCE}*}(\Gamma)$.

Proof (sketch): Consider a load balancing game with two jobs J_1, J_2, with weights 2 and 1 respectively, and two machines M_1, M_2 with latency functions

$$\begin{aligned}
&\ell_1(1) = 1, && \ell_1(2) = 2, \ell_1(3) = 3; \\
&\ell_2(1) = 1 + \epsilon, && \ell_2(2) = 2, \ell_2(3) = 4.
\end{aligned}$$

The optimal outcome $\mathbf{s}^*$ assigns J_1 to M_1 and J_2 to M_2, and has a social cost 3. For small enough ϵ, the best ACCE* dual certificate[3] is $z \approx 7/9$ which corresponds to a POA of $16/9 \approx 1.77$. For CCE* a dual certificate $(z_1, z_2) \approx (23/24, 5/12)$ exists, for a better POA bound of $49/30 \approx 1.63$. ■

Remark 1. There are games with an arbitrary gap between $\text{POA}_{\text{CCE}^*}$ and $\text{POA}_{\text{ACCE}^*}$, e.g., by changing the latency function ℓ_2 in the proof of Proposition 2 to $\ell_2(3) = H$, for a large enough H.

Remark 2. In contrast to Proposition 2, in symmetric games — where all players have the same strategy set and each player's cost depends only on its own strategy and the number of players that choose each strategy — the POA for ACCE* is equal to the POA for CCE*. We omit the easy argument.

References

1. Aumann, R.J.: Subjectivity and correlation in randomized strategies. Journal of Mathematical Economics 1(1), 67–96 (1974)
2. Blum, A., Even-Dar, E., Ligett, K.: Routing without regret: On convergence to Nash equilibria of regret-minimizing algorithms in routing games. In: Proceedings of the 25th Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 45–52 (2006)
3. Blum, A., Hajiaghayi, M., Ligett, K., Roth, A.: Regret minimization and the price of total anarchy. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC), pp. 373–382 (2008)
4. Blum, A., Mansour, Y.: Learning, regret minimization, and equilibria. In: Nisan, N., Roughgarden, T., Tardos, É., Vazirani, V. (eds.) Algorithmic Game Theory, ch. 4, pp. 79–101. Cambridge University Press, Cambridge (2007)
5. Christodoulou, G., Koutsoupias, E.: On the price of anarchy and stability of correlated equilibria of linear congestion games. In: Brodal, G.S., Leonardi, S. (eds.) ESA 2005. LNCS, vol. 3669, pp. 59–70. Springer, Heidelberg (2005)
6. Correa, J.R., Schulz, A.S., Stier Moses, N.E.: On the inefficiency of equilibria in congestion games. Games and Economic Behavior 64(2), 457–469 (2008)
7. Fabrikant, A., Papadimitriou, C.H., Talwar, K.: The complexity of pure Nash equilibria. In: Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC), pp. 604–612 (2004)
8. Hannan, J.: Approximation to Bayes risk in repeated plays. In: Dresher, M., Tucker, A., Wolfe, P. (eds.) Contributions to the Theory of Games, vol. 3, pp. 97–139. Princeton University Press, Princeton (1957)
9. Harks, T.: Stackelberg strategies and collusion in network games with splittable flow. In: Bampis, E., Skutella, M. (eds.) WAOA 2008. LNCS, vol. 5426, pp. 133–146. Springer, Heidelberg (2009)
10. Perakis, G.: The price of anarchy when costs are non-separable and asymmetric. Mathematics of Operations Research 32(3), 614–628 (2007)
11. Roughgarden, T.: Intrinsic robustness of the price of anarchy. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC), pp. 513–522 (2009)
12. Vetta, A.: Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions. In: Proceedings of the 43rd Annual Symposium on Foundations of Computer Science (FOCS), pp. 416–425 (2002)

[3] The equivalent, and optimal (λ, μ) pair for this upper bound is $(16/7, -2/7)$.

On the Competitive Ratio of Online Sampling Auctions

Elias Koutsoupias[1,⋆] and George Pierrakos[2]

[1] University of Athens
elias@di.uoa.gr
[2] UC Berkeley
georgios@cs.berkeley.edu

Abstract. We study online profit-maximizing auctions for digital goods with adversarial bid selection and uniformly random arrivals. Our goal is to design auctions that are constant competitive with $\mathcal{F}^{(2)}$; in this sense our model lies at the intersection of prior-free mechanism design and secretary problems. We first give a generic reduction that transforms any offline auction to an online one, with only a loss of a factor of 2 in the competitive ratio; we then present some natural auctions, both randomized and deterministic, and study their competitive ratio; our analysis reveals some interesting connections of one of these auctions with RSOP, which we further investigate in our final section.

1 Introduction

The design of mechanisms that maximize the auctioneer's profit is a well-studied question in mechanism design. Most of the relevant literature assumes a prior on the distribution of bidders' values and aims at maximizing the expected profit [16]; the question of designing a profitable auction with no assumptions about the bids' distribution has only recently been addressed during the past decade. In *prior-free mechanism design* [10] we assume that bids are picked by an adversary and we want to design auctions that are profitable for any such input bid sequence. To analyze such auctions, prior-free mechanism design adopts the model of competitive analysis and compares the profit of every auction to some well-behaved benchmark.

Most of the work in prior-free mechanism design assumes that the bids are known in advance [10,8,13,14]. Since almost all auctions today are happening online it makes sense to consider the online setting, where bidders arrive one at a time with a random order. In this setting, the design of a profitable, truthful auction reduces to making the "right" offer to every arriving bidder, using bids of previous bidders as the only information. We call such auctions *Online Sampling Auctions*.

This model bears a lot of similarities with the *secretary model*: the adversary picks the values of the elements, which are then presented in (uniformly) random

⋆ Supported in part by IST-2008-215270 (FRONTS).

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 327–338, 2010.

order, and we are called to design an algorithm that maximizes the probability of picking the largest element. There is an extensive literature about online auctions and generalized secretary problems (for a survey see [3]). The online auctions studied there are social-welfare maximizing auctions, and the overall focus is on the competitive analysis. Given an online algorithm one can turn it into a *truthful* mechanism very easily (at least when people cannot misreport their arrival times) by simply charging every bidder its threshold value; this of course makes the profit of such auctions very hard to analyze. Our approach is the opposite one: in order to design online profit-maximizing auctions, we start with the truthful offline setting of prior-free mechanism design, and turn it into an online setting. This way we ensure our auctions are both truthful and constant-competitive with the profit-benchmark.

The work closer in spirit to ours is [11]. This paper studies limited-supply online auctions, where an auctioneer has k items to sell and bidders arrive and depart dynamically; the analysis assumes worst-case input bids and random arrivals and the main result is an online auction that is constant-competitive for both efficiency and revenue. The profit-benchmark considered for $k > 1$ items, is essentially the same as the one here, namely the optimal single price sale profit that sells at least two items, $\mathcal{F}^{(2)}$. The authors present an auction that acts in two phases, very much in the spirit of secretary algorithms, that is 6338-competitive with respect to this benchmark. Our auctions achieve much better competitive ratios (below 10), and are arguably simpler to analyze; however in our model we do not address the issue of possible arrival times misreports.

Online auctions for digital goods have also been studied before in [5,7,6,4]. Their model is different from ours in that they do not assume random arrivals. Most of the algorithms presented in these papers are based on techniques from machine learning, and their performance depends on h, the ratio of the highest to the lowest bid. Our auctions are arguably more natural, and in most cases achieve better competitive ratios; however in our model auctions heavily rely on learning the actual values of past bids, and not just whether a bidder accepted or rejected the offer (as opposed to some of the auctions in [6]).

Finally, in an earlier work, Lavi and Nisan study worst case social-efficiency and profitability of online auctions for a different setting (not digital goods), taking the off-line Vickrey auction as a benchmark [15].

2 Our Model

We are going to study auctions of digital goods, where bidders arrive online. Formally we have n bidders with valuations $v_1, \ldots, v_n$ (where we assume $v_1 \geq \ldots \geq v_n$) and n identical items for sale. Bidders arrive with a random order, specified by the function $\pi : [n] \to [n]$, which is a permutation on $[n] = \{1, \ldots, n\}$; we assume uniform distribution over all different permutations of the n bids and adversarial (worst-case) choice of the values of the bids. In that sense our model is similar to the secretary model.

As each bidder arrives, we make her a take-it-or-leave-it offer for a copy of the item, for some price p. We want to make the offer before the bidder declares her bid (or equivalently we do not want our offer to depend on her declared bid) so that our auction is *truthful* (i.e. it is in the bidder's best interest to bid her true value v_i); hence, from now on we shall use $b_1, \ldots, b_n$ to refer to both bids and actual values of the players. Formally we want to make the j-th bidder b_{π_j}, an offer $p_j = p(b_{\pi_1}, \ldots, b_{\pi_{j-1}})$; the bidder will accept the offer if $b_{\pi_j} \geq p_j$ and will pay p_j.

Our goal is to maximize the expected profit of our auction, defined as $\mathbb{E}\left[\sum_{i=1}^{n} p_j \cdot \mathbb{I}(b_{\pi_j} \geq p_j)\right]$. We are going to consider both deterministic and randomized pricing rules $p(b_{\pi_1}, \ldots, b_{\pi_{j-1}})$; therefore the expectation is over all possible orderings of the input bids and –in the case of random pricings– over the randomization in our mechanism.

We are going to use the competitive framework proposed in [10] and compare the expected profit of our auctions to the profit of the best single price auction that sells at least two items, namely $\mathcal{F}^{(2)}(b_1, \ldots, b_n) = \max_{i \geq 2} i \cdot b_i$.[1] We say that an online auction is ρ-competitive if its expected profit is at least $\mathcal{F}^{(2)}/\rho$. Our goal is to design constant-competitive auctions (i.e. auctions where ρ is a constant).

3 Online Sampling Auctions

3.1 Randomized Competitive Online Sampling Auctions

Our first result establishes the existence of constant-competitive online sampling auctions. In fact we show the stronger result that any truthful offline auction gives rise to a truthful online sampling auction, with competitive ratio at most twice as large.

We start by noticing that any truthful (offline) auction for digital goods has the following format: every bidder i is given a take-it-or-leave-it offer p_i which is a function of the bids of the other players $f(b_{-i})$; if the bidder accepts she pays p_i otherwise nothing (this follows from Myerson's theorem [16]). Then we notice that every such truthful offline auction gives rise to an online auction if we simply set the price offered to the j-th arriving bidder to be $p_j = f(b_{\pi_1}, \ldots, b_{\pi_{j-1}})$, for the same function f; intuitively this means that we run the offline auction on the whole set of revealed bids, but actually charge only the bidder that has just arrived. Because we restrict our attention to truthful offline auctions, we know that the price offered to p_j will not depend on b_j and so we can offer the j-th bidder a price before she even reveals her bid. Our theorem now says that the resulting online auction has at most twice the competitive ratio of the offline auction.

Theorem 1. *If we turn an offline auction with competitive ratio ρ into an online auction, the competitive ratio of the online auction is at most 2ρ. More precisely, if b_k is the price of the optimal auction, then the competitive ratio of the online auction is at most $\rho \cdot k/(k-1)$.*[2]

[1] Notice however that in general our auctions will not be single-price auctions.

[2] We note that this looks very much like the $\frac{1}{4} \cdot \frac{k-1}{k}$ approximation ratio of [8], in a different model of course.

Proof. Consider the first t bids of the online auction. The online auction runs the offline auction on them. The expected profit of the offline auction from the whole set of bids would be at least $\frac{1}{\rho}\mathcal{F}^{(2)}(b_{\pi_1},\ldots,b_{\pi_t})$; by the random-order assumption about the input, the expected profit from every bid is equal and, in particular, the expected gain from b_{π_t} is at least:

$$\frac{1}{t}\frac{1}{\rho}\mathcal{F}^{(2)}(b_{\pi_1},\ldots,b_{\pi_t})$$

With probability $\binom{t}{m}\binom{n-t}{k-m}/\binom{n}{k}$ the first t bids have **exactly** m of the highest k bids which contribute to the optimum. Also, for $m \geq 2$, $\mathcal{F}^{(2)}(b_{\pi_1},\ldots,b_{\pi_t}) \geq mb_k$.[3] So, it follows that when $m \geq 2$, with the above probability the expected gain of the online auction from b_{π_t} is at least:

$$\frac{1}{t}\frac{1}{\rho}mb_k$$

So, the expected profit of the online auction is at least:

$$\begin{aligned}
&\sum_{t=2}^{n}\sum_{m=2}^{\min\{t,k\}}\frac{\binom{t}{m}\binom{n-t}{k-m}}{\binom{n}{k}}\frac{1}{t}\frac{1}{\rho}mb_k\\
&=\frac{1}{\rho}b_k\binom{n}{k}^{-1}\sum_{t=2}^{n}\sum_{m=2}^{k}\binom{t-1}{m-1}\binom{n-t}{k-m}\\
&=\frac{1}{\rho}b_k\binom{n}{k}^{-1}\sum_{t=1}^{n-1}\sum_{m=1}^{k-1}\binom{t}{m}\binom{n-1-t}{k-1-m}\\
&=\frac{1}{\rho}b_k\binom{n}{k}^{-1}\sum_{t=1}^{n-1}\left(\binom{n-1}{k-1}-\binom{n-1-t}{k-1}\right)\\
&=\frac{1}{\rho}b_k\binom{n}{k}^{-1}\left((n-1)\binom{n-1}{k-1}-\sum_{j=k-1}^{n-2}\binom{j}{k-1}\right)\\
&=\frac{1}{\rho}b_k\binom{n}{k}^{-1}\left((n-1)\binom{n-1}{k-1}-\binom{n-1}{k}\right)\\
&=\frac{k-1}{\rho}b_k,
\end{aligned}$$

where in the third equality we used the Chu-Vandermonde identity and in the second-to-last equality we used the identity $\sum_{j=k}^{n}\binom{j}{k}=\binom{n+1}{k+1}$; the Theorem now follows. □

[3] Notice that when $m=1$, there is no decent lower bound for $\mathcal{F}^{(2)}$; this is the reason that the online auction has larger competitive ratio than the offline auction.

We can now state our main Theorem:

Theorem 2. *The competitive ratio of Online Sampling Auctions is between 4 and 6.48.*

Proof. The upper bound is given by the online version of the (offline) auction presented in [12] which achieves a competitive ratio of 3.24.

For the lower bound, consider the case where the input chosen by the adversary consists of two bids: if the price offered to the second bidder is strictly greater than the first bid the adversary will pick two identical bids and the online auction will have zero profit. If the auction's offer to the second bidder is less than or equal to the first bid the adversary will pick as input the bids $h+\epsilon, h$ (for sufficiently large ϵ), in which case the optimal profit is $2h$ but the auction has expected profit at most $h/2$. □

At this point we note that the above Theorem greatly improves over the previously known bounds due to [11]. A natural question to ask now is whether we can bridge the gap between the lower and the upper bound. To this end we first study the competitive ratio that can be achieved by the online version of the *Sampling Cost Sharing* auction (SCS); this auction partitions bidders uniformly into two parts and extracts the optimal single price sale profit of each side from the other (if possible, otherwise it extracts no profit) [10]. We have the following:

Corollary 1. *The competitive ratio of the online version of SCS is at most 8. In the special case in which the optimal single-price auction for the whole set of bids sells the item to at least 5 buyers, the competitive ratio is at most 4.*

Proof. Using Theorem 1 and the bound on the competitive ratio of SCS proved in [10] we get that the online version of SCS will have competitive ratio at most $\frac{k}{k-1}\left(\frac{1}{2}-\binom{k-1}{\lfloor k/2\rfloor}2^{-k}\right)^{-1}$, which is less than 4 for $k \geq 5$. □

Notice that the worst-case inputs for this auction are when the optimal single price b_k is large, i.e. k is small. In the following section we show that this is not the case for all auctions.

3.2 A Deterministic Online Sampling Auction: BPSF$_r$

The two online auctions considered in the previous section are randomized, like their offline counterparts. In this section we shift our focus on deterministic online sampling auctions. For the offline setting the following theorem from [10] wipes out all hope for such an auction.

Theorem 3 ([10]). *We say an auction is* symmetric *if its outcome is independent of the order of the bids. It then holds that no symmetric, deterministic, truthful auction is constant-competitive against $\mathcal{F}^{(2)}$.*

There exist asymmetric, deterministic auctions with constant competitive ratio, but most of them result from derandomization of randomized ones and are unnatural [1]. In the online setting where order matters anyway, we can hope to design a constant competitive and deterministic (truthful) auction, that is also natural.

To this end we define the Best-Price-So-Far auction: BPSF_r is the (family of) auction(s) which offer as price the bid among the highest r of the previous bids which maximizes the single price sale profit of past requests. We are going to focus our attention on two representatives of this family, BPSF_1 and BPSF_∞, henceforth denoted by BPSF. BPSF_1 is an interesting auction which offers as price the maximum revealed bid. BPSF is an auction that offers the j-th bidder the price $p_j = p(b_{\pi_1}, \ldots, b_{\pi_{j-1}}) = \arg\max_{i \le j-1} i \cdot b_{\pi_i}$.

Theorem 4. *The expected profit of $BPSF_1$ is exactly $\sum_{i=2}^{n} \frac{1}{i} b_i$. Furthermore, if b_k is the price of the optimal auction, then the competitive ratio of $BPSF_1$ is $\frac{k}{H_k - 1}$ where $H_k = 1 + 1/2 + \cdots + 1/k$ is the k-th harmonic number, and this is tight.*

Proof. Notice that b_j is going to be offered as price exactly when b_j appears before $b_1, \ldots, b_{j-1}$. Every such bid is accepted if there is a higher bid after b_j appears. Thus b_j is going to be accepted at some point when $j \ge 2$. The probability that b_j appears before $b_1, \ldots, b_{j-1}$ is exactly $1/j$. It follows that the expected profit of BPSF_1 is $\sum_{i=2}^{n} \frac{1}{i} b_i$.

For the second fact, simply observe that when b_k is the price of the optimal auction, the online profit is at least $\sum_{i=2}^{k} \frac{1}{i} b_i \ge \sum_{i=2}^{k} \frac{1}{i} b_k = (H_k - 1) b_k$. Since the optimal profit is $k b_k$, the claim follows.

Finally, it is easy to verify that the above bound is tight for any set of n bids with $b_1 > \ldots > b_n$ and $b_n \ge b_1 - \epsilon$, for sufficiently small ϵ. □

Corollary 2. *Let b_k, be the optimal single price for the whole set of bids. If $k \le 5$ then the competitive ratio of $BPSF_1$ is at most 4.*

Corollaries 1 and 2 show that if we knew in advance the number of buyers of the optimal single-price auction, we could achieve competitive ratio 4 against $\mathcal{F}^{(2)}$, thus matching the corresponding lower bound.

We saw that BPSF_1 is not constant competitive; it is also easy to see that the competitive ratio of BPSF_r can only decrease for larger r; the natural question to ask is if it will ever be constant. To this end we examine BPSF, which is arguably a very natural online auction: BPSF is the online version of the *Deterministic Optimal Price* (DOP) auction that offers bidder j the optimal single price of the other bidders, namely $p_j = p(b_{-j}) = \arg\max_{i \ne j} i \cdot b_i$. DOP is known not to be competitive [10]; we conjecture that BPSF on the contrary is constant-competitive:

Conjecture 1. The competitive ratio of BPSF is 4.

The competitive ratio of 4 is the same as the conjectured competitive ratio of RSOP. This is not a coincidence; in the next section we take a closer look into the similarities of RSOP and BPSF.

4 On the Competitive Ratio of BPSF and RSOP

One of the simplest competitive auctions, and arguably the most studied [10,9,2] is the *Random Sampling Optimal Price* auction (RSOP). In RSOP the bidders are uniformly partitioned into two parts, and the optimal single price of each part (i.e. $\arg\max i \cdot b_i$) is offered to the bidders of the other part. RSOP is conjectured to be 4-competitive; to date the best upper bound is 4.68 [2].

In what follows we analyze the competitive ratio of BPSF and RSOP in more detail. We see that the analyses of the two auctions bear a lot of similarities and we suggest a possible approach for both auctions. We believe that our approach may be a promising direction for proving both Conjecture 1 and that RSOP is 4-competitive as well.[4]

We first introduce some notation. Let $B = \{b_1, \ldots, b_n\}$ be the set of all bids and $B_2 = \{b_2, \ldots, b_n\}$. Given a specific partition of bids $b_1, \ldots, b_n$ in two parts, we use $(b_{j_1}, \ldots, b_{j_k})$ to denote the side of the partition that does not contain the highest bid b_1, i.e. by writing $(b_{j_1}, \ldots, b_{j_k})$ we assume implicitly that $j_1 \geq 2$ and also $b_{j_1} \geq \ldots \geq b_{j_k}$. Finally let

$$y(b_{j_1}, \ldots, b_{j_k}) = \max\{b_{j_1}, 2b_{j_2}, \ldots, kb_{j_k}\},$$

the optimal single price sale profit from $(b_{j_1}, \ldots, b_{j_k})$ and let $z(b_{j_1}, \ldots, b_{j_k})$ be the profit from offering the optimal price of $(b_{j_1}, \ldots, b_{j_k})$ to the other side.

We next show how to write the expected profits of RSOP and BPSF in terms of z and y.

For RSOP it is straightforward; just notice that the adversary can always pick a large enough b_1 so that the profit from the side of the partition not containing b_1 will always be 0 [9]. We then have:

$$RSOP = \sum_{S \subseteq B_2} z(S) 2^{-n+1}$$

For BPSF the expression is less straightforward. We have:

Lemma 1. *The expected profit of BPSF is* $\sum_{S \subseteq B_2} z(S) \binom{n-1}{|S|}^{-1} n^{-1}$.

Proof. Let $Profit(S, b_i)$ denote the profit we get if we offer the optimal single price for S to bid $b_i \notin S$. In what follows, the expectation operator is used to denote expectation over the non-uniform distribution on the collection of sets $S \subseteq B_2$ induced by the random arrival order of the bids.[5] We have:

$$\begin{aligned} BPSF &= \sum_{b_i} \mathbb{E}_{S \subseteq B_2, b_i \notin S} \left[Profit(S, b_i) \right] \\ &= \frac{1}{n} \cdot \sum_{b_i} \sum_{k=0}^{n-1} \mathbb{E}_{S \subseteq B_2, |S|=k, b_i \notin S} \left[Profit(S, b_i) \right] \\ &= \frac{1}{n} \cdot \sum_{b_i} \sum_{k=0}^{n-1} \sum_{S \subseteq B_2, |S|=k, b_i \notin S} \frac{Profit(S, b_i)}{\binom{n-1}{k}} \end{aligned}$$

[4] The very technical approach of [2], although coming very close, does not seem to be able to prove the Conjecture.

[5] As opposed to expectation taken over a uniformly random choice of a set $S \subseteq B_2$.

$$= \frac{1}{n} \cdot \sum_{b_i} \sum_{S \subseteq B_2, b_i \notin S} \frac{Profit(S, b_i)}{\binom{n-1}{|S|}}$$
$$= \frac{1}{n} \cdot \sum_{S \subseteq B_2} \frac{1}{\binom{n-1}{|S|}} z(S)$$

where in the second equality we used the fact that b_i will be in the $k+1$ position with probability $1/n$, in the third equality we used the fact that all orderings have the same probability and in the last equality we used the fact that $z(S) = \sum_{b_i \notin S} Profit(S, b_i)$. □

The next lemma establishes an interesting relation between the y values and the optimal single price sale profit $\mathcal{F}^{(2)}$.

Lemma 2. *For any $i \in [n], i \geq 2$ it holds that:*

$$\sum_{S \subseteq B_2 :\ b_2 \in S} y(S) \geq 2^{n-3}\ ib_i$$

Proof. Let b_i be the optimum single price for the whole set of bids, i.e. $\mathcal{F}^{(2)} = ib_i$ (although our result holds for any bid b_i).

We will introduce a mapping between the set of sequences $X = \{S \subseteq B_2 |\ b_2 \in S\ \&\ b_i \notin S\}$ and the set $Y = \{S \subseteq B_2 |\ b_2 \notin S\ \&\ b_i \in S\}$. Given a sequence of bids $S \in X$ let $t = \max\{j : j < i, b_j \in S\}$.[6] We then define the following mapping for each bid $b_j \in S$:

$$f(b_j) = \begin{cases} b_{j+i-t} & : \text{if } j < i \\ b_j & : \text{if } j > i \end{cases}$$

It is easy to see that the mapping $g : X \longrightarrow Y$ defined as $g(b_{j_1}, \ldots, b_{j_k}) = (f(b_{j_1}), \ldots, f(b_{j_k}))$ is in fact a bijection. Also note that $b_1 \geq \ldots \geq b_n$ implies that $y(S) \geq y(g(S))$. Hence we have:

$$\begin{aligned}
\sum_{S \subseteq B_2 :\ b_2 \in S} y(S) &= \sum_{S \subseteq B_2 :\ b_2 \in S, b_i \in S} y(S) + \sum_{S \subseteq B_2 :\ b_2 \in S, b_i \notin S} y(S) \\
&\geq \sum_{S \subseteq B_2 :\ b_2 \in S, b_i \in S} y(S) + \sum_{S \subseteq B_2 :\ b_2 \in S, b_i \notin S} y(g(S)) \\
&= \sum_{S \subseteq B_2 :\ b_2 \in S, b_i \in S} y(S) + \sum_{S \subseteq B_2 :\ b_2 \notin S, b_i \in S} y(S) \\
&= \sum_{S \subseteq B_2 :\ b_i \in S} y(S) \\
&= 2^{n-i} \sum_{j=0}^{i-2} \binom{i-2}{j} (j+1) \cdot b_i
\end{aligned}$$

[6] Notice that this is a non-empty set, as it contains $j = 2$.

For the last equality consider all possible positions of b_i in S. There can be j bids larger than b_i where j ranges from 0 to $i-2$; there are $\binom{i-2}{j}$ ways to pick these bids and 2^{n-i} ways to pick the bids that are smaller than b_i and for this specific position the coefficient of b_i is $(j+1)$.[7]

A straightforward calculation shows that $\sum_{j=0}^{i-2}\binom{i-2}{j}(j+1) = i2^{i-3}$, and the claim follows. □

It is now easy to see that the following claim implies that RSOP is indeed 4-competitive.

Conjecture 2

$$\sum_{S\subseteq B_2} z(S) \geq \sum_{S\subseteq B_2:\ b_2\in S} y(S)$$

The corresponding claim for BPSF is:

Conjecture 3

$$\sum_{S\subseteq B_2} z(S)\binom{n-1}{|S|}^{-1} n^{-1} \geq \sum_{S\subseteq B_2:\ b_2\in S} y(S)2^{-n+1}$$

We believe that Conjectures 2 and 3 both hold and that RSOP and BPSF are both 4-competitive. We attempted to prove the Conjectures using a number of relations between the z and y values; analytical and numerical simulations show that one can sum up individual relations between $z(S)$ and $y(S)$ for any set S of bids, like the ones presented in Appendix A, in order to get the result.

5 Conclusion

There is a number of open questions from this work: the obvious ones are to prove that BPSF is indeed 4-competitive and see what this proof implies for the competitive ratio of RSOP. Proving that BPSF is constant competitive for some other constant is also interesting, and probably much easier. Finally, it would be interesting to see if there is a natural online sampling auction with competitive ratio at most 4, for all values k of the optimal single price b_k.

Acknowledgments. We are grateful to an anonymous reviewer for a pointer to missing literature and for comments that helped us improve the presentation.

References

1. Aggarwal, G., Fiat, A., Goldberg, A.V., Hartline, J.D., Immorlica, N., Sudan, M.: Derandomization of auctions. In: STOC, pp. 619–625 (2005)
2. Alaei, S., Malekian, A., Srinivasan, A.: On random sampling auctions for digital goods. In: ACM Conference on Electronic Commerce, pp. 187–196 (2009)

[7] If b_i is an arbitrary bid with $i \geq 2$, rather than the optimum single price as stated in the beginning of the proof, then the last equality should be replaced with an inequality, and the claim still goes through.

3. Babaioff, M., Immorlica, N., Kempe, D., Kleinberg, R.: Online auctions and generalized secretary problems. SIGecom Exchanges 7(2) (2008)
4. Balcan, M.-F., Blum, A., Hartline, J.D., Mansour, Y.: Reducing mechanism design to algorithm design via machine learning. J. Comput. Syst. Sci. 74(8), 1245–1270 (2008)
5. Bar-Yossef, Z., Hildrum, K., Wu, F.: Incentive-compatible online auctions for digital goods. In: SODA, pp. 964–970 (2002)
6. Blum, A., Hartline, J.D.: Near-optimal online auctions. In: SODA, pp. 1156–1163 (2005)
7. Blum, A., Kumar, V., Rudra, A., Wu, F.: Online learning in online auctions. Theor. Comput. Sci. 324(2-3), 137–146 (2004)
8. Dhangwatnotai, P., Roughgarden, T., Yan, Q.: Revenue maximization with a single sample. In: ACM Conference on Electronic Commerce, pp. 129–138 (2010)
9. Feige, U., Flaxman, A.D., Hartline, J.D., Kleinberg, R.D.: On the competitive ratio of the random sampling auction. In: Deng, X., Ye, Y. (eds.) WINE 2005. LNCS, vol. 3828, pp. 878–886. Springer, Heidelberg (2005)
10. Goldberg, A.V., Hartline, J.D., Karlin, A.R., Wright, A., Saks, M.: Competitive auctions. In: Games and Economic Behavior, pp. 72–81 (2002)
11. Hajiaghayi, M.T., Kleinberg, R.D., Parkes, D.C.: Adaptive limited-supply online auctions. In: ACM Conference on Electronic Commerce, pp. 71–80 (2004)
12. Hartline, J.D., McGrew, R.: From optimal limited to unlimited supply auctions. In: ACM Conference on Electronic Commerce, pp. 175–182 (2005)
13. Hartline, J.D., Roughgarden, T.: Optimal mechanism design and money burning. In: STOC, pp. 75–84 (2008)
14. Hartline, J.D., Roughgarden, T.: Simple versus optimal mechanisms. In: ACM Conference on Electronic Commerce, pp. 225–234 (2009)
15. Lavi, R., Nisan, N.: Competitive analysis of incentive compatible on-line auctions. In: Proc. 2nd ACM Conf. on Electronic Commerce (EC 2000), pp. 233–241 (2000)
16. Myerson, R.B.: Optimal auction design. Discussion Papers 362, Northwestern University, Center for Mathematical Studies in Economics and Management Science (December 1978)

A Relation of z and y Values

In order to prove Conjectures 2 and 3 we need a strong lemma that captures the relation of the z and y values. In the appendix we list three such lemmata, of increasing strength. Numerical and analytical simulations in MAPLE12 have verified that Lemma 5 is enough to prove Conjecture 2 (by just summing up for all subsets $S \subseteq B_2$) for up to $n = 20$ bids.

Lemma 3

$$z(b_{j_1}, \ldots, b_{j_k}) \geq \min_{t=1,\ldots,k} \left(\frac{j_t - t}{t} \right) \cdot y(b_{j_1}, \ldots, b_{j_k})$$

Proof. Let b_{j_t} be the optimal price for $(b_{j_1}, \ldots, b_{j_k})$, i.e.

$$t \cdot b_{j_t} = \max\{b_{j_1}, 2b_{j_2}, \ldots, kb_{j_k})$$

Then

$$
\begin{aligned}
z(b_{j_1}, \ldots, b_{j_k}) &= (j_t - t) b_{j_t} \\
&= \frac{j_t - t}{t} \cdot t b_{j_t} \\
&= \frac{j_t - t}{t} \cdot y(b_{j_1}, \ldots, b_{j_k}) \\
&\geq \min_{t=1,\ldots,k} \left(\frac{j_t - t}{t} \right) \cdot y(b_{j_1}, \ldots, b_{j_k})
\end{aligned}
$$

□

Notice that the term $\left(\frac{j_t - t}{t}\right)$ is the same quantity as the one minimized in the random walk of [9] and it also appears in the analysis of [2].

The following relation is stronger, in that by summing up all for all S we immediately get $\sum_{S \subseteq B_2:\ b_2 \in S} y(S)$ and some more terms, whose sum we then need to show is positive.

Lemma 4

$$
z(b_{j_1}, \ldots, b_{j_k}) \geq y(b_{j_1}, \ldots, b_{j_k}) - \max\left(0, \max_{t=2,\ldots,k} \left\{ \frac{2t - j_t}{t-1} \right\}\right) \cdot y(b_{j_2}, \ldots, b_{j_k})
$$

Proof. Let b_{j_t} be the optimal price for $(b_{j_1}, \ldots, b_{j_k})$, i.e.

$$
t \cdot b_{j_t} = \max\{b_{j_1}, 2b_{j_2}, \ldots, k b_{j_k})
$$

Then

$$
\begin{aligned}
z(b_{j_1}, \ldots, b_{j_k}) &= (j_t - t) b_{j_t} \\
&= t b_{j_t} - (2t - j_t) b_{j_t} \\
&= t b_{j_t} - \frac{2t - j_t}{t - 1} (t-1) b_{j_t} \\
&= y(b_{j_1}, \ldots, b_{j_k}) - \frac{2t - j_t}{t-1} (t-1) b_{j_t} \\
&\geq y(b_{j_1}, \ldots, b_{j_k}) - \max\left(0, \max_{t=2,\ldots,k} \left\{ \frac{2t - j_t}{t-1} \right\}\right) \cdot y(b_{j_2}, \ldots, b_{j_k})
\end{aligned}
$$

where we need $\frac{2t - j_t}{t-1}$ to be positive for the inequalities to work correctly, which is why we take $\max\left(0, \max_{t=2,\ldots,k} \left\{ \frac{2t - j_t}{t-1} \right\}\right)$. □

In order to optimally handle the negative terms showing up in the RHS of Lemma 4 we used the following, more elaborate bound.

Lemma 5. *Let* $(b_{j_1}, \ldots, b_{j_k})$ *be a set of at least 2 bids and* λ *a real in* $[0, j_1 - 1]$*. We can bound* $z(b_{j_1}, \ldots, b_{j_k})$ *with*

$$
z(b_{j_1}, \ldots, b_{j_k}) \geq \lambda y(b_{j_1}, \ldots, b_{j_k}) + \mu y(b_{j_2}, \ldots, b_{j_k}), \tag{1}
$$

where μ is defined by

$$\mu = \begin{cases} \frac{k}{k-1} \min_{t=1,\ldots,k}\{\frac{j_t - t - \lambda t}{t}\}\}, & \textit{when } \min_{t=2,\ldots,k}\{j_t - t - \lambda t\} \geq 0 \\ \min_{t=2,\ldots,k}\{\frac{j_t - t - \lambda t}{t-1}\}, & \textit{otherwise} \end{cases}$$

Proof. Assume that

$$y(b_{j_1}, \ldots, b_{j_k}) = t \cdot b_{j_t}$$
$$y(b_{j_2}, \ldots, b_{j_k}) = (s-1) \cdot b_{j_s}$$

From these we get that $tb_{j_t} \geq sb_{j_s}$ and $(s-1)b_{j_s} \geq (t-1)b_{j_t}$. Notice that the latter holds even when $t = 1$.

Assume that $\min_{r=2,\ldots,k}\{j_r - r - \lambda r\} \geq 0$. We will show that inequality (1) is satisfied for $\mu = \frac{k}{k-1} \min_{r=1,\ldots,k}\{\frac{j_r - r - \lambda r}{r}\}\}$. We will use the fact that μ is nonnegative and the inequality $tb_{j_t} \geq sb_{j_s}$. Indeed we have,

$$\begin{aligned} \lambda y(b_{j_1}, \ldots, b_{j_k}) + \mu y(b_{j_2}, \ldots, b_{j_k}) &= \lambda t b_{j_t} + \mu(s-1)b_{j_s} \\ &\leq \lambda t b_{j_t} + \mu(s-1)\frac{t}{s} b_{j_t} \\ &\leq \lambda t b_{j_t} + \mu(k-1)\frac{t}{k} b_{j_t} \\ &\leq \lambda t b_{j_t} + \frac{k}{k-1} \frac{j_t - t - \lambda t}{t} (k-1) \frac{t}{k} b_{j_t} \\ &= (j_t - t) b_{j_t} \\ &= z(b_{j_1}, \ldots, b_{j_k}) \end{aligned}$$

Now we consider the case of $\min_{r=2,\ldots,k}\{j_r - r - \lambda r\} < 0$. Assume first that $t \geq 2$. We will now show that inequality (1) is satisfied for $\mu = \min_{r=2,\ldots,k}\{\frac{j_r - r - \lambda r}{r-1}\}\}$. We will use the fact that μ is now negative and the inequality $(t-1)b_{j_t} \leq (s-1)b_{j_s}$. Indeed we have,

$$\begin{aligned} \lambda y(b_{j_1}, \ldots, b_{j_k}) + \mu y(b_{j_2}, \ldots, b_{j_k}) &= \lambda t b_{j_t} + \mu(s-1)b_{j_s} \\ &\leq \lambda t b_{j_t} + \mu(t-1) b_{j_t} \\ &\leq \lambda t b_{j_t} + \frac{j_t - t - \lambda t}{t-1} (t-1) b_{j_t} \\ &= (j_t - t) b_{j_t} \\ &= z(b_{j_1}, \ldots, b_{j_k}) \end{aligned}$$

The case $t = 1$ must be handled separately because $t - 1$ appears in the denominator in the above. When $t = 1$ we have that

$$\begin{aligned} z(b_{j_1}, \ldots, b_{j_k}) &= (j_1 - 1)b_{j_1} \\ &= \lambda b_{j_1} + (j_1 - 1 - \lambda)b_{j_1} \\ &\geq \lambda y(b_{j_1}, \ldots, b_{j_k}) \\ &\geq \lambda y(b_{j_1}, \ldots, b_{j_k}) + \mu y(b_{j_2}, \ldots, b_{j_k}). \end{aligned}$$

Notice that we used the fact that $\lambda \leq j_1 - 1$ and that $\mu \leq 0$. □

Near-Strong Equilibria in Network Creation Games

Ola Rozenfeld[1] and Moshe Tennenholtz[2]

[1] Technion–Israel Institute of Technology
ola.rozenfeld@gmail.com
[2] Microsoft R&D Labs, Israel
moshet@ie.technion.ac.il

Abstract. We introduce a new solution concept for games, near-strong equilibrium, a variation of strong equilibrium. Previous work has shown the existence of 2-strong pure strategy equilibrium for network creation games with $1 < \alpha < 2$ and that k-strong equilibrium for $k \geq 3$ does not exist. In this paper we show that 3-near-strong equilibrium exists, and provide tight bounds on existence of k-near-strong equilibria for $k \geq 4$. Then we repeat our analysis for correlated mixed strategies, where we show that, surprisingly, 3-correlated-strong equilibrium exists, and also show bounds for existence of correlated k-strong equilibria. Moreover, the equilibrium profile can be arbitrarily close to the social optimum. For both pure and correlated settings, we show examples where no equilibrium exists. On the conceptual level, our work contributes to the recent literature of extensions of strong equilibrium, while providing positive results for stability against group deviations in one of the basic settings discussed in the algorithmic game theory literature.

1 Introduction

The Nash equilibrium is a solution concept most commonly used by game theorists. A known drawback of this concept is the assumption that agents do not cooperate in order to agree on a joint deviation. The strong equilibrium (SE), introduced in (Aumann, 1959), is an extension of NE that takes care of this problem: it is a strategy profile which is stable against joint deviations by coalitions of agents. Having an SE is highly desirable, but most games of interest do not possess such stability. Also, the concept might be too strong, in a sense: it does not take into account that deviations themselves might be unstable against sub-deviations. Indeed, if we begin by supposing that an agent will deviate from a proposed profile if a better choice is available, the same should hold if the profile is proposed to him by a deviating coalition of other agents. (Bernheim *et al.*, 1987) suggested the concept of *coalition-proof Nash equilibrium* (CPNE), which captures this idea: a profile is CPNE if it has no *self-enforcing* profitable group deviations, where self-enforcing means: has no self-enforcing profitable sub-deviations (recursive definition). A unilateral deviation is always self-enforcing. Although CPNE captures the above logic perfectly, it goes too far

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 339–353, 2010.

in a sense: people in real life are extremely unlikely to employ considerations of such complexity. In a general game, it will be probably computationally impossible to determine if a given group deviation is self-enforcing. Therefore, a simpler and stronger solution concept was suggested by (Milgrom and Roberts, 1994): to require self-enforcing group deviations to be stable only against *unilateral* sub-deviations[1].

We suggest to concentrate on the following (even stronger) solution concept, which is a variation of the above definition:

A profile of actions is a *near-strong equilibrium* (NSE) if for every beneficial joint deviation by a coalition of players, there exists a player in that coalition who, given that the rest of the coalition will stick to their deviation, would strictly prefer to betray them and return *to his original strategy*.

Our definition differs from (Milgrom and Roberts, 1994) only in that the betraying player's strategy is restricted to equal his original strategy. This difference is important, because it emphasizes the stability of the original profile from the agent-centric perspective: suppose a coalition K of players wishes to convince a member i of K to participate in a joint deviation. Their arguments are: all of them (the other players in K) agreed to play their corresponding new actions, and the new actions will result in a higher payoff for all $i \in K$, compared with the original profile. In a SE, that argumentation should suffice to convince each player $i \in K$ to join. However, player i might consider the following logic: either he trusts the rest of the coalition to behave according to the chosen deviation, or he does not. If he does not trust them to behave as agreed, there is certainly no incentive for him to deviate from the equilibrium; but if he does trust them, the equilibrium strategy still gives him a strictly higher payoff than the deviation strategy! So, in both cases, player i will actually be better off not joining the coalition than joining it. So, although NSE is formally weaker than strong equilibrium, it is conceptually similar: *an agent will prefer the original profile over all possible deviations, even when coalitions of agents can coordinate a joint deviation.*

The aim of this work is to explore how the concept of NSE can improve over SE: to show an interesting setting which does not possess SE, but in which NSE can be shown to exist. The setting we chose is the network creation game, introduced by (Fabrikant *et al.*, 2003) and since appearing in many works, including (Albers *et al.*, 2006; Andelman *et al.*, 2009). The basic version of the game is as follows: each player is associated with a vertex in a network, and the aim of all players is to be connected to all the other players with as small a distance as possible to each. However, each player controls his outgoing edges and must choose which edges to buy. Buying an edge has a fixed cost, α. In the simplest model, which we adopt, a player's cost is simply a sum of his distances to all other players in the resulting (undirected) graph, and the total price he paid for his chosen edges.

[1] Independently, the same concept was introduced in (Kaplan, 1992) as *semistrong equilibrium.*

When $\alpha \leq 1$, the game is not very interesting, since the socially optimal profile (a clique) is a strong equilibrium. Similarly, when $\alpha \geq 2$, the out-star (the strategy profile in which one node buys edges to all other nodes, while they buy nothing) is both socially optimal and an SE (Andelman *et al.*, 2009). However, when $1 < \alpha < 2$ even a 3-SE was shown not to exist for $n > 5$ (Andelman *et al.*, 2009); therefore, this is the case we concentrate upon in this paper.

We show:

1. The out-star is always a 3-NSE
2. An example where no 4-NSE exists
3. A bound of when the out-star is a k-NSE:
   (a) The out-star is a k-NSE if $\alpha \geq 2(1 - \frac{1}{k'})$, where $k' = 2\lfloor \frac{k}{2} \rfloor$
   (b) Otherwise, if $\alpha < 2(1 - \frac{1}{k'})$
      i. For $k \leq \lfloor \frac{n+1}{2} \rfloor$, the out-star is not a k-NSE
      ii. Otherwise, we show examples of n, α where the out-star still is a k-NSE, and other examples where it is not
4. Some empirical results for small n, e.g. the out-star is an NSE for $n \leq 6$

All the above analysis concerned only pure strategies. Next, we turn to the case when players can also use mixed strategies; since the solution concepts we deal with allow coordination by coalitions of players, it makes sense to allow the players to employ *correlated* mixed strategies. A natural extension of the strong equilibrium concept to the case of correlated mixed strategies is the correlated strong equilibrium (CSE), by (Rozenfeld and Tennenholtz, 2006).

Using correlated mixed strategies allows us to concentrate on symmetric strategy profiles – informally, profiles where all players are treated equally (and, in particular, incur the same costs). We concentrate on two intuitive symmetric strategy profiles: the ϵ-clique and the randomized out-star (which we abbreviate to "star"). For $0 < \epsilon << 1$ the ϵ-clique is a profile that can get arbitrarily close to a clique (with probability $1 - \epsilon$ the players form a fair clique), and the star is a an out-star where the root is chosen with uniform distribution.

We show:

1. The ϵ-clique is a k-CSE for $k < min\{n, 1 + \frac{\alpha}{\alpha - 1}\}$. In particular, this implies:
   – The ϵ-clique is a 3-CSE for $n \geq 4$.
   – The ϵ-clique is a 4-CSE for $\alpha < \frac{3}{2}, n \geq 5$.
2. We derive a tight bound on when the star is a k-CSE
   – In particular, it implies that the star is a 4-CSE for $\alpha \geq \frac{3}{2}, n \geq 10$
3. We present a sound algorithm for proving that no symmetric k-CSE exists for a given instance
   – In particular, using the algorithm we show that no symmetric 4-CSE exists for $n = 5$,$\alpha = \frac{3}{2}$
4. Monotonicity:
   (a) If the ϵ-clique is a k-CSE for n, k, α, then for all $\alpha' < \alpha$ it is still a k-CSE for n, α'
   (b) If the star is a k-CSE for n, k, α, then for all $\alpha' > \alpha$ it is still a k-CSE for n, α'

Recall that the clique is the social optimum in our setting ($1 < \alpha < 2$). Using only pure strategies, there was no hope to implement any profile with sufficiently many edges as even a NE. Surprisingly, when we allow correlated mixed strategies, a strong positive result emerges: we can implement a fair profile arbitrarily close to the social optimum as a k-CSE. Note also that the ϵ-clique gets more stable for smaller values of α, which is very good, since it is for small α that the clique yields a much lower social cost than the star. For big values of α, the social cost of the star approaches that of the clique; and in these cases, too, the closer to optimal the star gets, the more stable it becomes.

We can now extend our near-strong equilibrium concept in a similar way to define a correlated near-strong equilibrium (CNSE). Recall that the aim of this work is to compare the existence of NSE to that of SE. However, in the correlated setting it turns out that the concept of CNSE is just as strong as CSE: we show that a symmetric strategy profile is a CNSE if and only if it is a CSE.

Note: due to lack of space, some of the proofs in the following sections were omitted.

2 Model and Preliminaries

The network creation game was introduced in (Fabrikant *et al.*, 2003). A player is associated with a vertex in a network, who wishes to connect to other players. The set of players is $V = \{1, ..., n\}$, and a strategy of a player is to select the subset of other players to whom he buys an edge: $S_v = 2^{V\setminus\{v\}}$. For a set of players $K \subseteq V$, let $S_K = \prod_{v\in K} S_v$, and let $S = S_V$. A strategy profile $s \in S$ induces a directed graph $G(s) = (V, E)$, where $E = \{(v, u)|u \in s_v\}$.

The cost that a player incurs consists of two parts: the price of the edges he bought (each edge has a fixed cost of α) and his distances to the other players in the resulting network. Formally, $c_v(s) = \alpha|s_v| + Dist(v)$, where $Dist(v) = \sum_{u\in V} \delta_s(v, u)$, and $\delta_s(v, u)$ is the length of the shortest path from v to u in the *undirected* graph induced by $G(s)$. So, the directions of the edges serve only to visualize which player is paying for them; network-wise, the network is treated as an undirected graph. A player has to be connected to all other players; otherwise, $c_v(s) = \infty$. *For the remainder of this work, we assume that* $1 < \alpha < 2$.

We recall that a profile of actions $s \in S$ is a *strong equilibrium* (SE) if for every coalition $K \subseteq V$ and every joint choice of actions $t_K \in S_K$ there exists $v \in K$ for whom $c_v(t_K, s_{-K}) \geq c_v(s)$ (Aumann, 1959). For $1 \leq k \leq n$ we say that $s \in S$ is a *k-strong equilibrium* (k-SE) if the above condition holds for all $K \subseteq V$ s.t. $|K| \leq k$.

A profile of actions $s \in S$ is a *near-strong equilibrium* (NSE) if for every coalition $K \subseteq V$ and every joint choice of actions $t_K \in S_K$ such that $\forall v \in K \;\; c_v(t_K, s_{-K}) < c_v(s)$, there exists $v \in K$ for whom $c_v(t_{K\setminus\{v\}}, s_{-K\cup\{v\}}) < c_v(t_K, s_{-K})$. For $1 \leq k \leq n$ we say that $s \in S$ is a *k-near-strong equilibrium* (k-NSE) if the above condition holds for all $K \subseteq V$ s.t. $|K| \leq k$.

For a set A, let $\Delta(A)$ denote the set of all probability distributions over A. For a correlated strategy profile $s \in \Delta(S)$, let $C_v(s)$ denote the expected cost of player v in s. We say that $s \in \Delta(S)$ is a *correlated strong equilibrium* (CSE) if for every coalition $K \subseteq V$ and a deviation $t_K \in \Delta(S_K)$ there exists $v \in K$ s.t. $C_v(t_K \times s_{[-K]}) \geq C_v(s)$ (Rozenfeld and Tennenholtz, 2006). Here, $s_{[-K]}$ means the marginal probability induced by s on $V \setminus K$. For $1 \leq k \leq n$ we say that $s \in S$ is a *correlated k-strong equilibrium* (k-CSE) if the above condition holds for all $K \subseteq V$ s.t. $|K| \leq k$. Note that this definition means that the deviating players should be able to gain in expectation, without knowing *anything* about the pure realization of s.

A justification of the above version of the definition of CSE is, perhaps, in order. The definition of a correlated Nash equilibrium (Aumann, 1974) presumed that before deciding on a deviation each player could observe his *signal* (his action in the pure realization of the correlated strategy). When defining a correlated strong equilibrium, the first instinct is to extend Aumann's definition of correlated Nash equilibrium (CNE) and require the profile to be stable against *ex-post* deviations by coalitions of players. However, there is a caveat in doing this. In Aumann's definition, it is vital that each player is informed only about *his own* signal. The underlying assumption of this model could be, for example, that a trusted authority rolls the dice, and sends to each player his selected strategy over a private channel. But how would this work for a coalition of players, who need to deviate jointly? Presumably, they would have to share their signals. But here is the problem: if they do share signals, then each one of them *by himself* possesses more information than he is allowed to by the CNE concept! We would have to consider situations such as this: a single agent cannot beneficially deviate knowing his own possible signals, but two agents can each beneficially deviate alone if each also gets to know the other's signal – even though the two of them together do not possess a joint deviation beneficial for both! No matter how we classify such cases, the justification is not at all intuitive. We are not saying that such definition is impossible or not interesting (on the contrary, we are working on it), but we have to be careful and visualize, in detail, the flow of information in the situation that we are trying to model.

Our definition, much like Aumann's, presumes the existence of a mediator – a trusted third party that can roll the dice to select a pure realization of a correlated strategy profile. In our model, the mediator does not output the signals at all – rather, it selects actions according to a preset correlated profile *on behalf* of the players who chose to use it (much like in (Rozenfeld and Tennenholtz, 2007), only without punishments). Alternatively, one could imagine some other means to enforce the players to follow through on the selected pure realization, once it is chosen (e.g. a contract (Kalai *et al.*, 2007)). In the following, we assume that such means are available to any coalition of players wishing to implement a joint correlated strategy.

A profile $s \in \Delta(S)$ is a *correlated near-strong equilibrium* (CNSE) if for every coalition $K \subseteq V$ and a deviation $t_K \in \Delta(S_K)$ s.t. $\forall v \in K \;\; C_v(t_K \times s_{[-K]}) < C_v(s)$ there exists $v \in K$ for whom $C_v(t_{[K \setminus \{v\}]} \times s_{[-K \cup \{v\}]}) < C_v(t_K \times s_{[-K]})$. For $1 \leq k \leq n$ we say that $s \in S$ is a *correlated k-near-strong equilibrium* (k-CNSE) if the above condition holds for all $K \subseteq V$ s.t. $|K| \leq k$.

Now we want to define a symmetric profile. Intuitively, a strategy profile is symmetric if the names of the players don't matter. Formally, we will need some notation. Let π be a permutation of V. For a set $K \subseteq V$, we let $\pi(K) = \{\pi(v)|v \in K\}$. For a strategy profile $s \in S$, we denote by $\pi(s)$ the following strategy profile: for all $v \in V$, $\pi(s)_{\pi(v)} = \pi(s_v)$. For a correlated strategy profile $s \in \Delta(S)$, we denote by $\pi(s)$ the following correlated strategy profile: for all $z \in S$ $\pi(s)(\pi(z)) = s(z)$. Let Π be the set of all possible permutations of V. For a correlated strategy profile $s \in \Delta(S)$, let $Sym(s) \in \Delta(S)$ be the following correlated strategy: first select $\pi \in \Pi$ with uniform probability, and then play $\pi(s)$. We say that a correlated strategy profile $s \in \Delta(S)$ is *symmetric* if $s = Sym(s)$.

3 Pure Strategies

In (Andelman *et al.*, 2009) it was shown that a 2-SE always exists, but even a 3-SE does not exist for $n > 5$. In particular, consider the following strategy profile $o*$ (the out-star):

Let one player (the root, denoted by r) purchase an edge to every other player, while all other players (which we call leaves) purchase no edges. In this profile, the cost of the root, $c_r(o*) = \alpha(n-1) + n - 1 = (n-1)(\alpha+1)$ (he pays for $n-1$ edges, and his distance to any other node is 1). The cost of every other player is $c_v(o*) = 1 + 2(n-2) = 2n-3$ (his distance to r is 1, and his distance to any other node is 2).

It is easy to see that $o*$ is a NE: $1 < \alpha < 2$ means that no player wishes to purchase an edge in order to decrease a distance to a single player from 2 to 1. By enumerating the few possible cases one can also verify that the out-star is a 2-SE. The reason the out-star is not a 3-SE is that any three leaves can deviate to the following strategy: they form a triangle in which each one of them purchases one edge. For each one of them, the deviation decreased his distances from the two other deviators by 1, decreasing his overall distance cost by 2; since his edge cost increased by $\alpha < 2$, the player strictly gains from the deviation.

But is this deviation stable? "Betraying" the deviators and returning to the original profile simply means buying no edges at all. Clearly, such strategy is more beneficial than the deviation – dropping the edge (v, u) decreases the edge cost of v by α while increasing his distance to u from 1 to 2, not affecting his distances to other players. So this deviation is not stable. As we will now show, this holds for every other beneficial deviation as well:

Lemma 1. *The out-star is a 3-NSE.*

In order to prove that, we first need the following:

Lemma 2. *Any stable deviation of k players from the out-star must include the root.*

Proof: Suppose for contradiction that r does not deviate, and let v be any node purchasing at least one edge in the deviation. Since r still buys an edge to every node, v has a path of length 2 to any other node, regardless of his chosen strategy. Therefore, each edge (v, u) decreases v's total distance cost by exactly 1, while increasing his edge cost by $\alpha > 1$. Therefore, v will be strictly better off not joining the deviation. □

Now we can prove lemma 1:

Proof: By lemma 2, the set of deviators is $K = \{r, v, u\}$. Suppose they have a beneficial deviation, $s_K \in S_K$. Now v has to purchase at least one edge – otherwise, $o*$ is not a 2-SE. Since $\alpha > 1$, v has to reduce his total distance cost by at least 2 in order to gain from the deviation. That means someone (other than r) has now to purchase an edge to v. That only leaves u. But then, by the same logic, u cannot gain from the deviation (there is no one left to purchase an additional edge to u). □

Is the out-star a 4-NSE? In general, no. Let $n = 7$, $\alpha = 1.1$. Consider the deviation depicted in Fig.1:

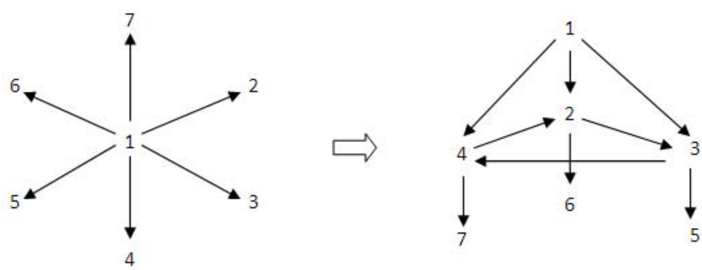


Fig. 1. Here, $n = 7$, $\alpha = 1.1$, and $K = \{0, 1, 2, 3\}$

Here, the root (1) benefits from the deviation, because he now has a distance of 2 to the nodes 5,6 and 7, instead of buying a direct link, so he saves $3(\alpha - 1) > 0$. Obviously, he would not benefit from returning to the original profile, where he buys edges to all nodes. The other deviators purchase two edges each, while decreasing their distances by 3. Since $\alpha < 1.5$, they benefit from the deviation; since each one of them is responsible for connecting a node to the graph, he would suffer a cost of ∞ if returned to play his original strategy (not buying any edges). Therefore, the deviation is stable.

However, for $\alpha \geq 1.5$, the out-star is a 4-NSE. In general, we can show:

Theorem 3. *Let $2 < k \leq n$, and $k' = 2\lfloor \frac{k}{2} \rfloor$. If $\alpha \geq 2(1 - \frac{1}{k'})$, the out-star is a k-NSE.*

Thm. 3 raises two questions. Firstly, is the bound tight? Secondly, suppose the out-star is not a k-NSE. But maybe some other profile is? Are there examples where it can be proved that no k-NSE exists?

The following results address the first question:

Theorem 4. *Let $4 \leq k \leq \lfloor \frac{n+1}{2} \rfloor$, and let $\alpha < 2(1 - \frac{1}{k'})$, with $k' = 2\lfloor \frac{k}{2} \rfloor$. Then the out-star is not a k-NSE.*

Proof: It is sufficient to show a beneficial and stable deviation for a coalition of size k'. Let us denote the root by r and let $K' = K \setminus \{r\}$. The deviation is a generalized version of Fig. 1, and is defined as follows:

- The nodes in K' buy a fair clique between themselves; that is, each buys exactly $(k'-2)/2$ edges to other deviators and they form a clique. It is possible since the clique size $(k'-1)$ is odd.
- Each member of K' is assigned a distinct node in $N \setminus K$, and buys an edge to it. This is possible, because $2|K'| \leq n-1$.
- The root buys edges to all members of K', and buys edges to all the nodes in $N \setminus K$ that are not connected to a node in K'.

Let us show that the deviation is profitable. For r, the difference is that he dropped the edges to some of the nodes in $N \setminus K$ (saving α from each), and as result, his distance to these nodes increased to 2. Since $\alpha > 1$, he gained from the deviation. Each $v \in K'$ gained $k' - 2 + 1 = k' - 1$ in distances (from forming a clique and having a direct connection to a node in $N \setminus K$). He paid $\alpha \left(\frac{k'-2}{2} + 1 \right) = \alpha \frac{k'}{2} < k' - 1$ for his edges. His distance to r stayed 1 and his distances to $N \setminus K$ stayed 2, therefore he gains by deviating. Now, let us show that no one would benefit from going back to the original strategy. For r, we have already seen that he is worse off if he buys edges to all the nodes. For $v \in K'$, dropping his edges leaves him disconnected from his appropriate node in $N \setminus K$, so he cannot deviate. □

So for $k \leq \lfloor \frac{n+1}{2} \rfloor$, the bound of Thm. 3 is tight. Otherwise, it is tight for some cases and not tight for others:

Example 1. Let $n = 15, k = 11, \alpha < \frac{9}{5}$. Suppose that $r = 1$. Consider the following deviation s by $\{1, ..., 11\}$: $s_1 = \{2, ..., 11\}$
The 9 players $\{2, ..., 10\}$ form a fair clique, with each player purchasing 4 edges, and in addition they all point to 11. 11 points to $V \setminus K$:

$s_2 = \{3, 4, 5, 6, 11\}$ $\quad$ $s_6 = \{7, 8, 9, 10, 11\}$
$s_3 = \{4, 5, 6, 7, 11\}$ $\quad$ $s_7 = \{8, 9, 10, 2, 11\}$
$s_4 = \{5, 6, 7, 8, 11\}$ $\quad$ $s_8 = \{9, 10, 2, 3, 11\}$
$s_5 = \{6, 7, 8, 9, 11\}$ $\quad$ $s_9 = \{10, 2, 3, 4, 11\}$
$s_{10} = \{2, 3, 4, 5, 11\}$ $\quad$ $s_{11} = \{12, 13, 14, 15\}$

One can easily verify that the deviation is beneficial and stable. Here, unlike in the proof of Thm. 4, the deviation is stable not because each deviator is responsible for connecting a node to the graph (that holds only for node 11), but because dropping the edge to 11 will increase the player's total distances by 5, making betrayal not beneficial for him for $\alpha < \frac{9}{5}$.

So in this case, the bound of Thm. 3 is tight.

Example 2. Let $n = 7, k = 7, \frac{3}{2} \leq \alpha < \frac{5}{3}$. Here, the out-star is an NSE.

Proof: By checking all possible deviations with NSESAT (see below). □

So in this case the bound of Thm. 3 is not tight.

NSESAT. During the course of this work we often ran into the problem of classifying small instances. Even for number of agents as small as 5, it was often not easy to answer questions such as "Is a given profile a k-NSE (or k-SE)?". The question of existence of k-NSE is even more difficult, since even for very small instances there is a huge number of possible profiles to check. Therefore we developed a computer program for these tasks, called NSESAT. The main idea was to reduce the problem of finding a stable beneficial deviation of at most k players to SAT, and then use a known SAT solver to solve it (we used MiniSat, introduced in (Eén and Sörensson, 2003)). We also added some simple optimizations to break symmetry in symmetric strategy profiles. Despite the fact that the total number of deviations of k players in an n vertex graph is order of $2^{(n-1)k}$, MiniSat handled the decision problem extremely well, allowing us to solve instances for n as high as 11! Using NSESAT, we quickly established the following fact:

Fact 5. *The out-star is an NSE for* $n < 7$.

But this fact raised a problem – it meant that in order to find an example where 4-NSE does not exist, we had to start with n of at least 7. The number of non-isomorphic directed graphs on 7 nodes is 882033440 (Sloane, N. J. A. Sequence A000273 in "The On-Line Encyclopedia of Integer Sequences"). Even with MiniSat deciding on each graph in under 1 second, the straightforward approach of checking all possible graphs was infeasible. Fortunately, we could use the following two necessary conditions for Nash equilibria to narrow the search:

1. The diameter of the undirected graph has to equal 2 (otherwise, if it equals 1, an agent will want to drop an edge, and if there exists a shortest path of at least 3, an agent will want to add an edge).
2. Lemma B.2 in (Andelman *et al.*, 2009) provided a structural property of NE for $1 < \alpha < 2$, which could be efficiently verified on the underlying undirected graph.

Using the above conditions, we were able to start with all the simple undirected graphs of 7 nodes (1044) and reduce the number of potentials for 4-NSE to only 46. All the possible ways to direct these graphs gave us an order of 120000 strategy profiles, which we were able to check in under 10 hours. Our conclusion:

Fact 6. *For* $n = 7, \alpha = 1.1$ *a 4-NSE does not exist.*

Some additional facts that we were able to establish using NSESAT are: for $\alpha \geq 1.5$ the out-star is an NSE for $n \leq 10$. For $n = 11$, the out-star is an NSE for $\alpha \geq \frac{5}{3}$. There are examples where the out-star is not an NSE, but another profile is (such as the in-star, where all leaves buy edges to the root).

Overall, it appears impossible to achieve a better bound on existence of NSE than the one we have with the out-star. Which is unfortunate; not because the bound is not good enough, but rather because the out-star has several serious drawbacks, which would probably make it impractical as recommended strategy profile. Firstly, the profile is extremely unfair – one agent has to incur all the costs of the network. Secondly, its social cost is very high – with $1 < \alpha < 2$, the clique is the social optimum, and a star is the worst NE possible.

In order to address the first issue, it is intuitive to consider correlated mixed strategies (if the players can roll a dice to choose the root, then, at least in terms of expected cost, the profile will be fair). As we will show in the next section, surprisingly, allowing for correlated mixed strategies addresses the second problem as well – we will often be able to implement nearly optimal fair strategy profiles as correlated strong equilibrium.

4 Correlated Mixed Strategies

Let us again recall the problem with implementing good social outcomes as strong equilibrium. Suppose we recommend the agents to form a clique. What is the best response of a single agent? It is to buy no edges at all. Since some of the other agents still connect to him, dropping his edges only increases his corresponding distances from 1 to 2, but saves him α. But what if the agents form a *randomized clique*? A randomized clique (denoted by r-#) is a symmetric correlated strategy profile where all (undirected) edges are bought with probability 1, and for each edge (u, v), the buyer of the edge (u or v) is selected independently and uniformly. Recall that in our model, a deviating agent does not possess the "signal" to his selected strategy; he only knows what randomized strategy the other players will play.

Not buying any edges is no longer a best response of a single agent u to a randomized clique. Why not? Buying an edge (u, v) is now even less beneficial for u than in a pure clique – now with probability 0.5 v will buy the edge himself, so the *expected distance* to v if no edge is bought is 1.5 instead of 2. It is clearly better to lose 0.5 (increasing the distance from 1 to 1.5) than to spend α. However, there is one problem. Since the buyer of each edge is selected independently, there is a small chance that u is selected to buy all his edges. Since the cost of not being connected to another agent is ∞, then, no matter how small the chance, u cannot afford to take it! He must purchase at least one edge (however, one edge will suffice – the randomized clique is not a 1-CSE).

Let us now define a profile we call the randomized out-star (or, simply, the star, denoted by $r*$): the agents select one agent uniformly to be the root. The root purchases edges to all other agents, while the other agents purchase nothing.

Let $0 < \epsilon << 1$. The ϵ-clique is a symmetric correlated strategy profile (denoted by ϵ-#) defined as follows: with probability $1 - \epsilon$, the randomized clique is played, and with probability ϵ, the star is played.

Theorem 7. *For $k < min\{n, 1 + \frac{\alpha}{\alpha-1}\}$, there exists $\epsilon > 0$ s.t. ϵ-# is a k-CSE.*

Proof: Firstly, note that ϵ-#, as well as any other symmetric profile with less than optimal social cost, can never be is an n-CSE, because the n players could always deviate to the optimum, r-#. (But, in a sense, this observation does not take away from the result – the whole point is to implement a socially good outcome, so if the only problem with it is that the agents can deviate to another outcome that is even better for everyone, then it is not really a problem).

Let $k < n$, and K be the set of the deviating agents. Since with positive probability, $\epsilon\left(1-\frac{k}{n}\right) > 0$, the agents in $V \setminus K$ do not buy any edges, any profitable deviation of K must include links to all the agents in $V \setminus K$ (since the deviators do not possess the signals to the chosen pure realization of the correlated strategy, they must consider the worst case). This is the key idea that allows us to show that the agents will not be able to reduce their total cost.

Let us denote by C_{OPT} the summary cost of k agents in a fair clique:

$$C_{OPT} = \alpha\frac{k(n-1)}{2} + k(n-1)$$

The deviation that minimizes the total cost of the agents (their collective best response strategy) is to form a clique among themselves and to purchase an edge to all of $V \setminus K$. Since we are interested in their summary costs, assume w.l.o.g. that a single agent buys all the edges to $V \setminus K$ (obviously, the agents can easily share these costs by selecting this agent uniformly). Let us denote this deviation by g^k, and let us denote by C_{DEV} the total cost the deviating agents incur when playing g^k when the other agents play r-#:

$$C_{DEV} = \alpha\left(\frac{k(k-1)}{2} + n - k\right) + n - 1 + (k-1)(k-1+1.5(n-k))$$

Here, $n-1$ are the total distances of the agent who bought the additional edges, and $k-1+1.5(n-k)$ are the total distances of every other deviator. Recall that 1.5 is the expected distance to any node in $V \setminus K$, since they play r-#. The summary cost of K using the above deviation against the ϵ-# will be slightly higher than C_{DEV}. But, as we will now show, $C_{DEV} > C_{OPT}$:

$$C_{DEV} - C_{OPT} = \alpha(\frac{k(k-1)}{2} + n - k) + n - 1 + (k-1)(k-1+1.5(n-k)) - \alpha\frac{k(n-1)}{2} + k(n-1) =$$

$$= \frac{\alpha}{2}(n-k)(2-k) + \frac{1}{2}(n-k)(k-1) = \frac{1}{2}(n-k)(\alpha(2-k) + k - 1) > 0 \quad \Leftrightarrow$$

$$\alpha(2-k) + k - 1 > 0 \quad \Leftrightarrow \quad k < 1 + \frac{\alpha}{\alpha - 1}$$

Hence, there exists $\epsilon > 0$ s.t. the payoff of k players in ϵ-# will be below C_{DEV}, and therefore, the players will not have a beneficial joint deviation. □

In particular, ϵ-# is a 3-CSE for all $n \geq 4$, and 4-CSE for $n \geq 5, \alpha < 1.5$. Note that the bound on k increases as α approaches 1, meaning that ϵ-# can be implemented as a more stable equilibrium. Formally,

Corollary 8. *If the ϵ-clique is a k-CSE for n, k, α, then for all $\alpha' < \alpha$ it is still a k-CSE for n, α'.*

But what about when α is big? Turns out that in these cases $r*$ (the randomized star) becomes a k-CSE.

Let us denote by C_* the total cost of any k players in $r*$:

$$C_* = \frac{k}{n}(\alpha(n-1) + n - 1 + (n-1)(1 + 2(n-2))) = \frac{k(n-1)}{n}(\alpha + 2n - 2)$$

Let us denote by C_{DEV*} the total cost of k players in their joint best response deviation (which is g^k, the same as against r-#):

$$C_{DEV*} = \alpha(\frac{k(k-1)}{2} + n - k) + \frac{k}{n}(n-1+(k-1)(k-1+2(n-k))) + \frac{n-k}{n}(n-1+(k-1)(k-1+2(n-k))-(k-1)) =$$
$$= \frac{\alpha}{2}(k^2 - 3k + 2n) - k^2 - n + 2nk - k + 1 + \frac{k(k-1)}{n}$$

Theorem 9. *The star, $r*$, is a k-CSE if and only if $C_{DEV*} \geq C_*$.*

Unfortunately, unlike in Thm. 7, the bound cannot be simplified to eliminate one of the variables. However, it can be used to derive various bounds for fixed values of one or two variables. For example, it is simple to see that for $\alpha \geq 1.5$ and $n \geq 10$ the star is a 4-CSE. Also, it is easy to derive the following result:

Corollary 10. *If the star is a k-CSE for n, k, α, then for all $\alpha' > \alpha$ it is still a k-CSE for n, α'.*

Note that when α approaches 2, the social cost of star approaches the social optimum; so Thms. 7 and 9 together imply that as α approaches its bounds, it becomes easier to implement a near-optimal outcome; the intermediate values of α are the most problematic. For example, for $\alpha = 1.5, 4 < n < 10$ neither ϵ-# nor $r*$ are a 4-CSE. But what about other profiles? Can we prove that no 4-CSE exists? We will now show a way to do this for symmetric profiles.

For $s \in \Delta(s)$, let $edges(s)$ denote the expected number of bought edges in s: $edges(s) = \sum_{z \in S} s(z) \sum_v |z(v)|$. Let $maxEdges = \frac{n(n-1)}{2}$. Let $C_G(s)$ denote the total cost of k agents when they deviate to g^k from a symmetric correlated strategy profile s.

Lemma 11. $C_G(s) = \alpha(\frac{k(k-1)}{2} + n - k) + n - 1 + (k-1)(k-1+(2-0.5p)(n-k))$, *where* $p = \frac{edges(s)}{maxEdges}$.

Proof: Consider the expected distance from any node $v \in K$ who does not buy the $n-k$ additional edges to any node $u \in V \setminus K$. We need to show that $E_{s_{[-K]}}\delta(v, u) = 2 - 0.5p$, and the result will follow. The profile is symmetric, therefore the edge (v, u) is bought by u with probability $0.5p$, in which case the distance is 1, and is not bought with probability $1 - 0.5p$, in which case the distance is 2. On average, it is $0.5p + 2(1 - 0.5p) = 2 - 0.5p$. □

Since $C_G(s)$ depends only on $edges(s)$, we will abuse notation and denote it by $C_G(t)$, with $t = edges(s)$.

Lemma 12. *For* $t \in [n-1, maxEdges]$*, let* $v(t) = \frac{n}{k}(C_G(t) - C_{OPT})$*, and* $t' = maxEdges - \frac{v(t)}{2-\alpha}$*. If* $s \in \Delta(S)$ *is a symmetric k-CSE then:*

1. $v(edges(s)) > 0$
2. $edges(s) \geq t'$

Proof

1. Follows from the fact that the total cost of k players in s is strictly above C_{OPT} (since, as we know, the clique is not a k-CSE), and s is a k-CSE (therefore k players cannot beneficially deviate to $Sym(g^k)$).
2. Let $t = edges(s)$. Since the deviation to $Sym(g^k)$ is not beneficial to K, $C_v(s) \leq \frac{C_G(t)}{k} = \frac{C_{OPT}}{k} - \frac{v(t)}{n}$, and therefore $\sum_{v \in V} C_v(s) \leq OPT - v(t)$ (where OPT is the optimal social cost). Each edge removed from the optimal profile (the clique) increases the social cost by at least $2 - \alpha$. Therefore, in order to satisfy the cost bound, s must remove at most $\frac{v(t)}{2-\alpha}$ edges. □

Now, suppose we want to prove that no symmetric k-CSE exists for given n, k, α. We know that any CSE has to have at least $t_0 = n - 1$ edges, because the graph has to be connected; therefore, we can apply Lemma 12 and derive a new lower bound, t_1, for the expected number of edges in a symmetric k-CSE. If $t_1 \leq t_0$, we don't have any additional information, and the process stops (we cannot prove anything). But if $t_1 > t_0$, we can apply Lemma 12 again! This gives us a new bound, t_2. So, we can continue the process and keep deriving lower bounds $t_3, ..., t_j, ...$ for $edges(s)$ in any symmetric k-CSE s. If, at any point in this process, it holds that $C_G(t_j) \leq C_{OPT}$, we have our proof – a symmetric k-CSE does not exist (by Lemma 12, part 1). Similarly, if $t_j \leq t_{j+1}$, the process stops and we cannot prove anything. The only other option is that the series t_j converge. In this case, if $lim_{j \to \infty} t_j = maxEdges$, we have our proof – no symmetric CSE exists (because this means that no profile s with $edges(s) < maxEdges$ can be a k-CSE; and we already know that the clique is not a k-CSE).

Some empirical results: for $5 \leq n \leq 20, 4 \leq k \leq n$, and $\alpha \in 1, 1.05, ..., 1.95$, we tested which of the following holds in each case: ϵ-# is a k-CSE, r* is a k-CSE, both of these profiles are k-CSE, or no symmetric k-CSE exists. In all these runs, we have not encountered a case where neither ϵ-# nor r* were a k-CSE, but the iterative proof that no k-CSE exists failed. These empirical results strongly suggest the following:

Conjecture. For every n, k, α, if a symmetric k-CSE exists, then at least one of ϵ-#, r* is a k-CSE.

Also, although the assumption of symmetry was crucial to our negative result, we conjecture that in this setting, the existence of k-CSE implies the existence of a symmetric k-CSE. Whether this conjecture is correct is an interesting open question for future work.

Now that we have explored the existence of (symmetric) k-CSE, we would like to find out whether k-CNSE might exist in cases where k-CSE does not. Unfortunately, the following result implies otherwise:

Lemma 13. *Let $s \in \Delta(S)$ be a symmetric strategy profile and let $t_K \in \Delta(S_K)$ be a profitable deviation by players $K \subseteq V$. Then there exists a deviation $q_K \in \Delta(S_K)$ which is both profitable and stable.*

Proof: Let $q_K \in \Delta(S_K)$ be defined as follows: with probability $1-\epsilon$ the players play t_K, and with probability ϵ they play the following profile: choose a root $v \in K$ uniformly; the root buys edges to all the other nodes; nodes in $K \setminus \{v\}$ buy no edges. $0 < \epsilon < 1$ is selected so that $C_v(q_K \times s_{[-K]}) < C_v(s)$ for any $v \in K$ (this is possible, since $C_v(t_K \times s_{[-K]}) < C_v(s)$). Since every node in K is now responsible for connecting himself to other nodes with positive probability, the deviation is stable (the only case where $v \in K$ can betray without incurring a cost of ∞ is if he purchases all edges to $K \setminus \{v\}$ in s with probability 1; due to symmetry of s, this is impossible). □

The proof uses the same idea that allowed us to implement good outcomes as k-CSE – using a small probability of having a player disconnected from the graph. An interesting idea for future work is to try and work with a more realistic model, where a player does not incur a cost of ∞ for being disconnected.

References

[Albers *et al.*, 2006] Albers, S., Eilts, S., Even-dar, E., Mansour, Y., Roditty, L.: On nash equilibria for a network creation game. In: Proc. of SODA, pp. 89–98 (2006)

[Andelman *et al.*, 2009] Andelman, N., Feldman, M., Mansour, Y.: Strong price of anarchy. Games and Economic Behavior 65(2), 289–317 (2009)

[Aumann, 1959] Aumann, R.J.: Acceptable points in general cooperative n-person games. In: Tucker, A.W., Luce, R.D. (eds.) Contribution to the Theory of Games, IV. Annals of Mathematics Studies, vol. 40, pp. 287–324 (1959)

[Aumann, 1974] Aumann, R.J.: Subjectivity and correlation in randomized strategies. Journal of Mathematical Economics 1, 67–96 (1974)

[Bernheim *et al.*, 1987] Bernheim, B.D., Peleg, B., Whinston, M.: Coalition proof nash equilibrium i: Concepts. Journal of Economic Theory 42(1), 1–12 (1987)

[Eén and Sörensson, 2003] Eén, N., Sörensson, N.: An extensible SAT-solver. In: Giunchiglia, E., Tacchella, A. (eds.) SAT 2003. LNCS, vol. 2919, pp. 502–518. Springer, Heidelberg (2004)

[Fabrikant *et al.*, 2003] Fabrikant, A., Luthra, A., Maneva, E., Papadimitriou, C.H., Shenker, S.: On a network creation game. In: PODC 2003: Proceedings of the Twenty-Second Annual Symposium on Principles of Distributed Computing, pp. 347–351. ACM, New York (2003)

[Kalai *et al.*, 2007] Kalai, A.T., Kalai, E., Lehrer, E., Samet, D.: Voluntary commitments lead to efficiency. Discussion Papers 1444, Northwestern University, Center for Mathematical Studies in Economics and Management Science (April 2007), http://ideas.repec.org/p/nwu/cmsems/1444.html

[Kaplan, 1992] Kaplan, G.: Sophisticated outcomes and coalitional stability. M.sc. thesis, Dept. of Statistics, Tel. Aviv. University (1992)

[Milgrom and Roberts, 1994] Milgrom, P., Roberts, J.: Strongly coalition-proof equilibria in games with strategic complementarities. Working Papers 95002, Stanford University, Department of Economics (1994)

[Rozenfeld and Tennenholtz, 2006] Rozenfeld, O., Tennenholtz, M.: Strong and Correlated Strong Equilibria in Monotone Congestion Games. In: Proc. of 2nd International Workshop on Internet & Network Economics (2006)

[Rozenfeld and Tennenholtz, 2007] Rozenfeld, O., Tennenholtz, M.: Routing Mediators. In: Proc. 20th International Joint Conference on Artificial Intelligence (2007)

You Share, I Share: Network Effects and Economic Incentives in P2P File-Sharing Systems

Mahyar Salek[1], Shahin Shayandeh[2,⋆], and David Kempe[1]

[1] Department of Computer Science, University of Southern California, CA 90089-0781, USA
{dkempe,salek}@usc.edu
[2] Microsoft Corporation, One Microsoft Way, Redmond, WA, 98052-6399, USA
shahins@microsoft.com

Abstract. We study the interaction between network effects and external incentives on file sharing behavior in Peer-to-Peer (P2P) networks. Many current or envisioned P2P networks reward individuals for sharing files, via financial incentives or social recognition. Peers weigh this reward against the cost of sharing incurred when others download the shared file. As a result, if other nearby nodes share files as well, the cost to an individual node decreases. Such positive network sharing effects can be expected to increase the rate of peers who share files.

In this paper, we formulate a natural model for the network effects of sharing behavior, which we term the "demand model." We prove that the model has desirable concavity properties, meaning that the network benefit of increasing payments decreases when the payments are already high. This result holds quite generally, for submodular objective functions on the part of the network operator.

In fact, we show a stronger result: the demand model leads to a "coverage process," meaning that there is a distribution over graphs such that reachability under this distribution exactly captures the joint distribution of nodes which end up sharing. The existence of such distributions has advantages in simulating and estimating the performance of the system. We establish this result via a general theorem characterizing which types of models lead to coverage processes, and also show that all coverage processes possess the desirable submodular properties. We complement our theoretical results with experiments on several real-world P2P topologies. We compare our model quantitatively against more naïve models ignoring network effects. A main outcome of the experiments is that a good incentive scheme should make the reward dependent on a node's degree in the network.

1 Introduction

Peer-to-Peer (P2P) file sharing systems have become an important platform for the dissemination of files, music, and other content. The basic idea is very simple: individuals make files available for download from their own machine. Other users can search for files they desire and download them from a peer who has made the file available. Naturally, designing systems such that the search and download of files are efficient poses many research challenges, which have received a lot of attention in the literature [2,19].

⋆ Work done while the author was at the University of Southern California.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 354–365, 2010.

A second, and somewhat orthogonal, issue is how to ensure sufficient participation (sharing). Unless enough content is provided by individuals, the utility of membership will be very small. If free-riding [8] is too prevalent, the system may exhibit a quick decrease in membership common to public-goods type economic settings [20].

Thus, the P2P system must be designed with incentives in mind to encourage file sharing. These incentives can take the form of monetary payments or redeemable "points" [10], download privileges, or simply recognition. From the system designer's perspective, these payments should be "small," while ensuring enough participation.

On the other hand, from a peer's perspective, the payments need to be weighed against the cost incurred by sharing a file. In this paper, we assume that the content is shared legally and the system is designed with security in mind: hence, the user's main cost is the upload bandwidth used by another peer to download a file from this node.

Nodes will in general choose to download from nearby peers (in terms of bandwidth or latency). Therefore, as additional nearby peers share the same files, the load will get distributed among more nodes, and the cost to each individual node will decrease. Thus, not only will we expect cascading effects of sharing based on social dynamics [11], but we would also expect these cascading effects to be based on a network structure determined by point-to-point latencies and bandwidths.

Our contribution in this paper is the definition and analysis (both theoretical and experimental) of a natural model for peers' sharing behavior in P2P systems, in the presence of economic incentives. In our model, we focus only on sharing one file; in practice, the model can be applied separately for each file of interest. The basic premise of the model is that each node has a certain *demand* for the file. Furthermore, the network determines which percentage of the demand will be met by downloading from each peer sharing the file[1]. The crucial implication of this model is that the more nearby peers are sharing a file, the more evenly the demand will be distributed among them.

The upload bandwidth cost is compensated by a *payment* to the peers who make the file available. Again, our model is agnostic about whether these payments are monetary, recognition, or take other forms. In our model, the payments can be explicitly based on the network degree of peers, since high-degree nodes presumably serve a key role in propagating sharing behavior.

We argue that this model captures the essential dynamics of P2P systems in which a peer can join the network and download files without sharing; hence, availability of files is not the only incentive for sharing. The FastTrack P2P protocol, used by KaZaA, Grokster, and iMesh, is an example where this assumption holds; hence, our model should be a reasonable approximation for these services in terms of its incentives.

The network operator is interested in maximizing a social welfare function W, which grows monotonically as a function of the set of nodes that share the file. This function could be the total number of sharing nodes, the number of nodes with at least one uploading neighbor, or the total download bandwidth available to peers under various natural models of downloading.

[1] In practice, we could expect these percentages to correlate strongly with network latency or available bandwidth, but our model is agnostic about the derivation.

After defining this model formally (in Section 2), we prove strong and general concavity properties about it (in Section 3). In particular, we show that whenever W is monotone and submodular, the network's social welfare as a function of the payments is monotone and concave. In other words, while increasing payments will always increase social welfare, the *rate* of increase decreases when payments are already high.

To prove this result, we consider a slightly different model, wherein payments are combined with giving the network operator the ability to "force" some set S of peers to share. By first proving certain local submodularity properties for this modified model, the desired concavity properties are implied by the general result of Mossel and Roch [15]. However, we derive a similar result to [15] for a broad subclass of submodular functions which we call coverage functions. It consists of the functions for which in the underlying process, the distribution of nodes sharing the file is equivalent to the distribution of nodes reachable from S in an appropriately defined random graph model. We establish this equivalence via a general and non-trivial theorem characterizing all functions that can be obtained by counting reachable nodes under random graph models. As a corollary, our approach provides a much simpler proof of the main result from [15] for coverage processes. Moreover, the fact that the propagation of sharing behavior is a coverage process is useful for the purpose of simulating the process and estimating the parameters of the system, allowing more efficient algorithms. Finally, our characterization can be of independent interest in the study of submodular set-functions.

While the bulk of our paper focuses on a theoretical analysis of the demand model, we complement the theoretical results by an experimental evaluation of our model (in Section 4), using two network topologies derived from real-world data sets [12,18,17], and a regular two-dimensional grid topology. We first show that network effects are significant by comparing our demand model with one in which peers are not aware of changes in load due to nearby sharing peers. We then evaluate different payment schemes, in particular regarding their dependence on nodes' degrees. We evaluate these both in terms of the fraction of peers that end up sharing, and the amount paid by the network operator per sharing node.

1.1 Related Work

There is a large body of work on incentive mechanisms in P2P file-sharing systems. (See [7] for a thorough overview and [23] for a recent generalized analysis framework.) Incentive mechanisms can be classified in three categories: barter-based mechanisms, reputation-based mechanisms, and currency-based mechanisms.

Barter-based methods [1] (e.g., BitTorrent) enforce repeated transactions among peers by matching each peer to only a small subset of the network, hence raising the survival chance for strategies based on reciprocation. This method only works when we have a small and popular set of files.

Reputation-based mechanisms have an excellent track record at facilitating cooperation in very diverse settings, from evolutionary biology to marketplaces like eBay. These systems keep tally of the contribution of each peer; the past contributions determine which peers obtain more of the system's resources in the future. However, such mechanisms are susceptible to sybil and whitewashing attacks [5,8].

Inspired by markets, a P2P system can also deploy a currency scheme to facilitate resource contributions by rational peers. Generally, peers earn currency by contributing resources to the system, and spend the currency to obtain resources from the system. Karma [21] is one example of this kind. Depending on the policy toward newcomers, such systems may also suffer from sybil and whitewashing attacks.

In [8], file sharing is modeled as a social phenomenon, akin to those discussed by Schelling [20]. Users consider whether or not to contribute files based on the number of other users who contribute. Our model is different in that it explicitly models the costs incurred by contributing nodes, rather than simply positing an intrinsic generosity parameter for each user.

2 Models and Preliminaries

We consider a peer-to-peer network with n servers (or *nodes* or *peers*), and focus on the behavior of sharing one particular file. Thus, each peer v may either choose to share the file or to not share it. We also call sharing peers *active*, and the other ones *inactive*. The set of all peers who share is denoted by V^+.

2.1 The Demand Model

Each peer has a local *demand* d_v for the file: this demand will originate from individual users on the server v (who themselves might not possess the file or be in a position to make it available). The demand d_v should be served by downloading the file from other servers $u \in V^+$. The *quality* of the connection between v and u is captured by a matrix P: the larger $p_{v,u}$, the larger a fraction of v's demand will be served by u (assuming that u shares the file). Specifically, the demand that $u \in V^+$ will see from v is $d_v \cdot \frac{p_{v,u}}{\sum_{w \in V^+} p_{v,w}}$. The matrix P will in practice depend on network latencies or bandwidth, as well as explicit download agreements. It need not be symmetric. For the purpose of the general model, we are agnostic to the derivation of P; in Section 4, we will derive P from measured network latencies by positing a latency threshold which individuals are willing to tolerate.

A node $u \in V^+$ sharing the file will incur a *cost* of c_u per unit of demand that it serves; this cost is the result of using upload bandwidth, machine processing time, or similar resources. To encourage peers to share the file despite this cost, the P2P network administrator offers payments π_u to the nodes $u \in V^+$. These payments need not be the same for all nodes, and can be derived from the network structure, e.g., a node's degree.

Different nodes may have different (and unknown) tradeoffs between money and upload bandwidth. We model this fact by assuming that each node u has a tradeoff factor λ_u, drawn independently and uniformly at random from $[0, 1]$, which captures how many units of bandwidth one unit of money is worth to the node. Thus, the *sharing utility* of an active node $u \in V^+$ is

$$U(u) = \lambda_u \pi_u - c_u \sum_v \frac{d_v p_{v,u}}{\sum_{w \in V^+} p_{v,w}},$$

while the sharing utility of non-sharing nodes is 0. (A non-sharing node does not get paid and incurs no upload costs.) We assume that agents are rational, and thus choose whether to share or not to share so as to maximize their own utility.

2.2 Payment Schemes, Sharing Process, and Administrator's Objective

The network administrator's choice is how to set the payment offers π_u. In doing so, the administrator balances two competing goals: low overall payments and high utility for the participants in the system. In this paper, we study the impact of payment schemes on these objectives.

In order to provide enough incentives for sharing, the network administrator should ensure that $\pi_u \geq C_u := c_u \cdot \sum_v d_v$. Otherwise, even a node u with $\lambda_u = 1$ (i.e., the highest possible utility for money) would have no incentive to share the file if no other peers are sharing the file.

The full model is thus as follows: after the administrator decides on the payments π_u for all nodes u, the random tradeoffs λ_u between money and bandwidth are determined independently for all nodes u. Subsequently, the process proceeds in iterations. In each iteration, all peers simultaneously decide whether to share the file or not, based on the payments, costs, and previous decisions of all other peers. The process continues until an equilibrium is reached. Notice that because the cost to a peer is monotone decreasing in the set V^+ of currently sharing peers, the set of sharing peers can only become larger from iteration to iteration. In particular, this implies that the process will eventually terminate with some set V^+ of active peers. We call this the *sharing process* or *activation process*.

The network administrator is in general interested in increasing access to the file while keeping the payments low. This general objective may be captured using various metrics. In general, we allow for any overall social welfare function W which increases monotonically in the set S of sharing nodes. Notice that since the set S itself is the result of a random process, the administrator's goal will be to maximize $\mathrm{E}[W(S)]$, where S is derived from the random activation process in the demand model. Several social welfare functions W suggest themselves naturally such as the number of active nodes, the number of *serviced* nodes (with at least one active neighbor), or the social utility $\sum_{v \in V^+} p_{u,v}$. Notice that the social welfare function W may also include the utilities of the sharing nodes.

3 Theoretical Analysis of the Model

The main analytical contribution of this paper is based on *coverage processes*[2], defined formally in Definition 2. Informally, a coverage process is a random process such that the distribution over sets of ultimately active nodes is also the distribution of reachable nodes under a suitably chosen distribution of random graphs. Our results on coverage processes are twofold: (1) We give a general characterization of coverage processes, and show that the activation process for P2P systems is a coverage process. (2) We give a significantly simplified proof (compared to the general result of [15]) showing that under coverage processes, the expected social welfare is a concave function of the payments so long as the social welfare is a submodular function of the active nodes.

[2] We thank Bobby Kleinberg for this naming suggestion, and also note here that Theorem 2 was derived independently by him.

A set function f is *submodular* if $f(S+v) - f(S) \geq f(T+v) - f(T)$ whenever $S \subseteq T$, i.e., if the addition of an element to a larger set causes no larger increase in the function value than to a smaller set. Submodularity is the discrete analogue of concavity, and intuitively corresponds to "diminishing returns." An easy inductive proof (on the size of X) shows that submodularity is equivalent to the condition that for all sets X,

$$f(S \cup X) - f(S) \geq f(T \cup X) - f(T) \text{ whenever } S \subseteq T. \tag{1}$$

The two main contributions of our paper imply the following theorem as a corollary:

Theorem 1. *Let $\overline{W}(\pi_1, \ldots, \pi_n) = \mathrm{E}[W(S)]$ be the expected social welfare when S is obtained from the sharing process of the demand model with payments $\pi_1, \ldots, \pi_n$.*
If $W(S)$ is submodular, then $\overline{W}(\pi_1, \ldots, \pi_n)$ is a concave function of $\pi_1, \ldots, \pi_n$.

Concavity in payments intuitively means that the additional benefit in social welfare that can be derived from increasing the payment to a peer u decreases as the peers' current payments increase.

The proof of Theorem 1 is based on analyzing the following *Seed Set Model*, which we define mainly for the purpose of analysis.

Definition 1 (Seed Set Model). *For each node, the payment offered is $\pi_u = C_u$. Besides payments, we have a* seed set *S of peers that will always share regardless of the payments. Subsequently, the process unfolds exactly according to the sharing process.*

The main technical step is to show that the Seed Set Model is a *coverage process*, in the following sense.

Definition 2 (Coverage Process). *Let $\phi(S)$ be the random variable describing the set of nodes active at the end of a process starting from the set S of nodes active. The process is called a* coverage process *if there exists a distribution D over graphs G such that for each set T of nodes, $\mathrm{Prob}[\phi(S) = T]$ equals the probability that exactly T is reachable starting from S in G if G is drawn from the distribution D.*

Remark 1. Without using our nomenclature, [13] showed submodularity for the Cascade and Threshold models of innovation diffusion [11,9] by establishing that both gave rise to coverage processes. Subsequently, [14] showed that there are natural diffusion processes which are not coverage processes, yet have a submodular function $\mathrm{E}[|\phi(S)|]$.

We prove that the Seed Set Model is a coverage process in two steps. First, in Section 3.1, we give a general and complete characterization of coverage processes. This characterization may be of interest in its own right, as coverage processes have a practical advantage: they can be simulated easily and efficiently, by first generating a random graph according to D, and then simply finding the set of reachable nodes.

Then, in Section 3.2, we show that the Seed Set Process satisfies the conditions established in Section 3.1. Finally, in Section 3.3, we give a simple proof that for *any* coverage process and any submodular social welfare function, the expected social welfare under the process is also submodular. This immediately implies concavity as a function of the payments.

Remark 2. The fact that the tradeoffs λ_u between money and bandwidth are uniformly random in $[0,1]$ is important to ensure the submodularity and concavity properties. If the λ_u are not random but fixed, then the concavity and submodularity properties cease to hold. Furthermore in the Seed Set Model, the optimization problem of finding the best seed set S of at most k nodes becomes inapproximable (via a reduction from SET COVER); details are in the full version of this paper. (The full version is available from the authors' web sites.) This contrasts with Corollary 1 below for the optimization problem with random tradeoffs.

3.1 Characterization of Coverage Processes

Our setting is exactly as in the recent paper on Viral Marketing in Social Networks by Mossel and Roch [15]: each node u has an activation function f^u, which is monotone non-decreasing and satisfies $f^u(\emptyset) = 0$. Each node independently chooses a threshold $\theta_u \in [0,1]$ uniformly at random, and becomes active when $f^u(S) \geq \theta_u$, where S is the previously active set of nodes.

In order to express our results concisely, we use the following discrete equivalent of a derivative (see, e.g., [22]). For a function f defined on sets, we define inductively:

$$\begin{aligned} f_\emptyset(S) &= f(S) \\ f_{R\cup\{v\}}(S) &= f_R(S\cup\{v\}) - f_R(S). \end{aligned}$$

It is not difficult to verify that this notion is well-defined, i.e., independent of *which* element v is chosen at which stage. Moreover, an easy induction shows that discrete derivatives can be expressed non-recursively as follows: For all sets T, we have that

$$f_T(W) = \sum_{S\subseteq T}(-1)^{|T|-|S|} f(W\cup S).$$

Theorem 2. *The following conditions are necessary and sufficient for the process to be a coverage process.*

- *For all sets T of odd cardinality $|T|$, as well as for $T=\emptyset$, and each node u, we have $f^u_T(\overline{T}) \geq 0$.*
- *For all sets T of positive even cardinality $|T|$, and each node u, we have $f^u_T(\overline{T}) \leq 0$.*
- *$f^u(\emptyset) = 0$ for all u.*

To prove this theorem, we begin with the following reasoning. Focus on one node u, and its activation function f^u. If there were an equivalent graph distribution D, then it would have to define a probability $q_u(T)$ for the presence of edges from exactly the set T of vertices to u. These probabilities need to satisfy the following property: if a set S of nodes is active, then the probability of u having *at least* one incoming edge from S must equal $f^u(S)$. Thus, a necessary and sufficient condition for being a coverage function is that for each node u, there exists a distribution $q_u(T)$ over sets T such that

$$f^u(S) = \sum_{T:T\cap S\neq\emptyset} q_u(T). \tag{2}$$

We can express this requirement more compactly. Let $\mathbf{f}_u$ be the (2^n-1)-dimensional vector of all entries of $f^u(S)$ for $S\neq\emptyset$. Similarly, let $\mathbf{q}_u$ be the (2^n-1)-dimensional vector of all $q_u(S)$ for $S \neq \emptyset$. Let A be the $((2^n - 1) \times (2^n - 1))$-dimensional matrix indexed by non-empty subsets such that $A_{S,T} = 1$ if and only if $S \cap T \neq \emptyset$, and $A_{S,T} = 0$ otherwise. (A is called an *incidence matrix* [4].) Then, Equation 2 states the requirement that for each node u, there exists a distribution $\mathbf{q}_u$ such that $A \cdot \mathbf{q}_u = \mathbf{f}_u$.

For the analysis, it will be useful to fix a canonical ordering of subsets. Specifically, if the current (sub-)universe consists of k nodes indexed $\{1, 2, \ldots, k\}$, their canonical ordering is defined recursively as first containing all subsets of $\{1, 2, \ldots, k-1\}$ in canonical order, then the set $\{k\}$, followed by the sets $T \cup \{k\}$, where the sets $T \subseteq \{1, 2, \ldots, k-1\}$ appear in canonical order.

In order to find out when the distribution $\mathbf{q}_u$ exists, we want to solve the equation $A \cdot \mathbf{q}_u = \mathbf{f}_u$, or $\mathbf{f}_u = A^{-1} \cdot \mathbf{q}_u$. While the inverses of some incidence matrices have been studied before (see, e.g., [3]), we are not aware of any source explicitly giving the inverse of the matrix A. Hence, we establish here:

Lemma 1. *The inverse of A is the matrix B defined by*

$$b_{S,T} := \begin{cases} 0 & \textit{if } S \cup T \neq \{1, \ldots, n\} \\ (-1)^{|S \cap T|+1} & \textit{otherwise} \end{cases}$$

The next lemma shows that so long as all $q_u(S)$ are non-negative, by setting $q_u(\emptyset)$ appropriately, we can always obtain a probability distribution.

Lemma 2. *With $q_u(S)$ defined as $\mathbf{q}_u = B \cdot \mathbf{f}_u$, we have $\sum_S q_u(S) \leq 1$.*

(Due to space-constraints, the proofs of the above lemmas are deferred to the full version of the paper.) By Lemma 1, we know that $\mathbf{q}_u = B \cdot \mathbf{f}_u$. And by Lemma 2, the entries sum up to at most 1. Thus, it remains to show that the entries of $\mathbf{q}_u$ are non-negative if and only if f^u satisfies the conditions of Theorem 2.

Proof of Theorem 2. Fix any node u, and define $\mathbf{q}_u = B \cdot \mathbf{f}_u$. The discrete derivative of f^u at $\overline{T}$ can be expressed as $f^u_T(\overline{T}) = \sum_{S \subseteq T} (-1)^{|T|-|S|} f^u(\overline{T} \cup S)$. If $|T|$ is odd, then $(-1)^{|T|-|S|} = (-1)^{|S|+1}$, so we can rewrite the above as

$$\textstyle\sum_{S \subseteq T} (-1)^{|S|+1} f^u(\overline{T} \cup S) = \sum_{W \supseteq \overline{T}} (-1)^{|W \cap T|+1} f^u(W) = q_u(T).$$

Similarly, if $|T|$ is even, then $(-1)^{|T|-|S|} = (-1)^{|S|}$, so we can rewrite the discrete derivative as

$$\textstyle\sum_{S \subseteq T} (-1)^{|S|} f^u(\overline{T} \cup S) = \sum_{W \supseteq \overline{T}} (-1)^{|W \cap T|} f^u(W) = -q_u(T).$$

Thus, the $q_u(T)$ are all non-negative (and the probability distribution thus well-defined) if and only if $f^u_T(\overline{T}) \geq 0$ for $|T|$ odd, and $f^u_T(\overline{T}) \leq 0$ for $|T| > 0$ even. ■

3.2 Coverage Property of the Seed Set Process

In this section, we establish the following theorem.

Theorem 3. *The Seed Set Process is a coverage process.*

Proof. In order to prove this theorem, we want to apply Theorem 2. To do so, we need show that the local decisions of nodes about sharing can be cast in terms of submodular threshold functions. Specifically, we define $f^u(S) = 1 - \frac{1}{C_u} \cdot c_u \cdot \sum_v \frac{d_v p_{v,u}}{\sum_{w \in S \cup \{u\}} p_{v,w}}$ and let $\theta_u = 1 - \frac{\lambda_u \pi_u}{C_u}$.

A node u becomes active if doing so has positive utility, i.e., if $\lambda_u \pi_u > c_u \cdot \sum_v \frac{d_v p_{v,u}}{\sum_{w \in S \cup \{u\}} p_{v,w}}$. Dividing both sides by C_u, and subtracting from 1 shows that this is equivalent to saying that $1 - \frac{\lambda_u \pi_u}{C_u} < 1 - \frac{1}{C_u} \cdot c_u \cdot \sum_v \frac{d_v p_{v,u}}{\sum_{w \in S \cup \{u\}} p_{v,w}}$. Since $\lambda_u \pi_u$ is uniformly random in $[0, C_u]$ by the definition of π_u in the Seed Set Model, this condition is equivalent to saying that $\theta_u < f^u(S)$. Thus, we have shown that the activation process can be equivalently recast in terms of threshold activations functions.

Finally, we need to show that for every node u, all derivatives $f^u_T(S)$ are non-negative when $|T|$ is odd and non-positive when $|T| > 0$ is even. (The fact that $f^u(S) = f^u_\emptyset(S)$ is non-negative follows directly by definition.) Let

$$\hat{f}^u(x_1, \ldots, x_n) = 1 - \frac{1}{C_u} \cdot c_u \cdot \sum_v \frac{d_v p_{v,u}}{\sum_{v_i \in V} p_{v,v_i} x_i}$$

be the continuous equivalent of the local influence function f^u. For a set S, let $\mathbf{y}^{(S)}$ denote the n-dimensional vector with $y_i^{(S)} = 1$ if $v_i \in S \cup \{u\}$ and $y_i^{(S)} = 0$ otherwise. Then, $f^u(S) = \hat{f}^u(\mathbf{y}^{(S)})$. Notice that by definition, there is no division by zero.

Writing $dY_T = dy_{i_1} dy_{i_2} \cdots dy_{i_{|T|}}$, where $T = \{i_1, i_2, \ldots, i_{|T|}\}$, an easy inductive proof first shows that $f^u_T(S) = \int_0^1 \ldots^{|T|} \int_0^1 \frac{d\hat{f}^u(\mathbf{y}^{(S)})}{dY_T} dY_T$. It remains to show that each term inside the integration is non-negative for odd $|T|$ and non-positive for even $|T|$. We accomplish this by showing that

$$\frac{d\hat{f}^u(\mathbf{y}^{(S)})}{dY_T} = (-1)^{|T|+1} |T|! \frac{c_u}{C_u} \sum_v \frac{d_v p_{v,u} \prod_{t \in T} p_{v,t}}{(\sum_{v_i \in V} p_{v,v_i} y_i^{(S)})^{|T|+1}}.$$

The proof is by induction. The base case $|T| = 1$ can be verified easily. Assume that the claim holds for $|T| = i - 1$. We have

$$\begin{aligned}
\frac{d\hat{f}^u(\mathbf{y}^{(S)})}{dY_T dy_i} &= \frac{d}{dy_i} (-1)^{|T|+1} |T|! \frac{c_u}{C_u} \sum_v \frac{d_v p_{v,u} \prod_{t \in T} p_{v,t}}{(\sum_{v_i \in V} p_{v,v_i} y_i)^{|T|+1}} \\
&= (-1)(-1)^{|T|+1} |T|! \frac{c_u}{C_u} \cdot \sum_v \frac{(|T|+1) p_{v,v_i} d_v p_{v,u} \prod_{t \in T} p_{v,t} (\sum_{v_i \in V} p_{v,v_i} y_i)^{|T|}}{(\sum_{v_i \in V} p_{v,v_i} y_i)^{2|T|+2}} \\
&= (-1)^{|T|+2} |T+1|! \frac{c_u}{C_u} \sum_v \frac{d_v p_{v,u} \prod_{t \in T \cup \{v_i\}} p_{v,t}}{(\sum_{v_i \in V} p_{v,v_i} y_i)^{|T|+2}}.
\end{aligned}$$

This completes the inductive proof, and thus the proof of Theorem 3. ■

While we defined the Seed Set Process primarily as a tool for analysis, we remark here that using a theorem of Nemhauser et al. [6, 16], Theorem 3 has a direct consequence for the optimization problem of maximizing the expected total number of active nodes at the end of the process, subject to a size constraint on the seed set S.

Corollary 1. *The best starting set S for the Seed Set Process can be approximated within $(1 - 1/e - \epsilon)$ in polynomial time, for any $\epsilon > 0$.*

3.3 Concavity of Expected Social Welfare

Finally, we use the machinery of coverage processes to show submodularity and concavity of social welfare. Consider an arbitrary coverage process. When the coverage process starts with the set T, let $\phi(T)$ be a random variable describing the set of nodes active at the end of the process. Thus, the distribution of $\phi(T)$ for all T precisely characterizes the coverage process. Our main theorem is now the following:

Theorem 4. *Let $h(S)$ be any monotone submodular function of S. Then, $\mathrm{E}[h(\phi(T))]$ is a monotone submodular function of T, where the expectation is taken over the randomness in $\phi(T)$.*

This theorem follows from the general result of [15], since all coverage processes are locally submodular, and our utility function is submodular with respect to the set of sharing neighbors. However, we derive it more simply using reachability in graphs and the fact that ϕ is a coverage process. For coverage processes, instead of generating random thresholds and simulating a dynamic process, we can generate a random graph and then simply use Dijkstra's algorithm to find the number of reachable nodes. The proof of Theorem 4 is in the full version of the paper.

The final piece of the proof of Theorem 1 is to show that monotonicity and submodularity of the Seed Set Model imply concavity for the original model.

Lemma 3. *Let f be a non-negative, monotone, submodular function on sets. Consider the function g defined as follows: Each element u is included in S independently with probability $q_u(\pi_u)$, where q_u is an increasing and concave function of π_u. Define $g(\pi) = \mathrm{E}[f(S)]$. Then, g is monotone and concave.*

The proof of Lemma 3 is virtually identical to the second half of the proof of Theorem 6.1 in [13], and we therefore omit it here. With Theorem 3 and Lemma 3, we can now complete the proof of Theorem 1.

Proof of Theorem 1. Consider one node u. The probability that it becomes active initially is $p_u^0 = \mathrm{Prob}[\lambda_u \pi_u \geq C_u] = 1 - \frac{C_u}{\pi_u}$. Recall that $C_u = c_u \cdot \sum_v d_v$, and $\pi_u \geq C_u$ in our model, so this number is always non-negative. Clearly, p_u^0 is also a monotone increasing function of π_u. Moreover, the second derivative is $\frac{-2C_u}{(\pi_u)^3}$, and thus non-positive, so p_u^0 is concave.

Now, consider all the nodes u which did not initially become active. This is equivalent to saying that $\lambda_u \pi_u \leq C_u$. But subject to this bound, $\lambda_u \pi_u$ is uniformly random, so we are in the situation of having an initially active set S, and for each remaining node u, the payment is independently and uniformly random in $[0, C_u]$. By Theorems 3 and 4, the expected social welfare $\overline{W}(S)$ is a monotone and submodular function of the seed set S, so long as W is submodular in the set of active nodes. We can therefore apply Lemma 3 to $\mathrm{E}[h(\phi(T))]$, which implies that $\overline{W}(\pi_1, \ldots, \pi_n)$ is concave in its arguments. ■

It is not difficult to verify that all of the social welfare functions listed in Section 2 are monotone and submodular in the set of active nodes. (Due to space constraints, we defer the formal proofs to the full version of this paper.) Thus, for all of these objective functions, the total social welfare is a monotone concave function of the payments.

4 Experimental Evaluation

In this section, we summarize our observations based on simulations both on synthetic and real-world P2P networks. The detailed discussion with plots is relegated to the full version of the paper due to space constraints.

We defined two other models with no or limited network effects and compared them against the demand model. In the *No-Network* Model, the peers completely ignore other sharing peers. Thus, a node u assumes that if it shares the file, then it will see a fraction $p_{v,u}$ of the demand originating with node u. In the *One-Hop* Model, the peers are aware of network effects in a very limited way: node u assumes that any node v sharing the file will contribute toward serving both v's and u's demand, but not toward serving the demand of any other node $w \neq u, v$. We developed a simulator for the three models and evaluated several social welfare functions defined in Section 2.

In addition, we calculated the total payments and the average payment per active and per serviced node. These numbers are averaged over 1000 iterations, each with different random λ. Based on the simulations, the following were our main observations:

1. Our results show a significant difference between the models in their prediction of sharing: while the fraction of sharing nodes is qualitatively similar, the predictions ignoring network effects can be off by about 15%–25%. This results in up to 10% depreciation in the number of serviced peers.
2. We observe that the denser the network, the higher the rate of participation, given fixed incentives. This holds across grid and realistic Internet topologies.
3. Our experiments suggest that the payments π_u for realistic topologies should be proportional to u's degree to give high overall participation at low cost.

5 Conclusions

There are several natural directions for future work. A very interesting question arises when taking payments by "reputation" or download priorities into account. While monetary payments can (in principle) be increased arbitrarily, reputation is inherently constant-sum: if some peers are recognized as outstanding sharers, then others will receive less recognition. Similarly, download priorities come at the expense of other peers, and can thus not be arbitrarily increased for all members of the network. As a result, the process of sharing will not necessarily be monotone: peers may choose to stop sharing once too many other peers are active. A first question is then whether stable (equilibrium) states even exist. If so, it would be interesting what fraction of the peers will be sharing, what the social welfare is, and how these quantities will depend on the network structure.

From a more practical viewpoint, it would be desirable to evaluate how accurately our model (or a variation thereof) captures the actual behavior of participants in a P2P system. This would likely be a difficult experiment to perform, as many of the parameters, such as file demands and latency, are inherently transient, and in a realistic system, payments cannot be changed constantly to evaluate the impact of such changes.

Finally, our work lies among various applications in economics for which there are positive or negative externalities among agents in a neighborhood. Our results suggest we should always consider the cascading effects of agents' strategies over the network on different economic metrics such as revenue or social welfare.

Acknowledgments. We would like to thank Bobby Kleinberg and Jan Vondrák for useful discussions, and anonymous referees for useful feedback.

References

1. Anagnostakis, K.G., Greenwald, M.B.: Exchange-based incentive mechanisms for peer-to-peer file sharing. In: Proc. of ICDCS 2004, Washington, DC, USA, pp. 524–533. IEEE Computer Society, Los Alamitos (2004)
2. Aperjis, C., Freedman, M.J., Johari, R.: Peer-assisted content distribution with prices. In: Proc. of CONEXT, pp. 1–12. ACM, New York (2008)
3. Bapat, R.: Moore-Penrose inverse of set inclusion matrices. Linear Algebra and its Applications 318(1), 35–44 (2000)
4. Brualdi, R., Ryser, H.: Combinatorial Matrix Theory. Cambridge University Press, Cambridge (1991)
5. Cheng, A., Friedman, E.: Sybilproof reputation mechanisms. In: Proc. of P2PECON 2005, pp. 128–132. ACM, New York (2005)
6. Cornuejols, G., Fisher, M., Nemhauser, G.: Location of bank accounts to optimize float. Management Science 23, 789–810 (1977)
7. Feldman, M., Chuang, J.: Overcoming free-riding behavior in peer-to-peer systems. SIGecom. Exch. 5(4), 41–50 (2005)
8. Feldman, M., Papadimitriou, C., Chuang, J., Stoica, I.: Free-riding and whitewashing in peer-to-peer systems. In: Proc. ACM PINS 2004 (2004)
9. Goldenberg, J., Libai, B., Muller, E.: Talk of the network: A complex systems look at the underlying process of word-of-mouth. Marketing Letters 12, 211–223 (2001)
10. Golle, P., Leyton-Brown, K., Mironov, I., Lillibridge, M.: Incentives for sharing in peer-to-peer networks. In: Fiege, L., Mühl, G., Wilhelm, U.G. (eds.) WELCOM 2001. LNCS, vol. 2232, pp. 75–87. Springer, Heidelberg (2001)
11. Granovetter, M.: Threshold models of collective behavior. American J. of Sociology 83, 1420–1443 (1978)
12. Gummadi, K.P., Saroiu, S., Gribble, S.D.: King: Estimating latency between arbitrary internet end hosts. In: Proc. 21st ACM SIGCOMM (2002)
13. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence in a social network. In: Proc. 9th KDD, pp. 137–146 (2003)
14. Kempe, D., Kleinberg, J.M., Tardos, É.: Influential nodes in a diffusion model for social networks. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 1127–1138. Springer, Heidelberg (2005)
15. Mossel, E., Roch, S.: On the submodularity of influence in social networks. In: Proc. 38th ACM STOC, pp. 128–134 (2007)
16. Nemhauser, G., Wolsey, L., Fisher, M.: An analysis of the approximations for maximizing submodular set functions. Mathematical Programming 14, 265–294 (1978)
17. Network Coordinate Research at Harvard
18. Parallel & Distributed Operating Systems Group at MIT
19. Saroiu, S., Gummadi, P., Gribble, S.: A measurement study of peer-to-peer file sharing systems. In: Proc. of MMCN (2002)
20. Schelling, T.: Micromotives and Macrobehavior. Norton (1978)
21. Vishnumurthy, V., Chandrakumar, S., Sirer, E.: Karma: A secure economic framework for peer-to-peer resource sharing. In: 1st Workshop on Economics of Peer-to-Peer Systems (2003)
22. Vondrák, J.: Optimal approximation for the submodular welfare problem in the value oracle model. In: Proc. 39th ACM STOC, pp. 67–74 (2008)
23. Zhao, B.Q., Lui, J.C.S., Chiu, D.-M.: Analysis of adaptive protocols for p2p networks. In: Proc. 28th IEEE INFOCOM (2009)

The Complexity of Equilibria in Cost Sharing Games

Vasilis Syrgkanis*

Department of Computer Science, Cornell University, Ithaca, NY 14853
vasilis@cs.cornell.edu

Abstract. We study Congestion Games with non-increasing cost functions (Cost Sharing Games) from a complexity perspective and resolve their computational hardness, which has been an open question. Specifically we prove that when the cost functions have the form $f(x) = c_r/x$ (Fair Cost Allocation) then it is PLS-complete to compute a Pure Nash Equilibrium even in the case where strategies of the players are paths on a directed network. For cost functions of the form $f(x) = c_r(x)/x$, where $c_r(x)$ is a non-decreasing concave function we also prove PLS-completeness in undirected networks. Thus we extend the results of [7, 1] to the non-increasing case. For the case of Matroid Cost Sharing Games, where tractability of Pure Nash Equilibria is known by [1] we give a greedy polynomial time algorithm that computes a Pure Nash Equilibrium with social cost at most the potential of the optimal strategy profile. Hence, for this class of games we give a polynomial time version of the Potential Method introduced in [2] for bounding the Price of Stability.

Keywords: Cost Sharing, PLS-completeness, Price of Stability, Congestion Games.

1 Introduction

The rapid and overwhelming expansion of the Internet has transformed it into a completely new economic arena where a large number of self-interested players interact. The lack of central coordination has rendered classic optimization problems insufficient to capture Internet interactions and has given rise to new game-theoretic models. In this work we study a general such model, namely Congestion Games with non-increasing cost functions (Cost Sharing Games), from a complexity perspective and we resolve the computational hardness of computing a Pure Nash Equilibrium (PNE) in such games, which has been an open problem. The computational hardness is an important aspect of an equilibrium concept since it indicates whether it is a reasonable outcome in real world settings.

We start with a motivating example: a group of Internet Service Providers (ISP) wants to create a new network on a set of nodes (possibly different set for each provider). Each ISP's goal is that any two of his nodes are connected

* Supported in part by NSF grant CCF-0729006.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 366–377, 2010.

by a path. For practical reasons, each provider can build edges only between two nodes in his set and his clients can use a link only if the ISP helped build it. Moreover, we assume that ISPs are clever enough and when they decide to build the same link as others then they all build one link and share the cost. The moment we add this last specification, the problem faced by an ISP is no longer an optimization problem and the setting becomes a game which from now on we will call the ISP Network Creation Game. ISP Network Creation Games can be easily modeled as Cost Sharing Games.

Congestion Games in general has been a widely studied game theoretic model. In Congestion Games a set of players allocate some set of shared resources. The cost incurred from using a resource is a function of the number of players that have allocated the resource and the total cost of a player is the sum of his costs on all the resources he has allocated. A reasonable outcome of such a setting is a Pure Nash Equilibrium (PNE): a strategy profile such that no player can profit from deviating unilaterally. In a seminal paper, Rosenthal [13] gives a proof that Congestion Games always possess a PNE. To achieve this, he introduces a potential function and shows that the change in the potential induced by a unilateral move of some player is equal to the change of that player's utility. Several aspects of the PNE of Congestion Games have been studied in the literature.

An interesting research area has been the complexity of computing a PNE in Congestion Games. In a seminal paper Fabrikant et al. [7] proved that the above problem is PLS-complete even in the case where the strategies of the players are paths in a directed network. Later, Ackermann et al. [1] extended the above result to the case of undirected networks with linear cost functions. However, both results use cost functions on the resources that are non-decreasing (delays) and do not carry over to Cost Sharing Games. The complexity of computing a PNE in Cost Sharing Games has been an open question.

Another interesting line of research has been measuring the inefficiency that arises from selfishness. An important concept in that direction (especially in the case of Cost Sharing Games) has been the Price of Stability (PoS), which is the ratio of the quality (sum of players' costs) of the best PNE over the socially optimal outcome ([2]). One major motivation for the PoS is that it is the socially optimal solution subject to the constraint of unilateral stability. If there was a third-party that could propose to players a solution to their problem, then the optimal stable solution he could propose would be the best PNE. This motivation raises an interesting open question: Given an upper bound on the PoS for a class of games, is there a polynomial-time algorithm for computing a PNE with cost comparable to that bound?

In this work we make significant progress in both directions described above. We prove the first PLS-hardness results for Cost Sharing Games. Our results show that the non-increasing case is not easier than the non-decreasing. Moreover, we give the first polynomial-time algorithm that computes a PNE with quality equal to the known bound on the PoS for a significant class of Cost Sharing Games that contains, for example, the ISP Network Creation Game.

Results

- Our first main result is that a greedy approach leads to a polynomial time algorithm that computes a PNE of any Matroid Cost Sharing Game, with cost equal to the potential of the socially optimal solution. The quality of such a PNE is no worse than any bound on the PoS that can be proved via the Potential Method. Hence, for this class we give a polynomial time equivalent to the Potential Method. Matroid Cost Sharing Games are Cost Sharing Games where the strategy space of each player is exactly the set of bases of a player-specific matroid. The existence of algorithms like the one given here has been an interesting open question [18]. From previous work [5], we know that computing the global potential minimizer is NP-hard even for Singleton Cost Sharing Games. Also we note here that the same holds for the minimum social cost PNE. Hence it is surprising that we can achieve such an efficiency guarantee.
- The above result directly implies the logarithmic bound on the PoS for Matroid Cost Sharing Games with cost functions of the form $f(x) = c_r(x)/x$, where $c_r(x)$ is a nondecreasing concave function.
- For the case of Singleton Cost Sharing Games our algorithm does not output just a PNE but a Strong Nash Equilibrium. Hence this extends the results in [6] on the existence of Strong Nash Equilibria in Cost Sharing Games.
- Our second main result is that computing a PNE in the class of Network Cost Sharing Games where the cost functions come from the Shapley Cost Sharing Mechanism, $f(x) = c_r/x$ (Fair Cost Allocation) is PLS-complete. The hardness results are based on a tight PLS-reduction from MAX CUT. The result is not restrictive to Fair Cost Allocation and holds for almost any reasonable set of decreasing functions.
- The tightness of our reduction also shows that there exist instances of Network Cost Sharing Games with Fair Cost Allocation, where best response dynamics will certainly need exponential time to reach a PNE. This gives a negative answer to an interesting open question proposed in [2] of whether there exist a scheduling of best response moves that lead to a PNE in polynomially many steps.
- For cost functions of the form $f(x) = c_r(x)/x$, where $c_r(x)$ is a nondecreasing concave function we also prove PLS-completeness for the case of Undirected Network Cost Sharing. This result is not restricted to the above class of functions but generalizes to any class of cost functions that contains almost constant functions.
- The new techniques that we introduce can be used to simplify the existing reductions for the non-decreasing case.

Techniques. To prove our main hardness result we introduce a new class of Congestion Games called k-Congestable Congestion Games. In a k-Congestable Congestion Game the resources of any two strategies of a player are disjoint and each resource is contained in some strategy of at most k players. Thus at most k players can share a resource in any strategy profile. These games generalize k-Threshold Games introduced by Ackermann et al. [1].

We show how to reduce the computation of a PNE in a 2-Congestable Congestion Game with cost functions that satisfy certain assumptions, to the same problem in a Network Congestion Game with the same set of cost functions. If the class of cost functions is general enough to contain almost constant functions with arbitrary high cost, then we can reduce 2-Congestable Congestion Games to Undirected Network Games. We notice that the MAX CUT reduction of Fabrikant et al [7] constructs a 2-Congestable Congestion Game hence our techniques can be applied to simplify the PLS-completeness proofs for the non-decreasing case.

A lot of the proofs of our results had to be omitted due to space limitations. See the full version of the paper for the proofs that had to get deleted from this proceedings version.

Related Work

Complexity of Equilibria. Apart from the results mentioned in the introduction [7, 1], there has been several works on the relation between PLS and PNE. Skopalik et al [16] proves that even computing an approximate PNE is PLS complete for Congestion Games. Their techniques can also be used to prove PLS-completeness of approximate equilibria in Bottleneck Games (player cost is maximum of cost of allocated resources) as noted independently by Syrgkanis [17] and Harks et al [9].

For Cost Sharing Games Chekuri et al [5] prove that it is NP-hard to compute the global potential minimizer for Multicast Games with Fair Cost Allocation. Hansen et al [8] give an exponential sequence of best response moves for the case of Metric Facility Location Games and provide a polynomial time algorithm for computing approximate equilibria in that class.

On the positive side, Ieong et al. [10] give a dynamic programming algorithm for computing the optimal PNE in the class of Symmetric Singleton Games with arbitrary cost functions. Moreover, Ackermann et al [1] introduce Matroid Congestion Games as a class of games where best response dynamics converge in polynomially many steps.

Quality of Equilibria. Cost Sharing Games in the form studied in this work were introduced by Anshelevich et al. [2]. One of their main results is that the PoS is $O(\log(n))$ (where n is the number of players) for Cost Sharing Games where the cost functions have the form $f(x) = c_r(x)/x$, where $c_r(x)$ is a nondecreasing concave function. Their proof introduces the Potential Method, a way of bounding the PoS by showing that the global minimizer of Rosenthal's potential (which is a PNE) has cost close to the optimal.

Several other works have dealt with Cost Sharing Games from the perspective of bounding the inefficiency that arises from selfishness. Chekuri et al [5] study Multicast and Facility Location Games when players arrive sequentially and then perform best response. They prove that the quality of the resulting PNE is at most $O(\sqrt{n}\log^2 n)OPT$. Later Charikar et al. [4] improve this bound to $O(\log^3 n)OPT$ and also make progress for the case when best response and sequential arriving is interleaved. Epstein et al [6] study the quality of Strong Nash Equilibria of Cost Sharing Games with Fair Cost Allocation. Strong Nash Equilibria allow for group moves of players, therefore they are a solution concept robust to collusion. However, they do not always exist in Cost Sharing Games. When they exist Epstein et al [6] show that their worst case quality matches the PoS bound of H_n. Balcan et al [3] study Cost Sharing Games with Fair Cost Allocation under the perspective of Learning Agents. They prove that if players perform best response but at each step with a small fixed probability chose their strategy in a nearly optimal outcome, then the expected quality of the resulting PNE is $O(\log(n)\log(n|F|))OPT$, where $|F|$ is the number of resources.

2 Definitions and Notation

Definition 1. *A **Congestion Game**(CG), denoted $\langle N, F, (S_i)_{i\in N}, (r_f)_{f\in F}\rangle$, consists of: A set of N players and a set of facilities F. For each player i a set of strategies $S_i \subseteq 2^F$. For each facility f a cost function $r_f(x)$. Given a strategy profile s, the cost of a player i is $C_i(s_i, s_{-i}) = \sum_{f\in s_i} r_f(n_f(s))$, where $n_f(s)$ (congestion) is the number of players using facility f in strategy profile s. The Social Cost of s is: $SC(s) = \sum_{i\in[N]} C_i(s)$ and the Potential of s is: $\Phi(s) = \sum_{f\in F}\sum_{k=1}^{n_f(s)} r_f(k)$.*

Definition 2. *A **Cost Sharing Game** (CSG) is a CG where the facility cost functions $r_f(x)$ are non-increasing. Any Cost Sharing Game may also be augmented by the property of **Fair Cost Allocation** which means that the cost functions have the specific form of $r_f(x) = \frac{c_f}{x}$.*

Definition 3. *A **Network Cost Sharing Game** is a CSG, where the strategy space of each player i is the set of all possible paths between two nodes (s_i, t_i) on a directed network $G = (V, E)$. If we assume an undirected network then we have the class of **Undirected Network Cost Sharing Games**. If all players share a common sink then we have the case of a **Multicast Cost Sharing Game** either on a directed or undirected network.*

Definition 4. *A **Matroid Cost Sharing Game** is a CSG, where for each player $i \in [N]$, S_i is the set of bases of a matroid $\mathcal{M}_i = (F, l_i)$, where l_i is the set of independent sets [15]. Additionally we denote by $rk(G) = max_{i\in[N]} rk(\mathcal{M}_i)$ the rank of the game G, where $rk(\mathcal{M}_i)$ is the rank of matroid $rk(\mathcal{M}_i)$.*

Definition 5. *A **Singleton Cost Sharing Game** is a CSG, where for each player $i \in [N]$, S_i consists of singleton sets. For this class of games we will use an equivalent model that consists of: n jobs and m machines and an arbitrary bipartite graph $\mathcal{G}$ on nodes $[n] \cup [m]$. The jobs are the players of the game and their strategies is to choose one of the machines they are connected to in $\mathcal{G}$. We denote with M_j the set of neighbors of job j, i.e. the set of possible machines job j can choose from. We denote with N_i the set of neighbors of machine i, i.e. the set of jobs machine i can be picked by.*

3 Computing a Good Pure Nash Equilibrium

Matroid Cost Sharing Games is a subclass of Matroid Congestion Games, hence by Ackermann et al. [1] we have that best response dynamics converge to a PNE in at most $n^2m\ rk(G)$ steps. Thus one polynomial algorithm that gives a PNE starts from an arbitrary configuration and performs a best response in each step.

However, one interesting question is whether we can find a good quality PNE, for example the one that globally minimizes the potential function or at least a PNE with good social cost characteristics. Chekuri et al. [5] prove that computing the global potential minimizer for the class of Singleton Cost Sharing Games with cost functions of the form $f(x) = 1/x$ is NP-hard. The proof is based on a gap introducing reduction by Lund and Yannakakis [12]. We remark here that this reduction can also be used to prove that computing the socially optimal PNE is NP-hard.

Nevertheless, these hardness results do not exclude the possibility of computing a PNE with social cost comparable with the bound on the PoS produced by the Potential Method. Specifically the Potential Method works as follows: Suppose that for any profile s: $SC(s) \leq \Phi(s) \leq \alpha SC(s)$. Then if we find the global potential minimizer $\hat{s}$ and denoting with s^* the optimal, we get that: $SC(\hat{s}) \leq \Phi(\hat{s}) \leq \Phi(s^*) \leq \alpha SC(s^*)$, hence the PoS is at most α. Now we know that computing the global potential or social cost minimizer is NP-hard, however if we manage to find a PNE $\hat{s}'$ such that: $SC(\hat{s}') \leq \Phi(s^*)$, then we would have: $SC(\hat{s}') \leq \alpha SC(s)$ and we would get the same upper bound. This is the guarantee that we will achieve for the algorithms that follow.

3.1 Singleton Cost Sharing Games

In this section we present the polynomial time algorithm that computes a good PNE for the class of Singleton Cost Sharing Games. We start from Singleton Cost Sharing Games to give a clear intuition for the case of Matroid Cost Sharing Games that will be a generalization of the results presented in this section.

Singleton Cost Sharing Games can also be viewed as a Multicast Cost Sharing Game on a directed network. Given an instance of our model we create a multicast game as follows: Set a common sink t. Create a machine node v_i for each machine i and connect it with t with an edge of cost r_i. Create one source node s_j for each job j. Then for each $j \in N_i$ create an edge of cost 0 from source node s_j to machine node v_i. It could also be viewed as a Multicast Cost Sharing Game on an undirected network. Instead of setting the edge costs of edges (s_j, v_i) to 0 we set it to some huge number q. However, the Price of Anarchy and PoS bounds do not carry over in this case.

In the special case of Fair Cost Allocation, the social cost is the sum of the costs of the machines used and the optimization problem of computing a strategy profile with minimum social cost is a problem equivalent to SET COVER.

Our algorithm (Alg. 1) works as follows: Each time pick the machine that incurs the minimum player cost if it is assigned all the possible unassigned jobs and assign to that machine all possible jobs. We iterate until all jobs are assigned.

Algorithm 1. Poly-time algorithm for good PNE

Require: An instance of $G = \langle \mathcal{G} = ([n] \cup [m], E), (r_i)_{i \in [m]} \rangle$
  $\mathcal{G}^1 = \mathcal{G}$
  for $t = 1$ to m **do**
    Let $d_i^t = |N_i^t|$ be the degree of machine i in $\mathcal{G}^t$. Let $i^t = \arg\min_{i \in [m]} r_i(d_i^t)$
    For all $j \in N_{i^t}^t$ set $s_j = i^t$
    Remove nodes $N_{i^t}^t \cup i^t$ from $\mathcal{G}^t$ to obtain $\mathcal{G}^{t+1}$
  end for
  return Strategy profile s

Theorem 1. *For any instance of Singleton Cost Sharing Games, Algorithm (1) computes a PNE that is as good as the potential of the optimal allocation.*

Sketch of Proof. Suppose in the end of the algorithm, some job j wants to move from his current machine i to some i'. Assume j was assigned to i at time step t_0, i.e. $i = i^{t_0}$. Therefore it was not connected to any machine i^t for $t < t_0$. Thus i' must correspond to some machine i^{t_1} for $t_1 > t_0$. Since j was not assigned to i' it means that at t_0, the degree of i' dropped by at least 1. Moreover, at each subsequent time step the degree of machine i' can only drop. Thus $d_{i'}^{t_1} \leq d_{i'}^{t_0} - 1$. Moreover, since i was selected at t_0 it means that: $r_i(d_i^{t_0}) \leq r_{i'}(d_{i'}^{t_0}) \leq r_{i'}(d_{i'}^{t_1} + 1)$, where the last inequality holds from the fact that $d_{i'}^{t_0} \geq d_{i'}^{t_1} + 1$ and r_i are non-increasing functions. In the end of the algorithm $n_i = d_i^{t_0}$ and $n_{i'} = d_{i'}^{t_1}$, hence $r_i(n_i) \leq r_{i'}(n_{i'} + 1)$, which is a contradiction.

For the efficiency guarrantee we work with a price scheme. Consider a machine i in the optimum solution and assume d players are assigned to it in OPT. Order these players in the order that the algorithm assigns them to some other machine. When the j-th player was assigned to another machine i' we know that at least $d - j + 1$ players can still be assigned to i. Since we assign all possible players to i' and we choose the machine with the smallest possible player cost we know that j-th player pays a cost of at most $r_i(d - j + 1)$ for being assigned to i'. Iterating this for all j and for all machines in the optimum solution we get the desired result. □

For the case of Fair Cost Allocation, Algorithm (1) is exactly the greedy H_n-approximation for SET COVER. Hence we immediately get a good efficiency guarantee. In addition it is interesting to notice that the tight example for the H_n-approximation given in Example 2.5 of [19] for the greedy approximation algorithm for SET COVER is the identical analogue of the tight example for the PoS given in [2].

We also note that the above algorithm actually computes a Strong Nash Equilibrium. For Matroid Cost Sharing Games such a guarantee cannot be achieved since the example of [6] that possesses no Strong Nash Equilibrium can be easily transformed into a Matroid Game.

3.2 Matroid Cost Sharing Games

In this section we present a generalization of Algorithm (1) that computes a good PNE for the class of Matroid Cost Sharing Games.

Algorithm 2. Poly-time algorithm for good PNE in Matroid Cost sharing Games

Require: An instance of $\langle N, F, (S_i)_{i\in[N]}, (r_f)_{f\in F}\rangle$
  $\forall i \in [N] : s_i^0 = \emptyset;\ \ \forall f \in F : N_f^0 = \{i \in [N] \mid f \in l_i\},\ d_f^0 = |N_f^0|,\ t = 0$
  while $\exists f \in F : d_f^t > 0$ **do**
    $f^t = \arg\min_{f\in F} r_f(d_f^t)$
    $\forall i \in N_{f^t}^t$ set $s_i^t = s_i^{t-1} \cup \{f^t\}$. $t = t+1$
    $N_f^t = \{i \in [N] \mid s_i^{t-1} \cup \{f\} \in l_i\},\ d_f^t = |N_f^t|$
  end while
  return Strategy profile s^{t-1}

The algorithm (Alg. 2) works as follows: At each point we keep a temporary strategy for each player, starting from the empty strategy. At each iteration t we compute for each resource to how many players' strategy it could be added (d_f^t). Then we choose the resource that has minimum player cost if added to the strategy of all possible players ($\min_{f\in F} r_f(d_f^t)$) and we perform this addition. This happens until no player's strategy can be further augmented.

Assuming that checking whether some set is in l_i takes polynomial time in the size of the input, then it is clear that the above algorithm runs in polynomial time since the while loop is executed at most $n \cdot rk(G)$ times and during each time step we go over all the resources. For example the above property is true for the case where the strategy space of each player is the set of spanning trees on a set of nodes, like the ISP Network Creation Game.

Theorem 2. *For any Matroid Cost Sharing Game, Algorithm (2) computes a PNE with social cost at most the potential of the optimal allocation.*

Sketch of Proof. To prove that the resulting allocation is a PNE we use the matroid property that a base is minimum if and only if there is no $(1,1)$ exchange of a facility that results to a better strategy. Then we argue that no profitable $(1,1)$ exchange can exist in a player's strategy due to the way the algorithm works. To prove the efficiency guarantee we construct for every player a 1-1 mapping of the facilities in the algorithm's allocation and those in the optimal allocation with the following property: whenever a facility is assigned to the player then its mapping in the optimal allocation is also still an option at that time step. In this way we are able to simulate the same logic that we used in the proof of Theorem 1. □

4 Intractability of Cost Sharing Games

In this section we provide the PLS-hardness results. For a more detailed exposition of the class PLS and definitions and properties of PLS reductions the reader is redirected to the initial papers on PLS [11, 14] and to previous work on PLS hardness of congestion games [7, 1].

4.1 General Cost Sharing Games

We will prove that finding a PNE in the class of Cost Sharing Games is PLS-complete via a reduction from MAX CUT under the flip neighborhood.

Definition 6. *We say that a class of functions has the property $\mathcal{P}_1$ if for arbitrary $a > 0$ it contains a function $f(x)$ such that $f(1) = f(2) + a$.*

Theorem 3. *Computing a PNE for the class of Cost Sharing Games with a class of cost functions that has property $\mathcal{P}_1$ is PLS-complete.*

Proof. Assume an instance of MAX CUT on weighted graph $G = (V, E, (w_e)_{e\in E})$. Assume that $(i, j) \notin E \Rightarrow w_{ij} = 0$. We will create an instance of a Cost Sharing Game $\langle N, F, (S_i)_{i\in[N]}, (r_f)_{f\in F}\rangle$ such that from every PNE of the game we can construct in polynomial time a local maximum cut of the MAX CUT instance.

For each node $i \in V$ we add a player $P_i \in [N]$. We assume an ordering of the players $P_1, \ldots, P_N$. For each unordered pair of players $\{i, j\}$ $(i < j)$ we add two facilities f_{ij}^1 and f_{ij}^2 in the set F, each with cost function r_{ij}, such that $r_{ij}(1) = r_{ij}(2) + w_{ij}$, which can be achieved due to the $\mathcal{P}_1$ property.

$$s_i^A = \{f_{ji}^2 \mid j \in \{1 \ldots i-1\}\} \cup \{f_{ij}^1 \mid j \in \{i+1 \ldots N\}\}$$
$$s_i^B = \{f_{ji}^1 \mid j \in \{1 \ldots i-1\}\} \cup \{f_{ij}^2 \mid j \in \{i+1 \ldots N\}\}$$

In other words for each pair of players $\{i, j\}$ if player i has facility f_{ij}^1 in his s_i^A strategy then player j has facility f_{ij}^2 in his s_j^A strategy and correspondingly for the B strategies.

Now given any strategy profile s we consider the following partition of the initial graph: If player $P_i \in N$ is playing s_i^A then place node i in partition $V_A(s)$ and to partition $V_B(s)$ otherwise. For every node $i \in V$ denote with $w_i = \sum_{j\in V} w_{ij}$, $w(i, V_A) = \sum_{j\in V_A} w_{ij}$, $w(i, V_B) = \sum_{j\in V_B} w_{ij}$.

Given any strategy profile s the cost of player P_i for playing each strategy is:

$$C_i(s_i^A, s_{-i}) = \sum_{j\in V_A(s)} r_{ij}(1) + \sum_{j\in V_B(s)} r_{ij}(2) = w(i, V_A(s)) + \sum_{j\neq i} r_{ij}(2)$$
$$C_i(s_i^B, s_{-i}) = \sum_{j\in V_A(s)} r_{ij}(2) + \sum_{j\in V_B(s)} r_{ij}(1) = w(i, V_B(s)) + \sum_{j\neq i} r_{ij}(2)$$

Thus if s is a PNE and P_i is playing s_i^A then: $C_i(s_i^A, s_{-i}) \leq C_i(s_i^B, s_{-i}) \implies w(i, V_A(s)) \leq w(i, V_B(s))$. Hence, switching node i from partition $V_A(s)$ to $V_B(s)$ will not increase the weight of the cut. Similarly if P_i is playing s_i^B we get the opposite inequality. Therefore, for any PNE s the corresponding partition $(V_A(s), V_B(s))$ is a local maximum of the initial MAX CUT instance. $\square$

Corollary 1. *Computing a PNE for the class of Cost Sharing Games with Fair Cost Allocation is PLS-complete.*

4.2 Extending to Network Games

We will introduce k-Congestable Games. We will then notice that the game instance created in the MAX CUT reduction belongs to the class of 2-Congestable Cost Sharing Games with Fair Cost Allocation. We will then show how from any instance of a 2-Congestable Game we can create a Network Congestion Game that preserves the PNE.

Definition 7. *A k-Congestable Congestion Game is a Congestion Game where: (1) Any facility is used by at most k players. (2) The facilities on the different strategies of a player are disjoint. There is no restriction on the cost functions.*

Definition 8. *A class of cost functions has property $\mathcal{P}_2$ if for arbitrary $H > 0$ it contains a function $f(x)$ such that $\min_{k\in[1..N]} f(k) > H$ and any member of the class has bounded maximum in a finite integer range $[1..N]$.*

Theorem 4. *Given an instance of a 2-Congestable Congestion Game with cost functions from a class with property $\mathcal{P}_2$, we can create an instance of a Network Congestion Game on a directed network, where any PNE of the latter corresponds to a PNE of the former and the conversion can be computed in polynomial time. Moreover, the reduced game contains the same set of cost functions as the initial.*

It is easy to see that the game created in the MAX CUT reduction is a 2-Congestable Game and that the class of functions of the form $f(x) = c/x$ satisfy property $\mathcal{P}_2$ (see full version of the paper). Hence, we have the following corollary:

Corollary 2. *Computing a PNE in the class of Network Cost-Sharing Games with Fair Cost Allocation is PLS complete.*

An example of a reduction of MAX CUT on a three node graph to General Cost Sharing and Network Cost Sharing is depicted in Figure 1.

Now we describe for which classes of functions we can have PLS-completeness in undirected networks too (see full version for proofs).

Definition 9. *A class of functions has property $\mathcal{P}_3$ if for any $a, \epsilon > 0$ it contains a function $f(x)$ such that $f(1) = a$ and $\max_{k\in[1..N]} f(k) - \min_{k\in[1..N]} f(k) \leq \epsilon$.*

Theorem 5. *Given an instance of a 2-Congestable Congestion Game with cost functions from a class that has properties $\mathcal{P}_2$ and $\mathcal{P}_3$, we can create an instance of an Undirected Network Congestion Game, where any PNE of the latter corresponds to a PNE of the former and the conversion can be computed in polynomial time. Moreover, the reduced game contains the same set of cost functions as the initial.*

Corollary 3. *Computing a PNE in the class of Undirected Network Cost Sharing Games with functions of the form $f(x) = c_r(x)/x$, where $c_r(x)$ is a non-decreasing concave function, is PLS-complete.*

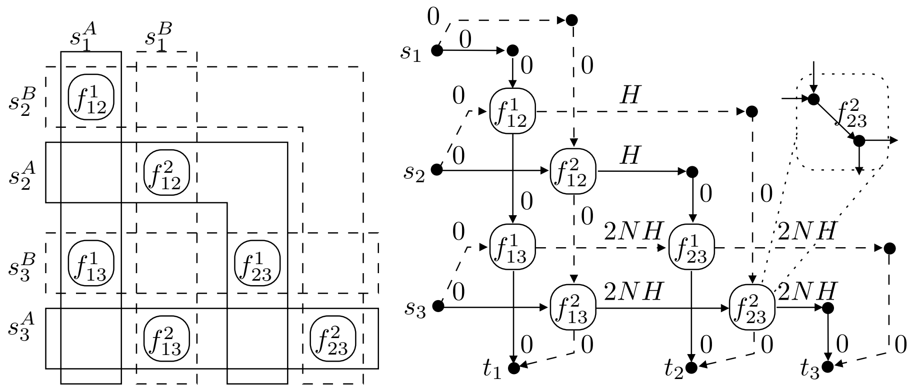


Fig. 1. Reduction from MAX CUT to Cost Sharing (left) and Network Cost Sharing (right) with Fair Cost Allocation. The facilities f_{ij}^k have cost the weight of the edge (i, j). The cost of the rest of the facilities-edges is depicted on the figure. H is a number much bigger than any cost imposed by facilities f_{ij}^k.

4.3 Tightness of PLS-Reductions

It is easy to observe that all the PLS reductions used in the previous sections are tight reductions as defined in [14]. From the initial works on PLS [11, 14] we know that tight reductions do not decrease the distance of an initial solution from a local optimum through local improvement steps. Moreover, we know that there exist instances of MAX CUT with initial configurations that have exponential distance from any local maximum. This directly implies that there exist instances of the class of games for which we prove PLS-completeness, with strategy profiles such that any sequence of best response moves needs exponential time to reach a PNE. Moreover, the tightness of our reductions shows that for the class of games we cope with it is PSPACE-complete to compute a PNE that is reachable from a specific initial strategy profile through best response moves.

5 Discussion and Further Results

Another interesting fact that might be useful in other reductions is the following:

Theorem 6. *Computing a PNE in General Congestion Games where all players have two strategies and each facility is used by at most two players can be reduced to computing a PNE of a 2-Congestable Congestion Game. If the initial game contains a cost function $r_f(x)$ then the reduced game might contain cost functions of the form $r_f(x + k)$ for arbitrary k.*

Last, we observe that our reductions from 2-congestable games also show how one can conclude PLS completeness of Undirected Network Congestion Games with linear cost functions, directly from the MAX CUT reduction of [7] without introducing 2-threshold congestion games. It is easy to observe that the Congestion Game created in the reduction of [7] is a 2-Congestable Game and linear functions is a class that satisfies properties $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$.

Acknowledgements. I would like to deeply thank Eva Tardos for the many fruitful and insightful discussions on the subject. I would also like to thank the reviewers for the helpful comments.

References

[1] Ackermann, H., Röglin, H., Vöcking, B.: On the impact of combinatorial structure on congestion games. J. ACM 55(6) (2008)
[2] Anshelevich, E., Dasgupta, A., Kleinberg, J.M., Tardos, É., Wexler, T., Roughgarden, T.: The price of stability for network design with fair cost allocation. In: FOCS, pp. 295–304. IEEE Computer Society, Los Alamitos (2004)
[3] Balcan, M.-F., Blum, A., Mansour, Y.: Circumventing the price of anarchy: Leading dynamics to good behavior. In: ICS, pp. 200–213 (2010)
[4] Charikar, M., Karloff, H.J., Mathieu, C., Naor, J., Saks, M.E.: Online multicast with egalitarian cost sharing. In: SPAA, pp. 70–76. ACM, New York (2008)
[5] Chekuri, C., Chuzhoy, J., Lewin-Eytan, L., Naor, J., Orda, A.: Non-cooperative multicast and facility location games. IEEE Journal on Selected Areas in Communications 25(6), 1193–1206 (2007)
[6] Epstein, A., Feldman, M., Mansour, Y.: Strong equilibrium in cost sharing connection games. In: EC, pp. 84–92. ACM, New York (2007)
[7] Fabrikant, A., Papadimitriou, C.H., Talwar, K.: The complexity of pure nash equilibria. In: STOC, pp. 604–612. ACM, New York (2004)
[8] Hansen, T.D., Telelis, O.: Improved bounds for facility location games with fair cost allocation. In: COCOA 2009. LNCS, vol. 5573, pp. 174–185. Springer, Heidelberg (2009)
[9] Harks, T., Hoefer, M., Klimm, M., Skopalik, A.: Computing pure nash and strong equilibria in bottleneck congestion games. In: de Berg, M., Meyer, U. (eds.) ESA 2010. LNCS, vol. 6347, pp. 29–38. Springer, Heidelberg (2010)
[10] Ieong, S., McGrew, R., Nudelman, E., Shoham, Y., Sun, Q.: Fast and compact: A simple class of congestion games. In: AAAI, pp. 489–494 (2005)
[11] Johnson, D.S., Papadimitriou, C.H., Yannakakis, M.: How easy is local search? J. Comput. Syst. Sci. 37(1), 79–100 (1988)
[12] Lund, C., Yannakakis, M.: On the hardness of approximating minimization problems. J. ACM 41(5), 960–981 (1994)
[13] Rosenthal, R.W.: A class of games possessing pure-strategy nash equilibria. International Journal of Game Theory 2(1), 65–67 (1973)
[14] Schäffer, A.A., Yannakakis, M.: Simple local search problems that are hard to solve. SIAM J. Comput. 20(1), 56–87 (1991)
[15] Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency. Matroids, Trees, Stable Sets, vol. B (2003)
[16] Skopalik, A., Vöcking, B.: Inapproximability of pure nash equilibria. In: STOC, pp. 355–364. ACM, New York (2008)
[17] Syrgkanis, V.: Equilibria in Congestion Game Models. Undergraduate Diploma Thesis (2009), http://www.cs.cornell.edu/~vasilis/thesis.pdf
[18] Tardos, E., Kleinberg, J.: Algorithm Design, ch. 12. Addison-Wesley, Reading (2005)
[19] Vazirani, V.V.: Approximation Algorithms. Springer, Heidelberg (2001)

Practical and Efficient Approximations of Nash Equilibria for Win-Lose Games Based on Graph Spectra

Haralampos Tsaknakis[1] and Paul G. Spirakis[1,2]

[1] Research Academic Computer Technology Institute (RACTI), Greece
{tsaknak,spirakis}@cti.gr
[2] Dept. of Computer Eng. and Informatics, Patras University, Patras Greece

Abstract. It is shown here that the problem of computing a Nash equilibrium for two-person games can be polynomially reduced to an indefinite quadratic programming problem involving the spectrum of the adjacency matrix of a strongly connected directed graph on n vertices, where n is the total number of players' strategies. Based on that, a new method is presented for computing approximate equilibria and it is shown that its complexity is a function of the average spectral energy of the underlying graph. The implications of the strong connectedness properties on the energy and on the complexity of the method is discussed and certain classes of graphs are described for which the method is a polynomial time approximation scheme (PTAS). The worst case complexity is bounded by a subexponential function in the total number of strategies n and a comparison is made with a previously reported method with subexponential complexity.

1 Introduction

The problem of computing an exact Nash equilibrium has been shown to be PPAD-complete even for two-player games. Furthermore, the seemingly easier problem of computing an ε-approximate equilibrium in time polynomial in $1/\varepsilon$ and n is also PPAD-complete. It turns out that there can be no fully polynomial time approximation algorithm (FPTAS) for the problem unless P = PPAD. These results are established in [5] and [6]. For the definition and insight into the complexity class PPAD and its connection with the more general Nash equilibrium problem for more than two players, the interested reader is referred to [12], [2] and [7].

In view of such complexity results, attention has been focused in the past few years on the problem of efficiently computing ε-approximate equilibria for fixed ε. However, despite considerable efforts in this direction, it has not been possible so far to achieve in polynomial time guaranteed constant approximations better than $\varepsilon = 0.3393$ for general bimatrix games and $\varepsilon = 1/4$ for win-lose games ([13] and [14]). Computing an ε-approximate equilibrium in polynomial time for any fixed $\varepsilon > 0$ (i.e. a PTAS) is still open for general problems and for win-lose problems as well.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 378–390, 2010.

This work is an effort toward the direction of obtaining a PTAS for the game equilibrium problem. We focus on win-lose game problems (where each entry of the payoff matrices is either 0 or 1) and we show that it is possible to represent each such problem by a strongly connected directed graph (digraph) on n vertices, where n is the total number of players' strategies. We believe that the graph representation of win-lose games provides valuable insight into the complexity of the equilibrium computation problem, not only for win-lose games but for general games as well.

On the other hand, we show that it is possible to formulate the search for an equilibrium as an indefinite quadratic optimization problem involving the eigenvalues and eigenvectors of the adjacency matrix of the graph which is the undirected version of the underlying digraph. An algorithm for obtaining an approximate equilibrium is presented, based on searching among stationary points of a regret function (as described in Section 3), and its complexity is derived as a function of the energy of the graph. Although the worst case complexity that we can prove so far (based on the results presented here) is subexponential in n, there are certain important classes of graphs discussed here for which the algorithm is a PTAS.

Considering win-lose games is no loss of generality in terms of the complexity of computing exact Nash equilibria. In [1] it is proven that there is a polynomial time reduction from a two-player game with rational payoffs to a win-lose game. However, in terms of approximate equilibria, it is not clear that an ε-approximation for a win-lose problem implies a similar quality of approximation for the original problem. Nevertheless, the investigation of equilibrium approximation for win-lose games can lead to results that are applicable to general games as well. In fact, the general game problem can be represented by considering weighted graphs (with rational weights) following techniques similar to the ones used to represent win-lose games by ordinary graphs. Also, the method presented here for approximating an equilibrium is general enough to include all kinds of payoff matrices and graph structures and the complexity bound that is derived as a function of the energy of the graph is valid for weighted graphs as well. The only difference between general games and win-lose ones is that in the latter case we have better insight and we can make more clear cut structural characterizations of the underlying graphs.

We adopt the following notation throughout:

- For any positive integer k, $[k]$ denotes the set of integers from 1 to k. Also, e denotes the column vector of appropriate dimension having all its entries equal to 1.
- $\Delta_k = \{u : u \in \mathcal{R}^k, u \geq 0, e^\tau u = 1\}$ is the set of k-dimensional probability vectors (superscript τ denotes transpose).
- $supp(u) = \{i \in [k] : u_i \neq 0\}$ denotes the subset of integers (indices) in $[k]$ constituting the support of a vector $u \in \mathcal{R}^k$.
- $suppmax(u) = \{i \in [k] : u_i \geq u_j \ \forall j \in [k]\}$ is the subset of indices in $[k]$ for which the corresponding entries of a vector $u \in \mathcal{R}^k$ are equal to the maximum entry value of u.

– $\max(u) = \{u_i : u_i \geq u_j, \text{for all} j\}$ is the value of the maximum entry of a vector u.
– $\max_S(u) = \{u_i, i \in S : u_i \geq u_j, \text{for all } j \in S\}$ is the value of the maximum entry of a vector u within an index subset $S \subset [k]$.
– The symbol $|.|$ will denote absolute value of a number or the cardinality of a set depending on context.

2 Graph Representation

For any two-person win-lose game with row and column payoff matrices R and C respectively (where R and C have the same dimensions and each entry is either 0 or 1), the problem of computing a Nash equilibrium can be reduced to a problem of computing a symmetric equilibrium of a symmetric game of the form (A, A^τ), where A is the square matrix defined as:

$$A = \begin{bmatrix} 0 & C^\tau \\ R & 0 \end{bmatrix}$$

Actually, this is true not just for win-lose games but for any game (e.g., see [2], Part I, Ch. 2) with appropriately normalized payoff matrices in the interval $[0, 1]$ (which can always be done without loss of generality). Furthermore, this result can be extended (for the general case) to the problem of computing approximate equilibria as well (e.g., see [16]), in the sense that an ε-approximation of the latter symmetric problem provides an $O(\epsilon)$-approximation for the initial problem.

It can be easily verified that the problem of finding a symmetric equilibrium of the symmetric game (A, A^τ) is equivalent to finding a probability vector $x \in \Delta_n$ such that $supp(x) \subseteq suppmax(Ax)$. Alternatively, it is equivalent to finding a probability vector x such that $f_A(x) = 0$, where, the function $f_A(x)$ maps Δ_n into $[0, 1]$ and is defined as follows:

$$f_A(x) = \max(Ax) - x^\tau Ax$$

In fact, the function $f_A(.)$ (we call it **regret function**) provides a measure of closeness to an equilibrium of any given probability vector. A probability vector x for which $f_A(x) \leq \varepsilon$ is an ε-approximate equilibrium.

Let n be a positive integer and let A be a square $n \times n$ matrix defined as above for a win-lose game. Obviously, the diagonal entries of such a matrix A are all 0 and each off-diagonal entry is either 0 or 1. In order to avoid trivial cases, such as the existence of pure Nash equilibria (in which case our search for an equilibrium ends successfully), it is necessary to impose the requirement that for any pair of pure strategies of the row and column players (say (k, l)) the corresponding entries of the matrices R and C (i.e. R_{kl} and C_{kl}) are not both equal to 1. This immediately implies that for any $i \in [n]$ and $j \in [n]$ the entries A_{ij} and A_{ji} of the matrix A cannot be both equal to 1. Under such a requirement, it can be easily seen that the matrix A can be considered as the adjacency matrix of a simple directed graph, or digraph (no loops and no repeated arcs) with n nodes representing the total number of pure strategies of both players, and arcs being the ordered pairs of nodes (strategies) (i, j) for which the corresponding payoff A_{ij} is equal to 1. Notice that the symmetric matrix $A + A^{\tau}$ is the adjacency matrix of the undirected version of the digraph A (i.e. the graph obtained by keeping the same edges but ignoring their directions). Also, notice that the graph constructed this way is a bipartite graph. However, the bipartite nature of the graph is not crucial for the derivations and results presented here.

We denote by $D = (V, E)$ the underlying directed graph for a win-lose game, where, V is the set of nodes with cardinality $|V| = n$ and E is the set of ordered pairs of nodes representing the arcs. The undirected version of the graph is denoted by $UD = (V, UE)$, where, UE is the set of the corresponding unordered pairs of nodes. In the sequel, a game will be characterized either by the adjacency matrix A of the digraph or by the digraph itself.

Additional requirements should be imposed on the digraph with adjacency matrix A to avoid other trivial cases (from the standpoint of computing a Nash equilibrium). In fact, each node should have at least one incoming arc and at least one outgoing arc. Indeed, if a node has no outgoing arcs (i.e. arcs leaving the node), then, the corresponding row of A is an all 0's vector and the node can be omitted reducing thus the size of the problem. On the other hand, if a node has no incoming arcs (i.e. arcs entering the node), then, the corresponding column of A is an all 0's vector and a trivial equilibrium can be found by considering that node alone.

All the above requirements on the structure of the digraph representing meaningful nontrivial game problems are special cases of a more general category of digraphs, namely, strongly connected digraphs. Theorem 1 below essentially states that if we are looking for a Nash equilibrium we can restrict attention, without loss of generality, to win-lose games whose underlying digraph is strongly connected. Before that, we briefly summarize some standard definitions and results from the theory of graphs (e.g., see [3]).

A digraph D is called simply connected if its undirected version UD is connected, i.e. if there is a path between any two distinct nodes of UD. A digraph is called strongly connected if there is a directed path between any two distinct nodes. A standard result here is that any simply connected digraph can be decomposed into node-disjoint strongly connected subgraphs (components), where, the connections among the strongly connected components form an acyclic directed graph. In such a decomposition, among all the strongly connected components (if there are more than one) there is at least one component that is a source (i.e. there are only outgoing arcs from it and no incoming arcs), and at least one component that is a sink (i.e. there are only incoming arcs into it and no outgoing arcs). The decomposition of any simply connected digraph into strongly connected components can be performed in polynomial time. Also, if the digraph is not simply connected (i.e. its undirected version is not connected), then, it can be decomposed into simply connected components in polynomial time.

Some additional considerations are necessary: We say that a node i of a digraph $D = (V, E)$ is redundant if the set of its outgoing neighbors (the tails of the arcs leaving i) is a subset (proper or not) of the set of outgoing neighbors of some other node of the digraph. It is not difficult to see that such a node can be removed (reducing thus the size of the game) without affecting the overall problem of searching for an equilibrium. In fact, if such a node is removed, any Nash equilibrium of the resulting reduced (by one) game is also a Nash equilibrium of the initial game. It is evident that the process of identifying and removing redundant nodes, one at a time, until no further reduction is possible, is a polynomial time process.

We now express the following theorem:

Theorem 1. *Let $D = (V, E)$ be the underlying digraph of a win-lose game and let A be its adjacency matrix. Let $D_s = (V_s, E_s)$ be a strongly connected component of D that is a source subgraph with adjacency matrix A_s. Then, any Nash equilibrium of the subgame with adjacency matrix A_s (and underlying graph the strongly connected subgraph D_s) is also a Nash equilibrium of the initial game A.*

Proof. The adjacency matrix A_s of the subgraph D_s is a principal submatrix of A and has dimensions $|V_s| \times |V_s|$. Since the subgraph D_s is a source it has no incoming arcs, hence the matrix A can be written in block triangular form (via an appropriate permutation of the nodes) as follows:

$$A = \begin{bmatrix} A_s & F \\ 0 & H \end{bmatrix}$$

where, H is some $(n-|V_s|)\times(n-|V_s|)$ square matrix and F is some $|V_s|\times(n-|V_s|)$ matrix, both representing the rest of the arcs of the adjacency matrix A. Now, let x be any probability vector such that $supp(x) \subseteq V_s$. Then, using the representation of A above, we should have $max(Ax) = max_{V_s}(Ax) = max_{V_s}(A_s x)$. If, in addition, x is a Nash equilibrium of the subgame with adjacency matrix A_s, then we should have $max_{V_s}(A_s x) = x^\tau A_s x = x^\tau Ax$. Therefore $max(Ax) = x^\tau Ax$ which implies that x is also a Nash equilibrium of the initial game A. □

We conclude this section by summarizing the results in the theorem below.

Theorem 2. *The problem of computing a Nash equilibrium for any bimatrix game can be polynomially reduced to a problem of computing a minimum of a function of the form $f_A(x) = \max(Ax) - x^\tau Ax$ over $x \in \Delta_n$, where A is the adjacency matrix of a strongly connected directed graph containing no redundant nodes.*

3 Stationary Points of Descent Algorithms

Since the regret function $f_A(x)$ is (in general) non-convex, any gradient descent procedure that seeks to minimize it will converge to a stationary point that is not in general a global minimum (recall that any global minimum is a Nash equilibrium for which $f_A(x) = 0$). Stationary points and gradient descent algorithms related to game equilibria have been studied in [13] and [14] as well as in [15]. Following the analysis, results and terminology presented there and adapted here for the case of symmetric games (which makes things simpler in terms of notation), we formally define stationary points of the function $f_A(x)$ and summarize their properties in definitions 1, 2 and 3 and theorems 3, 4, 5 and 6 in this section. Proofs of the results summarized in theorems 3, 4 and 5 can be found in the above mentioned references (equivalent versions of them), while theorem 6 and its corollary are direct consequences of the previous ones.

Definition 1. *The **gradient** $D_A(x', x)$ of the function $f_A(x) = \max(Ax) - x^\tau Ax$ at $x \in \Delta_n$ along a direction $x' \in \Delta_n$ is defined as follows:*

$$D_A(x', x) \equiv \lim_{\epsilon \to 0} \frac{1}{\epsilon}(f_A((1-\epsilon)x + \epsilon x') - f_A(x))$$

Theorem 3. *The limit in the above definition always exists for any $x \in \Delta_n$ and $x' \in \Delta_n$. Furthermore, for given x it defines a convex function $D_A(x', x)$ of x' involving the $\max(.)$ of an affine function of x' and given by the following equation: $D_A(x', x) = \max_{suppmax(Ax)}(Ax') - x^\tau Ax' - (x')^\tau Ax + x^\tau Ax - f_A(x)$.*

Definition 2. *A probability vector $x^\star \in \Delta_n$ is called a **stationary point** of the function $f_A(x) = \max(Ax) - x^\tau Ax$, if $D_A(x, x^\star) \geq 0, \forall x \in \Delta_n$.*

Theorem 4. *A stationary point always exists and can be approximated as closely as desired in polynomial time through an iterative descent algorithm applied to the function $f_A(x)$. The algorithm can start from an arbitrary $x_0 \in \Delta_n$ and involves solving a linear programming problem of the form $\min_{x \in \Delta_n}(D_A(x, x_k))$ at each step k. Furthermore, every Nash equilibrium is a stationary point.*

Theorem 5. *Let $x^\star \in \Delta_n$ be a stationary point and define $S(x^\star) \equiv suppmax$ $(Ax^\star)$. Then, the following relationship (called here **stationarity condition**) holds: $\max_{S(x^\star)}(Ax) - (x^\star)^\tau Ax - x^\tau Ax^\star + (x^\star)^\tau Ax^\star - f_A(x^\star) \geq 0, \forall x \in \Delta_n$, or, equivalently:*

$$f_A(x^\star) - f_A(x) + (\max(Ax) - \max_{S(x^\star)}(Ax)) \leq (x - x^\star)^\tau A(x - x^\star), \forall x \in \Delta_n \quad (1)$$

Definition 3. *For a given convex set $\mathcal{K} \subseteq \mathcal{R}^n$, a probability vector $x^\star \in \mathcal{K} \cap \Delta_n$ is called a **constrained stationary point** of the function $f_A(x) = \max(Ax) - x^\tau Ax$, if $D_A(x, x^\star) \geq 0, \forall x \in \mathcal{K} \cap \Delta_n$.*

A constrained stationary point can be obtained efficiently just like an unconstrained one by imposing the additional (convex) constraints $x \in \mathcal{K}$ at each step k of the descent process: $\min_{x \in \mathcal{K} \cap \Delta_n}(D_A(x, x_k))$. Notice that when dealing with constrained stationary points $x^\star$ in some convex set $\mathcal{K}$, the stationarity condition in equation (1) should be modified to hold $\forall x \in \mathcal{K} \cap \Delta_n$.

A direct consequence of the previous relationships involving stationary points (constrained or not) is summarized in the theorem below.

Theorem 6. *For any two stationary points $x_1^\star, x_2^\star$, the following relationship holds:*

$$|f_A(x_1^\star) - f_A(x_2^\star)| \leq (x_1^\star - x_2^\star)^\tau A(x_1^\star - x_2^\star)$$

As a corollary of the last theorem, if one of the two stationary points, say $x_2^\star$, is a Nash equilibrium, then:

$$f_A(x_1^\star) \leq (x_1^\star - x_2^\star)^\tau A(x_1^\star - x_2^\star) \tag{2}$$

4 Spectral Representation

Consider the spectral representation of the matrix $A + A^\tau$. Since this matrix is symmetric, all its eigenvalues are real and the eigenvectors are mutually orthogonal. Let $\lambda_1, \lambda_2, \ldots, \lambda_m$ be the m positive eigenvalues of $A + A^\tau$ $(m < n)$ and $-|\lambda_{m+1}|, -|\lambda_{m+2}|, \ldots, -|\lambda_n|$ the non-positive ones. Let $z_i, i = 1, 2, \ldots, n$ be the corresponding normalized eigenvectors satisfying $\|z_i\| = 1$ for all $i \in [n]$ and $z_i^\tau z_j = 0$ for all i, j such that $i \neq j$. By $\|.\|$ we denote the usual Euclidean L_2 norm. Assume that the eigenvalues are indexed in descending order, i.e. $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$.

Consider the representation of $A + A^\tau$ in terms of its eigenvalues and eigenvectors: $A + A^\tau = \sum_{i=1}^m \lambda_i z_i z_i^\tau - \sum_{j=m+1}^n |\lambda_j| z_j z_j^\tau$. Then, the function $f_A(x)$ can be written as follows:

$$f_A(x) = \max(Ax) + \frac{1}{2} \sum_{j=m+1}^{n} |\lambda_j| (z_j^\tau x)^2 - \frac{1}{2} \sum_{i=1}^{m} \lambda_i (z_i^\tau x)^2 \tag{3}$$

The complexity of finding a solution to the problem $\min_{x \in \Delta_n} f_A(x)$ is due exclusively to the last term of the above equation that involves the projection of the probability space Δ_n on the m-dimensional linear subspace spanned by the eigenvectors $z_i, i = 1, 2, \ldots, m$ that correspond to the positive eigenvalues of the matrix $A + A^\tau$.

The assumptions on matrix A expressed in Section 2 (as the adjacency matrix of a strongly connected digraph) have some implications on the spectrum of $A + A^\tau$. In particular, since $A + A^\tau$ is a non-negative irreducible matrix (because A is non-negative and irreducible), by the Perron-Frobenius theorem (e.g. see [4]) we deduce that the largest eigenvalue λ_1 has maximum absolute value among all eigenvalues (i.e. $\lambda_1 \geq max_{i \in [n]} |\lambda_i|$), it has multiplicity 1 and, also, all the entries of the corresponding eigenvector z_1 are strictly positive. Therefore, any other eigenvector (which is orthogonal to z_1) should have at least one positive entry and at least one negative entry as well.

In order to approximate an equilibrium, we will take advantage of the properties of stationary points, in particular equation (2), as well as the properties of the eigenvalues and eigenvectors of the spectral representation of the regret function $f_A(x)$.

5 Approximating an Equilibrium

Consider the linear m-dimensional subspace spanned by the eigenvectors $z_i, i = 1, 2, \ldots, m$ that correspond to the positive eigenvalues and the metric $d(.,.)$: $d^2(a,b) = \sum_{i=1}^{m} \lambda_i (z_i^\tau (a-b))^2$ defined on this subspace. Consider the orthogonal projection of the feasible region Δ_n on this subspace and denote it by $P_m(\Delta_n)$. Also, let $P_m(x)$ denote the projection on this subspace of any $x \in \Delta_n$. It can be easily verified that $P_m(\Delta_n)$ is the convex hull of the projections of all the n vertices of Δ_n on the m-dimensional subspace. In general, the number of vertices of the convex hull should be less than or equal to n. Notice that the projection of any probability vector in Δ_n on this subspace is an m-dimensional vector that can be expressed as a convex combination of at most $m+1$ vertices of the convex hull (by Caratheodory's theorem). Also, notice that the vertices of the convex hull can all be computed in polynomial time.

Let ε be a positive approximation parameter such that $1/\varepsilon < m$. Let us consider a set of regions in $P_m(\Delta_n)$ each consisting of convex combinations of no more that $1/\varepsilon$ vertices of $P_m(\Delta_n)$. Since each vertex is the projection of some vertex of Δ_n, the set of such regions consist of the projections of all n-dimensional probability vectors with support no more than $1/\varepsilon$. The total number of such subsets of vertices of $P_m(\Delta_n)$ is $\leq n^{1/\varepsilon}$ and so is the total number of the corresponding regions. Let us denote the set of all such regions by $L(\varepsilon)$. The crucial question is how well does $L(\varepsilon)$ approximate the entire $P_m(\Delta_n)$, i.e. what is the largest distance of a point in $P_m(\Delta_n)$ from $L(\varepsilon)$ with respect to the metric $d(.,.)$. In regard to the latter question we can express the following theorem:

Theorem 7. *For any $y \in \Delta_n$, the closest (with respect to the metric $d(.,.)$) point $x \in \Delta_n$ whose projection $P_m(x)$ belongs to $L(\varepsilon)$, satisfies the relationship: $d^2(P_m(x), P_m(y)) \leq \varepsilon\xi(m)$, where, $\xi(m) = \sum_{i=1}^{m} \lambda_i / n$.*

Proof. Consider the matrix $A_+ = \sum_{i=1}^{m} \lambda_i z_i z_i^\tau$. This is a nonnegative definite symmetric $n \times n$ matrix with rank m which represents the positive part of the spectrum of $A + A^\tau$. For any given $y \in \Delta_n$ we should have the relationship:

$$d^2(P_m(x), P_m(y)) = (x-y)^\tau A_+ (x-y)$$

Let the scalars $\bar{z}_i$ for $i = 1, 2, \ldots, n$ be defined as $\bar{z}_i = z_i^\tau y$. Define a new symmetric nonnegative definite matrix $A'_+ = \sum_{i=1}^{m} \lambda_i (z_i - \bar{z}_i e)(z_i - \bar{z}_i e)^\tau = (I - ey^\tau) A_+ (I - ye^\tau)$ (e is the all 1's vector). Then, since x and y are probability vectors we can write the previous equation as:

$$d^2(P_m(x), P_m(y)) = x^\tau A'_+ x$$

Let $\mu_1, \mu_2, \ldots$ be the eigenvalues of A'_+. By construction, the sum of μ_i's (that is to say the trace of the matrix A'_+) should be $\leq \sum_{i=1}^{m} \lambda_i$ (the trace of matrix A_+). Notice that the set $supp(x)$ can be any subset of size $|supp(x)| \leq 1/\varepsilon$. So, the minimum of $d^2(P_m(x), P_m(y))$ with respect to x is over all principal submatrices of A'_+ of size $1/\varepsilon \times 1/\varepsilon$. Let $S(\varepsilon)$ be a subset of indices in $[n]$ defining such a submatrix and let $G(\varepsilon)$ be the submatrix itself. Then, $d^2(P_m(x), P_m(y)) = x^\tau G(\varepsilon)x$. It can be verified that, given an $S(\varepsilon)$, the minimum of the latter expression with respect to $x \in \Delta_n$ with $supp(x) \subset S(\varepsilon)$ is given by an expression of the form $1/e^\tau G_\varepsilon^{-1} e$, where, e here is the all 1's vector with support $S(\varepsilon)$ and G_ε^{-1} is the inverse of a principal submatrix of A'_+ of size $1/\varepsilon \times 1/\varepsilon$ if it exists, or it can be replaced by the pseudo inverse (generalized inverse) without loss of generality. It turns out that $d^2(P_m(x), P_m(y))$ can be bounded by an expression of the form $1/\sum_{i\in S(\varepsilon)} \frac{1}{\mu'_i}$, where, $\mu'_i, i \in S(\varepsilon)$ are the eigenvalues of the submatrix $G(\varepsilon)$. Using the harmonic-arithmetic mean inequality, the latter expression is bounded from above by $\sum_{i\in S(\varepsilon)} \mu'_i/|S(\varepsilon)|^2 = \varepsilon \sum_{i\in S(\varepsilon)} \mu'_i/|S(\varepsilon)| = \varepsilon tr(G_\varepsilon)/|S(\varepsilon)|$. Since all submatrices G_ε of size $1/\varepsilon \times 1/\varepsilon$ are considered, there is one whose average trace is minimum, which implies $tr(G_\varepsilon)/|S(\varepsilon)| \le tr(A'_+)/n$. So, the bound becomes:

$$d^2(P_m(x), P_m(y)) \le \varepsilon tr(A'_+)/n \le \varepsilon \sum_{i=1}^{m} \lambda_i/n = \varepsilon\xi(m)$$

Finally, the claim of the theorem follows from the last relationship. □

Based on the above theorem, we have the following result:

Theorem 8. *Consider a game whose underlying digraph has adjacency matrix A. Then, for any $\varepsilon > 0$, there is an algorithm to find an ε-approximate Nash equilibrium in time $n^{\xi(m)/\varepsilon}$, where, $\xi(m) = \sum_{i=1}^{m} \lambda_i/n$ and $\lambda_1, \lambda_2, \ldots, \lambda_m$ are the positive eigenvalues of $A + A^\tau$.*

Proof. Since all points y in $P_m(\Delta_n)$ can be covered by balls of the form $d^2(x, y) \le \varepsilon\xi(m)$, for $x \in \Delta_n$ such that $P_m(x) \in L(\varepsilon)$, we can consider all $n^{1/\varepsilon}$ points of $L(\varepsilon)$ as starting points for the descent algorithm and compute constrained stationary points within each such ball. One of them will be $\varepsilon\xi(m)$-close to a Nash equilibrium.

Also, since the parameter ε can be chosen arbitrarily, one can choose $\varepsilon/\xi(m)$ in its place to get an ε-approximation in $n^{\xi(m)/\varepsilon}$ time. □

As an immediate consequence of the above theorems, we can express the following:

Theorem 9. *There is a PTAS for a class of games for which the positive eigenvalues satisfy the relationship $\sum_{i=1}^{m} \lambda_i/n =$ constant.*

A general upper bound for $\xi(m)$ for all instances of games and corresponding graphs is given in the theorem below.

Theorem 10. *The following relationship holds:*

$$\xi(m) = \sum_{i=1}^{m} \lambda_i/n \le \sqrt{m} \tag{4}$$

Proof. Since the λ_i's are the eigenvalues of the adjacency matrix $A + A^\tau$ of the undirected graph, the sum of squares of the λ_i's is equal to the trace of the matrix $(A + A^\tau)^2$ which is equal to the total number of walks of length 2 in the graph, i.e. $2|E|$, where $|E|$ is the number of edges. Using this fact and standard inequalities we obtain the series of relationships: $(\sum_{i=1}^m \lambda_i)^2 \leq m(\sum_{i=1}^m \lambda_i^2) \leq m(\sum_{i=1}^n \lambda_i^2) = 2m|E|$. Finally, in view of the fact that $|E| \leq n(n-1)/2$, we obtain:

$$\sum_{i=1}^{m} \lambda_i/n \leq \sqrt{m}\frac{\sqrt{2|E|}}{n} \leq \sqrt{m}$$

□

As a result of the last theorem, an ε-approximate equilibrium can be computed in time bounded by $n^{\sqrt{m}/\varepsilon}$, where m is the number of positive eigenvalues of the matrix $A + A^\tau$. Notice that the parameter $\xi(m)$ is related to the energy of the undirected version of the graph A. The **energy** of a graph, say $\Xi(A)$, is defined as the sum of the absolute values of all eigenvalues of the adjacency matrix $A + A^\tau$ (e.g., see [11]). Since $A + A^\tau$ has zero trace, the energy of the graph is twice the sum of the positive eigenvalues, hence, $\xi(m) = \Xi(A)/(2n)$, i.e. it is proportional to some kind of an "average energy" of the graph per node.

6 Special Cases

In general, for all game equilibrium problems for which the underlying graphs have energy of the order of $O(n)$, or equivalently the average energy is a constant, the approach presented here is a PTAS for computing an equilibrium. In addition, there are other categories of graphs whose energy may or may not be $O(n)$ but for which, nevertheless, our methodology is either a PTAS or polynomial. The latter categories include graphs for which a stationary point is always a Nash equilibrium (hence a NE can be computed in polynomial time) and a sufficient condition for that to happen is that the matrix A is "almost negative definite" as defined below:

Definition 4. *For an n-node graph, its $n \times n$ adjacency matrix A is called almost negative definite if $u^\tau Au \leq 0$ for every $u \in \mathcal{R}^n$ such that $e^\tau u = 0$ where e is the all ones vector.*

The most notable examples of almost negative definite adjacency matrices appear in constant sum games for which the underlying graphs are fully connected. Also, any graph derived from a fully connected one by removing any number of node-disjoint links, or, more generally, by removing any number of node-disjoint cliques, has an almost negative definite adjacency matrix. These statements can be easily verified by manipulating quadratic expressions of the form $u^\tau Au$ for vectors u whose sum of entries is equal to 0.

In addition to the above cases, we give below an indicative list of special classes of games for which our methods are PTAS or polynomial. The characterizations are given in terms of the underlying graphs (directed or undirected version).

In all cases, the underlying digraph is assumed to be strongly connected (or, equivalently, the underlying undirected graph is connected and has no bridge) with no redundant nodes.

- Sparse graphs. The number of arcs of such graphs is (by definition) $O(n)$, hence the sum of the positive eigenvalues of the adjacency matrix is also $O(n)$ which implies that the average energy is a constant and according to Theorem 9 we have PTAS.
- Graphs consisting of a fixed number of arbitrarily intersecting cycles and paths. In such graphs the maximum degree is fixed, therefore the number of arcs is $O(n)$ and we have a sparse graph as above.
- Power law degree distribution graphs with exponent $\beta > 1$. For such graphs the sum of the degrees of the nodes is given by an expression of the form $O(n) \sum_{i \geq 1} i^{-\beta}$ which is bounded by $O(n)$, hence we have a sparse graph.
- Highly connected regular graphs with total degree $n-k$ for some fixed integer k. Indeed, for such graphs the largest eigenvalue λ_1 is equal to $n-k$ and the sum of the two largest eigenvalues $\lambda_1 + \lambda_2$ is $< n$ (e.g. see [8]). Therefore, λ_2 is bounded by a constant which implies that the average energy of the graph is also bounded by a constant, so by Theorem 9 we have PTAS.
- Fixed rank games, where, the adjacency matrix $A + A^\tau$ of the underlying (undirected) graph has fixed rank.

Fixed rank games are special cases of games for which the number m of positive eigenvalues is fixed (hence the average energy of the underlying graph is fixed). Indeed, the rank of $A + A^\tau$ is equal to the number of non-zero eigenvalues and m is obviously bounded from above by the rank. Such fixed rank games have been studied in [9] where it is shown that an ε-approximate equilibrium can be found in polynomial time without, however, giving a formula for the complexity bound as a function of the rank.

More special cases arise when the matrix A of the underlying directed graph has fixed rank (which implies that $A + A^\tau$ has fixed rank but the reverse statement is not in general true). In such special cases, as shown in [17] where a similar problem is studied involving Arrow-Debreu-Leontief equilibria, there is a strongly polynomial time algorithm to compute an exact equilibrium (not just an approximate one) and a formula for the complexity is also given. A similar remark is made in [10] for bimatrix games where it is shown that low rank implies small support exact equilibria.

7 Comparison with an Existing Approximation Result

A comparison is made with an existing subexponential scheme presented in [10] which is based on probabilistic arguments to prove the existence of an algorithm to find an ε-approximate equilibrium in time $n^{O(\ln n/\varepsilon^2)}$. Considering as performance indicators the two parameters representing the approximation to a Nash equilibrium and the size of the game (i.e. the approximation parameter ε and the number of strategies n respectively), there is no uniform comparison between the complexity bound of the latter method and the one presented here: For a given ε, the result in [10] outperforms the result we present here for arbitrarily large n, larger than a threshold $n_0(\varepsilon)$. On the other hand, for a given n our result outperforms the result in [10] for arbitrarily small ε, smaller than a threshold $\varepsilon_0(n) = n_0^{-1}(n)$. By taking into account also the constants involved in the exponents of the complexity bounds of the two methods (for the former method the constant multiplying $\ln n/\varepsilon^2$ is 12 and for the method we present here the corresponding constant is less than 1), the threshold curve is given approximately by the equation $\varepsilon \approx 12 \ln n/\sqrt{n}$.

It turns out that even for modest values of ε, the values of the threshold n_0 (above which the method of [10] outperforms our method) are simply too large, large enough to render any subexponential scheme totally unrealistic anyway. Indicatively, for $\varepsilon = 1/3$ we have $n_0(1/3) \approx 2 \times 10^5$ and for $\varepsilon = 0.15$ we have $n_0(0.15) \approx 1.2 \times 10^6$ and this threshold increases very fast when ε becomes even smaller. So, the method presented here is more efficient than the one presented in [10] for all practical purposes.

8 Discussion and Future Work

There are some issues whose further investigation along the lines presented here could lead to improved results. One problem is how to improve the grid used to approximate the space of probability vectors. Another problem is how to improve the $\sqrt{m}$ bound on the average spectral energy by exploring the connectivity of the underlying graph and the possibility of further reductions of the equilibrium problem. The following two problems appear mostly relevant to further research for efficient equilibrium computation algorithms in the framework of the optimization approach:

1. Given a simplex $\Delta(t_0, t_1, \ldots, t_n)$ in $\mathcal{R}^n$ defined by $n+1$ points (vertices) $t_0, t_1, \ldots, t_n$ and a positive δ, what is the minimum size of a grid of points in $\mathcal{R}^n$ (as a function of $t_0, t_1, \ldots, t_n$ and δ) such that every point of the simplex has Euclidean distance from some point of the grid no more than δ? How complex is it to compute such grid points?

2. In addition to the reductions of the general bimatrix game equilibrium problem presented here, is it possible to obtain further (polynomial) reduction to a win-lose game whose underlying graph has fixed average spectral energy?

References

1. Abbot, T., Kane, D., Valiant, P.: On the Complexity of Two-Player Win-Lose Games. In: FOCS 2005 (2005)
2. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, Cambridge (2007)
3. Bang-Jensen, J., Gutin, G.: Digraphs Theory, Algorithms and Applications. Springer, Heidelberg (2007)
4. Bollobas, B.: Modern Graph Theory. Springer, New York (1998)

5. Chen, X., Deng, X., Teng, S.: Computing Nash Equilibria: Approximation and smoothed complexity. In: Proc. of the 47th IEEE Symp. on Foundations of Comp. Sci (FOCS 2006), pp. 603–612. IEEE Press, Sci (FOCS (2006)
6. Chen, X., Deng, X.: Settling the complexity of 2-player Nash equilibrium. In: Proc. of the 47th IEEE Symp. on Foundations of Comp. Sci. (FOCS 2006), pp. 261–272. IEEE Computer Society Press, Los Alamitos (2006)
7. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The Complexity of Computing a Nash Equilibrium. In: Proceedings of STOC (2006)
8. Ebrahimi, J., Mohar, B., Nikiforov, V., Sheikn, A.: On the sum of two largest eigenvalues of a symmetric matrix. Linear Algebra and its Applications 429, 2781–2787 (2008)
9. Kannan, R., Theobald, T.: Games of fixed rank: A hierarchy of bimatrix games. In: Proc. Symposium on Discrete Algorithms, New Orleans, LA (2007)
10. Lipton, R.J., Markakis, E., Mehta, A.: Playing large games using simple strategies. In: EC 2003: Proceedings of the 4th ACM Conference on Electronic Commerce, pp. 36–41 (2003)
11. Nikiforov, V.: The energy of graphs and matrices. Journal of Mathematical Analysis and Applications 326, 1472–1475 (2007)
12. Papadimitriou, C.H.: On the complexity of the parity argument and other inefficient proofs of existence. J. Comput. Syst. Sci. 48(3), 498–532 (1994)
13. Tsaknakis, H., Spirakis, P.G.: An Optimization Approach for Approximate Nash Equilibria. In: Deng, X., Graham, F.C. (eds.) WINE 2007. LNCS, vol. 4858, pp. 42–56. Springer, Heidelberg (2007); Also, in ECCC TR07-067, 2nd revision (2007)
14. Tsaknakis, H., Spirakis, P.G.: An Optimization Approach for Approximate Nash Equilibria. Internet Mathematics 5(4), 363–382 (2008)
15. Tsaknakis, H., Spirakis, P.G., Kanoulas, D.: Performance Evaluation of a Descent Algorithm for Bi-matrix Games. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 222–230. Springer, Heidelberg (2008)
16. Tsaknakis, H., Spirakis, P.G.: A Graph Spectral Approach for Computing Approximate Nash Equilibria. In: Electronic Colloquium on Computational Complexity (ECCC), TR09-096 (2009)
17. Ye, Y.: Computational Economy Equilibrium and its Application: Progresses on computing Arrow-Debreu-Leontief Competitive Equilibria. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, p. 14. Springer, Heidelberg (2008) (invited talk)

Market Communication in Production Economies

Christopher A. Wilkens*

Computer Science Division,
University of California at Berkeley, CA, 94720 USA
cwilkens@cs.berkeley.edu

Abstract. We study the information content of equilibrium prices using the market communication model of Deng, Papadimitriou, and Safra [4]. We show that, in the worst case, communicating an exact equilibrium in a production economy requires a number of bits that is a quadratic polynomial in the number of goods, the number of agents, the number of firms, and the number of bits used to represent an endowment.

1 Introduction

In the European Union, prices are typically expressed in whole-Euro amounts (or as "nice" decimals when they are small). In contrast, buyers and sellers in the United states cling to every penny and advertise prices to the $\frac{1}{100}$-th of a dollar. Does such accuracy serve a computational purpose? We study this question in the case of market equilibrium: how many bits of information must prices express in order to ensure that the economy achieves equilibrium?

The *market communication* model of Deng et al. [4] highlights the unusual properties of communication in markets. In standard market models, communication often comes from central authority, such as a market maker or Walras's fictitious auctioneer [14]. This omniscient authority must broadcast enough information (e.g. prices) for each agent to decide his own behavior without further communication — because each agent has private information (e.g. an endowment), it may be that agents are ignorant of others' equilibrium allocations. By comparison, in Yao's basic two-party model [15], two players follow a protocol (where both may send information) to communicate enough information that both players know the answer to the problem. Here, we study the communication requirements of reaching equilibrium in the market communication model.

Classical economic treatment of communication costs studies the *dimensionality of the message space* required to communicate a Pareto-efficient outcome. In standard convex economies, the seminal work of Arrow and Debreu [1] may be interpreted as a proof that $(m-1)$ real numbers — i.e. normalized prices — are sufficient. Subsequent work [6,10] shows that normalized prices are optimal. A priori, the results for convex economies are powerful because the amount of

* Supported by a fellowship from the University of California at Berkeley and NSF grant CCF-0635319.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 391–401, 2010.

communication is independent of the number of agents and firms. Many subsequent works have sharpened and extended these results [7,2,13]. Of particular relevance, Calsamiglia's introduction of parametric communication [2] precisely captures the notion that communication may leverage private information to reduce communication.

Our work focuses on the bit-wise communication requirements for reaching equilibrium — while $(m-1)$ real numbers may be dimensionally optimal, they may hide many bits. Since most real-world applications communicate a price with fixed precision, we follow Deng et al. [4] in believing that bit-wise communication bounds are important. Related communication complexity results [12,11] consider the problem of communicating preferences or complete allocations, while most research on market equilibria has focused on developing efficient algorithms (e.g. [5,8,3]). To the best of our knowledge, Deng et al. give the only result specifically applicable to this model.

Our main result gives a lower bound on the number of bits of information that must be communicated in an Arrow-Debreu market with production. We show that the number of bits depends polynomially on the number of agents, the number of firms, and the amount of private information they hold.

Our bound is significantly stronger than the bound of Deng at al. [4]. First, Deng et al. need $\Theta(\frac{n}{m})$-bit numbers to show a $poly(n)$ lower bound, i.e. they give each agent polynomially many bits of private information. We achieve the same lower bound with a logarithmic number of such bits. Second, Deng et al. must relax the standard non-satiation requirement on utility functions;[1] we do not. Thirdly, our bound is more general because it considers a production economy.

The main shortcoming of our bound is that it critically exploits the fact that real numbers rarely sum to the same value, even if they are very close. Thus, it is unlikely to extend to approximate equilibria.

2 Markets and Market Communication

Market communication complexity aims to study the amount of information that prices must encode to induce equilibrium in an Arrow-Debreu economy [1].

2.1 Arrow-Debreu Markets

An Arrow-Debreu market with production consists of n agents, m goods, and l production firms (indexed by i, j, and k respectively). A bundle of goods is a vector $\boldsymbol{x} \in \mathbb{R}^m$ where x_j represents a quantity of good j.

Each agent has a utility function and an endowment. The utility function $u_i(\boldsymbol{x}_i) : \mathbb{R}^M \to \mathbb{R}$ maps bundles of goods to utilities, and the endowment $\boldsymbol{e}_i \in \mathbb{R}^m$ is a bundle of goods. In order to guarantee the existence of an equilibrium, it is sufficient to assume that u_i is strictly concave in x.

[1] They call it "strict concavity." Nonsatiation is required for Arrow and Debreu's proof of the existence of equilibrium [1].

A production firm is specified by a set of net production possibilities $Y_k \subset \mathbb{R}^M$. A vector $\boldsymbol{y}_k \in Y_k$ represents the net quantities of goods produced: a positive value $y_{j,k}$ represents an output of good j, and a negative value $y_{j,k}$ represents an input. Notice that at prices $\boldsymbol{\pi}$, the profit of firm k may be written as $\boldsymbol{\pi} \cdot y_k$. Again, the sets Y_k must satisfy convexity requirements. In particular, it is sufficient to assume the following: Y_k is closed, convex, and contains the 0 vector, and if $y \in \bigcup_k Y_k$, then $-y \in \bigcup_k Y_k$ if and only if $y = 0$.

To link production to consumption, a firm is owned by agents. Agent i may own a share $\alpha_{i,k} \in [0, 1]$ of the profits of firm k, i.e. at prices $\boldsymbol{\pi}$, agent i's budget will be the value of his endowment plus the profit derived from firms he owns, i.e.

$$M_i = \boldsymbol{\pi} \cdot \boldsymbol{e}_i + \sum_{k \in [l]} \sigma_{i,k} \boldsymbol{\pi} \cdot \boldsymbol{y}_k \quad . \tag{1}$$

Since $\sigma_{i,k}$ denotes a share of firm k, it must be that $\sum_{i \in [n]} \sigma_{i,k} = 1$. We omit the precise restrictions on production sets and utility functions for brevity.

The following economic definitions are standard [9]:

Definition 1. *An* economic allocation *is a tuple* $(\{\boldsymbol{x}_i\}, \{\boldsymbol{y}_k\})$ *specifying the bundle* $\boldsymbol{x}_i$ *consumed by each agent and the production vector* $\boldsymbol{y}_k$ *chosen by each firm.*

Definition 2. *An economic allocation is* feasible *if* $\boldsymbol{x}_i \geq 0$, $\boldsymbol{y}_k \in Y_k$, *and the total demand is less than or equal to the total supply, i.e.*

$$\sum_{i \in [n]} \boldsymbol{x}_i \leq \sum_{i \in [n]} \boldsymbol{e}_i + \sum_{j \in [m]} \boldsymbol{y}_k \tag{2}$$

Definition 3. *A* competitive equilibrium *(hereafter equilibrium) in an Arrow-Debreu market is a set of prices* $\boldsymbol{\pi} \in \mathbb{R}^M$ *and a feasible allocation* $(\{\boldsymbol{x}_i\}, \{\boldsymbol{y}_k\})$ *such that agents maximize their utilities and firms maximize their profits at current prices, i.e.*

$$\boldsymbol{x}_i \in \operatorname*{arg\,max}_{\boldsymbol{x} \in \{\boldsymbol{x} | \boldsymbol{x} \cdot \boldsymbol{\pi} \leq \boldsymbol{e}_i \cdot \boldsymbol{\pi}\}} u_i(\boldsymbol{x}) \tag{3}$$

$$\boldsymbol{y}_k \in \operatorname*{arg\,max}_{\boldsymbol{y} \in Y_k} \boldsymbol{\pi} \cdot \boldsymbol{y} \quad . \tag{4}$$

2.2 Market Communication

Deng et al. [4] define the market communication model as follows:

Definition 4. Market Communication*: n agents $[n]$ have private information $x_i \in X_i$ (the sets X_i are common knowledge). Agent i wishes to compute the function $f_i(x_1, \ldots x_n)$. Another agent, agent 0 (the "invisible hand"), knows $(x_1, \ldots x_n)$.*

A protocol is a set of functions $(g_0(\cdot), g_1(\cdot), \ldots g_n(\cdot))$ where $g_0 : X_1 \times \ldots X_n \to X_0$, $g_{i \in [n]} : X_0 \times X_i \to \mathbb{R}$, and $g_i(g_0(x_1, \ldots x_n), x_i) = f_i(x_1, \ldots x_n)$. The amount of market communication is the number of bits in $x_0 = g_0(x_1, \ldots x_n)$.

In essence, the omniscient agent 0 computes $x_0 = g_0(x_1, \ldots x_n)$ and broadcasts x_0 to agents $i \in [n]$. Next, each agent privately uses x_i to compute $g_i(x_0, x_i) = f_i(x_1, \ldots x_n)$.

The Power of Market Communication. The addition of an omniscient agent substantially increases the model's power: it is as powerful as standard nondeterministic communication.

Theorem 1. *Assume communication costs are measured in bits. Then any problem* $f(x_1, \ldots x_n)$ *in* NP^{CC} *has an efficient market communication protocol.*

Proof. By assumption, there is a communication sequence σ of poly-logarithmic length that solves the problem. Let $T = \{(i_t, \sigma_t)\}$ be a transcript of the communication, i.e. agent i_t sent σ_t at time t.

Note that agent 0 may compute T because she is omniscient. Thus, in the market communication protocol, agent 0 computes T and broadcasts it to the agents. Each agent then simulates his behavior based on T to solve the problem. The size of i_t is $\log n$, so $|T| = \Theta(|\sigma| \log n)$, thus giving an efficient market communication protocol. □

Market Communication in Arrow-Debreu Markets. We wish to discuss the number of bits of private information an agent or firm receives; however, such private information is often given in terms of real numbers or functions. To generate a meaningful measure of each agent's private information, we assume that utility functions and production sets are drawn from finite sets.

Specifically, an agent's utility function u_i is drawn from a finite set $\mathbb{U}$. Also, an agent's endowment is a vector of dimension m in which each coordinate is represented in β bits. Similarly, a firm's production set Y_k is drawn from a finite set $\mathbb{Y}$. Our bound will be a function of the number of possible utility functions $|\mathbb{U}|$ and the number of possible production sets $|\mathbb{Y}|$.

The goal of an agent or firm is to compute its consumption vector $\boldsymbol{x}_i$ or production vector $\boldsymbol{y}_k$. Thus, if E represents all private information in the economy, we have $\boldsymbol{g}_i = \boldsymbol{x}_i(E)$ for the agents and $\boldsymbol{g}_k = \boldsymbol{y}_k(E)$ for the firms. (While the definition of an equilibrium includes prices, we take the position that prices are merely a communication tool and that, at the end of the day, we only care if each agent chooses the correct allocation. Thus, we do not explicitly require agents to compute prices as part of g_i.)

For example, a trivial protocol might broadcast each agent's utility function and each firm's production set, using $O(n \log |\mathbb{U}| + l \log |\mathbb{Y}|)$ bits of communication. Agents would then know everything and, therefore, could compute an equilibrium.

3 A Lower Bound for the Arrow-Debreu Model

Our main result shows that the number of bits is polynomial in the number of goods, agents, firms, and bits of private information.

Theorem 2. *In the worst case, communicating a market equilibrium in the market communication model requires at least*

$$\frac{m}{2}\left(\beta + \lg(n-1)\right) + n + l - O(1) \tag{5}$$

bits of communication to reach equilibrium, where n is the number of agents, m is the number of goods, l is the number of firms, and β is the number of bits used to represent a value in an agent's endowment.

The $(n+l)$ term is the most significant — it implies that the number of bits is linear in the number of agents and production firms. From a practical perspective, it is unrealistic to believe that prices contain $(n+l)$ bits of information.

The $\frac{m}{2}(\beta + \lg(n-1))$ term corresponds to communicating the total global endowment of resources. The total endowment of each resource is, in general, a $(\beta + \lg n)$-bit number. Thus, this term roughly corresponds to communicating the total endowment of $\frac{m}{2}$ goods.

Instead of proving the theorem directly, we prove a more general lemma that allows us to compare our bound to Deng et al. [4]. They achieve an $\Omega(n \log(m+n))$ lower bound using an exponentially large set of utility functions that require $poly(m,n)$-bit numbers, i.e. $|\mathbb{U}| = \Omega\left(2^{poly(m,n)}\right)$ and $\beta = poly(m,n)$. By comparison, taking the number of utility functions and production functions to be polynomial in our construction (i.e. $|\mathbb{U}| = |\mathbb{Y}| = poly(m+n)$) gives the same $\Omega(n \log(m+n))$ bound.

Moreover, if we allow players and firms to value arbitrary bundles of goods, then we get $|\mathbb{U}| = |\mathbb{Y}| = \Omega(2^m)$ and thus a lower bound of

$$\Omega\left(\frac{m}{2}(\beta + \lg(n-1)) + m \cdot (n+l) - m\right) \ . \tag{6}$$

In this case, each item requires $(n+l)$ bits of information in its price. (While attributing an arbitrary bundle value to an item is slightly unrealistic, it is a common worst case setting in areas such as combinatorial auctions.)

Lemma 1. *Communicating a market equilibrium in the market communication model requires at least*

$$\frac{m}{2}(\beta + \lg(n-1)) + (n - 1 - |\mathbb{U}|^{\frac{4}{m}}) \lg |\mathbb{U}| + (l - 1 - |\mathbb{Y}|^{\frac{4}{m}}) \lg |\mathbb{Y}| \tag{7}$$

bits of communication in the worst case, where $|\mathbb{U}|$ is the number of possible utility functions u_i, and $|\mathbb{Y}|$ is the number of possible production sets Y_k.

Moreover, there are specific sets $\mathbb{U}$ and $\mathbb{Y}$ such that the economy requires

$$\frac{m}{2}(\beta + \lg(n-1)) + n + l - O(1) \tag{8}$$

bits of communication to reach equilibrium.

Proof. We construct an economy with m goods, n agents, and l firms. The main trick is to make each combination of utility functions (or production functions) correspond to a unique prime factorization. Thus, no two combinations of utility functions (or production functions) will have the same optimal allocation.

The second trick is to leave one agent (and firm) without any private information, so the number of communication sequences is trivially lower-bounded by the number of possible equilibrium choices she may make.

The Economy. Assume m is divisible by 4, $n \geq 2$, and $l \geq 2$.

Partition the goods into four groups modulo 4, i.e.

$$M_a = \{j | j \in [m] \text{ and } j \equiv a \mod 4\} \tag{9}$$

The sets will serve the following purposes:

- Goods in M_0 are production inputs and goods in M_1 are outputs. Nobody wants goods in M_0. Consequently, the entire supply of goods M_0 is converted to goods in M_1. Goods in M_1 are indistinguishable to the agents, so Pareto-optimality will imply that producers maximize the total output of goods in M_1.
- Goods in M_2 and M_3 are traded among agents. Goods are paired such that an agent balances the quantity of a good $m_2 \in M_2$ with some good $m_3 \in M_3$ to match marginal utilities.

Agents $i > 1$ have utility functions of the form

$$u_i(\boldsymbol{x}_i) = \sum_{j \in M_3} \left(2\sqrt{x_{i,j} \lg c_{i,j}} + x_{i,j-1}\right) + \sum_{j \in M_1} x_{1,j} \tag{10}$$

where $c_{i,j} \in C$, and the set C will be determined later. Agents $i > 1$ are endowed with goods from M_0, M_2 and M_3 only, i.e.

$$e_{i,j} = \begin{cases} e_{i,j}, & j \in M_0 \cup M_3 \\ \bar{e}, & j \in M_2 \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

For endowed goods, $e_{i,j} \in [2^\beta]$ is a β-bit integer and $\bar{e} = n \cdot 2^\beta$ is a large number (large enough that, in equilibrium, an agent will always keep a positive quantity of each good in M_2).

Agent 1 has the utility function

$$u_1(\boldsymbol{x}_1) = \sum_{j \in M_3} \left(2\sqrt{x_{1,j}} + x_{1,j-1}\right) \tag{12}$$

(the first term is equivalent to setting $c_{1,j} = 2$). She is endowed with 1 unit each of goods in M_2, and M_3, i.e.

$$e_{1,j} = \begin{cases} 1, & j \in M_2 \cup M_3 \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

Note that agent 1 has no private information.

The firms have technology to convert goods in M_0 to goods in M_1. Like agents' utilities, the production functions are parameterized by coefficients $c_{j,k} \in C$. (We will take C to be the same for both parts of the economy, but this is certainly not necessary.) We define the production of firm k in terms of a production function, i.e. firm k may transform $y_{j-1,k}$ units of good $(j-1)$ into $y_{j,k}$ units of good j according to the following function $f_{j,k}$:

$$y_{j,k} = f_{j,k}(y_{j-1,k}) = 2\sqrt{y_{j-1,k} \cdot \lg c_{j,k}} \tag{14}$$

This function is translated to a set of vectors to match the model.[2] In order to create a firm with no private information, we require that $c_{j,1} = 2$. For simplicity, we also specify that all firms are owned by agents $i > 1$.

Analysis. First, we show that agent 1 must be able to select

$$\left((n-1)2^{\beta}\right)^{\frac{m}{4}} |\mathbb{U}|^{n-1-|\mathbb{U}|^{\frac{4}{m}}} \tag{15}$$

distinct consumption vectors. A similar proof gives a lower bound for the production side of the economy.

Consider a good $j \in M_3$. (Note that good $(j-1)$ is in M_2.) Let $p_j = \frac{\pi_j}{\pi_{j-1}}$ be the relative price of good j compared to good $(j-1)$. Note that each agent has a term of the form $2\sqrt{x_{i,j} \lg c_{i,j}} + x_{i,j-1}$ in u_i. In equilibrium, we know that agent i does not wish to sell good $j-1$ to get good j (or vice-versa). Thus, agent i must balance her marginal utilities from the $2\sqrt{x_{i,j} \lg c_{i,j}}$ and $x_{i,j-1}$ terms. This gives the relation

$$\frac{\partial\left(2\sqrt{x_{i,j} \lg c_{i,j}}\right)}{\partial x_{i,j}} = \frac{\partial\left(p_j x_{i,j-1}\right)}{\partial x_{i,j-1}} \tag{16}$$

$$\sqrt{\frac{\lg c_{i,j}}{x_{i,j}}} = p_j \tag{17}$$

$$x_{i,j} = \frac{\lg c_{i,j}}{(p_j)^2} \tag{18}$$

(Note that by construction, i.e. by choice of $\bar{e}$, this is always possible.) Since goods in M_2 and M_3 do not involve production, we know that

$$\sum_{i \in [n]} x_{i,j} = \sum_{i \in [n]} e_{i,j} \ . \tag{19}$$

Let $\alpha_j = \sum_{i \in [n]} e_{i,j}$. Using this constraint and the equations $x_{i,j} = \frac{\lg c_{i,j}}{(p_j)^2}$, it follows that

$$p_j^2 = \frac{1}{\alpha_j} \sum_{j \in M_3} \lg c_{i,j} = \frac{1}{\alpha_j} \lg\left(\prod_{j \in M_3} c_{i,j}\right) \ , \tag{20}$$

and thus

$$x_{i,j} = \frac{\alpha_j \lg c_{i,j}}{\lg\left(\prod_{j \in M_3} c_{i,j}\right)} \ . \tag{21}$$

[2] The only trick to converting $f_{j,k}$ to a set is to allow firm k to produce any amount of good j between 0 and $f_{j,k}$. In equilibrium, production will always occur on the boundary defined by $f_{j,k}$, so this change is inconsequential.

For agent 1, we get

$$x_{1,j} = \frac{1}{\lg\left(\left(\prod_{j\in M_3\setminus 1} c_{i,j}\right)^{\frac{1}{\alpha_j}}\right)} \quad . \tag{22}$$

To show a lower bound, we want to show that we can choose the set C such that the number of possible values for $x_{1,j}$ is large. We take C to be the $|C|$ smallest primes. (Note that this implies the total number of possible utility functions is $|\mathbb{U}| = |C|^{\frac{m}{4}}$.) To count the number of possible values for $x_{1,j}$, consider the value

$$\prod_{j\in M_3\setminus 1} (c_{i,j})^{\frac{1}{\alpha_j}} \quad . \tag{23}$$

This represents a fractional prime factorization of a number where the prime factors are in C. Thus, the number of distinct values is the number of distinct sets of the form

$$\left\{\frac{k_{c,j}}{\alpha_j}\right\} \tag{24}$$

where $k_{c,j}$ is the number of times a prime c occurs in the factorization and $\sum_i k_{c,j} = n$. Note that if two sets $\left\{\frac{k_{c,j}}{\alpha_j}\right\}$ and $\left\{\frac{k'_{c,j}}{\alpha'_j}\right\}$ are the same, then $\sum \frac{k_{c,j}}{\alpha_j} = \sum \frac{k'_{c,j}}{\alpha'_j}$. Since $\sum_i k_{c,j} = n$ is fixed, the only way for two sets to be the same is if $\alpha_j = \alpha'_j$. Thus, since there are $(n-1)2^\beta$ possible values for α_j, there are

$$(n-1)2^\beta \frac{|C|^{n-1}}{|C|!} \geq (n-1)2^\beta |C|^{n-|C|-1} \tag{25}$$

possible values for this quantity, and, therefore, the same number of possible values for $x_{1,j}$. Counting over all $\frac{m}{4}$ goods in M_3 and assuming that C is the same for all j (thus $|\mathbb{U}| = |C|^{\frac{m}{4}}$), it follows that agent 1 has at least

$$\left((n-1)2^\beta |C|^{n-|C|-1}\right)^{\frac{m}{4}} = \left((n-1)2^\beta\right)^{\frac{m}{4}} |\mathbb{U}|^{n-1-|\mathbb{U}|^{\frac{4}{m}}} \tag{26}$$

possible choices. Moreover, note that we may derive this bound for other sizes $|\mathbb{U}|$ by fixing all c_{ij} for some j. Assume that we fix the values of c_{ij} for $\left(\frac{m}{4} - k\right)$ goods (i.e. k goods still have c_{ij} drawn from C), then we get

$$\left((n-1)2^\beta\right)^{\frac{m}{4}} |\mathbb{U}|^{n-1-|\mathbb{U}|^{\frac{1}{k}}} \tag{27}$$

where $|\mathbb{U}| = |C|^k$. (This will be useful when we wish to set $|\mathbb{U}| = 2$.)

The analysis for the firms is similar: firm 1 must be able to select

$$\left((n-1)2^\beta\right)^{\frac{m}{4}} |\mathbb{Y}|^{l-1-|\mathbb{Y}|^{\frac{4}{m}}} \tag{28}$$

distinct production vectors. First, we characterize optimal production. Consider a single good $j \in M_1$ and observe that

$$\sum_{k \in [l]} y_{j-1,k} = \sum_{i \in [n]} e_{i,j-1} = \alpha_j \quad . \tag{29}$$

Let $p_{j-1} = \frac{\pi_{j-1}}{\pi_j}$ be the equilibrium price of good $(j-1)$ relative to good j. Then we know that firm k maximizes

$$y_{j,k} - p_{j-1} y_{j-1,k} = 2\sqrt{y_{j-1,k} \cdot \lg c_{j,k}} - p_{j-1} y_{j-1,k} \tag{30}$$

Taking the first derivative with respect to $y_{j-1,k}$ implies that $y_{j,k} = \frac{\lg c_{j,k}}{(p_{j-1})^2}$, so we repeat the analysis used for agent 1. This shows that firm 1 must be able to make at least

$$\left((n-1)2^{\beta}\right)^{\frac{m}{4}} |\mathbb{Y}|^{l-1-|\mathbb{Y}|^{\frac{4}{m}}} \tag{31}$$

different choices.

Because the choices of agent 1 and firm 1 are independent, all combinations of choices are possible. Thus, the total number of communication sequences must be at least

$$\left((n-1)2^{\beta}\right)^{\frac{m}{2}} |\mathbb{U}|^{n-1-|\mathbb{U}|^{\frac{4}{m}}} |\mathbb{Y}|^{l-1-|\mathbb{Y}|^{\frac{4}{m}}} \tag{32}$$

and the total number of bits of communication is at least

$$\frac{m}{2}\left(\beta + \lg(n-1)\right) + (n - 1 - |\mathbb{U}|^{\frac{4}{m}}) \lg |\mathbb{U}| + (l - 1 - |\mathbb{Y}|^{\frac{4}{m}}) \lg |\mathbb{Y}| \tag{33}$$

If we take $|\mathbb{U}| = |\mathbb{Y}| = 2$ (i.e. the set $C = \{2, 3\}$ for one good, fixed at $c_{ij} = 2$ otherwise), then we get a lower bound of

$$\frac{m}{2}\left(\beta + \lg(n-1)\right) + n + l - O(1) \tag{34}$$

bits of communication. □

4 Conclusion

Our main theorem diminishes the power of prices, with the caveat that we demand an exact equilibrium. While $(m-1)$ prices are sufficient, the amount of information they contain may be highly dependent on the parameters of the market.

Most significantly, the number of bits of information they must communicate is linear in the number of agents and firms in the worst case (the $(n+l)$ term). This implies that even though a price is supposed to be "universal," prices must contain a unique bit of information for every agent in the economy. In the context of decimal prices, this roughly translates to one digit for every four buyers of a good. This is quite impractical.

It remains an open problem to give tight bounds. For example, we currently do not have any nontrivial upper bounds. Also, there are a few reasons why our lower bound may not be tight. First, instead of a multiplicative factor of $\frac{m}{2}$, one might expect a multiplicative factor of $(m-1)$ since that is the number of prices that must be communicated. Second, the multiplicative $\log(m+n)$ factor shown by Deng et al. [4] arises from an effect not present in our construction.

A more significant open problem is to give lower bounds for communicating approximate equilibria. Since our construction is highly dependent on the fact that two sets of irrational numbers rarely sum to the same value, it is unlikely to survive when an approximate equilibrium is sufficient. In particular, it is plausible that the polynomial dependence on n and l fundamentally requires $poly(n, l)$-bit numbers.

Furthermore, lower bounds for approximate equilibria would be more realistic. Because market clearing is also measured to finite precision, a lower bound approximate equilibria would give a stronger result on the amount of precision required in prices.

References

1. Arrow, K.J., Debreu, G.: Existence of an equilibrium for a competitive economy. Econometrica 22(3), 265–290 (1954)
2. Calsamiglia, X.: Informational requirements of parametric resource allocation processes. In: Groves, T., Radner, R., Reiter, S. (eds.) Information, Incentives, and Economic Mechanisms. Basil Blackwell (1988)
3. Cole, R., Fleischer, L.: Fast-converging tatonnement algorithms for one-time and ongoing market problems. In: STOC 2008: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, pp. 315–324. ACM, New York (2008)
4. Deng, X., Papadimitriou, C., Safra, S.: On the complexity of equilibria. In: STOC 2002: Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing, pp. 67–71. ACM, New York (2002)
5. Devanur, N.R., Papadimitriou, C.H., Saberi, A., Vazirani, V.V.: Market equilibrium via a primal-dual-type algorithm. In: FOCS 2002: Proceedings of the 43rd Symposium on Foundations of Computer Science, Washington, DC, USA, pp. 389–395. IEEE Computer Society, Los Alamitos (2002)
6. Hurwicz, L.: Optimality and informational efficiency in resource allocation processes. In: Arrow, K.J., Karlin, S., Suppes, P. (eds.) Mathematical Methods in the Social Sciences, pp. 27–46. Stanford University Press, Stanford (1960)
7. Jordan, J.S.: The competitive allocation process is informationally efficient uniquely. Journal of Economic Theory 28(1), 1–18 (1982)
8. Kapoor, S., Mehta, A., Vazirani, V.: An auction-based market equilibrium algorithm for a production model. Theor. Comput. Sci. 378(2), 153–164 (2007)
9. Mas-Colell, A., Whinston, M.D., Green, J.R.: Microeconomic Theory. Oxford University Press, Oxford (June 1995)
10. Mount, K., Reiter, S.: The informational size of message spaces. Journal of Economic Theory 8(2), 161–192 (1974)
11. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, New York (2007)
12. Nisan, N., Segal, I.: The communication requirements of efficient allocations and supporting prices. Journal of Economic Theory 129(1), 192–224 (2006)

13. Tian, G.: On the informational requirements of decentralized pareto-satisfactory mechanisms in economies with increasing returns (October 2006)
14. Walras, L.: Elements of pure economics, or, The theory of social wealth / Leon Walras; translated by William Jaffe. American Economic Association and the Royal Economic Society, Allen, Unwin, London (1954)
15. Yao, A.C.-C.: Some complexity questions related to distributive computing (preliminary report). In: STOC 1979: Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing, pp. 209–213. ACM, New York (1979)

Strategy-Proof Voting Rules over Multi-issue Domains with Restricted Preferences

Lirong Xia and Vincent Conitzer

Department of Computer Science
Duke University
Durham, NC 27708, USA
{lxia,conitzer@cs.duke.edu}

Abstract. In this paper, we characterize strategy-proof voting rules when the set of alternatives has a multi-issue structure, and the voters' preferences are represented by acyclic CP-nets that follow a common order over issues. Our main result is a simple full characterization of strategy-proof voting rules satisfying non-imposition for a very natural restriction on preferences in multi-issue domains: we show that if the preference domain is lexicographic, then a voting rule satisfying non-imposition is strategy-proof if and only if it can be decomposed into multiple strategy-proof local rules, one for each issue and each setting of the issues preceding it. We also obtain the following variant of Gibbard-Satterthwaite: when there are at least two issues and each of the issues can take at least two values, then there is no non-dictatorial strategy-proof voting rule that satisfies non-imposition, even when the domain of voters' preferences is restricted to linear orders that are consistent with acyclic CP-nets following a common order over issues. This impossibility result follows from either one of two more general new impossibility results we obtained, which are not included in this paper due to the space constraint.

Keywords: Voting, multi-issue domains, strategy-proofness, lexicographic domains.

1 Introduction

When agents have conflicting preferences over a set of alternatives, and they want to make a joint decision, a natural way to do so is by *voting*. Each agent (voter) is asked to report his or her preferences. Then, a *voting rule* is applied to the vector of submitted preferences to select a winning alternative. However, in some cases, a voter has an incentive to submit false preferences in order to change the winner to a more preferable alternative (to her). An instance of such misreporting is called a *manipulation*, and the perpetrating voter is called a *manipulator*. If there is no manipulation under a voting rule, then the rule is said to be *strategy-proof*.

Unfortunately, there are some very natural properties that are satisfied by no strategy-proof voting rule, according to the Gibbard-Satterthwaite theorem [16,27]. The theorem states that when there are three or more alternatives, and any voter can choose *any* linear order over alternatives to represent her preferences, then no non-dictatorial voting rule

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 402–414, 2010.

that satisfies non-imposition is strategy-proof. A voting rule is dictatorial if the same voter's most-preferred alternative is always chosen; it satisfies non-imposition if for every alternative, there exist *some* reported preferences that make that alternative win.

There are several approaches to circumventing this impossibility result. One that has received significant attention from computer scientists in recent years is to consider whether finding a manipulation is computationally hard under some rules. If so, then even though a manipulation is guaranteed to exist, it will perhaps not occur because the manipulator(s) cannot find it. Indeed, it has been shown that finding a manipulation is computationally hard (more precisely, NP-hard) for various rules, for various definitions of the manipulation problem (*e.g.*, [6,5,13,17,14,36]). On the other hand, NP-hardness is a *worst-case* notion of hardness, so that it may very well be the case that *most* manipulations are easy to find. Various recent results suggest that this is indeed the case [25,12,24,15,37,31,30,28,34,29,18]. This paper does not fall under this line of research.

Instead, this paper falls under another, older, line of research on circumventing the Gibbard-Satterthwaite result. This line, which has been pursued mainly by economists, is to restrict the domain of preferences. That is, we assume that voters' preferences always lie in a restricted class. An example of such a class is that of *single-peaked* preferences [7]. For single-peaked preferences, desirable strategy-proof rules exist, such as the *median* rule. Other strategy-proof rules are also possible in this preference domain: for example, it is possible to add some artificial (*phantom*) votes before running the median rule. In fact, this characterizes all strategy-proof rules for single-peaked preferences [22]. On the other hand, preferences have to be significantly restricted to obtain such positive results: Aswal *et al.* [1] extend the Gibbard-Satterthwaite theorem, showing that if the preference domain is *linked*, then with three or more alternatives the only strategy-proof voting rule that satisfies non-imposition is a dictatorship.

In real life, the set of alternatives often has a multi-issue structure. That is, there are multiple *issues* (or *attributes*), each taking values in its respective domain, and an alternative is completely characterized by the values that the issues take. For example, consider a situation where the inhabitants of a county vote to determine a government plan. The plan is composed of multiple sub-plans for several interrelated issues, such as transportation, environment, and health [10]. Clearly, a voter's preferences for one issue in general depend on the decisions taken on the other issues: if a new highway is constructed through a forest, a voter may prefer a nature reserve to be established; but if the highway is not constructed, the voter may prefer that no nature reserve is established. As another example, in each US presidential election year, the president as well as members of the Senate and the House must be elected. In principle, a voter's preferences for a senator can depend on who is elected as president, for example if the voter prefers a balance of power between the Democratic and Republican parties. A straightforward way to aggregate preferences in multi-issue domains is *issue-by-issue* (a.k.a. *seat-by-seat*) voting, which requires that the voters explicitly express their preferences over each issue separately, after which each issue is decided by applying issue-wise voting rules independently. This makes sense if voters' preferences are *separable*, that is, each voter's preferences over a single issue are independent of her preferences over other issues. However, if preferences are not separable, it is not clear how the voter should vote in such an issue-by-issue election. Indeed, it is known that natural strategies for voting in such a context can lead to very undesirable results [10,20].

The problem of characterizing strategy-proof voting rules in multi-issue domains has already received significant attention. Strategy-proof voting rules for high-dimensional single-peaked preferences (where each dimension can be seen as an issue) have been characterized [8,2,3,23]. Barbera *et al.* [4] characterized strategy-proof voting rules when the voters' preferences are separable, and each issue is binary (that is, the domain for each issue has two elements). Ju [19] studied multi-issue domains where each issue can take three values: "good", "bad", and "null", and characterized all strategy-proof voting rules that satisfy *null-independence*, that is, if a voter votes "null" on an issue i, then her preferences over other issues do not affect the value of issue i.

The prior research that is closest to ours was performed by Le Breton and Sen [11]. They proved that if the voters' preferences are separable, and the restricted preference domain of the voters satisfies a *richness* condition, then, a voting rule is strategy-proof if and only if it is an issue-by-issue voting rule, in which each issue-wise voting rule is strategy-proof over its respective domain.

Despite its elegance, the work by Le Breton and Sen is limited by the restrictiveness of separable preferences: as we have argued above, in general, a voter's preferences on one issue depend on the decision taken on other issues. On the other hand, one would not necessarily expect the preferences for one issue to depend on every other issue. CP-nets [9] were developed in the artificial intelligence community as a natural representation language for capturing limited dependence in preferences over multiple issues. Recent work has started to investigate using CP-nets to represent preferences in voting contexts [26,21,35,32]. If there is an order over issues such that every voter's preferences for "later" issues depend only on the decisions made on "earlier" issues, then the voters' CP-nets are acyclic, and a natural approach is to apply issue-wise voting rules *sequentially* [21]. While the assumption that such an order exists is still restrictive, it is much less restrictive than assuming that preferences are separable (for one, the resulting preference domain is exponentially larger [21]). Recent extensions of sequential voting rules include order-independent sequential voting [35], as well as frameworks for voting when preferences are modeled by general (that is, not necessarily acyclic) CP-nets [32,33]. However, in this paper, we only study acyclic CP-nets that are consistent with a common order over the issues.

Our results. In this paper, we focus on multi-issue domains that are composed of at least two issues with at least two possible values each.[1] We first show that over *lexicographic* preference domains (where earlier issues dominate later issues in terms of importance to the voters), the class of strategy-proof voting rules that satisfy non-imposition is exactly the class of voting rules that can be decomposed into multiple strategy-proof local rules, one for each issue and each setting of the issues preceding it. Technically, it is exactly the class of all *conditional rule nets (CR-nets)*, defined later in this paper but analogous to CP-nets, whose local (issue-wise) entries are strategy-proof voting rules. CR-nets represent how the voting rule's behavior on one issue depends on

[1] This is the standard assumption for studying voting in multi-issue domains, because otherwise either the domain can be simplified (by removing issues that only take one value), or it has no multi-issue structure (when there is only one issue).

the decisions made on all issues preceding it. Conceptually, this is similar to how acyclic CP-nets represent how a voter's preferences on one issue depend on the decisions made on all issues preceding it.

Then, we prove an impossibility theorem, which is the following variant of Gibbard-Satterthwaite. When there are at least two issues with at least two values each, the only strategy-proof voting rule that satisfies non-imposition is a dictatorship. This result assumes that each voter is free to choose any linear order that corresponds to an acyclic CP-net that follows a common order over the issues. This impossibility result follows from either one of two more general new impossibility results that we do not include in this paper due to the space constraint.

We are not aware of any previous characterization or impossibility results for strategy-proof voting rules when voters' preferences display dependencies across issues (that is, when they are modeled by CP-nets).

2 Preliminaries

In a voting setting (not necessarily one with multiple issues), let $\mathcal{X}$ be the set of *alternatives* (or *candidates*). A linear order V on $\mathcal{X}$ is a transitive, antisymmetric, and total relation on $\mathcal{X}$. The set of all linear orders on $\mathcal{X}$ is denoted by $L(\mathcal{X})$. An n-voter profile P on $\mathcal{X}$ consists of n linear orders on $\mathcal{X}$. That is, $P = (V_1, \ldots, V_n)$, where for every $1 \leq j \leq n$, $V_j \in L(\mathcal{X})$. The set of all profiles on $\mathcal{X}$ is denoted by $P(\mathcal{X})$. In this paper, we let n denote the number of voters. A *(voting) rule* r is a mapping from the set of all profiles on $\mathcal{X}$ to $\mathcal{X}$, that is, $r : P(\mathcal{X}) \to \mathcal{X}$. For example, the *plurality* rule (also called the *majority* rule, when there are only two alternatives) chooses the alternative that is ranked in the top position in the most votes (with a tie-breaking mechanism, for example, ties are broken in alphabetical order—in this paper, it does not matter which tie-breaking mechanism we use). A voting rule r satisfies

- *unanimity*, if $top(V) = c$ for all $V \in P$ implies $r(P) = c$.
- *non-imposition*, if for any $c \in \mathcal{X}$, there exists an n-voter profile P such that $r(P) = c$.
- *(strong) monotonicity*, if for any pair of profiles $P = (V_1, \ldots, V_n)$, $P' = (V'_1, \ldots, V'_n)$ such that for any alternative c and any $1 \leq j \leq n$, we have $c \succ_{V'_j} r(P) \Rightarrow c \succ_{V_j} r(P)$, then, $r(P') = r(P)$.
- *strategy-proofness*, if there does not exist a pair (P, V'_j), where P is a profile, and V'_j is a false vote of voter j, such that $r(P_{-j}, V'_j) \succ_{V_j} r(P)$. That is, there is no profile where a voter can misrepresent her preferences to make herself better off.

In this paper, the set of all alternatives $\mathcal{X}$ is a *multi-issue domain*. That is, let $\mathcal{I} = \{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$ be a set of *issues*, where each issue $\mathbf{x}_i$ takes values in a *local domain*, denoted by D_i. An alternative is uniquely identified by its values on all issues, that is, $\mathcal{X} = D_1 \times \cdots \times D_p$.

Example 1. *A group of people must make a joint decision on the menu for dinner (the caterer can only serve a single menu to everyone). The menu is composed of two issues: the main course (*$\mathbf{M}$*) and the wine (*$\mathbf{W}$*). There are three choices for the main course:*

beef (b), fish (f), or salad (s). The wine can be either red wine (r), white wine (w), or pink wine (p). The set of alternatives is a multi-issue domain: $\mathcal{X} = \{b, f, s\} \times \{r, w, p\}$.

CP-nets [9] are a compact representation that captures dependencies across issues. In this paper, we use them not for their representational compactness, but rather as useful mathematical notation for describing preferences in multi-issue domains, where preferences over one issue can depend on the values of earlier issues.

A CP-net $\mathcal{N}$ over $\mathcal{X}$ consists of two parts: (a) a directed graph $G = (\mathcal{I}, E)$ and (b) a set of conditional linear preferences $\succeq^i_{\boldsymbol{d}}$ over D_i, for each setting $\boldsymbol{d}$ of the parents of $\mathbf{x}_i$ in G. Let $CPT(\mathbf{x}_i)$ be the set of the conditional preferences of a voter on D_i; this is called a *conditional preference table (CPT)*.

A CP-net $\mathcal{N}$ captures dependencies across issues in the following sense. $\mathcal{N}$ induces a partial preorder $\succeq_{\mathcal{N}}$ over the alternatives $\mathcal{X}$ as follows: for any $a_i, b_i \in D_i$, any setting $\boldsymbol{d}$ of the set of parents of $\mathbf{x}_i$ (denoted by $Par_G(\mathbf{x}_i)$), and any setting $\boldsymbol{z}$ of $\mathcal{I} \setminus (Par_G(\mathbf{x}_i) \cup \{\mathbf{x}_i\})$, $(a_i, \boldsymbol{d}, \boldsymbol{z}) \succeq_{\mathcal{N}} (b_i, \boldsymbol{d}, \boldsymbol{z})$ if and only if $a_i \succeq^i_{\boldsymbol{d}} b_i$. In words, the preferences over issue $\mathbf{x}_i$ only depend on the setting of the parents of $\mathbf{x}_i$ (but not on any other issues). For any $1 \leq i \leq p$, $\text{CPT}(\mathbf{x}_i)$ specifies conditional preferences over $\mathbf{x}_i$. Now, if we obtain an alternative $\boldsymbol{d}'$ from $\boldsymbol{d}$ by only changing the value of the ith issue of $\boldsymbol{d}$, we can look at $\text{CPT}(\mathbf{x}_i)$ to conclude whether the voter prefers $\boldsymbol{d}'$ to $\boldsymbol{d}$, or vice versa. In general, however, from the CP-net, we will not always be able to conclude which of two alternatives a voter prefers, if the alternatives differ on two or more issues. This is why $\mathcal{N}$ usually induces a partial preorder rather than a linear order.

We note that when the graph of $\mathcal{N}$ is acyclic, $\succeq_{\mathcal{N}}$ is transitive and asymmetric, that is, a strict partial order. Let $\mathcal{O} = \mathbf{x}_1 > \cdots > \mathbf{x}_p$. We say that a CP-net $\mathcal{N}$ is *compatible* with (or, *follows*) $\mathcal{O}$, if $\mathbf{x}_i$ being a parent of $\mathbf{x}_j$ in the graph implies that $i < j$. That is, preferences over issues only depend on the values of earlier issues in $\mathcal{O}$. A CP-net is *separable* if there are no edges in its graph, which means that there are no preferential dependencies among issues.

Example 2. *Let* $\mathcal{X}$ *be the multi-issue domain defined in Example 1. We define a CP-net* $\mathcal{N}$ *as follows:* $\mathbf{M}$ *is the parent of* $\mathbf{W}$*, and the CPTs consist of the following conditional preferences:* $CPT(\mathbf{M}) = \{b \succ f \succ s\}, CPT(\mathbf{W}) = \{b : r \succ p \succ w, f : w \succ p \succ r, s : p \succ w \succ r\}$*, where* $b : r \succ p \succ w$ *is interpreted as follows: "when* $\mathbf{M}$ *is* b*, then,* r *is the most preferred value for* $\mathbf{W}$*,* p *is the second most preferred value, and* w *is the least preferred value."* $\mathcal{N}$ *and its induced partial order* $\succeq_{\mathcal{N}}$ *are illustrated in Figure 1.* $\mathcal{N}$ *is compatible with* $\mathbf{M} > \mathbf{W}$*.* $\mathcal{N}$ *is not separable.*

A linear order V over $\mathcal{X}$ *extends* a CP-net $\mathcal{N}$, denoted by $V \sim \mathcal{N}$, if it extends the partial order that $\mathcal{N}$ induces. (This is merely saying that V is consistent with the preferences implied by the CP-net $\mathcal{N}$.) V is *separable* if it extends a separable CP-net. The set of all linear orders that extend CP-nets that are compatible with $\mathcal{O}$ is denoted by $Legal(\mathcal{O})$. Throughout the paper, we make the following assumption about multi-issue domains and the voters' preferences.

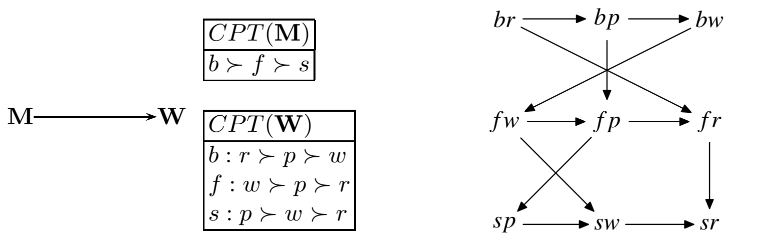
Assumption 1. *In this paper, each multi-issue domain is composed of at least two issues (*$p \geq 2$*), and each issue can take at least two values. Moreover, all CP-nets are compatible with* $\mathcal{O} = \mathbf{x}_1 > \cdots > \mathbf{x}_p$*, and the voters' preferences are always in*

$Legal(\mathcal{O})$ (that is, a voter's preferences over an issue do not depend on the values of later issues).

To present our results, we will frequently use notations that represent the projection of a vote/CP-net/profile to an issue $\mathbf{x}_i$ (that is, the voter's local preferences over $\mathbf{x}_i$) given the setting of all issues preceding $\mathbf{x}_i$, defined as follows. For any issue $\mathbf{x}_i$, any setting $\boldsymbol{d}$ of $Par_G(\mathbf{x}_i)$, and any linear order V that extends $\mathcal{N}$, we let $V|_{\mathbf{x}_i:\boldsymbol{d}}$ and $\mathcal{N}|_{\mathbf{x}_i:\boldsymbol{d}}$ denote the the projection of V (or, equivalently $\mathcal{N}$) to $\mathbf{x}_i$, given $\boldsymbol{d}$. That is, each of these notations evaluates to the linear order $\succeq^i_{\boldsymbol{d}}$ in the CPT associated with $\mathbf{x}_i$. For example, let $\mathcal{N}$ be the CP-net defined in Example 2. $\mathcal{N}|_{\mathbf{W}:b} = r \succ p \succ w$. For any $\mathcal{O}$-legal profile P, $P|_{\mathbf{x}_i:\boldsymbol{d}}$ is the profile over D_i that is composed of the projections of all votes in P on $\mathbf{x}_i$, given $\boldsymbol{d}$. That is, $P|_{\mathbf{x}_i:\boldsymbol{d}} = (V_1|_{\mathbf{x}_i:\boldsymbol{d}}, \ldots, V_n|_{\mathbf{x}_i:\boldsymbol{d}}) = (\mathcal{N}_1|_{\mathbf{x}_i:\boldsymbol{d}}, \ldots, \mathcal{N}_n|_{\mathbf{x}_i:\boldsymbol{d}})$, where $P = (V_1, \ldots, V_n)$, and for any $1 \leq i \leq p$, V_i extends $\mathcal{N}_i$.

The *lexicographic extension* of a CP-net $\mathcal{N}$, denoted by $Lex(\mathcal{N})$, is a linear order V over $\mathcal{X}$ such that for any $1 \leq i \leq p$, any $\boldsymbol{d}_i \in D_1 \times \cdots \times D_{i-1}$, any $a_i, b_i \in D_i$, and any $\boldsymbol{y}, \boldsymbol{z} \in D_{i+1} \times \cdots \times D_p$, if $a_i \succ_{\mathcal{N}|_{\mathbf{x}_i:\boldsymbol{d}_i}} b_i$, then $(\boldsymbol{d}_i, a_i, \boldsymbol{y}) \succ_V (\boldsymbol{d}_i, b_i, \boldsymbol{z})$. Intuitively, in the lexicographic extension of $\mathcal{N}$, $\mathbf{x}_1$ is the most important issue, $\mathbf{x}_2$ is the next important issue, etc; a desirable change to an earlier issue always outweighs any changes to later issues. We note that the lexicographic extension of any CP-net is unique w.r.t. the order $\mathcal{O}$. We say that $V \in L(\mathcal{X})$ is *lexicographic* if it is the lexicographic extension of a CP-net $\mathcal{N}$. For example, let $\mathcal{N}$ be the CP-net defined in Example 2. We have $Lex(\mathcal{N}) = br \succ bp \succ bw \succ fw \succ fp \succ fr \succ sp \succ sw \succ sr$. A profile P is $\mathcal{O}$-legal/separable/lexicographic, if each of its votes is in $Legal(\mathcal{O})$/ is separable/ is lexicographic.

Given a vector of *local rules* $(r_1, \ldots, r_p)$ (that is, for any $1 \leq i \leq p$, r_i is a voting rule on D_i), the *sequential composition* of $r_1, \ldots, r_p$ w.r.t. $\mathcal{O}$, denoted by $Seq(r_1, \ldots, r_p)$, is defined for all $\mathcal{O}$-legal profiles as follows: $Seq(r_1, \ldots, r_p)(P) = (d_1, \ldots, d_p) \in \mathcal{X}$, so that for any $1 \leq i \leq p$, $d_i = r_i(P|_{\mathbf{x}_i:d_1 \cdots d_{i-1}})$. That is, the winner is selected in p steps, one for each issue, in the following way: in step i, d_i is selected by applying the local rule r_i to the preferences of voters over D_i, conditioned on the values $d_1, \ldots, d_{i-1}$ that have already been determined for issues that precede $\mathbf{x}_i$. When the input profile is separable, $Seq(r_1, \ldots, r_p)$ becomes an *issue-by-issue* voting rule.



(a) A CP-net $\mathcal{N}$. (b) The partial order induced by $\mathcal{N}$.

Fig. 1. A CP-net $\mathcal{N}$ and its induced partial order

3 Conditional Rule Nets (CR-Nets)

We now move on to the contributions of this paper. In a sequential voting rule, the local voting rule that is used for a given issue is always the same, that is, the local voting *rule* does not depend on the decisions made on earlier issues (though, of course, the voters' *preferences* for this issue do depend on those decisions).

However, in many cases, it makes sense to let the local voting rules depend on the values of preceding issues. For example, let us consider again the setting in Example 1, and let us suppose that the caterer is collecting the votes and making the decision based on some rule. Suppose the order of voting is $\mathbf{M} > \mathbf{W}$. Suppose the main course is determined to be beef. One would expect that, conditioning on beef being selected, most voters prefer red wine (*e.g.*, $r \succ p \succ w$). Still, it can happen that even conditioned on beef being selected, surprisingly, slightly more than half the voters vote for white wine ($w \succ p \succ r$), and slightly less than half vote for red ($r \succ p \succ w$). In this case, the caterer, who knows that in the general population most people prefer red to white given a meal of beef, may "overrule" the preference for white wine among the slight majority of the voters, and select red wine anyway. While this may appear somewhat snobbish on the part of the caterer, in fact she may be acting in the best interest of social welfare if we take the non-voting agents (who are likely to prefer red given beef) into account.

In this section, we introduce *conditional rule nets (CR-nets)* to model voting rules where the local rules depend on the values chosen for earlier issues. A CR-net is defined similarly to a CP-net—the difference is that CPTs are replaced by conditional rule tables (CRTs), which specify a local voting rule over D_i for each issue $\mathbf{x}_i$ and each setting of the parents of $\mathbf{x}_i$.[2]

Definition 1. *An (acyclic)* conditional rule net (CR-net) $\mathcal{M}$ *over* $\mathcal{X}$ *is composed of the following two parts.*

1. A directed acyclic graph G *over* $\{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$.

2. A set of conditional rule tables *(CRTs) in which, for any variable* $\mathbf{x}_i$ *and any setting* $\boldsymbol{d}$ *of* $Par_G(\mathbf{x}_i)$*, there is a* local conditional voting rule $\mathcal{M}|_{\mathbf{x}:\boldsymbol{d}}$ *over* D_i.

A CR-net encodes a voting rule over all $\mathcal{O}$-legal profiles (we recall that we fix $\mathcal{O} = \mathbf{x}_1 > \cdots > \mathbf{x}_p$ in this paper). For any $1 \leq i \leq p$, in the ith step, the value d_i is determined by applying $\mathcal{M}|_{\mathbf{x}_i:d_1\cdots d_{i-1}}$ (the local rule specified by the CR-net for the ith issue given that the earlier issues take the values $d_1 \cdots d_{i-1}$) to $P|_{\mathbf{x}_i:d_1\cdots d_{i-1}}$ (the profile of preferences over the ith issue, given that the earlier issues take the values $d_1 \cdots d_{i-1}$). Formally, for any $\mathcal{O}$-legal profile P, $\mathcal{M}(P) = (d_1, \ldots, d_p)$ is defined as follows: $d_1 = \mathcal{M}|_{\mathbf{x}_1}(P|_{\mathbf{x}_1})$, $d_2 = \mathcal{M}|_{\mathbf{x}_2:d_1}(P|_{\mathbf{x}_2:d_1})$, etc. Finally, $d_p = \mathcal{M}|_{\mathbf{x}_p:d_1\cdots d_{p-1}}(P|_{\mathbf{x}_p:d_1\cdots d_{p-1}})$.

A CR-net $\mathcal{M}$ is *separable* if there are no edges in the graph of $\mathcal{M}$. That is, the local voting rule for any issue is independent of the values of all other issues (which corresponds to a sequential voting rule).

[2] It is not clear how a cyclic CR-net could be useful, so we only define acyclic CR-nets.

4 Restricting Voters' Preferences

We now consider restrictions on preferences. A restriction on preferences (for a single voter) rules out some of the possible preferences in $L(\mathcal{X})$. Following the convention of [11], a *preference domain* is a set of all admissible profiles, which represents the restricted preferences of the voters. Usually a preference domain is the Cartesian product of the sets of restricted preferences for individual voters. A natural way to restrict preferences in a multi-issue domain is to restrict the preferences on individual issues. For example, we may decide that $r \succ w \succ p$ is not a reasonable preference for wine (regardless of the choice of main course), and therefore rule it out (assume it away). More generally, which preferences are considered reasonable for one issue may depend on the decisions for the other issues. Hence, in general, for each i, for each setting $\boldsymbol{d}_i$ of the issues before issue $\mathbf{x}_i$, there is a set of "reasonable" (or: possible, admissible) preferences over $\mathbf{x}_i$, which we call $\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$. Formally, *admissible conditional preference sets*, which encode all possible conditional preferences of voters, are defined as follows.

Definition 2. *An* admissible conditional preference set $\mathcal{S}$ *over* $\mathcal{X}$ *is composed of multiple* local conditional preference sets*, denoted by* $\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$*, such that for any* $1 \leq i \leq p$ *and any* $\boldsymbol{d}_i \in D_1 \times \cdots \times D_{i-1}$*,* $\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$ *is a set of (not necessarily all) linear orders over* D_i*.*

That is, for any $1 \leq i \leq p$ and any $\boldsymbol{d}_i \in D_1 \times \cdots \times D_{i-1}$, $\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$ encodes the voter's local language over issue i, given the preceding issues taking values $\boldsymbol{d}_i$. In other words, if $\mathcal{S}$ is the admissible conditional preference set for a voter, then we require the voter's preferences over $\mathbf{x}_i$ given $\boldsymbol{d}_i$ to be in $\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$.

An admissible conditional preference set restricts the possible CP-nets, preferences, and lexicographic preferences. We note that Le Breton and Sen [11] defined a similar structure, which works specifically for separable votes.

Now we are ready to define the restricted preferences of a voter over $\mathcal{X}$. Let $\mathcal{S}$ be the admissible conditional preference set for the voter. A voter's admissible vote can be generated in the following two steps: first, a CP-net $\mathcal{N}$ is constructed such that for any $1 \leq i \leq p$ and any $\boldsymbol{d}_i \in D_1 \times \cdots D_{i-1}$, the restriction of $\mathcal{N}$ on $\mathbf{x}_i$ given $\boldsymbol{d}_i$ is chosen from $\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$; second, an extension of $\mathcal{N}$ is chosen as the voter's vote. By restricting the freedom in either of the two steps (or both), we obtain a set of restricted preferences for the voter. Hence, we have the following definitions.

Definition 3. *Let* $\mathcal{S}$ *be an admissible conditional preference set over* $\mathcal{X}$*.*

- *CPnets*$(\mathcal{S}) = \{\mathcal{N} : \mathcal{N}$ *is a CP-net over*$\mathcal{X}$*, and* $\forall i \forall \boldsymbol{d}_i \in D_1 \times \cdots \times D_{i-1}, \mathcal{N}|_{\mathbf{x}_i:\boldsymbol{d}_i} \in \mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}\}$.
- *Pref*$(\mathcal{S}) = \{V : V \sim \mathcal{N}, \mathcal{N} \in$ *CPnets*$(\mathcal{S})\}$.
- $LD(\mathcal{S}) = \{Lex(\mathcal{N}) : \mathcal{N} \in$ *CPnets*$(\mathcal{S})\}$.

That is, CPnets$(\mathcal{S})$ is the set of all CP-nets over $\mathcal{X}$ corresponding to preferences that are consistent with the admissible conditional preference set $\mathcal{S}$. Pref$(\mathcal{S})$ is the set of all linear orders that are consistent with the admissible conditional preference set $\mathcal{S}$. $LD(\mathcal{S})$, which we call the *lexicographic preference domain*, is the subset of linear orders in Pref$(\mathcal{S})$ that are lexicographic. For any $L \subseteq$ Pref$(\mathcal{S})$, we say that L *extends* $\mathcal{S}$ if for any CP-net in CPnets($\mathcal{S}$), there exists at least one linear order in L consistent with that CP-net. It follows that $LD(\mathcal{S})$ extends $\mathcal{S}$; in this case, for any CP-net $\mathcal{N}$ in CPnets($\mathcal{S}$), there exists exactly one linear order in $LD(\mathcal{S})$ that extends $\mathcal{N}$. Lexicographic preference domains are natural extensions of admissible conditional preference sets, but they are also quite restrictive, since any CP-net only has one lexicographic extension.

We now define a notion of richness for admissible conditional preference sets. This notion says that for any issue, given any setting of the earlier issues, each value of the current issue can be the most-preferred one.[3]

Definition 4. *An admissible conditional preference set $\mathcal{S}$ is* rich *if for each $1 \leq i \leq p$, each valuation $\boldsymbol{d}_i$ of the preceding issues, and each $a_i \in D_i$, there exists $V^i \in \mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}$ such that a_i is ranked in the top position of V^i.*

We remark that richness is a natural requirement, and it is also a very weak restriction in the following sense. It only requires that when a voter is asked about her (local) preferences over $\mathbf{x}_i$ given $\boldsymbol{d}_i$, she should have the freedom to at least specify her most preferred local alternative in D_i at will. We note that $|\mathcal{S}|_{\mathbf{x}_i:\boldsymbol{d}_i}|$ can be as small as $|D_i|$ (by letting each alternative in D_i be ranked in the top position exactly once), which is in sharp contrast to $|L(D_i)| = |D_i|!$ (when all local orders are allowed).

A CR-net $\mathcal{M}$ is *locally strategy-proof* if all its local conditional rules are strategy-proof over their respective local domains (we recall that the voters' local preferences must be in the corresponding local conditional preference set). That is, for any $1 \leq i \leq p$, $\boldsymbol{d}_i \in D_1 \times \cdots \times D_{i-1}$, $\mathcal{M}|_{\mathbf{x}_i:\boldsymbol{d}_i}$ is strategy-proof over $\prod_{j=1}^{n} \mathcal{S}_j|_{\mathbf{x}_i:\boldsymbol{d}_i}$.

5 Strategy-Proof Voting Rules in Lexicographic Preference Domains

In this section, we present our main theorem, which characterizes strategy-proof voting rules that satisfy non-imposition, when the voters' preferences are restricted to lexicographic preference domains. Our main theorem states the following: if each voter's preferences are restricted to the lexicographic preference domain for a rich admissible conditional preference set, then a voting rule that satisfies non-imposition is strategy-proof if and only if it is a locally strategy-proof CR-net. We recall that in this paper, there are at least two issues with at least two possible values each, and the lexicographic preference domain for a rich admissible conditional preference set $\mathcal{S}$ is composed of all lexicographic extensions of the CP-nets that are constructed from $\mathcal{S}$.

Theorem 1. *Under Assumption 1, for any $1 \leq j \leq n$, suppose $\mathcal{S}_j$ is a rich admissible conditional preference set, and voter j's preferences are restricted to the lexicographic preference domain of $\mathcal{S}_j$. Then, a voting rule r that satisfies non-imposition is strategy-proof if and only if r is a locally strategy-proof CR-net.*

Sketch of Proof. The "if" part is easy. The "only if" part is proved by induction on p (the number of issues). More precisely, suppose the theorem holds for p issues. For $p+1$

[3] This is *not* the same richness notion as the one proposed by Le Breton and Sen, which applies to preferences over all alternatives rather than to admissible conditional preference sets.

issues, let r be a strategy-proof voting rule that satisfies non-imposition. We first prove that r can be decomposed in the following way: there exists a local rule r_1 over D_1 and a voting rule $r_{\mathbf{x}_{-1}:a_1}$ over $D_2 \times \cdots \times D_{p+1}$ for each $a_1 \in D_1$, such that for any profile P, the first component of $r(P)$ is determined by applying r_1 to the projection of P on $\mathbf{x}_1$, and the remaining components are determined by applying $r_{\mathbf{x}_{-1}:a_1}$ to the restriction of P on the remaining issues given $\mathbf{x}_1 = a_1$, where a_1 is the first component of $r(P)$ (just determined by r_1). Moreover, we prove that r_1 and $r_{\mathbf{x}_{-1}:a_1}$ (for all $a_1 \in D_1$) satisfy non-imposition and strategy-proofness. Therefore, by the induction hypothesis, for each $a_1 \in D_1$, $r_{\mathbf{x}_{-1}:a_1}$ is a locally strategy-proof CR-net over $D_2 \times \cdots \times D_{p+1}$. It follows that r is a locally strategy-proof CR-net over $D_1 \times \cdots \times D_{p+1}$, in which the (unconditional) rule for $\mathbf{x}_1$ is r_1, and given any $a_1 \in D_1$, the sub-CR-net conditioned on $\mathbf{x}_1 = a_1$ is $r_{\mathbf{x}_{-1}:a_1}$. □

The proofs of all theorems are omitted due to the space constraint. All proofs can be found in the long version of this paper on the first author's website.

It follows from Theorem 1 that any sequential voting rule that is composed of locally strategy-proof voting rules is strategy-proof over lexicographic preference domains, because a sequential voting rule is a separable CR-net. Specifically, when the multi-issue domain is binary (that is, for any $1 \leq i \leq p$, $|D_i| = 2$), the sequential composition of majority rules is strategy-proof when the profiles are lexicographic. It is interesting to view this in the context of previous works on the strategy-proofness of sequential composition of majority rules: Lacy and Niou [20] and Le Breton and Sen [11] showed that the sequential composition of majority rules is strategy-proof when the profile is restricted to the set of all separable profiles; on the other hand, Lang and Xia [21] showed that this rule is not strategy-proof when the profile is restricted to the set of all $\mathcal{O}$-legal profiles.

The restriction to lexicographic preferences is still limiting. Next, we investigate whether there is any other preference domain for the voters on which the set of strategy-proof voting rules that satisfy non-imposition is equivalent to the set of all locally strategy-proof CR-nets. The answer to this question is "No," as shown in the next result. More precisely, over any preference domain that extends an admissible conditional preference set, the set of strategy-proof voting rules satisfying non-imposition and the set of locally strategy-proof CR-nets satisfying non-imposition are identical *if and only if* the preference domain is lexicographic.

Theorem 2. *Under Assumption 1, for any $1 \leq j \leq n$, suppose $\mathcal{S}_j$ is a rich admissible conditional preference set, $L_j \subseteq$ Pref$(\mathcal{S}_j)$, and L_j extends $\mathcal{S}_j$. If there exists $1 \leq j \leq n$ such that L_j is not the lexicographic preference domain of $\mathcal{S}_j$, then there exists a locally strategy-proof CR-net $\mathcal{M}$ that satisfies non-imposition and is not strategy-proof over $\prod_{j=1}^{n} L_j$.*

6 An Impossibility Theorem

In this section, we present an impossibility theorem for strategy-proof voting rules when voters' preferences are restricted to be $\mathcal{O}$-legal.

Theorem 3. *When the set of alternatives is a multi-issue domain, if each voter can choose any linear order in $Legal(\mathcal{O})$ to represent her preferences, then there is no strategy-proof voting rule that satisfies non-imposition, except a dictatorship.*

This impossibility theorem is a variant of the Gibbard-Satterthwaite theorem. We emphasize that there are at least two issues with at least two possible values each, and $Legal(\mathcal{O})$ is much smaller than the set of all linear orders over $\mathcal{X}$. Therefore, the theorem does *not* follow directly from Gibbard-Satterthwaite. It follows directly from either of the two stronger impossibility theorems proved in the full version of the paper: one is for extensions of lexicographic domains, and the other is for extensions of the "rich" domains defined by Le Breton and Sen [11]. Due to the space constraint and the heavy technicality and notation of the two impossibility theorems, we omit them.

We recall that Lang and Xia [21] showed that a specific sequential voting rule (the sequential composition of majority rules) is not strategy-proof when each voter can choose any linear order in $Legal(\mathcal{O})$ to represent her preferences. Theorem 3 is much stronger, in that it states that over such a preference domain, not only does the sequential composition of majority rules fail to be strategy-proof, but in fact all non-dictatorial voting rules that satisfy non-imposition fail to be strategy-proof; moreover, this holds for non-binary multi-issue domains as well.

7 Conclusion

In settings where a group of agents needs to make a joint decision, the set of alternatives often has a multi-issue structure. In this paper, we characterized strategy-proof voting rules when the voters' preferences are represented by acyclic CP-nets that follow a common order over issues. We showed that if each voter's preferences are restricted to a lexicographic preference domain, then a voting rule satisfying non-imposition is strategy-proof if and only if it is a locally strategy-proof CR-net. We then proved that if the profile is allowed to be any $\mathcal{O}$-legal profile, then the only strategy-proof voting rules satisfying non-imposition are dictatorships.

Our result for lexicographic preferences is quite positive; however, beyond that, our results do not inspire much hope for desirable strategy-proof voting rules in multi-issue domains. Of course, it is well known that it is difficult to obtain strategy-proofness in voting settings in general, and this does not mean that we should abandon voting as a general method. Similarly, difficulties in obtaining desirable strategy-proof voting rules in multi-issue domains should not prevent us from studying voting rules for multi-issue domains altogether. From a mechanism design perspective, strategy-proofness is a very strong criterion, which corresponds to implementation in dominant strategies. It may well be the case that rules that are not strategy-proof still result in good outcomes in practice—or, more formally, in (say) Bayes-Nash equilibrium.

Acknowledgements

We thank Jérôme Lang and the anonymous reviewers for EC and WINE for helpful discussions and comments. Lirong Xia is supported by a James B. Duke Fellowship and

Vincent Conitzer is supported by an Alfred P. Sloan Research Fellowship. This work is also supported by NSF under award numbers IIS-0812113 and CAREER 0953756.

References

1. Aswal, N., Chatterji, S., Sen, A.: Dictatorial domains. Economic Theory 22(1), 45–62 (2003)
2. Barberà, S., Gul, F., Stacchetti, E.: Generalized median voter schemes and committees. Journal of Economic Theory 61(2), 262–289 (1993)
3. Barberà, S., Masso, J., Neme, A.: Voting under constraints. Journal of Economic Theory 76(2), 298–321 (1997)
4. Barberà, S., Sonnenschein, H., Zhou, L.: Voting by committees. Econometrica 59(3), 595–609 (1991)
5. Bartholdi III, J., Orlin, J.: Single transferable vote resists strategic voting. Social Choice and Welfare 8(4), 341–354 (1991)
6. Bartholdi III, J., Tovey, C., Trick, M.: The computational difficulty of manipulating an election. Social Choice and Welfare 6(3), 227–241 (1989)
7. Black, D.: On the rationale of group decision-making. Journal of Political Economy 56(1), 23–34 (1948)
8. Border, K.C., Jordan, J.S.: Straightforward elections, unanimity and phantom voters. The Review of Economic Studies 50(1), 153–170 (1983)
9. Boutilier, C., Brafman, R., Domshlak, C., Hoos, H., Poole, D.: CP-nets: A tool for representing and reasoning with conditional ceteris paribus statements. Journal of Artificial Intelligence Research 21, 135–191 (2004)
10. Brams, S., Kilgour, D., Zwicker, W.: The paradox of multiple elections. Social Choice and Welfare 15(2), 211–236 (1998)
11. Le Breton, M., Sen, A.: Separable preferences, strategyproofness, and decomposability. Econometrica 67(3), 605–628 (1999)
12. Conitzer, V., Sandholm, T.: Nonexistence of voting rules that are usually hard to manipulate. In: Proc. of AAAI, pp. 627–634 (2006)
13. Conitzer, V., Sandholm, T., Lang, J.: When are elections with few candidates hard to manipulate? Journal of the ACM 54(3), Article 14, 1–33 (2007)
14. Faliszewski, P., Hemaspaandra, E., Schnoor, H.: Copeland voting: Ties matter. In: Proc. of AAMAS, pp. 983–990 (2008)
15. Friedgut, E., Kalai, G., Nisan, N.: Elections can be manipulated often. In: Proc. of FOCS, pp. 243–249 (2008)
16. Gibbard, A.: Manipulation of voting schemes: a general result. Econometrica 41, 587–602 (1973)
17. Hemaspaandra, E., Hemaspaandra, L.A.: Dichotomy for voting systems. Journal of Computer and System Sciences 73(1), 73–83 (2007)
18. Isaksson, M., Kindler, G., Mossel, E.: The geometry of manipulation – a quantitative proof of the Gibbard Satterthwaite theorem. In: Proc. of FOCS (2010)
19. Ju, B.-G.: A characterization of strategy-proof voting rules for separable weak orderings. Social Choice and Welfare 21(3), 469–499 (2003)
20. Lacy, D., Niou, E.M.S.: A problem with referendums. Journal of Theoretical Politics 12(1), 5–31 (2000)
21. Lang, J., Xia, L.: Sequential composition of voting rules in multi-issue domains. Mathematical Social Sciences 57(3), 304–324 (2009)
22. Moulin, H.: On strategy-proofness and single peakedness. Public Choice 35(4), 437–455 (1980)

23. Nehring, K., Puppe, C.: Efficient and strategy-proof voting rules: A characterization. Games and Economic Behavior 59(1), 132–153 (2007)
24. Procaccia, A.D., Rosenschein, J.S.: Average-case tractability of manipulation in voting via the fraction of manipulators. In: Proc. of AAMAS (2007)
25. Procaccia, A.D., Rosenschein, J.S.: Junta distributions and the average-case complexity of manipulating elections. Journal of Artificial Intelligence Research 28, 157–181 (2007)
26. Rossi, F., Venable, K., Walsh, T.: mCP nets: Representing and reasoning with preferences of multiple agents. In: Proc. of AAAI, pp. 729–734 (2004)
27. Satterthwaite, M.: Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. Journal of Economic Theory 10, 187–217 (1975)
28. Walsh, T.: Where are the really hard manipulation problems? The phase transition in manipulating the veto rule. In: Proc. of IJCAI, pp. 324–329 (2009)
29. Walsh, T.: An empirical study of the manipulability of single transferable voting. In: Proc. of ECAI (2010)
30. Xia, L., Conitzer, V.: Generalized scoring rules and the frequency of coalitional manipulability. In: Proc. of EC, pp. 109–118 (2008)
31. Xia, L., Conitzer, V.: A sufficient condition for voting rules to be frequently manipulable. In: Proc. of EC, pp. 99–108 (2008)
32. Xia, L., Conitzer, V., Lang, J.: Voting on multiattribute domains with cyclic preferential dependencies. In: Proc. of AAAI, pp. 202–207 (2008)
33. Xia, L., Conitzer, V., Lang, J.: Aggregating preferences in multi-issue domains by using maximum likelihood estimators. In: Proc. of AAMAS, pp. 399–408 (2010)
34. Xia, L., Conitzer, V., Procaccia, A.D.: A scheduling approach to coalitional manipulation. In: Proc. of EC, pp. 275–284 (2010)
35. Xia, L., Lang, J., Ying, M.: Strongly decomposable voting rules on multiattribute domains. In: Proc. of AAAI (2007)
36. Xia, L., Zuckerman, M., Procaccia, A.D., Conitzer, V., Rosenschein, J.: Complexity of unweighted coalitional manipulation under some common voting rules. In: Proc. of IJCAI (2009)
37. Zuckerman, M., Procaccia, A.D., Rosenschein, J.S.: Algorithms for the coalitional manipulation problem. Artif. Intell. 173(2), 392–412 (2009)

Optimal Iterative Pricing over Social Networks (Extended Abstract)

Hessameddin Akhlaghpour[1], Mohammad Ghodsi[1,2,⋆], Nima Haghpanah[3], Vahab S. Mirrokni[4], Hamid Mahini[1], and Afshin Nikzad[5]

[1] Computer Engineering Department, Sharif University of Technology
akhlaghpour@ce.sharif.edu, ghodsi@sharif.edu,
mahini@ce.sharif.edu
[2] Institute for Research in Fundamental Sciences (IPM), P.O. Box: 19395-5746, Tehran, Iran
[3] Northwestern University, EECS Department
nima.haghpanah@eecs.northwestern.edu
[4] Google Research NYC, 76 9th Ave, New York, NY 10011
mirrokni@gmail.com
[5] Carnegie Mellon University, Tepper School of Business
anikzad@andrew.cmu.edu

Abstract. We study the optimal pricing for revenue maximization over social networks in the presence of positive network externalities. In our model, the value of a digital good for a buyer is a function of the set of buyers who have already bought the item. In this setting, a decision to buy an item depends on its price and also on the set of other buyers that have already owned that item. The revenue maximization problem in the context of social networks has been studied by Hartline, Mirrokni, and Sundararajan [4], following the previous line of research on optimal viral marketing over social networks [5,6,7].

We consider the Bayesian setting in which there are some prior knowledge of the probability distribution on the valuations of buyers. In particular, we study two iterative pricing models in which a seller iteratively posts a new price for a digital good (visible to all buyers). In one model, re-pricing of the items are only allowed at a limited rate. For this case, we give a FPTAS for the optimal pricing strategy in the general case. In the second model, we allow very frequent re-pricing of the items. We show that the revenue maximization problem in this case is inapproximable even for simple deterministic valuation functions. In the light of this hardness result, we present constant and logarithmic approximation algorithms when the individual distributions are identical.

1 Introduction

Despite the rapid growth, online social networks have not yet generated significant revenue. Most efforts to design a comprehensive business model for monetizing such social networks [9,10], are based on contextual display advertising [12]. An alternative way to monetize social networks is viral marketing, or advertising through word-of-mouth.

⋆ This author's work has been partially supported by IPM School of CS (contract: CS 1388-2-01).

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 415–423, 2010.

This can be done by understanding the *externalities* among buyers in a social network. The increasing popularity of these networks has allowed companies to collect and use information about inter-relationships among users of social networks. In particular, by designing certain experiments, these companies can determine how users influence each others' activities.

Consider an item or a service for which one buyer's valuation is influenced by other buyers. In many settings, such influence among users are positive. That is, the purchase value of a buyer for a service increases as more people use this service. In this case, we say that buyers have *positive externalities* on each other. Such phenomena arise in various settings. For example, the value of a cell-phone service that offers extra discounts for calls among people using the same service, increases as more friends buy the same service. Such positive externality also appears for any high-quality service through positive reviews or the word-of-mouth advertising.

By taking into account the positive externalities, sellers can employ forward-looking pricing strategies that maximize their long-term expected revenue. For this purpose, there is a clear trade-off between the revenue extracted from a buyer at the beginning, and the revenue from future sales. For example, the seller can give large discounts at the beginning to convince buyers to adopt the service. These buyers will, in turn, influence other buyers and the seller can extract more revenue from the rest of the population, later on. Other than being explored in research papers [4], this idea has been employed in various marketing strategies in practice, e.g., in selling TiVo digital video recorders [11].

Preliminaries. Consider a case of selling multiple copies of a digital good (with no cost for producing a copy) to a set V of n buyers. In the presence of network externality, the *valuation* of buyer i for the good is a function of buyers who already own that item, $v_i : 2^V \rightarrow R$, i.e., $v_i(S)$ is the value of the digital good for buyer i, if set S of buyers already own that item. We say that users have *positive externality* on each other, if and only if $v_i(S) \leq v_i(T)$ for each two subsets $S \subseteq T \subseteq V$. In general, we assume that the seller is not aware of the exact value of the valuation functions, but she knows the distribution $f_{i,S}$ with an accumulative distribution $F_{i,S}$ for each random variable $v_i(S)$, for all $S \in V$ and any buyer i. Also, we assume that each buyer is interested only in a single copy of the item. The seller is allowed to post different prices at different time steps and buyer i buys the item in a step t if $v_i(S_t) - p_t \geq 0$, where S_t is the set of buyers who own the item in step t, and p_t is the price of the item in that step. Note that $v_i(\emptyset)$ does not need to be zero; in fact $v_i(\emptyset)$ is the value of the item for a user before any other buyer owns the item and influence him.

We study optimal iterative pricing strategies without price discrimination during k *time steps*. In particular, we assume an *iterative posted price* setting in which we post a *public price* p_i at each step i for $1 \leq i \leq k$. The price p_i at each step i is visible to all buyers, and each buyer might decide to buy the item based on her valuation for the item and the price of the item in that time step. We consider myopic or impatient buyers who buy an item at the first time in which the offered price is less than their valuations. In order to formally define the problem, we should also define each time step. A time step can be long enough in which the influence among users can propagate completely, and we can not modify the price when there is a buyer who is interested to buy the item at the current price. On the other extreme, we can consider settings in which the price of the item changes fast enough that we do not allow the influence amongst buyers to propagate in the same time step. In this setting, as we change the price per time step, we assume the influence among buyers will be effective on the next time step (and not on the same time step). In the following, we define these two problems formally.

Definition 1. The Basic(k) **Problem.** *In the* Basic(k) *problem, our goal is to find a sequence* $p_1, \ldots, p_k$ *of* k *prices in* k *consecutive time steps or days. A buyer decides to buy the item during a time step as soon as her valuation is more than or equal to the price offered in that time step. In contrast to the* Rapid(k) *problem, the buyer's decision in a time step immediately affects the valuations of other buyers in the same time step. More precisely, a time step is assumed to end when no more buyers are willing to buy the item at the price at this time step.*

Definition 2. The Rapid(k) **Problem.** *Given a number* k*, the* Rapid(k) *problem is to design a pricing policy for* k *consecutive* days *or* time steps*. In this problem, a pricing policy is to set a public price* p_i *at the start of time step (or day)* i *for each* $1 \leq i \leq k$*. At the start of each time step, after the public price* p_i *is announced, each buyer decides whether to buy the item or not, based on the price offered on that time step and her valuation. In the* Rapid(k) *problem, the decision of a buyer during a time step is not affected by the action of other buyers in the same time step*[1].

One insight about the Rapid(k) model is that buyers react slowly to the new price and the seller can change the price before the news spreads through the network. On the other hand, in the Basic(k) model, buyers immediately become aware of the new state of the network (the information spreads fast), and therefore respond to the new state of the world before the seller is capable of changing prices. Note that in the Basic(k) problem, the price sequence will be decreasing. If the price posted at any time step is greater than the previous price, no buyer would purchase the product at that time step.

A common assumption studied in the context of network externalities is the assumption of *submodular influence functions*. This assumption has been explored and justified by several previous work in this framework [3,4,5,7]. In the context of revenue maximization over social networks, Hartline et. al. [4] state this assumption as follows: suppose that at some time step, S is the set of buyers who have bought the item. We use the notion of *optimal (myopic) revenue* of a buyer for S, which is $R_i(S) = \max_p p \cdot (1 - F_{i,S}(p))$. Following Hartline et.al [4], we consider the optimal revenue function as the *influence function*, and assume that the optimal revenue functions (or influence functions) are submodular, which means that for any two subsets $S \subset T$, and any element $j \notin S$, $R_i(S \cup \{j\}) - R_i(S) \geq R_i(T \cup \{j\}) - R_i(T)$. In other words, submodularity corresponds to a diminishing return property of the optimal revenue function which has been observed in the social network context [3,5,7].

Definition 3. *We say that all buyers have identical initial distributions if there exists a distribution* F_0 *so that the valuation of a player is the sum of two independent random variables, one from* F_0*, and another one from* $F_{i,S}$*, with* $F_{i,\emptyset} = 0$.

[1] We use the terms time step and day interchangeably.

Definition 4. *A probability distribution f with accumulative distribution F satisfies the monotone hazard rate condition if the function $h(p) = f(p)/(1 - F(p))$ is monotone non-decreasing.*

Our Contributions. We first show that the deterministic $\mathsf{Basic(k)}$ problem is polynomial-time solvable. Moreover, for the Bayesian $\mathsf{Basic(k)}$ problem, we present a fully polynomial-time approximation scheme. We study the structure of the optimal solution by performing experiments on randomly generated preferential attachment networks. In particular, we observe that using a small number of price changes, the seller can achieve almost the maximum achievable revenue by many price changes. In addition, this property seems to be closely related to the role of externalities. In particular, the density of the random graph, and therefore the role of network externalities increases, fewer number of price changes are required to achieve almost optimal revenue. We show our experiments in the full version.

Next we show that in contrast to the $\mathsf{Basic(k)}$ problem, the $\mathsf{Rapid(k)}$ problem is intractable. For the $\mathsf{Rapid(k)}$ problem, we show a strong hardness result: we show that the $\mathsf{Rapid(k)}$ problem is not approximable within any reasonable approximation factor even in the deterministic case unless P=NP. This hardness result holds even if the influence functions are submodular and the probability distributions satisfy the monotone hazard rate condition. In the light of this hardness result, we give an approximation algorithm using a minor and natural assumption. We show that the $\mathsf{Rapid(k)}$ problem for buyers with submodular influence functions and probability distributions with the monotone hazard rate condition, and *identical initial distributions* admits logarithmic approximation if k is a constant and a constant-factor approximation if $k \geq n^{\frac{1}{c}}$ for any constant c.

Related work. Optimal viral marketing over social networks have been studied extensively in the computer science literature [6]. For example, Kempe, Kleinberg and Tardos [5] study the following algorithmic question (posed by Domingos and Richardson [3]): How can we identify a set of k influential nodes in a social network to influence such that after convincing this set to use this service, the subsequent adoption of the service is maximized? Most of these models are inspired by the dynamics of adoption of ideas or technologies in social networks and only explore influence maximization in the spread of a *free* good or service over a social network [3,5,7]. As a result, they do not consider the effect of pricing in adopting such services. On the other hand, the pricing (as studied in this paper) could be an important factor on the probability of adopting a service, and as a result in the optimal strategies for revenue maximization.

In an earlier work, Hartline, Mirrokni, and Sundararajan [4] study the optimal marketing strategies in the presence of such positive externalities. They study optimal adaptive ordering and pricing by which the seller can maximize its expected revenue. However, in their study, they consider the marketing settings in which the seller can go to buyers one by one (or in groups) and offer a price to those specific buyers. Allowing such price discrimination makes the implementation of such strategies hard. Moreover, price discrimination, although useful for revenue maximization in some settings, may result in a negative reaction from buyers [8].

2 The Basic(k) Problem

We define $B^1(S,p) := \{i|v_i(S) \geq p\} \cup S$. Assume a time step where at the beginning, we set the global price p, and the set S of players already own the item. So $B^1(S,p)$ specifies the set of buyers who immediately want to buy (or already own) the item. As $B^1(S,p)$ will own the item before the time step ends, we can recursively define $B^k(S,p) = B^1(B^{k-1}(S,p),p)$ and use induction to reason that $B^k(S,p)$ will own the item in this time step. Let $B(S,p) = B^{\hat{k}}(S,p)$, where $\hat{k} = \max\{k|B^k(S,p) - B^{k-1}(S,p) \neq \emptyset\}$, knowing that all buyers in $B(S,p)$ will own the item before the time step ends. One can easily argue that the set $B(S,p)$ does not depend on the order of users who choose to buy the item.

Solving Deterministic Basic(1)**.** In the Basic(1) problem, the goal is to find a price p_1 such that $p_1 \cdot |B(\emptyset,p_1)|$ is maximized. Let $\beta_i := sup\{p|i \in B(\emptyset,p)\}$ and $\beta := \{\beta_i|1 \leq i \leq n\}$. WLOG we assume that $\beta_1 > \beta_2 \ldots > \beta_n$. Player i will buy the item if and only if the price is set to be less than or equal to β_i.

Lemma 1. *The optimal price* p_1 *is in the set* β*.*

Now we provide an algorithm to find p_1 by finding all elements of the set β and considering the profit $\beta_i \cdot B(\emptyset,\beta_i)$ of each of them, to find the best result. Throughout the algorithm, we will store a set S of buyers who have bought the item and a global price g. In the beginning $S = \emptyset$ and $g = \infty$. The algorithm consists of $|\beta|$ steps. At the i-th step, we set the price equal to the maximum valuation of remaining players, considering the influence set to be S. We then update the state of the network until it stabilizes, and moves to the next step. Our main claim is as follows. At the end of the i-th step, the set who own the item is $B(\emptyset,\beta_i)$, and the maximum valuation of any remaining player is equal to β_{i+1}.

Generalization to Deterministic Basic(k)**.** We attempt to solve the Basic(k) problem by executing the Basic(1) algorithm. We are looking for an optimal sequence $(p_1,p_2,\ldots,p_k)$ in order to maximize $\sum_{i=1}^{k} |B(\emptyset,p_i) - B(\emptyset,p_{i-1})| \cdot p_i$. We claim that an optimal sequence exists such that for every i, $p_i = \beta_j$ for some $1 \leq j \leq |\beta|$. This can be shown by a proof similar to that of lemma 1. Thus the problem Basic(k) can be solved by considering the subproblem $A[k',m]$ where we must choose a non-increasing sequence π of k' prices from the set $\{\beta_1,\beta_2,\ldots\beta_m\}$, to maximize the profit, and setting the price at the last day to β_m. This subproblem can be solved using the following dynamic program: $A[k',m] = \max_{1\leq t<m} A[k'-1,t] + |B(\emptyset,\beta_m) - B(\emptyset,\beta_t)| \cdot \beta_m$.

FPTAS for the Bayesian setting. For the Bayesian (or probabilistic) Basic(k) problem, we run a similar dynamic program, but the main difficulty for this problem is that the space of prices is continuous, and we do not have the same set of candidate prices as we have for the deterministic case. To overcome this issue, we employ a natural idea of discritizing the space of prices. Then we estimate the expected revenue by a sampling technique.

3 The Rapid(k) Problem

As we will see in theorem 2, the Rapid(k) problem is hard to approximate even with submodular influence functions and probability distributions satisfying the monotone hazard rate condition. So we consider the Rapid(k) problem with submodular influence functions and probability distributions satisfying the monotone hazard rate condition, and buyers have identical initial distributions. For this problem, we present an approximation algorithm whose approximation factor is logarithmic for a constant k and its approximation factor is constant for $k \geq n^{\frac{1}{c}}$ for any constant $c > 0$ (See Algorithm 1).

Algorithm 1. Approximation algorithm for Rapid(k) problem

1: Compute a price p_0 which maximizes $p(1 - F_0(p))$ and let R_0 be this maximum value.
2: Compute a price $p_{1/2}$ such that $F_0(p_{1/2}) = 0.5$.
3: With probability $\frac{1}{2}$, let $c = 1$, otherwise $c = 2$.
4: **if** $c = 1$ **then**
5: Set the price to the optimal myopic price of F_0 (i.e, p_0) on the first time step and terminate the algorithm after the first time step.
6: **else** $\{c = 2\}$
7: Post the price $p_{1/2}$ on the first time step.
8: Let $\overline{S}$ be the set of buyers that do not buy in the first day, and let their optimal revenues be $R_1(V - \overline{S}) \geq R_2(V - \overline{S}) \geq \ldots \geq R_{|\overline{S}|}(V - \overline{S})$.
9: Let p_j be the price which achieves $R_j(V - \overline{S})$, and Pr_j be the probability with which j accepts p_j for any $1 \leq j \leq |\overline{S}|$. Thus we have $R_j(V - \overline{S}) = p_j Pr_j$.
10: Let $d_1 < d_2 < \ldots < d_{k-1}$ be the indices returned by lemma 6 as an approximation of the area under the curve $R(V - \overline{S})$ maximizing $\sum_{j=1}^{k-1}(d_j - d_{j-1}) \cdot R_{d_j}(V - S)$.
11: Sort prices $\frac{p_{d_j}}{e}$ for $1 \leq j \leq k - 1$, and offer them in non-increasing order in days 2 to k.
12: **end if**

To analyze the expected revenue of the algorithm, we need the following lemmas:

Lemma 2. *Let S be the set formed by sampling each element from a set V independently with probability at least p. Also let f be a submodular set function defined over V, i.e., $f : 2^V \rightarrow R$. Then we have $E[f(S)] \geq pf(V)$ [4].*

Lemma 3. *If the valuation of a buyer is derived from a distribution satisfying the monotone hazard rate condition, she will accept the optimal myopic price with probability at least $1/e$ [4].*

Lemma 4. *Suppose that f is a probability distribution satisfying the monotone hazard rate condition, with expected value μ and myopic revenue $R = \max_p p(1 - F(p))$. Then we have $R(1 + e) \geq \mu$.*

Lemma 5. *Let i be the index maximizing ia_i in the set $\{a_1, a_2, \ldots, a_m\}$. Then we have $ia_i \geq \sum_{j=1}^{m} a_j / (\lceil \log(m + 1) \rceil)$.*

Lemma 6. *For a set $\{a_1 \geq a_2 \geq \ldots \geq a_n\}$, let $D = \{d_1 \leq d_2 \leq \ldots \leq d_k\}$ be the set of indices maximizing $S(D) = \sum_{j=1}^{k}(d_j - d_{j-1})a_{d_j}$ (assuming $d_0 = 0$), over all sequences of size k. Then we have $S(D) \in \Theta(\frac{\sum_i a_i}{\log_k n})$.*

Proof idea. We present an algorithm that iteratively selects rectangles, such that after the m-th step the total area covered by the rectangles is at least $m/\log n$ using $4^m - 1$ rectangles. At the start of the m-th step, the uncovered area is partitioned into 4^{m-1} independent parts. In addition, the length of the lower edge of each of these parts is e_p which is at most $n/(2^{m-1})$. The algorithm solves each of these parts independently as follows. We use 3 rectangles for each part in each step. First, using lemma 5 we know that we can use a single rectangle to cover at least $1/\log e_p$ of the total area of part. Then, we cover the two resulting uncovered parts by two rectangles, which each equally divide the lower edge of the corresponding part.

Theorem 1. *The expected revenue of the algorithm 1 is at least* $\frac{1}{8e^2(e+1)\log_k n}$ *of the optimal revenue.*

Proof. For simplicity assume that we are allowed to set $k+1$ prices. In case $c = 1$, we set the optimal myopic price of all players and therefore achieve the expected revenue of nR_0. If $c = 2$, consider the second day of the algorithm. By lemma 3, we know that each remaining buyer accepts her optimal myopic price with probability at least $1/e$, so for every j we have $Pr_j \geq 1/e \geq Pr_i/e$. In addition, we know that for each $j \leq i$, $R_j(V - \overline{S}) \geq R_i(V - \overline{S}) \geq p_i/e$. We also know that $R_j(V - \overline{S}) \leq p_j$. As a result, $p_j \geq p_i/e$, for each $j \leq i$. Therefore, if we offer the player $j \leq i$ the price p_i/e, she will accept it with probability at least Pr_i/e (she would have accepted p_j with probability at least $Pr_j \geq Pr_i/e$; offering a lower price of p_i/e will only increase the probability of acceptance).

For now suppose that we are able to partition players to k different groups, and offer each group a distinct price. Ignore the additional influence that players can have on each other. In that case, we can find a set $d_1 < d_2 < \ldots < d_k$ maximizing $\sum_{j=1}^{k}(d_j - d_{j-1}) \cdot R_{d_j}(V - \overline{S})$. Assume that D_i is the set of players y with $d_{i-1} < y \leq d_i$. As we argued above, if we offer each of these players the price p_{d_i}/e, she will accept it with probability at least Pr_{d_i}/e. So the expected value of each of the players in D_i when offered p_{d_i}/e is at least $Pr_{d_i}/e \cdot p_{d_i}/e = R_{d_i}(V - \overline{S})/e^2$. The total expected revenue in this case will be $\sum_{j=1}^{k}(d_i - d_{i-1}) \cdot R_{d_j}(V - \overline{S})/e^2$, which, using lemma 6 is at least $\sum_i R_i(V - \overline{S})/(e^2 \log_k n)$. An important observation is that, if the expected revenue of a player when she is offered a price p is R, her expected revenue will not decrease when she is offered a non-increasing price sequence P which contains p . As a result, we can sort the prices that are offered to different groups, and offer them to *all players* in non-increasing order.

Finally, using Lemma 2, and since every player buys at the first day independently with probability 1/2, we conclude that any buyer i that remains at the second day observe an expected influence of $R_i(V)/2$ from all other buyers.

As a result, the expected revenue of our algorithm is $nR_0/2$ (from setting p_0 with probability 1/2 in the first day) plus $\sum_i R_i(V) \cdot (1/8) \cdot (1/(e^2 \log_k n))$. Since we set $p_{1/2}$ with probability 1/2, a player does not buy at first day with probability 1/2, and we achieve $1/(e^2 \log_k n)$ of the value of remaining players in the second day. We also know that the expected revenue that can be extracted from any player i is at most $E(F_0) + E(F_{i,V})$. Thus, using lemma 4, we conclude that the approximation factor of the algorithm is $8e^2(e+1)\log_k n$.

At last, we prove the hardness of the Rapid(k) problem even in the deterministic case with additive (modular) valuation functions. Specifically, we consider the following special case of the problem: (i) $k = n$; (ii) The valuations of the buyers are deterministic, i.e., $f_{i,S}$ is an impulse function, and its value is nonzero only at $v_i(S)$; and finally (iii) The influence functions are additive; $\forall i, j, S$ such that $i \neq j$ and $i, j \notin S$ we have $v_i(S \cup \{j\}) = v_i(S) + v_i(\{j\})$, also each two buyers $i \neq j$, $v_i(\{j\}) \in \{0, 1\}$, and each buyer has a non-negative initial value, i.e, $v_i(\emptyset) \geq 0$.

We use a reduction from the *independent set* problem; We show that using any $\frac{1}{n^{1-\epsilon}}$-approximation algorithm for the specified subproblem of Rapid(k), any instance of the *independent set* problem can be solved in polynomial time. We discard details here and show how to construct an instance of Rapid(k) from an instance of the independent set problem in the full version.

Theorem 2. *The* Rapid(k) *problem with additive influence functions can not be approximated within any multiplicative factor unless P=NP.*

4 Concluding Remarks

In this paper, we introduce new models for studying the optimal pricing and marketing problems over social networks. We study two specific models and show a major difference between the complexity of the optimal pricing in these settings. This paper leaves many problems for future studies.

- We presented results for myopic buyers, but many problems remain open for strategic buyers. Studying optimal pricing strategies for strategic or patient buyers is an interesting problem. In fact, one can model the pricing problem for the seller and the optimal strategy for buyers as a game among buyers and the seller, and study equilibria of such a game. Two possible models have been proposed in [1,2].
- We studied a monopolistic setting in which a seller does not compete with other sellers. It would be nice to study this problem in the non-monpolistic settings in which other sellers may provide similar items over time, and the seller should compete with other sellers to attract parts of the market.

References

1. AhmadiPourAnari, N., Ehsani, S., Ghodsi, M., Haghpanah, N., Immorlica, N., Mahini, H., Mirrokni, V.S.: Equilibrium pricing with positive externalities. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 424–431. Springer, Heidelberg (2010)
2. Candogan, O., Bimpikis, K., Ozdaglar, A.: Optimal pricing in the presence of local network effects. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 118–132. Springer, Heidelberg (2010)
3. Domingos, P., Richardson, M.: Mining the network value of customers. In: KDD 2001, pp. 57–66. ACM, New York (2001)
4. Hartline, J., Mirrokni, V.S., Sundararajan, M.: Optimal marketing strategies over social networks. In: WWW, pp. 189–198 (2008)
5. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: KDD 2003, pp. 137–146. ACM, New York (2003)

6. Kleinberg, J.: Cascading behavior in networks: algorithmic and economic issues. Cambridge University Press, Cambridge (2007)
7. Elchanan, M., Sebastien, R.: On the submodularity of influence in social networks. In: STOC 2007, pp. 128–134. ACM, New York (2007)
8. Oliver, R.L., Shor, M.: Digital redemption of coupons: Satisfying and dissatisfying e®ects of promotion codes. Journal of Product and Brand Management 12, 121–134 (2003)
9. Oswald, E.: http://www.betanews.com/article/Google_Buy_MySpace_Ads_for_900m/1155050350
10. Seeyle, K.Q.: http://www.nytimes.com/2006/08/23/technology/23soft.html
11. Walker, R.: http://www.slate.com/id/1006264/
12. Weber, T.: http://news.bbc.co.uk/1/hi/business/6305957.stm?lsf

Equilibrium Pricing with Positive Externalities (Extended Abstract)

Nima Anari[1], Shayan Ehsani[1], Mohammad Ghodsi[1,2,⋆], Nima Haghpanah[3], Nicole Immorlica[3], Hamid Mahini[1], and Vahab S. Mirrokni[4]

[1] Computer Engineering Department, Sharif University of Technology
anari@eecs.berkeley.edu, {ehsani,mahini}@ce.sharif.edu, ghodsi@sharif.edu

[2] Institute for Research in Fundamental Sciences (IPM), P.O. Box: 19395-5746, Tehran, Iran

[3] Northwestern University, EECS Department
{nima.haghpanah,nicimm}@gmail.com

[4] Google Research NYC, 76 9th Ave, New York, NY 10011
mirrokni@gmail.com

Abstract. We study the problem of selling an item to strategic buyers in the presence of positive *historical externalities*, where the value of a product increases as more people buy and use it. This increase in the value of the product is the result of resolving bugs or security holes after more usage. We consider a continuum of buyers that are partitioned into *types* where each type has a valuation function based on the actions of other buyers. Given a fixed sequence of prices, or *price trajectory*, buyers choose a day on which to purchase the product, i.e., they have to decide whether to purchase the product early in the game or later after more people already own it. We model this strategic setting as a game, study existence and uniqueness of the equilibria, and design an FPTAS to compute an approximately revenue-maximizing pricing trajectory for the seller in two special cases: the *symmetric* settings in which there is just a single buyer type, and the *linear* settings that are characterized by an initial type-independent bias and a linear type-dependent influenceability coefficient.

1 Introduction

Many products like software, electronics, or automobiles evolve over time. When a consumer considers buying such a product, he faces a tradeoff between buying a possibly sub-par early version versus waiting for a fully functional later version. Consider, for example, the dilemma faced by a consumer who wishes to purchase the latest Windows operating system. By buying early, the consumer takes full advantage of all the new features. However, operating systems may have more bugs and security holes at the beginning, and hence a consumer may prefer to wait with the rationale that, if more people already own the operating system, then more bugs will have already been uncovered and corrected. The key observation is, the more people that have already used the operating system, or any product for that matter, the more inherent value it accrues. In

⋆ This author's work has been partially supported by IPM School of CS (contract: CS 1388-2-01).

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 424–431, 2010.

other words, the product exhibits a particular type of externality, a so-called *historical externality*[1].

How should a company price a product in the presence of historical externalities? A low introductory price may attract early adopters and hence help the company extract greater revenue from future customers. On the other hand, too low a price will result in significant revenue loss from the initial sales. Often, when faced with such a dilemma, a company will offer an initial promotional price at the product's release in a limited-time offer, and then raise the price after some time. For example, when releasing Windows 7, Microsoft announced a two-week pre-order option for the Home Premium Upgrade version at a discounted price of $50; thereafter the price rose to $120, where it has remained since the pre-sale ended on July 11th, 2009. Additionally, beta testers, who can be interpreted as consumers who "bought" the product even prior to release, received the release version of Windows 7 for free (as is often the case with software beta-testers).

We study this phenomenon in the following stylized model: a monopolistic seller wishes to derive a pricing and marketing plan for a product with historical externalities. To this end, she commits to a price trajectory. Potential consumers observe the price trajectory of the seller and make simultaneous decisions regarding the day on which they will buy the product (and whether to buy at all). The payoff of a consumer is a function of the day on which he bought the product, the price on that day, and the set of consumers who bought before him. We compute the equilibria of the resulting sequential game and observe that the revenue-maximizing price trajectories for the seller are increasing, as in the Windows 7 example above.

A few words are in order about our model. First, we focus on settings in which the seller has the ability to commit to a price trajectory. Such commitments are observed in many settings especially at the outset of a new product (see the Windows 7 example described above) and have been assumed in prior models in the economics literature on pricing as well as in other games in the form of Stackelberg strategies [10]. Further, commitment increases revenue: clearly a seller who commits to price trajectories can extract at least as much revenue as a seller who does not (or can not). We further observe via example in the full version of the paper that in fact commitment enables a seller to extract unboundedly higher revenue than in settings without commitment. Second, we assume a consumer's payoff is only a function of past purchases; i.e., consumers have no utility for future purchases. We motivate this in the Windows 7 example by arguing that bugs are resolved in proportion to usage rates. Of course, strictly speaking, consumers of Windows 7 benefit from future purchases as well via software updates and the like. However, this forward-looking benefit is substantially dampened in comparison to past benefits by safety and security risks, and time commitments involved in updates. Another justification for our payoff model comes from consumers' uncertainties regarding products. In many settings, consumers have signals regarding the value of a product (say an electronic gadget like the iPad for example), but do not observe its precise value until the time of purchase. Past purchases and the ensuing online reviews may help consumers improve their estimates of their values prior to purchase, an especially important factor for risk-adverse buyers.

[1] Note that this is different from the more well-studied notion of externalities in the computer science literature where a product (e.g., a cell phone) accrues value as more consumers buy it simply because the product is used in conjunction with other consumers.

We focus on the non-atomic setting in which we have a continuum of consumers so that each consumer is infinitesimally small and therefore his own action has a negligible effect on the actions of others. Consumers are drawn from a (possibly infinite) set of types. These types capture varying behavior among consumer groups. We study a sequential game in which the seller first commits to a price trajectory and then the consumers simultaneously choose when and whether to buy in the induced normal-form game among them. We study subgame perfect equilibria. We first observe that equilibria exist due to a slight generalization of a paper of Mas-Colell [8] (see the full version of the paper). We then turn to the question of uniqueness. We focus on *well-behaved equilibria* in which consumers with non-negative utility always purchase the product (thus indifferent consumers purchase the product). In general multiple such equilibria may exist. However, in an aggregate model in which the value function of each consumer type depends only on the *aggregate* behavior of the population (i.e., the total fraction of potential consumers that have bought the product and not the total fraction of various types), then we are able to show that when they exist the well-behaved equilibria of this game are unique in the sense that the fraction of purchases per-type-per-day is fixed among all equilibria. This enables us to search for the revenue-maximizing price trajectory. We address this question in settings in which we either have just one type or there are multiple types whose valuation functions are linear in the aggregate, both of which are special cases of the aggregate model discussed above. For each price trajectory, we define its revenue to be the amount of money consumers spend on the product. We then design an FPTAS to find the revenue-maximizing price trajectory for a monopolistic seller in these settings. We do this via a reduction to a novel rectangle covering problem in which we must find the discounted area-maximizing set of rectangles that fit underneath a given curve.

As an interesting consequence of our result, we find that the revenue maximizing price trajectory is an increasing and convex function, matching the intuition that the seller should attract a few early adopters with a low introductory price and then exploit the value they add by offering high prices to remaining consumers. We also note that the distribution of sales in the revenue maximizing equilibrium matches this intuition as well – it is also increasing and convex.

1.1 Related Work

Our work falls in the long line of literature investigating pricing and marketing of products that exhibit externalities [1,2,3,4,5,6,7,9]. Among these, the paper of Bensaid and Lesne [2] is most closely related to our own work. Bensaid and Lesne [2] analyzed the two and infinite period pricing problems in the presence of linear historical externalities and they study equilibria of the induced games both with and without commitment. They observe, as we do, that optimal price trajectories are increasing. The historical externalities that we study generalize the externalities of Bensaid and Lesne [2], and in this more general model, we solve for the optimal price sequence for any fixed number of price periods. Most of the remaining externalities literature studies externalities in which consumers care about the total population of users of a product and hence their utility is affected by future sales as well as past sales. Although the phenomenon studied is different from ours, some of the modeling assumptions in these papers are similar to ours. For example, in the economics literature, Cabral, Salant, and Woroch [3] also consider a seller that commits to a price trajectory and then observe that the revenue-maximizing price sequence with fully rational consumers (playing a Bayesian equilibrium) is increasing. Similar to our model, they study the pricing problem in the presence of a continuum of consumers.

In the computer science literature Akhlaghpour et al. [1] and Hartline et al. [5] study algorithmic questions regarding revenue maximization over social networks for products with externalities. However, their models assume naive behavior for consumers. Namely, they assume consumers act myopically, buying the product on the first day in which it offers them positive utility without reasoning about future prices and sales that could affect optimal buying behavior and long-term utility. Furthermore, Hartline et al. [5] allow the seller to use adaptive price discrimination. In contrast, we model consumers as fully rational agents that strategically choose the day on which to buy based on full information regarding all future states of the world and a sequence of public posted prices. While the correct model of pricing and consumer behavior probably lies somewhere between these two extremes, we believe studying fully rational consumers is an important first step in relaxing myopic assumptions.

2 Model

We wish to study the sale of a good by a monopolistic seller over k days to a set of potential consumers or buyers. We model our setting as a sequential game whose players consist of the monopolistic seller and a continuum of potential consumers or buyers $b \in [0, 1]$. In our game, the seller moves first, selecting a *price trajectory* $p = (p_1, \ldots, p_k)$ where $p_i \in \Re$ assigning a (possibly negative) price p_i to each day i. The buyers move next, selecting a day on which to buy the product given the complete price trajectory, as described below.

The buyers are partitioned into n types $T_1, \ldots, T_n$ where each T_t is a subinterval of $[0, 1]$.[2] The strategy set $A = \{1, \ldots, k\} \cup \{\emptyset\}$ indicates the day on which the product is bought ($\emptyset$ is used to indicate that the product was not purchased). Hence the strategy profile of the buyer population can be represented by a $(k + 1) \times n$ matrix $X = \{X_{i,t}\}_{i=1,\ldots,k+1;t=1,\ldots,n}$ where entry $X_{i,t}$ indicates the fraction of buyers that are of type t and buy the product *before* day i, and we define $X_{1,t} = 0$ for all t. Note that by normalization $\sum_t X_{k+1,t} \leq 1$ and $1 - \sum_t X_{k+1,t}$ is the fraction of buyers that don't buy the product at any time. Corresponding to this matrix X we also define the *marginal strategy profile* matrix $x = \{x_{i,t}\}_{i=1,\ldots,k;t=1,\ldots,n}$ where $x_{i,t} = X_{i+1,t} - X_{i,t}$ is the fraction of buyers who are of type t and buy on day i. In the special case when there is only 1 type, we use X_i as a scalar to denote the fraction of buyers who bought before day i and x_i as a scalar to denote the fraction of buyers who buy on day i.

Given a strategy profile X, we define the value of buyers of type t buying on day i by a value function $F_i^t(X_i)$ where X_i is the i'th row of X (hence buyers are indifferent to future buying decisions). Note the explicit dependence of F on time, which allows $F_i^t(X_i)$ to be different than $F_j^t(X_j)$, for $i \neq j$. The revenue-maximization results in Section 4 further assume that the dependence of $F_i^t(X_i)$ on i is of the form

[2] Later, we will generalize this to infinitely many types.

$F_i^t(X_i) = \beta^i F^t(X)$ for $\beta \in [0,1]$. This special case is of particular interest as the β factor models settings in which the value degrades over time due to, for example, a reduction in the novelty of the product.

Given a strategy profile X, the payoff of buyers of type t who buy on day i is defined to be $F_i^t(X_i) - p_i$. We additionally allow buyers to have a discount factor α such that their payoff is $(1-\alpha)^i(F_i^t(X_i) - p_i)$. Thus α represents the way in which agents discount future payoffs with respect to present payoffs. We say that a strategy profile X is a Nash equilibrium of the induced subgame given by price trajectory p, or equivalently $X \in NE(p)$, if for any buyer of type t who buys on day i we have $i \in \arg\max_j (F_j^t(X_j) - p_j)(1-\alpha)^j$, and the strategy is $\emptyset$ whenever the maximum is negative (in which case the buyer's payoff is zero). We call an equilibrium *well-behaved* if all indifferent buyers buy, i.e., a buyer does not buy if and only if his payoff $(1-\alpha)^i(F_i^t(X_i) - p_i)$ is negative on all days $1 \leq i \leq k$. We say that (p, X) is a (well-behaved) equilibrium if the profile X is a (well-behaved) Nash equilibrium for the subgame of price trajectory p. Equivalently, a marginal strategy profile x is a (well-behaved) Nash equilibrium for the subgame of price trajectory p if for any type t and day i we have $x_{i,t} > 0$ only if $i \in \arg\max_j (F_j^t(X_j) - p_j)(1-\alpha)^j$ and the value of this maximum is non-negative.

Given a price trajectory p and a marginal strategy profile x that arises in the subgame induced by p, we define the payoff of the seller to be the *revenue* of x for p, which is $R(p,x) = \sum_{i=1}^k \sum_{t=1}^n x_{i,t} p_i (1-\alpha)^i$. A *subgame perfect equilibrium* of the sequential game is then a price trajectory p^* and a set of marginal strategy profiles x_p for each possible price trajectory p such that: (1) x_p is a Nash equilibrium of the subgame induced by p, and (2) p^* maximizes $R(p, x_p)$. The *outcome* of this subgame perfect equilibrium is (p^*, x_{p^*}) and its revenue is $R(p^*, x_{p^*})$.

We are interested in computing the outcome in a revenue-maximizing subgame perfect equilibrium. To do so, we must compute a price trajectory which maximizes the revenue of the seller in equilibrium. Note that this is equal to finding the best response of the seller given the strategies $\{x_p\}$ of the buyers. We solve this problem for special settings in which there exist revenue-maximizing well-behaved equilibria in $NE(p)$ for any price trajectory p, allowing us to maximize over them. These settings are as follows. For the purpose of these definitions, we will allow each buyer to have a unique type and hence there are infinitely many types. We will use $b \in [0,1]$ to denote type of buyer b.

Definition 1. *The Aggregate Model: The value function of each type in this model is a function of the aggregate behavior of the population and is invariant with respect to the behavior of each separate type. That is, the value function of buyer b is a function of X_i only, where X_i is a scalar indicating the total fraction of all buyers who buy before day i. In this instance, we overload the notation for the value function and let $F_i^b(X_i)$ indicate the value of buyer b (hence $F_i^b(\cdot)$ now maps the unit interval to the non-negative reals).*

Definition 2. *The Linear Model: This is a special case of the aggregate model which is defined by a function F_i, an initial bias I, and a function C so that the value of buyer b is $F_i^b(X_i) = I + C(b) \cdot F_i(X_i)$. We further define the commonly-known distribution $\mathcal{C} : \mathbb{R} \to [0,1]$ such that $\mathcal{C}(c^*)$ indicates the fraction of buyers b with $C(b) \leq c^*$.*

Definition 3. *The Symmetric Model: In this version we only have one type, that is, $F_i^b = F_i$ for all b.*

We note that alternatively, one could model this pricing game as a sequential game with multiple stages where in each day i the seller selects a price p_i and then buyers simultaneously choose whether to buy or not. Such a model is appropriate when it is not possible for a seller to commit to a price trajectory in advance. Again, in this setting, one could study the subgame perfect equilibria and analyze the resulting revenue. Clearly the revenue with commitment is at least as high as that without commitment. Also, there are examples in which the revenue without commitment can be unboundedly less.

3 Uniqueness of Equilibria

We prove that if there exists a *well-behaved equilibrium*, that is an equilibrium in which everyone with non-negative utility buys on some day, then it is unique. We show this for an infinite number of types in the aggregate model which generalizes both the linear and symmetric models.

Recall that we allow for each buyer $b \in [0, 1]$ to have a unique type in the aggregate model such that the valuation function of buyer b is F_i^b. We will show that in all of the well-behaved equilibrium points the fraction of people buying on each day is the same. In turn, it implies that the revenue of all well-behaved equilibrium points is the same and hence the well-behaved equilibria are revenue-unique. In what follows, we consider the equilibria of a fixed price sequence p. We start with a definition: Consider two well-behaved equilibria x and y. Partition the set of k days to two sets as follows: We call a day i a *level 1* day, and denote it by $i \in D_1(x, y)$, if $X_i < Y_i$. Otherwise, if $X_i \geq Y_i$, we call i a *level 2* day and denote it by $i \in D_2(x, y)$.

Lemma 1. *Assume that there exist two distinct well-behaved equilibria x and y. Then there exists a buyer whose strategy in x is a day i such that $i \in D_1(x, y)$ and whose strategy in y is $j \in D_2(x, y)$.*

Theorem 1. *Let $F_i^b(X)$ be a strictly increasing function for each buyer b and day i. For a price sequence p and two well-behaved equilibrium points x and y, we have $X_i = Y_i$, i.e. the fraction of buyers who have bought the product before day i is unique.*

Proof. Assume for contradiction that we have two well-behaved equilibrium points x and y and a day i for which $X_i \neq Y_i$. Again assume without loss of generality that $X_i < Y_i$. By lemma 1 we know that there exists a buyer b who buys on a level 1 day in x and buys on a level 2 day in y. Assume that b buys on day i in x and on day j in y. Then $F_i^b(X_i) - p_i \geq F_j^b(X_j) - p_j$ and $F_j^b(Y_j) - p_j \geq F_i^b(Y_i) - p_i$. Adding the two inequalities we get: $F_i^b(X_i) + F_j^b(Y_j) \geq F_j^b(X_j) + F_i^b(Y_i)$. On the other hand since i is a level 1 day, $X_i < Y_i$; hence by monotonicity $F_i^b(X_i) < F_i^b(Y_i)$. Since j is a level 2 day, $X_j \geq Y_j$; hence $F_j^b(Y_j) \leq F_j^b(X_j)$. The addition of these two inequalities contradicts the previous one.

4 Revenue Maximization

In this section, we solve the revenue-maximizing problem in two special cases: the discounted version of the symmetric model, and the general linear model without discount factors. In both cases, we provide an FPTAS to compute the revenue-maximizing price sequence. We do this by first showing that in both cases, the revenue maximizing equilibria are well-behaved ones, and then considering the problem of maximizing over well-behaved equilibria. We characterize the set of well-behaved equilibria in each section, and then use novel reductions of the problem into a new problem, called the *Rectangular Covering Problem* (RCP). The RCP is to maximize the discounted area covered by a certain number of rectangles that are fit under a given curve.

Definition 4. Rectangular Covering Problem (RCP). *Given an increasing function F and an integer k, find a sequence p of size at most k that maximizes the discounted total area of the rectangles fit under the graph of F, that is, $p \in \arg\max_{p'} \sum_t (F^{-1}(p'_{t+1}) - F^{-1}(p'_t))p'_t\gamma^t$.*

We provide an FPTAS for the RCP in the full version of the paper. Given the reductions from the revenue maximization problem to rectangular covering problem, this directly gives us FPTASs for the two versions of the problem.

4.1 Symmetric Setting

We start by characterizing the equilibria. Since all players in this model have the same valuation function F, the marginal strategy profile matrix will reduce to the vector $x = (x_1, \ldots, x_k)$. Also, fixing p and x, the utility of buyer b for the item on day i is $F_i^b(X_i) = F(X_i)\beta^i(1-\alpha)^i - p_i(1-\alpha)^i$, and the revenue $R(p, x) = \sum_i x_i p_i (1-\alpha)^i$. By renaming $q_i = p_i(1-\alpha)^i$ and $\gamma = \beta(1-\alpha)$, the utility of buyer b for the item on day i will be $F(X_i)\gamma^i - q_i$, and the revenue becomes $\sum_i x_i q_i$. Using this new notation, we may assume without loss of generality that the only discount factor is γ. For convenience, we use p for the discounted prices q.

Since we only have one type in this model, we know that the utility of buying in day i is equal among all players. We use the term *utility of a day* i, denoted by u_i, for $u_i = F(X_i)\gamma^i - p_i$. Define $u(p, x) = \max_i u_i$. Consider a price sequence and its equilibrium strategy profile x. We get the following properties immediately from the facts that players are utility maximizing: (i) players are allowed to choose inaction and have utility zero, (ii) they choose to buy if there is a day with a strictly positive utility. First, if there is an i with $x_i > 0$, then $u(p, x) \geq 0$ and $u_i = u(p, x)$. Second, if there is a day i with $x_i > 0$, then $\sum_{i=1}^k x_i = 1$.

Lemma 2. *Let $\hat{p}$ be the revenue-maximizing price vector that results in equilibrium $\hat{x}$. Then $u(\hat{p}, \hat{x}) = 0$.*

We use lemma 2 to find a closed form for the revenue of a price sequence. Assume that there is a price sequence p with equilibrium x and $u(p, x) = 0$ such that for some day i, we have $x_i = 0$ and $x_{i+1} > 0$. Then we can define a new price sequence $\tilde{p}$ which is equal to p except that $\tilde{p}_j = p_{j+1}/\gamma$ for each $j \geq i$. Also define the vector $\tilde{x}$ to be equal to x except that $\tilde{x}_j = x_{j+1}$ for each $j \geq i$, and $\tilde{x}_k = 0$. One can observe that the pair $(\tilde{p}, \tilde{x})$ is an equilibrium with no less revenue. So we can assume WLOG that for a revenue maximizing price sequence $\hat{p}$ associated with $\hat{x}$, there exists a $k' \leq k$ such that $x_i \neq 0$ if and only if $i \leq k'$. For such a price sequence, lemma 2 shows that $F(X_i)\gamma^i - p_i = 0$ for each $1 \leq i \leq k'$. As a result, we have $X_i = F^{-1}(p_i/\gamma^i)$, which is well-defined as F is increasing. Now set $p'_t = p_t/\gamma^t$. The fraction of people buying on day i and paying price p_i is equal to $x_i = F^{-1}(p'_{i+1}) - F^{-1}(p'_i)$. So the revenue is $\sum_i x_i p_i = \sum_i (F^{-1}(p'_{i+1}) - F^{-1}(p'_i))p'_i \gamma^i$. The revenue maximization problem therefore reduces to the rectangular covering problem.

4.2 Linear Version

Similar to the symmetric model, we reduce the linear model to rectangular covering problem by first characterizing the set of well-behaved equilibria. The sketch of this more technical proof is as follows. We show that in each equilibria all the purchases are sorted by the C_b coefficient, i.e., a player with a lower C_b buys earlier than one with higher such coefficient. We then argue that each equilibria is characterized by a sequence of thresholds $(s_1, s_2, \ldots, s_k)$ such that each person b with $C_b \in [s_{i-1}, s_i]$ buys in day i. The problem is then to optimize the sequence $(s_1, s_2, \ldots, s_k)$ to maximize revenue. Using this characterization, we provide a closed form of the optimum revenue in the following lemma.

Lemma 3. *If x and p correspond to the revenue-maximizing equilibrium, the total revenue can be expressed by the following formula $R(p,x) = I + \sum_{i=2}^{k}(1-X_i)\mathcal{C}^{-1}(X_i) \times (F(X_i) - F(X_{i-1}))$.*

We then show how this problem can be reduced to RCP in the following lemma.

Lemma 4. *The problem of maximizing $\sum_{i=2}^{k}(1 - X_i)\mathcal{C}^{-1}(X_i)(F(X_i) - F(X_{i-1}))$ can be reduced to the Rectangular Covering Problem.*

References

1. Akhlaghpour, H., Ghodsi, M., Haghpanah, N., Mahini, H., Mirrokni, V., Nikzad, A.: Optimal iterative pricing over social networks. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 118–132. Springer, Heidelberg (2010)
2. Bensaid, B., Lesne, J.-P.: Dynamic monopoly pricing with network externalities. International Journal of Industrial Organization 14(6), 837–855 (1996)
3. Cabral, L., Salant, D., Woroch, G.: Monopoly pricing with network externalities. Industrial Organization 9411003, EconWPA (Novemeber 1994)
4. Farrell, J., Saloner, G.: Standardization, compatibility, and innovation. RAND Journal of Economics 16(1), 70–83 (Spring 1985)
5. Hartline, J., Mirrokni, V.S., Sundararajan, M.: Optimal marketing strategies over social networks. In: WWW, pp. 189–198 (2008)
6. Katz, M.L., Shapiro, C.: Network externalities, competition, and compatibility. American Economic Review 75(3), 424–440 (1985)
7. Kempe, D., Kleinberg, J., Tardos, É.: Influential nodes in a diffusion model for social networks. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 1127–1138. Springer, Heidelberg (2005)
8. Mas-Colell, A.: On a theorem of schmeidler. Journal of Mathematical Economics 13(3), 201–206 (1984)
9. Mason, R.: Network externalities and the coase conjecture. European Economic Review 44(10), 1981–1992 (2000)
10. von Stackelberg, H.R.: Marktform und gleichgewicht. Springer, Heidelberg (1934)

The Good, The Bad and The Cautious: Safety Level Cooperative Games

Yoram Bachrach[1], Maria Polukarov[2], and Nicholas R. Jennings[2]

[1] Microsoft Research, Cambridge, UK
yobach@microsoft.com
[2] University of Southampton, UK
mp3,nrj@ecs.soton.ac.uk

Abstract. We study *safety level coalitions* in competitive games. Given a normal form game, we define a corresponding *cooperative* game with transferable utility, where the value of each coalition is determined by the safety level payoff it derives in the original—*non-cooperative*—game. We thus capture several key features of agents' behavior: (i) the possible monetary transfer among the coalition members; (ii) the solidarity of the outsiders against the collaborators; (iii) the need for the coalition to optimize its actions against the worst possible behavior of those outside the coalition. We examine the concept of safety level cooperation in *congestion games*, and focus on computing the value of coalitions, the core and the Shapley value in the resulting safety level cooperative games. We provide tractable algorithms for *anonymous* cooperative games and for safety level cooperative games that correspond to *symmetric* congestion games with singleton strategies. However, we show hardness of several problems such as computing values in games with multi-resource strategies or asymmetric strategy spaces.

1 Introduction

Game theory analyzes interactions of selfish rational agents. An agent may not follow a "prescribed" behavior if deviating from it improves its utility, so *stable* outcomes are central in game theory. In *non-cooperative* games, where agents take individual actions, the prominent stability concept is the *Nash equilibrium*—a strategy profile where no agent has a beneficial unilateral deviation. However, it does not take into account collective deviations by groups of agents; The *strong equilibrium* [2]—a strategy profile with no profitable agent subset deviations—extends Nash equilibrium to coalitions. *Cooperative* games consider how coalitions of agents cooperate, focusing on how the utility is distributed among the agents. In a non-cooperative game, agents act independently based on their individual interests. In many scenarios traditionally modeled as non-cooperative games (e.g., auctions, network and congestion games), a coalition can *jointly* decide on a collective action and make monetary transfers to *share* the gains. This requires applying tools from cooperative game theory to such domains. Hayrapetyan et al [10] modeled coalitions in *congestion games*. In a congestion game [15], a set of agents shares a set of resources, and an agent's strategy is to choose a subset of resources to use to minimize the sum of congestion-dependent costs over its selected resources. In [10], the authors assume that agents may collude to maximize their collective welfare. Their model allows monetary transfers but uses a different perspective

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 432–443, 2010.

than cooperative game theory, focusing on the negative effect of collusion on the social welfare. Other papers examine coalition formation in multi-unit auctions [3], assuming non-colluders bid *truthfully* or fostering cooperation through *external subsidies* [4,14].

In contrast, we study *safety level* coalitions in competitive games. As opposed to standard cooperative games, the utility of a coalition depends not only on the action the members take, but also on the actions taken by the non-members. In the worst case, the outsiders may "punish" the coalition members and take actions that minimize the collaborators' utility. A coalition then may decide to maximize its total) utility under the worst case action of the non-members—we call this a *joint safety level* strategy. In our model the collaborators are "good" to each other by coordinating actions and sharing gains and the non-collaborators are "bad" adversaries who reduce the collaborators' utility. If the collaborators adopt this view of non-collaborators as adversaries, they must be "cautious" and prepare for the worst-case choice of the non-collaborators, using their joint safety level strategy. To do this, they can agree on monetary transfers through an enforceable contract for distributing the gains. Solution concepts such as the core [9] and the Shapley value [18] can be used to predict what transfers would occur. Several works [7,8,21,5] consider computing the core and Shapley value in various domains.

We examine safety level cooperation in congestion games. These games where selfish agents choose from a common set of resources and derive individual utilities that depend on the total congestion on each resource, are fundamental to many applications [13,15,16]. Such games have Nash equilibria in pure strategies [15], and some restricted classes have strong equilibria in pure strategies [11,17]. Recent work focuses on specific subclasses that are computationally tractable [1,12]. One subclass which we also examine is *resource selection games*, where each agent chooses a *single* resource.

We distinguish between *symmetric* settings where agents choose strategies from a common space, and *asymmetric* ones where each agent has its own collection of strategies. While in the non-cooperative context, both symmetric and asymmetric models are anonymous, asymmetric models lose anonymity when monetary transfers are allowed. For anonymous settings, we show that testing core emptiness, constructing a core imputation and testing whether an imputation is in the core are in P when the computation of the coalitional values in the game is in P; we also show that the Shapley value is in the core if it is not empty, and can be computed in polynomial time. These results hold for *all* anonymous cooperative games[1]—not only those based on safety level coalitions. For congestion games, we show that computing a coalition's value is in P for singleton strategies and NP-hard for multiple-resource strategies, while for non-anonymous settings computing the value of even a singleton or the grand coalition are NP-hard.

1.1 Preliminaries

A *non-cooperative game in normal form* is given by an agent set $\mathbf{N} = \{1, \ldots, N\}$, and for each agent $i \in \mathbf{N}$, a strategy space $\mathbf{S}_i$ of its pure strategies and a payoff function $U_i : \times_{i \in \mathbf{N}} \mathbf{S}_i \rightarrow \mathbb{R}$ specifying the reward an agent gets. Denote by $\mathbf{S}_C$ the set of partial strategy profiles of a subset of agents $C \subseteq \mathbf{N}$, and by $\mathbf{S}_{-C} = \mathbf{S}_{\mathbf{N} \setminus C}$ the set

[1] Not to be confused with the *anonymity-proof* solutions [20] which are robust under "false name" manipulations. We refer to games where the characteristic function is not "sensitive" to the agents' identities so equal size coalitions get equal values.

of strategy combinations of all the agents outside C; for a single agent $i \in \mathbf{N}$, denote $\mathbf{S}_{-i} = \mathbf{S}_{\mathbf{N}\setminus\{i\}}$. A strategy profile $s \in \mathbf{S}$ is a *Nash equilibrium* if for each agent $i \in \mathbf{N}$ and for each its strategy $s'_i \in \mathbf{S}_i$ the following holds: $U_i(s) \geq U_i(s_{-i}, s'_i)$. A strategy profile is a *strong Nash equilibrium* if it is stable against deviations by coalitions: for any $C \subseteq \mathbf{N}$ and $s'_C \in \mathbf{S}_C$, there exists $i \in C$ such that $U_i(s) \geq U_i(s_{-i}, s'_i)$. The *safety level strategy* for agent $i \in \mathbf{N}$, s_i^{SL}, is the strategy maximizing its *guaranteed* utility, no matter what the other agents play: $s_i^{SL} \in \arg\max_{s_i \in \mathbf{S}_i} \min_{s_{-i} \in \mathbf{S}_{-i}} U_i(s_i, s_{-i})$.

Some utility functions ignore the identities of the agents, and only take into account *the number of times* each strategy is played. Settings where identities are irrelevant are *anonymous* . Given a set of strategies $\mathbf{S} = \{1, \ldots, S\}$, a strategy $s \in \mathbf{S}$ and an agent $i \in \mathbf{N}$, the utility of i playing s in an anonymous maps the set of partitions $\left\{(x_1, \ldots, x_S) \,|\, x_j \in \{1, \ldots, N\}, \sum_{j=1}^{S} x_j = N - 1\right\}$ to real numbers. A related important subclass is *symmetric games*, where the payoffs for playing a particular strategy are the same for different agents and depend only on the other strategies employed, so one can change the identities of the agents without changing the payoffs to the strategies. A game with strategy spaces $\mathbf{S}_1 = \ldots = \mathbf{S}_N = \mathbf{S}$ is symmetric if for any permutation π over $\mathbf{N}$ and agent $i \in \mathbf{N}$, we have $U_i(s_1, \ldots, s_i, \ldots, s_N) = U_{\pi(i)}\left(s_{\pi(1)}, \ldots, s_{\pi(i)}, \ldots, s_{\pi(N)}\right)$, where $s_j = s_{\pi(j)}$ for $j = 1, \ldots, N$.

A *transferable utility cooperative game* has a set $\mathbf{N}$ of N agents, and a characteristic function $v : 2^{\mathbf{N}} \rightarrow \mathbb{R}$ mapping any subset (coalition) of agents to a real value, indicating the total utility these agents achieve together. We denote all agents except i as $\mathbf{N}_{-i} = \mathbf{N} \setminus \{i\}$. A coalitional game is *monotone* if $v(C') \leq v(C)$ for any $C' \subseteq C$.

The characteristic function only indicates the total gains a coalition can achieve, but does not specify how these gains are distributed among the agents who formed it. An *imputation* $(p_1, \ldots, p_N)$ defines a division of the gains of the grand coalition among its agents, where $p_i \in \mathbb{R}$, such that $\sum_{i=1}^{N} p_i = v(\mathbf{N})$. We call p_i the payoff of agent i, and denote the payoff of a coalition C as $p(C) = \sum_{i \in C} p_i$. A basic requirement for a good imputation is *individual rationality*: for any agent $i \in \mathbf{N}$, $p_i \geq v(\{i\})$ (otherwise, this agent is incentivized to work alone). Similarly, we say a coalition B *blocks* imputation $(p_1, \ldots, p_N)$ if $p(B) < v(B)$. If a blocked payoff vector is chosen, the coalition is somewhat unstable. The most prominent solution concept based on such *stability* is the core [9]. The *core* is the set of all imputations $(p_1, \ldots, p_N)$ not blocked by any coalition, so that for any coalition $C \subseteq \mathbf{N}$ holds $p(C) \geq v(C)$.

Another solution concept is the Shapley value [18] which focuses on *fairness* It fulfills several important fairness axioms [18] and has been used to fairly share gains or costs. It depends on the agent's marginal contribution to possible coalition permutations. We denote by π a permutation (ordering) of the agents, and by Π the set of all possible such permutations. Given a permutation $\pi = (i_1, \ldots, i_N) \in \Pi$, the *marginal worth vector*, $m^{\pi}(v) \in \mathbb{R}^N$, is defined by $m^{\pi}_{i_1} = v(\{i_1\})$ and $m^{\pi}_{i_k}(v) = v(\{i_1, i_2, \ldots, i_k\}) - v(\{i_1, i_2, \ldots, i_{k-1}\})$ for $k > 1$. The convex hull of all the marginal vectors is called the *Weber Set*, and contains the game's core. The *Shapley value* is the centroid of the marginal vectors: $\phi(v) = \frac{1}{N!} \sum_{\pi \in \Pi} m^{\pi}(v)$.

We analyze the core and the Shapley value of cooperative games that arise when considering safety-level coalitions in a given non-cooperative setting, and demonstrate this approach on congestion games. In a *congestion game* (CG) [15], every agent has to choose from a finite set of resources. The utility of an agent from using a particular resource depends on the number of agents using it, and its total utility is the sum of utilities on its used resources. Formally, a congestion game $\Gamma = \left(\mathbf{N}, \mathbf{R}, (u_r(\cdot))_{r\in\mathbf{R}}\right)$ is described by the following components: a set $\mathbf{N} = \{1, \ldots, N\}$ of agents; a set $\mathbf{R} = \{r_1, \ldots, r_R\}$ of resources; an assignment $u_r : \{1, \ldots, N\} \to \mathbb{R}$, $r \in \mathbf{R}$, of resource utility functions, where for any resource $r \in \mathbf{R}$, $u_r(k)$ is the resource utility (cost) for r when the total number of users of r is k. Each agent i is allowed to choose a (non-empty) bundle of resources $B \subseteq 2^{\mathbf{R}}$, from a certain set $\mathbf{S}_i = \{B_1^i, \ldots, B_{S_i}^i\}$ of allowed bundles (where each $B_j^i \subseteq \mathbf{R}$). We denote by $s_i \in \mathbf{S}_i$ the strategy (set of resources) chosen by agent i. Every N-tuple of strategies—a *strategy profile*—$s = (s_i)_{i\in\mathbf{N}}$ corresponds to an R-dimensional congestion vector $h(s) = (h_r(s))_{r\in\mathbf{R}}$ where $h_r(s)$ is the number of agents who select resource r (we simply write h_r when it's clear what profile we refer to). The utility of i from s is: $U_i(s) = \sum_{e\in s_i} u_r(h_r(s))$. A congestion game is a *resource selection* game (RSG) if the strategy space of every agent corresponds to a set of singletons. That is, agent i chooses a single resource from the given set, and its payoff from a strategy profile $s = (s_i)_{i\in N}$ is given by $U_i(s) = u_{s_i}(h_{s_i}(s))$.

Remark 1. In a congestion game an agent's utility only depends on the numbers of agents choosing each resource but not on their identities, so congestion games are anonymous. Since the utility from each resource is the same for each of its users, the utility any agent gets from a particular strategy depends only on the other strategies selected, but not on who has chosen them. Thus a congestion game is symmetric if (and only if) all agents in the game have identical strategy spaces. We refer to symmetric congestion and resource selection games as SCGs and SRSGs, respectively.

Congestion games always have a pure strategy Nash equilibrium [15]. Resource selection games with monotone utility functions also admit strong equilibria [11]. In fact, in RSGs with decreasing utilities, any Nash equilibrium is strong. However, we show that coalitional stability is no longer guaranteed if utility transfers are allowed.

2 Safety Level Cooperative Games

Let $\Gamma = \left(\mathbf{N}, (\mathbf{S}_i)_{i\in\mathbf{N}}, (U_i)_{i\in\mathbf{N}}\right)$ be a normal-form game, where $\mathbf{N}$ is the agent set, and $\mathbf{S}_i$ and U_i denote, respectively, strategy spaces and utility functions of *individual* agents. We are interested in scenarios where it makes sense to the agents to form coalitions and coordinate their actions to optimize their *collective* gains and take a *safety level* approach to analyzing gains of a coalition. We assume that the coalition members attempt to maximize the minimal utility they would get under any strategy choices of the non-members. We model coordination in the underlying normal-form game as a coalitional game, where coalitional values are determined by the safety-level payoffs of each coalition. We first extend the notion of a safety level to coalitional payoffs.

For coalition $C \subseteq \mathbf{N}$ and strategy profile $s = (s_i)_{i\in\mathbf{N}}$, let $U_C(s) = \sum_{i\in C} U_i(s)$ be the total utility C achieves under s. The coalition's utility depends not only on the strategies chosen by its members, but also on the choices of the non-members. Let $B = \mathbf{N} \setminus C$ denote the set of non-members. A profile s can be written as $s = (s_B, s_C)$, where $s_C = (s_i)_{i\in C}$ and $s_B = (s_j)_{j\in B}$ are partial strategy profiles. Given the non-members' strategy s_B, the coalition could optimize for the total value it can achieve, by choosing $s^*_C \in \arg\max_{s_C \in \mathbf{S}_C} U_C(s_B, s_C)$, where $\mathbf{S}_C = \times_{i \in C} \mathbf{S}_i$ is the set of coalitional strategies of C. This choice maximizes C's utility for a *specific* strategy profile of B. What should coalition C do without knowing how the non-members would behave? Staying on the "safe" side, C can optimize the utility guaranteed to it, no matters what the outsiders do, by maximizing its *safety level*, the worst case utility the coalition obtains under all possible actions of the non-members. The safety level of C when it chooses s_C is: $U_C^{SL}(s_C) = \min_{s_B \in \mathbf{S}_B} U_C(s_B, s_C)$, and the *safety level strategy* of a coalition C is the coalitional strategy $s^*_C \in \mathbf{S}_C$ that maximizes the safety level:

$$s^*_C \in \arg\max_{s_C \in \mathbf{S}_C} U_C^{SL}(s_C) = \arg\max_{s_C \in \mathbf{S}_C} \left(\min_{s_B \in \mathbf{S}_B} U_C(s_B, s_C) \right)$$

The *safety level value* of C is its minimal utility when using its safety level strategy:

$$U^*_C = \min_{s_B \in \mathbf{S}_B} U_C(s_B, s^*_C) = \max_{s_C \in \mathbf{S}_C} \left(\min_{s_B \in \mathbf{S}_B} U_C(s_B, s_C) \right)$$

A coalition's safety level value is the utility it can guarantee *as a whole* when its members cooperate. A key challenge is determining how the members would share this value. To answer this, we define a *safety level cooperative game* (SLC-game) for Γ:

Definition 1 (Safety Level Cooperative Game). *Given a (normal-form) game* $\Gamma = \left(\mathbf{N}, (\mathbf{S}_i)_{i \in \mathbf{N}}, (U_i)_{i \in \mathbf{N}}\right)$ *with agent set* $\mathbf{N}$*, strategy space* $\mathbf{S}_i$ *and utility function* U_i *for each* $i \in \mathbf{N}$*, the induced* safety level cooperative game (SLC-game) *is a cooperative game over the same set* $\mathbf{N}$ *of agents, where the characteristic function is the safety level value of coalitions in* Γ*: for each* $C \subseteq \mathbf{N}$*,* $v(C) = U^*_C$*.*

We write SLC^Γ to indicate that an SLC-game is induced by a game Γ. Regardless of their underlying games Γ, all SLC-games have the following property:

Lemma 1. *The SLC-games are monotonically increasing.*

Proof. We need to show that for any C', C such that $C' \subseteq C$ we have $v(C') \leq v(C)$. Intuitively, as C includes more agents than C' and the agents in $D = C \setminus C'$ are coalition members for C and outsiders for C', so they "help" the members of C and "punish" the members of C'. Hence, the safety level value of a larger coalition is greater then that of a smaller one. Formally, denote $B = \mathbf{N} \setminus C$, so $\mathbf{N} \setminus C' = B \cup D$. We have:

$$v(C') = U^*_{C'} = \max_{s_{C'} \in \mathbf{S}_{C'}} \left(\min_{s_{B \cup D} \in \mathbf{S}_{B \cup D}} U_{C'}(s_{B \cup D}, s_{C'}) \right) \leq \max_{s_{C'} \in \mathbf{S}_{C'}} \left(\min_{s_B \in \mathbf{S}_B} U_{C'}(s_B, s_D, s_{C'}) \right)$$

$$\leq \max_{s_{C'} \in \mathbf{S}_{C'}} \left(\min_{s_B \in \mathbf{S}_B} U_C(s_B, s_D, s_{C'}) \right) \leq \max_{s_C \in \mathbf{S}_C} \left(\min_{s_B \in \mathbf{S}_B} U_C(s_B, s_C) \right) = U^*_{C'} = v(C)$$

2.1 Safety Level Coalitions in Congestion Games

We analyze safety level coalitions in congestion games and resource selection games. We make a distinction between symmetric settings where agents derive strategies from a common space, and asymmetric settings where each agent has its own collection of strategies. While in the non-cooperative context both symmetric and asymmetric models are anonymous, asymmetric models lose anonymity under monetary transfers. We show anonymous and non-anonymous SLC-games differ computationally.

Consider a congestion game with agents $\mathbf{N}$ and resources $\mathbf{R}$ with resource utility functions $u_r(\cdot)$ for $r \in \mathbf{R}$, and a coalition $C \subseteq \mathbf{N}$. For any strategy profile $s = (s_i)_{i \in \mathbf{N}}$, the congestion on each resource is $h(s) = (h_r(s))_{r \in \mathbf{R}}$, and we can compute the utility $U_i(s)$ for each agent i. C's total utility under s is $U_C(s) = \sum_{i \in C} U_i(s) = \sum_{i \in C} \sum_{r \in s_i} u_r(h_r(s))$. Denote the number of C's members who use a resource r at a strategy profile s as $h_r^C(s) = |\{i \in C | r \in s_i\}|$. We can write: $U_C(s) = \sum_{r \in \mathbf{R}} h_r^C(s) \cdot u_r(h_r(s))$. The coalitional value of C in the corresponding SLC-game is:

$$v(C) = U_C^* = \max_{s_C \in \mathbf{S}_C} \left(\min_{s_B \in \mathbf{S}_B} \sum_{r \in \mathbf{R}} h_r^C(s) \cdot u_r(h_r(s)) \right)$$

Recall our notation of (S)CG and (S)RSG for (symmetric) congestion and resource selection games. Note that $SRSGs \subseteq RSGs \subseteq CGs$ and $SRSGs \subseteq SCGs \subseteq CGs$. Similar inclusions hold for the corresponding safety level game classes.

2.2 Anonymous Cooperative Games

We consider the properties of SLC-games induced by symmetric congestion games, where all agents use a common set of strategies. We start with Lemma 2 showing that these games satisfy *anonymity*. We say a cooperative game is *anonymous* if any two agents are equivalent—i.e., for every two agents $i \neq j$ and any coalition C such that $i \notin C$ and $j \notin C$ we have $v(C \cup \{i\}) = v(C \cup \{j\})$.

Lemma 2. *All SLC^{SCG}-games are anonymous.*

Proof. Consider a coalition C that contains neither i nor j. Since both i and j have identical strategy spaces, we get the same sets for min and max operators when computing coalitional safety level values of $C \cup \{i\}$ and $C \cup \{j\}$.

In anonymous games the Shapley value can be found in polynomial time and is in the core when it's not empty (proofs omitted for lack of space).

Lemma 3 (Core of Anonymous Games). *Let v be an anonymous cooperative game over N agents $\mathbf{N}$, with a non-empty core. Denote $q = \frac{v(\mathbf{N})}{N}$. Then the symmetric payoff distribution $(q, q, \ldots, q)$ is an imputation in the core.*

Lemma 4 (Shapley Value of Anonymous Games). *Let v be an anonymous cooperative game over N agents $\mathbf{N}$. Denote $q = \frac{v(\mathbf{N})}{N}$. Then the Shapley value is the symmetric payoff distribution $(q, q, \ldots, q)$. If the core exists, then the Shapley value is in the core.*

Lemmas 3 and 4 require the non-emptiness of the core. Some safety level games have empty cores (see Examples 1 and 2 below). Empty cores can occur even among the restricted class of SLC-games induced by symmetric, monotone resource selection games, which always possess strong equilibria, highlighting the difference between the cooperative safety level cooperative game's core and strong equilibrium.

In an anonymous game all agents are equivalent so the value of a coalition only depends on the *number* of agents in the coalition and not their identities. Thus, we can write the characteristic function v as a function mapping the size of a coalition to its value, so $v : \{0, 1, \ldots, N\} \rightarrow \mathbb{R}$. We use the standard convention that $v(0) = 0$.

Example 1 (there are SLC^{CG}-games with non-empty core). Consider a SLC^{SRSG}-game with N agents and R resources with *identical*, *constant* resource utility functions $u_r(k) = x \in \mathbb{R}$ for any $r \in \mathbf{R}$, $k = 1, \ldots, N$. This game has a non-empty core.

Proof. Note that the value of any coalition in this domain, no matter what the non-members do, only depends on the size of the coalition, so $v(k) = xk$. Thus, the simple payoff vector $p = (x, \ldots, x)$ is in the core, since given any coalition C of size $|C|$ we have $p(C) = x|C| = v(C)$, and all the core conditions hold.

Example 2 (SLC^{SRSG}-games may have empty core). Consider an SLC^{SRSG}-game with $N = 3$ agents and two resources $\{a, b\}$ with identical resource utility functions $u_r(1) = 2; u_r(2) = u_r(3) = 1$ for $r = a, b$. The core of this game is empty.

Proof. We have $v(0) = 0$. Now compute $v(1)$, the safety level of a *single* agent (out of 3 agents). No matter which resource, a or b, the agent chooses, the worst case outcome is when the other 2 agents also choose the same resource, giving the agent a utility of $u_r(3) = 1$; thus, we have $v(1) = 1$. Now consider the safety level of 2 agents. They can either choose to both use the same resource, or to each use a different resource. If they both are on the same resource, the worst case action of the remaining agent is to also join that resource, and the utility of the coalition is $2u_r(3) = 2$. If the collaborators choose different resources, any choice of the remaining agent results in having 2 agents (one member and one non-member of the coalition) on one resource and a single coalition member on the other resource, resulting in a utility of $u_r(2) + u_r(1) = 2 + 1 = 3$ for the coalition. Thus, the safety level of any pair of agents is $v(2) = 3$. A coalition of 3 agents is the grand coalition, whose best choice is to assign 2 agents on one resource, and 1 agent on the other resource, and so $v(3) = u_r(1) + 2u_r(2) = 2 + 2 = 4$. Thus, the characteristic function of this SLC-game is given by $v(0) = 0, v(1) = 1, v(2) = 3, v(3) = 4$. Due to Lemma 3, if the game has a non-empty core, the imputation $p = (\frac{4}{3}, \frac{4}{3}, \frac{4}{3})$ should be in the core. However, under this imputation the payoff for any two agents is less than the value of a coalition of the pair: i.e., $p(\{1, 2\}) = \frac{8}{3}$, but $v(\{1, 2\}) = v(2) = 3 > \frac{8}{3} = p(\{1, 2\})$, which violates the core constraints. Hence, the core is empty.

Moreover, restricting or expanding the sets of the agents' allowed strategies may cause the core to change from being empty to being non-empty and vice versa:

Example 3 (Strategy Sets and the Core). Consider the game with 3 agents and 2 resources $\{a, b\}$ from the previous example, where the resource utility function is given by $u_r(1) = 2; u_r(2) = u_r(3) = 1$ for $r = a, b$. The core of this game is empty. Now add a third resource c with a constant utility of $u_c(k) = 10$ for $k = 1, 2, 3$, and expand each agent's strategy set to allow selecting $\{c\}$. The resulting game is anonymous, with characteristic function $v(1) = 10, v(2) = 20, v(3) = 30$, and its core is not empty: the imputation $p = (10, 10, 10)$ is in the core. On the other hand, if we take this new game, and restrict each agent's strategy set to allow selecting only $\{a\}$ or $\{b\}$, we obtain the original game with an empty core. Thus, extending strategy sets makes the core non-empty, and restricting them may empty it. Now, consider the game with 3 resources $\{a, b, c\}$, where again $u_r(1) = 2; u_r(2) = u_r(3) = 1$ for $r = a, b$, but $u_c(k) = 0.1$ for $k = 1, 2, 3$. If the agents are restricted to choosing only c, i.e. $\mathbf{S}_1 = \mathbf{S}_2 = \mathbf{S}_3 = \{\{c\}\}$, we have an anonymous game where $v(1) = 0.1, v(2) = 0.2, v(3) = 0.3$ which has a non-empty core as $p = (0.1, 0.1, 0.1)$ belongs to it. If we extend the strategy sets to also include a and b, so that $\mathbf{S}_1 = \mathbf{S}_2 = \mathbf{S}_3 = \{\{a\}, \{b\}, \{c\}\}$, we get the game where $v(1) = 1, v(2) = 3, v(3) = 3$, whose core is empty. Thus, extending strategy sets may make the core empty, and shrinking them makes it non-empty.

Remark 2. Based on the above examples, one can see that the non-cooperative and cooperative concepts of coalitional stability are rather different. While strong Nash equilibria always exist for (monotone) resource selection games, the core of their corresponding SLC-games may be empty. The reason for that is the following: while for any coalition there could be no deviation guaranteeing a better payoff to any of the deviators, there might exist a coalition that can improve its total welfare—that is, even if some agents may obtain worse individual utilities after the deviation, this loss will be covered by the gains their co-deviators get.

In light of the above observations, testing the (non-)emptiness of the core in safety level cooperative games is an important issue. It follows from the next Theorem 1 regarding anonymous cooperative games, that for SLC^{SCG}-games this can be done efficiently if the computation of coalitional values is easy; moreover, in this case, the construction of a core imputation and verification if a given imputation is in the core are also computationally efficient:

Theorem 1 (Core Computation in Anonymous Games). *In anonymous cooperative games, if computing the value of any coalition can be performed in polynomial time, then the following problems are in P: testing for core emptiness, constructing a core imputation (if one exists) and testing if an imputation p is in the core.*

Proof. In anonymous games the characteristic function is given as $v : \{0, 1, \ldots, N\} \to \mathbb{R}$—the function that maps the size of a coalition to its value. This representation is simply a table, containing N numbers: therefore, if computing the value of each coalition can be performed in polynomial time, then finding the characteristic function is also so.

To fulfill the core constraints, the following must hold for an imputation p: $\sum_{i=1}^{N} p_i = v(\mathbf{N}) = v(N)$, and $\forall C, p(C) \geq v(C)$. Consider testing whether an imputation p satisfies this. It is easy to check if $\sum_{i=1}^{N} p_i = v(\mathbf{N}) = v(N)$. However, testing the condition $\forall C, p(C) \geq v(C)$ seemingly requires 2^N similar tests. Order the agents according to their payment, so that $p_{i_1} \leq p_{i_2} \leq \ldots \leq p_{i_N}$. Denote by C_k the coalition $C = \{i_1, i_2, \ldots, i_k\}$. Note that if the core constraint $p(C) \geq v(C)$ holds for $C_k = \{i_1, i_2, \ldots, i_k\}$, it must also hold for any coalition of size k, as C_k is the minimally paid coalition of size k. Thus, to test if p is a core imputation, it is enough to test whether $p(C_k) \geq v(C_k)$ for $k \in \{1, 2, \ldots, N\}$. If the core constrains hold for all $C_1, \ldots, C_N$, they hold for any coalition C, and if they do not, we have a violated constraint. Since there are only N such checks, this can be done in polynomial time.

Now consider testing for core-emptiness and constructing a core imputation. Due to Lemma 3, if the core is non-empty, the symmetric imputation $(q, q, \ldots, q)$ where $q = \frac{v(\mathbf{N})}{N}$ must be in the core. Since q can be computed in polynomial time, this imputation can also be computed in polynomial time. We can then test whether it is in the core. If it is in the core, we have a core imputation, otherwise the core is empty.

The computation of coalition values in SLC-games, can be difficult as the safety level strategies of the agents are not even robust to small changes of game parameters. For instance, we show that even changing only the total number of agents can result in very different safety level strategies, even in simple anonymous settings.

Example 4 (Number of Agents and Safety Level Strategies). Consider an SLC^{SRSG}-game with two resources $\{a, b\}$ with resource utility functions given by $u_a(k) = \epsilon$ for $k = 1, \dots, 5$ an a small positive ϵ, and $u_b(k) = 1 \; \forall k = 1, \dots, 4$, $u_b(5) = 0$. Assume there are $N = 4$ agents playing the game and compute the value of a coalition C of 3 agents (out of 4). Since $u_b(\cdot)$ is constant up to congestion of 4, any agent in C who choses b is guaranteed a utility of 1 on that resource. On the other hand, any agent in C who chooses a only gets a utility of ϵ on that resource. Thus $v(3) = 3 \cdot 1 = 3$, and the safety level strategy of C is to have all its agents choosing the resource b. Now, consider the same resources and resource utility functions when there are $N = 5$ agents, and consider again a coalition C of 3 agents. If C places all agents in b, a possible strategy for the remaining 2 agents is to both join b, resulting in a total utility of $3u_b(5) = 3 \cdot 0 = 0$ for the coalition. Alternatively, the coalition can have 2 agents using a and 1 agent using b. For this strategy in $\mathbf{S}_C$, any strategy in $\mathbf{S}_{\mathbf{N} \setminus C}$ of the remaining 2 agents results in all agents on a getting a utility of ϵ and all those on b getting 1, resulting in a total coalitional utility of $1 \cdot \epsilon + 2 \cdot 1 = 2 + \epsilon$. This is the safety level strategy for the coalition C, so $v(3) = 2 + \epsilon$.

However for SLC^{SRSG}-games we can compute a coalition's value in polynomial time:

Theorem 2. *For SLC^{SRSG}-games, computing safety level strategies is in P.*

Proof. We provide a dynamic programming algorithm. Given an SLC^{SRSG}-game with R resources, for any $k = 1, \dots, R$ let v_k denote the k-subgame, played on the first k resources: that is, v_k is the restriction of the original game where the agents are only allowed to select one of the first k resources—i.e., for each $i \in \mathbf{N}$ we have $\mathbf{S}_i = \{r_1, r_2, \dots, r_k\} \subseteq \mathbf{R}$. Note that v_k is also an SLC^{SRSG}-game. We denote by $v_{i,j,k}$ the value of a coalition of i agents in the k-subgame with $i + j$ agents: to compute $v_{i,j,k}$ we must find a safety level strategy for a coalition of i agents when there are additional j non-members, and the agents are only allowed to select one of the first k resources. We prove that the following recursive formula holds:

$$v_{i,j,k} = \max_{p \in \{1,2,\dots,i\}} \left(\min_{q \in \{1,2,\dots,j\}} (v_{i-p,j-q,k-1} + p \cdot u_k(p+q)) \right)$$

Consider a coalition C of i agents who use the safety level strategy in the $(k-1)$-subgame with additional j agents. The coalition assigns c_x agents to use resource x (where $x \leq k-1$), so that $\sum_{j=1}^{k-1} c_x = i$. The worst case response of the non-members in this subgame is assigning b_x resources to use resource x, (where $x \leq k-1$), so that $\sum_{j=1}^{k-1} b_x = j$. We can describe a strategy for C in the k-subgame in terms of *moving* some p agents from the first $k-1$ resources and assigning them to the resource k. Any strategy for C in the k-subgame can be described as having $p \leq i$ coalition members using resource k, and a partition of the $i-p$ remaining agents to the first $k-1$ resources (which is a strategy for a coalition C' of $i-p$ agents in the $k-1$-subgame), for some choice of $p \leq i$. Each such a partial strategy profile s_C implies a response from the non-members $\mathbf{N} \setminus C$ which similarly can be described as a choice of $q \leq j$ non-members using resource k and a partition of the $j - q$ remaining non-members to the first $k - 1$ resources (which corresponds to a strategy profile of a non-member agent set B' of $j - q$ agents in the $k - 1$-subgame). The safety level strategy for a coalition C in the k-subgame is therefore a composition of the safely level strategy for a coalition of $|C| - p$ in the $k - 1$-subgame and p agents using resource k for some $p \leq |C|$.

By Theorems 1 and 2, for SLC^{SRSG}-games we can efficiently test core non-emptiness, construct a core imputation or check if an imputation is in the core. These results do not extend to all $SLC^{(S)CG}$-games. When agents are allowed multiple-resource strategies (even if derived from a common set), computing coalitional values is hard.

Theorem 3. *Computing the value of a coalition in SLC^{SCG}-games is NP-hard.*

Proof. We reduce from Exact-Cover-By-3-Sets (X3C). Consider an X3C instance, with a set $S = \{1, 2, \ldots, 3m\}$ of $3m$ elements, and triplets $S_1, \ldots, S_n$ where $S_i \subset S$ and $|S_i| = 3$. We are asked whether there is an exact cover of S that uses exactly m (disjoint) triplets. We construct an SLC^{SCG}-game, where each element $r \in S$ corresponds to a resource r (that is, S corresponds to $\mathbf{R}$), where each resource r's utility function satisfies $u_r(1) = 1$; $u_r(k) = 0$ for $k \geq 2$. The SLC^{SCG}-game has $N = m$ agents, and an agent is allowed to choose any resource triplet, S_j, from the given collection of triplets—that is, the strategy space of any agent i is given by $\mathbf{S}_i = \{S_1, \ldots, S_n\}$. This is an anonymous game. Let $v(\mathbf{N})$ be the value of the grand coalition. We show that if the X3C is a "yes" instance, $v(\mathbf{N}) = 3m$ and if it is a "no" instance then $v(\mathbf{N}) < 3m$. Suppose the X3C is a "yes" instance, and let $S_{i_1}, S_{i_2}, \ldots, S_{i_m}$ be the triplets in the exact cover. Let agent x choose the resources in S_x (for $x \in \{1, \ldots, m\}$). Since S is an exact cover, each resource r is selected exactly once, so $v(\mathbf{N}) = 3m$. On the other hand, if the X3C is a "no" instance, any choice of m (or more) triplets $S_{i_1}, \ldots, S_{i_m}$ results in choosing at least one of the resources, r, more than once. Thus, the congestion on this resource results in a utility of 0 for all agents using it, so $v(\mathbf{N}) < 3m$.

2.3 Non-anonymous Settings

We now turn to consider general, asymmetric settings where agents may have different strategy spaces. First, we observe that though these settings are anonymous in the original—non-cooperative—context, their corresponding SLC-games are not such:

Lemma 5. *The SLC^{CG}-games are, in general, non-anonymous.*

Proof. To see this, consider a single agent i that has an exclusive right to use a special resource rewarding its user with a very high utility, H. Any coalition C that includes i guarantees itself a utility of at least H, regardless of what the rest of agents do, while any $C \setminus \{i\} \cup \{j\}$, $j \in \mathbf{N} \setminus C$, cannot achieve this value.

Next, we show that losing anonymity results in high complexity of computing safety level values even for "degenerate" coalitions consisting of only a single agent:

Theorem 4. *Computing values of singleton coalitions in SLC^{CG}-games is coNP-hard.*

Proof. We reduce from dominating-set (DS). In DS, we are given a graph $G = \langle V, E \rangle$, and have to decide if there is a dominating vertex set of size at most K. In a dominating vertex set $V' \subset V$ for every $v \in V$ either $v \in V'$ or $(u, v) \in E$ for some $u \in V'$. Denote $|V| = m$. We create an SLC^{CG}-game instance as follows: The resources correspond to the vertices V, and we add a resource, r^* (so, $R = m+1$). The congestion function for resources $r \in V$ is given by $u_r(1) = H$ where $H > 3m$ is a very high value; $u_r(k) = 0$ for $k \geq 2$, and for the "special" resource we have $u_{r^*}(k) = 2m-k+1$ for $k = 1, \ldots, m+1$; $u_{r^*}(k) = 0$ for $k \geq m+2$. For any vertex resource $r \in V$ we define an agent a_v, who can choose any single resource which is a neighbour of v or resource r^*, so $\mathbf{S}_{a_v} = \{\{u\} \,|\, u \in V, (u, v) \in E\} \cup \{\{r^*\}\}$. There is also additional agent, a^*, whose only strategy is to select all the resources, so $\mathbf{S}_{a^*} = \{\mathbf{R} = \{v \,|\, v \in V\} \cup \{r^*\}\}$. Since a^* must use its only strategy, the value it obtains depends only on the choices of the other agents. If in a strategy profile s, a^* there exists a vertex resource r so that a^* is its only user, then a^* obtains a value of at least H from s. Thus, to minimize a^*'s utility, each of the vertex resources must be used by some other agent. If there is no dominating set of size K, this requires more than K other agents, so at most $m - K - 1$ other agents can use r^* in such a profile, so $v(\{a^*\}) \geq 2m-(m-K-1) = m+K+1$. If there is a K dominating set, the K outsiders can choose this dominating set, so a^* obtains a utility of 0 from the vertex resources, and having $m - K$ outsiders on r^* results on the utility of $2m - (m-K) = m+K$ for a^* from r^*, so $v(\{a^*\}) = m+K$. Thus, $v(\{a^*\}) > m + K$ iff the DS instance is a "no" instance.

In non-anonymous settings even equal size coalitions may have different values. While by Theorems 3 and 4 computing coalition values is hard, one may seek the *maximal* value of coalitions of size (at most) k, where $1 \leq k \leq N$. We show this is also hard.

Theorem 5. *Finding the value of the grand coalition in SLC^{CG}-games is NP-hard.*

Proof. We reduce from MAX-SAT, where given a Boolean formula we are asked to find the maximum number of clauses that can be satisfied by any assignment. Given a MAX-SAT instance, we construct a SLC^{CG}-game. There is a resource for each clause, and an agent for each variable. An agent for variable x can either choose all clauses satisfied by x or all clauses satisfied by $\neg x$ (thus choosing an assignment for variable x). The resource utility function is $u_r(k) = \frac{1}{k}$, $k = 1, \ldots, N$, for each resource r. Thus, if k agents choose a clause, each of them gets the utility of $\frac{1}{k}$ from the clause, and all of them together get the total utility of 1 from that clause. Thereby, given a strategy profile, its value to the grand coalition is exactly the number of satisfied clauses.

3 Conclusions

We defined a safety level cooperative game induced by a normal form game, and examined this concept on the class of congestion games. A number of questions remain open for future research. First, other solution concepts should be investigated in the context of safety level cooperative games. Second, the application domain should be extended to non-congestion scenarios, such as auctions. Finally, an important task is finding tractable game classes, where the our hardness results do not hold. We intend to examine games where computing Nash equilibria can be done in polynomial time, such as matroid congestion games [1] and congestion-averse games [6,19].

References

1. Ackermann, H., Röglin, H., Vöcking, B.: On the impact of combinatorial structure on congestion games. Journal of the ACM 55(6) (2008)
2. Aumann, R.: Acceptable points in general cooperative n-person games. In: Cont. to the Theory of Games, Princeton, NJ, vol. IV, pp. 287–324 (1959)
3. Bachrach, Y.: Honor among thieves - collusion in multi-unit auctions. In: AAMAS (2010)
4. Bachrach, Y., Elkind, E., Meir, R., Pasechnik, D., Zuckerman, M., Rothe, J., Rosenschein, J.: The cost of stability in coalitional games. In: Mavronicolas, M., Papadopoulou, V.G. (eds.) SAGT 2009. LNCS, vol. 5814, pp. 122–134. Springer, Heidelberg (2009)
5. Bachrach, Y., Rosenschein, J.S.: Power in threshold network flow games. In: Autonomous Agents and Multi-Agent Systems (2009)
6. Byde, A., Polukarov, M., Jennings, N.R.: Games with congestion-averse utilities. In: Mavronicolas, M., Papadopoulou, V.G. (eds.) SAGT 2009. LNCS, vol. 5814, pp. 220–232. Springer, Heidelberg (2009)
7. Conitzer, V., Sandholm, T.: Computing Shapley values, manipulating value division schemes, and checking core membership in multi-issue domains. In: AAAI (2004)
8. Elkind, E., Goldberg, L.A., Goldberg, P., Wooldridge, M.: Computational complexity of weighted threshold games. In: AAAI (2007)
9. Gillies, D.B.: Some theorems on n-person games. PhD thesis, Princeton University (1953)
10. Hayrapetyan, A., Tardos, E., Wexler, T.: The effect of collusion in congestion games. In: STOC 2006 (2006)
11. Holzman, R., Law-Yone, N.: Strong equilibrium in congestion games. Games and Economic Behavior 21, 85–101 (1997)
12. Ieong, S., McGrew, R., Nudelman, E., Shoham, Y., Sun, Q.: Fast and compact: A simple class of congestion games. In: AAAI 2005, pp. 489–494 (2005)
13. Milchtaich, I.: Congestion models of competition. American Naturalist (1996)
14. Resnick, E., Bachrach, Y., Meir, R., Rosenschein, J.S.: The cost of stability in network flow games. In: Královič, R., Niwiński, D. (eds.) MFCS 2009. LNCS, vol. 5734, pp. 636–650. Springer, Heidelberg (2009)
15. Rosenthal, R.W.: A class of games possessing pure-strategy Nash equilibria. International Journal of Game Theory 2, 65–67 (1973)
16. Roughgarden, T., Tardos, E.: How bad is selfish routing. J. ACM 49(2), 236–259 (2002)
17. Rozenfeld, O., Tennenholtz, M.: Strong and correlated strong equilibria in monotone congestion games. In: Spirakis, P.G., Mavronicolas, M., Kontogiannis, S.C. (eds.) WINE 2006. LNCS, vol. 4286, pp. 74–86. Springer, Heidelberg (2006)
18. Shapley, L.S.: A value for n-person games. Cont. to the Theory of Games (1953)
19. Voice, T., Polukarov, M., Byde, A., Jennings, N.R.: On the impact of strategy and utility structures on congestion-averse games. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 600–607. Springer, Heidelberg (2009)
20. Yokoo, M., Conitzer, V., Sandholm, T., Ohta, N., Iwasaki, A.: Coalitional games in open anonymous environments. In: AAAI 2005, pp. 509–514 (2005)
21. Zuckerman, M., Faliszewski, P., Bachrach, Y., Elkind, E.: Manipulating the quota in weighted voting games. In: AAAI (2008)

Improved Hardness of Approximation for Stackelberg Shortest-Path Pricing

Patrick Briest[1,⋆], Parinya Chalermsook[2], Sanjeev Khanna[3], Bundit Laekhanukit[4], and Danupon Nanongkai[5]

[1] Department of Computer Science, University of Paderborn, Germany
patrick.briest@upb.de
[2] Department of Computer Science, University of Chicago, IL, USA
parinya@uchicago.edu
[3] Department of Computer and Information Science, University of Pennsylvania, PA, USA
sanjeev@cis.upenn.edu
[4] Department of Computer Science, McGill University, QC, Canada
blaekh@cs.mcgill.ca
[5] College of Computing, Georgia Tech, Atlanta, GA, USA
danupon@cc.gatech.edu

Abstract. We consider the *Stackelberg shortest-path pricing problem*, which is defined as follows. Given a graph G with fixed-cost and pricable edges and two distinct vertices s and t, we may assign prices to the pricable edges. Based on the predefined fixed costs and our prices, a customer purchases a cheapest s-t-path in G and we receive payment equal to the sum of prices of pricable edges belonging to the path. Our goal is to find prices maximizing the payment received from the customer. While Stackelberg shortest-path pricing was known to be APX-hard before, we provide the first explicit approximation threshold and prove hardness of approximation within $2 - o(1)$. We also argue that the nicely structured type of instance resulting from our reduction captures most of the challenges we face in dealing with the problem in general and, in particular, we show that the gap between the revenue of an optimal pricing and the only known general upper bound can still be logarithmically large.

1 Introduction

The notion of *algorithmic pricing* encompasses a wide range of optimization problems aiming to assign revenue-maximizing prices to some fixed set of items given information about the valuation functions of potential customers [1,13]. In a line of recent work the approximation complexity of this kind of problem has received considerable attention.

Without supply constraints, the very simple *single-price algorithm*, which reduces the search to the one-dimensional subspace of pricings assigning identical

⋆ Work done while the author was staying at Cornell University, supported by a scholarship of the German Academic Exchange Service (DAAD).

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 444–454, 2010.

prices to all the items, achieves an approximation guarantee of $\mathcal{O}(\log n + \log m)$, where n and m denote the number of item types and customers, respectively [4,7]. Corresponding hardness of approximation results of $\Omega(\log^{\varepsilon} m)$ for some $\varepsilon > 0$ are known to hold (under different complexity theoretic assumptions) even in the special cases that valuation functions are *single-minded* (items are *strict complements*) [12] or *unit-demand* (items are *strict substitutes*) [5,8,11]. In these cases, it is the potentially conflicting nature of different customers' valuations that constitutes the combinatorial difficulty of multi-dimensional pricing.

Another line of research has been considering so-called *Stackelberg pricing* problems [17], in which valuation functions are expressed implicitly in terms of some optimization problem. More formally, we are given a set of items, each of which has some fixed cost associated with it. In addition to these fixed costs, we may assign prices to a subset of the items. Given both fixed costs and prices, a single customer will purchase a min-cost subset of items subject to some feasibility constraints and we receive payment equal to the prices assigned to items purchased by the customer. As an example, we may think of items as being the edges of a graph and a customer aiming to buy a min-cost spanning tree, cheapest path, etc.

Clearly, as there is only a single customer in this type of problem, conflicting valuation functions can no longer pose a barrier for the design of efficient pricing algorithms and, indeed, there are several examples of algorithmic results breaking the logarithmic approximation barrier of the general case in situations where the optimization problem solved by the customer is of a certain type [7], the underlying graph is particularly well-structured [10] or the customer is restricted to applying a specific approximation algorithm solving her cost-minimization problem sub-optimally [6].

Yet, many central Stackelberg pricing problems - and in particular the aforementioned spanning tree and shortest path versions in their unrestricted form - have so far resisted all attempts at improving over the single-price algorithm's logarithmic approximation guarantee. At the same time, the best known hardness results to date only prove APX-hardness of both the spanning tree [9] and shortest path [14] cases without even deriving explicit constants.

1.1 Preliminaries

In the *Stackelberg shortest-path pricing problem* (StackSP), we are given a directed graph $G = (V, A)$, a cost function $c : A \to \mathbb{R}_0^+$, a distinguished set of *pricable edges* $\mathcal{P} \subset A$, $|\mathcal{P}| = m$, and two distinguished nodes $s, t \in V$. We may assign prices $p : \mathcal{P} \to \mathbb{R}_0^+$ to the pricable edges. Given these prices, a customer will purchase a shortest directed s-t-path P^* in G, i.e.,

$$P^* \in \operatorname{argmin}\Big\{\sum_{e \in P}(c(e) + p(e)) \,\Big|\, P \text{ is } s\text{-}t\text{-path}\Big\},$$

and we receive revenue $rev(p) = \sum_{e \in P^*} p(e)$. We assume w.l.o.g.[1] that in case of a tie, the customer selects from the above set a path maximizing our revenue.

[1] We can make this assumption since decreasing all prices by a factor arbitrarily close to 1 will break ties in favor of higher revenue paths.

We want to find a price assignment p maximizing $rev(p)$. Throughout the rest of this paper, we will w.l.o.g. only consider StackSP instances for which $c(e) = 0$ for all $e \in \mathcal{P}$, i.e., every edge is either pricable or fixed-cost, but never both.

1.2 Contributions

In this paper, we present the first explicit hardness of approximation result for the shortest path version of Stackelberg pricing, which we show to be hard to approximate within a factor of $2 - o(1)$. The result is based on a reduction that is somewhat similar to the ones previously described in [16] and [14] to derive NP-hardness and APX-hardness, respectively. Our contribution consists of identifying the right starting point for the reduction in order to utilize the full potential of the construction and applying a completely new analysis which yields the desired gap. This novel analysis parts completely from previous approaches, as it argues explicitly about the structure of solutions to the resulting path pricing instances. These results are found in Section 2. Despite their apparent simplicity, these instances, which we refer to as a *shortcut instances*, seem to capture most of the challenges we face in dealing with the problem in general. It is then a natural question to ask whether we can obtain improved approximation results by exploiting the special structure of these instances or the insights gained from our analysis of the structural properties of their solutions. Unfortunately, it turns out that this might not be an easy task, since we can prove that the gap between the optimal revenue and the upper bound used in all known algorithmic results can still be of essentially logarithmic size. These results are presented in Section 3.

2 Hardness of Approximation

Theorem 1. StackSP *cannot be approximated in polynomial time within a factor of* $2 - 2^{-\Omega(\log^{1-\varepsilon} m)}$ *for any* $\varepsilon > 0$*, unless NP* $\subseteq$ *DTIME(*$n^{\mathcal{O}(\log n)}$*).*

2.1 Proof of Theorem 1

The proof of the Theorem is based on a reduction from the *label cover problem* (LabelCover), which is defined as follows. Given a bipartite graph $G = (V, W, E)$, a set $L = \{1, \ldots, k\}$ of *labels* and a set $R_{(v,w)} \subseteq L \times L$ of satisfying label combinations for every edge $(v, w) \in E$, we want to find a label assignment $\ell : V \cup W \rightarrow L$ to the vertices of G satisfying the maximum possible number of edges, i.e., edges (v, w) with $(\ell(v), \ell(w)) \in R_{(v,w)}$. The following hardness result for LabelCover, which is an easy consequence of the PCP theorem [3] combined with Raz' parallel repetition theorem [15], is found, e.g., in the survey by Arora and Lund [2].

Theorem 2. *For* LabelCover *on graphs with* n *vertices,* m *edges and label set of size* $k = \mathcal{O}(n)$ *there exists no polynomial time algorithm to decide whether the maximum number of satisfiable edges is* m *or at most* $m/2^{\log^{1-\varepsilon} m}$ *for any* $\varepsilon > 0$*, unless NP* $\subseteq$ *DTIME(*$n^{\mathcal{O}(\log n)}$*).*

Reduction. Let an instance $G = (V, W, E)$ with label set $L = \{1, \ldots, k\}$ as in Theorem 2 be given. Denote $E = \{(v_1, w_1) \ldots, (v_m, w_m)\}$, where the ordering of the edges is chosen arbitrarily. Note that in our notation, v_i, v_j with $i \neq j$ may well refer to the same vertex (and the same is true for w_i, w_j). For ease of notation we denote by R_i the satisfying label combinations for edge (v_i, w_i).

We create a StackSP instance as follows. For every edge (v_i, w_i) we construct a gadget as depicted in Fig. 1. Essentially, the gadget consist of a set of parallel pricable edges, one for each satisfying label assignment $(\kappa, \lambda) \in R_i$ and an additional parallel fixed-cost edge of price 2.

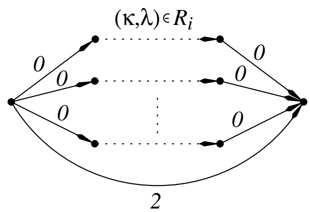


Fig. 1. Gadget for an edge (v_i, w_i) in the label cover instance. Each pricable edge corresponds to one satisfying label assignment (κ, λ) to vertices v_i, w_i.

These gadgets are joined together sequentially (see Fig. 2). Let $i < j$ and consider two pricable edges corresponding to label assignments $(\kappa, \lambda) \in R_i$ and $(\mu, \nu) \in R_j$. We connect the endpoint of the first edge with the start point of the second edge with a *shortcut edge* of cost $j - i - 1$, if the two label assignments are conflicting, i.e., if either $v_i = v_j$ and $\kappa \neq \mu$ or $w_i = w_j$ and $\lambda \neq \nu$. This construction is depicted in Fig. 2. Finally, we define the first node in the gadget corresponding to edge (v_1, w_1) and the last node in the gadget corresponding to (v_m, w_m) as nodes s and t the customer seeks to connect via a directed shortest path. We will refer to the gadgets by their indices $1, \ldots, m$ and denote the pricable edge corresponding to label assignment (κ, λ) in gadget i as $e_{i,\kappa,\lambda}$.

Completeness. Let ℓ be a label assignment satisfying all edges in G. We define a corresponding pricing p by setting for every pricable edge $p(e_{i,\kappa,\lambda}) = 2$ if $\ell(v_i) = \kappa$, $\ell(w_i) = \lambda$ and $p(e_{i,\kappa,\lambda}) = +\infty$ else.

The resulting shortest path from s to t cannot use any of the shortcut edges, because, as ℓ is a feasible label assignment, out of any two pricable edges corresponding to conflicting assignments, one must be priced at $+\infty$. Consequently, no path using a shortcut edge can have finite cost. On the other hand, since ℓ satisfies every edge, there is a pricable edge of cost 2 in each of the gadgets. It is then w.l.o.g. to assume that the customer purchases the shortest path using the maximum possible number of pricable edges and, hence, total revenue is $2m$.

Soundness. Let p be a given pricing resulting in overall revenue $m + c$ and let P denote the shortest path purchased by the customer given these prices. We will argue that there exists a label assignment ℓ satisfying $c/4$ of the edges in G.

First note that w.l.o.g. any pricable edge that is not part of path P has price $+\infty$ under price assignment p. In particular, this means that in every gadget i there is at most a single pricable edge with a finite price. We call this edge the *P-edge* of gadget i. We proceed by grouping gadgets into so-called *islands* as detailed below.

Islands. Let σ_1 be the first gadget with a P-edge and call σ_1 the *start point* of an *island*. Now for each σ_i find the maximum value of $j > \sigma_i$, such that gadget j has a P-edge and there exists a shortcut edge between the P-edges of gadgets σ_i and j. If such a j exists, define $\sigma_{i+1} = j$, else call σ_i an *end point* of an island, let $k > \sigma_i$ be the minimum value such that gadget k has a P-edge, define $\sigma_{i+1} = k$ and call σ_{i+1} a start point. If no such k exists, call σ_i an end point and stop. Let σ_r be the end point of the final island.

We call $\sigma_1, \ldots, \sigma_r$ the *significant gadgets*. Note that by construction every gadget with a P-edge is covered by some *island*, i.e., the interval defined by some consecutive start and end points.

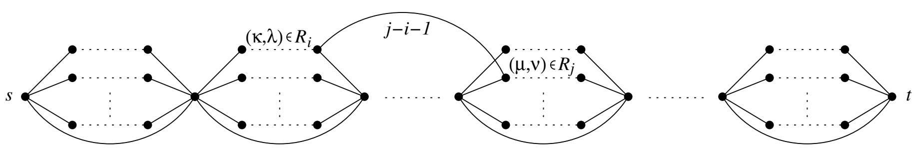


Fig. 2. Assembling the edge gadgets into a StackSP instance. Conflicting label assignments on two edges (v_i, w_i), (v_j, w_j) are connected by a shortcut of length $j-i-1$. All edges are directed from left to right.

Fact 1. *Consider an island $\sigma_\alpha, \ldots, \sigma_\omega$. Path P does not enter gadget σ_α or exit gadget σ_ω via a shortcut edge.*

Proof. If P exits σ_ω via a shortcut edge, then σ_ω could not have been declared an end point. If σ_α is entered via a shortcut edge, this shortcut must originate from a gadget $i < \sigma_\alpha$ which lies within the preceding island. As P cannot bypass the endpoint of the preceding island via a shortcut, i must in fact be the end point $\sigma_{\alpha-1}$ and so σ_α could not have become a start point. □

Consider now a single island $\sigma_\alpha, \ldots, \sigma_\omega$. By ℓ_i we denote the length of the shortcut edge between gadgets σ_i and σ_{i+1} for $\alpha \leq i \leq \omega - 1$. Furthermore, by in_i and out_i we refer to the lengths of the shortcut edges used by path P to enter and exit gadget σ_i, respectively, and set them to 0 if no shortcuts are used. From Fact 1 above it follows that $in_\alpha = out_\omega = in_{\alpha+1} = 0$. See Fig. 3 for an illustration.

For $\alpha \leq i \leq \omega$ let the cost of path P between shortcut edges out_i and in_{i+1} be $r_i + c_i$, where r_i denotes the cost due to pricable edges and c_i the cost due to fixed-cost edges, respectively. We are going to bound the expression $p_{\sigma_i} + r_i$, which is the sum of prices paid for the section of path P running from gadget σ_i to $\sigma_{i+1} - 1$.

We note that $\ell_\omega = 0$, since by the fact that gadget σ_ω is an endpoint, no shortcut edge connects its P-edge to the P-edge of another gadget. Similarly, we have $r_\omega = 0$, since path P does not use pricable edges between islands, as we have argued before.

Path P crosses the end node of the P-edge in gadget σ_i (node v_2 in Fig. 3) and the start node of the P-edge of gadget σ_{i+1} (node v_4 in Fig. 3) for $\alpha \leq i \leq \omega - 1$. The total cost of path P between these two vertices is $out_i + r_i + c_i + in_{i+1}$. An alternative path P_1 is obtained by replacing this part of P with the shortcut edge of length ℓ_i between σ_i and σ_{i+1}. By the fact that P is the shortest path we have $out_i + r_i + c_i + in_{i+1} \leq \ell_i$ and, thus,

$$r_i \leq \ell_i - out_i - in_{i+1} \qquad \text{for } \alpha \leq i \leq \omega, \tag{1}$$

where the bound on r_ω follows from the fact that for $i = \omega$ all summands in the above expression are 0. Similarly, the cost of path P between the start node of the shortcut edge into gadget σ_i (node v_1 in Fig. 3) and the end node of the shortcut edge exiting σ_i (node v_3 in Fig. 3) is $in_i + p_{\sigma_i} + out_i$ for $\alpha \leq i \leq \omega$. We obtain an alternative path P_2 by taking only fixed cost edges of cost 2 to bypass both shortcuts and gadget σ_i at total cost $2(in_i + out_i + 1)$. Again, since P is the shortest path, we get $in_i + p_{\sigma_i} + out_i \leq 2(in_i + out_i + 1)$, or

$$p_{\sigma_i} \leq 2 + in_i + out_i \qquad \text{for } \alpha \leq i \leq \omega\,. \tag{2}$$

Combining (1) and (2) yields

$$p_{\sigma_i} + r_i \leq 2 + \ell_i + in_i - in_{i+1} \quad \text{for } \alpha \leq i \leq \omega\,. \tag{3}$$

Finally, we have

$$\sum_{i=\alpha}^{\omega} \Big(p_{\sigma_i} + r_i\Big) \leq \sum_{i=\alpha}^{\omega} \Big(2 + \ell_i + in_i - in_{i+1}\Big) \tag{4}$$

$$= 2(\omega - \alpha + 1) + \sum_{i=\alpha}^{\omega} \ell_i + in_\alpha - in_{\omega+1} \tag{5}$$

$$= 2(\omega - \alpha + 1) + \sum_{i=\alpha}^{\omega} \ell_i, \tag{6}$$

where (6) holds since start points σ_α and $\sigma_{\omega+1}$ are not entered via shortcuts and, thus, $in_\alpha = in_{\omega+1} = 0$. Recall that $\sigma_1, \ldots, \sigma_r$ denote the significant gadgets across all islands. Assume now that there is a total number I of islands with start and end points $\sigma_{\alpha(1)}, \sigma_{\omega(1)}, \ldots, \sigma_{\alpha(I)}, \sigma_{\omega(I)}$. Summing over all islands we get that overall revenue of price assignment p is bounded by

$$\sum_{j=1}^{I} \sum_{i=\alpha(j)}^{\omega(j)} p_{\sigma_i} + r_i \leq \sum_{j=1}^{I} \Big(2\big(\omega(j) - \alpha(j) + 1\big) + \sum_{i=\alpha(j)}^{\omega(j)} \ell_i\Big) \leq 2r + m,$$

where the last inequality follows from the fact that $\alpha(j) = \omega(j-1)+1$ for $2 \leq j \leq I$, $\omega(I) = r$ and $\sum_{i=1}^{r} \ell_i \leq m$, since all shortcuts defining the ℓ_i are disjoint. Thus, we have $m + c \leq 2r + m$, or $r \geq c/2$.

So we have established a relation between the revenue of a price assignment and the number of its significant gadgets. It remains to show that a pricing's significant gadgets can be used to construct a label assignment satisfying a large number of the corresponding edges. Towards this end, consider the P-edges of the $\lceil r/2 \rceil$ gadgets $\sigma_1, \sigma_3, \sigma_5, \ldots$ and their corresponding label assignments (κ_i, λ_i). By definition, there are no shortcut edges between the P-edges of any of these gadgets and, thus, $(\kappa_1, \lambda_1), (\kappa_3, \lambda_3), \ldots$ define a non-conflicting label assignment satisfying at least $\lceil r/2 \rceil \geq c/4$ edges in G. More precisely, labels $(\kappa_1, \lambda_1), (\kappa_3, \lambda_3), \ldots$ can be extended into a complete label assignment satisfying $c/4$ edges by choosing all unspecified labels in an arbitrary fashion.

Finally, consider a label cover instance as in Theorem 2 and the path pricing instance resulting from our reduction above. If all edges can be satisfied, maximum path pricing revenue is $2m$. If no label assignment satisfies more than $m/2^{\log^{1-\varepsilon} m}$ edges, maximum path pricing revenue is bounded by $(1 + 4/2^{\log^{1-\varepsilon} m})m$. This completes the proof of Theorem 1. □

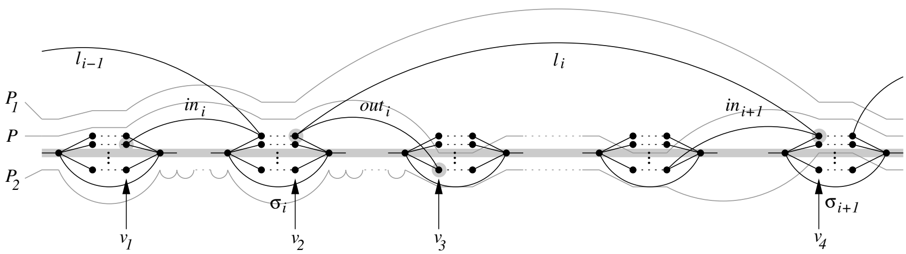


Fig. 3. Two consecutive significant gadgets σ_i, σ_{i+1} inside one island. The length of the shortcut edges used to enter and exit gadget σ_i (defined as 0 if no such shortcut exists) are denoted as in_i and out_i, respectively.

Tightness. We briefly mention that our analysis is tight in the following sense. It is easy to check that by assigning price 1 to all pricable edges we can make sure that w.l.o.g. the shortest s-t-path uses a pricable edge in each of the gadgets and, thus, we obtain revenue m. Since maximum possible revenue is bounded above by $2m$ (there is an s-t-path of that cost that does not use any pricable edges), it follows that it is trivial to achieve approximation guarantee 2 on the instances resulting from our reduction.

3 Shortcut Instances

In this section we take a closer look at the type of instances resulting from our reduction, which we believe present an important milestone in getting a further improvement in terms of hardness or algorithmic results. We will be interested in the family of so-called *shortcut instances*, which we define as follows.

We say that a *gadget* H consists of source and sink nodes connected by (i) a fixed cost edge from source to sink and (ii) node-disjoint paths of length three where each path is directed from the source to the sink, alternating between fixed-cost, pricable, and fixed-cost edges. For example, the graph in Figure 1 is a gadget with three node-disjoint paths of length three. The left and right nodes are source and sink, respectively. We call an input instance *shortcut instance*, if it can be constructed by the following two-step process:

1. Let $G_1, \ldots, G_{n'}$ be gadgets. Sequentially join them together by unifying the sink of each gadget G_i with the source of G_{i+1}. The source of G_1 is denoted as s, the sink of $G_{n'}$ as t.
2. For each pair of integers $i < j$, for each pricable edge (u, u') in G_i and each pricable edge (v, v') in G_j, we have a fixed cost edge (u', v). (Note, that we allow setting the price of edges to ∞, which is equivalent to removing them from the instance.) Edges created in this step are called *shortcuts*. The fixed cost edge of cost $j - i - 1$ in Figure 2 is an example of shortcut.

Clearly, the instances resulting from our reduction in the previous section are examples of shortcut instances. It is a natural question to ask whether one can exploit the special structure of shortcut instances to beat the $\mathcal{O}(\log n)$ approximation guarantee known for the general path pricing problem. In fact, getting a better approximation ratio for shortcut graphs would probably even yield insights into potential approaches to improving the general case.

It is, however, not clear at all how to exploit the structure of these seemingly simple instances. This is so, because in dealing with the shortcut graphs, one faces the same main barrier currently encountered in the general case: all known algorithms for the problem rely on the same upper bounding technique, which yields bounds as large as $\Theta(\log n \cdot \mathsf{OPT})$. Unfortunately, it turns out that this is also the case for shortcut graphs.

The upper bound used by previous algorithms is the quantity $f_\infty(G) - f_0(G)$ where $f_x(G)$, $x \in \mathbb{R}_0^+$, is defined to be the shortest path length in G when $p(e) = x$ for all pricable edges e.[2] It is known that $f_\infty(G) - f_0(G)$ can be as large as $\Omega(\log n \cdot \mathsf{OPT})$ and therefore one cannot hope for a better approximation guarantee using this upper bound. We show that the same problem occurs in the family of shortcut instances.

Theorem 3. *For infinitely many n, there exists a shortcut graph G of n nodes with*

$$f_\infty(G) - f_0(G) = \Omega((\log n / \log\log n) \cdot \mathsf{OPT}) .$$

We first describe an explicit construction of the shortcut graphs from Theorem 3 and then prove the claim.

Construction. Let $\alpha \geq 2$ be any integer and let $n = \alpha^\alpha$. We construct a graph G of $\Theta(n)$ nodes as follows.

[2] Intuitively, $f_\infty(G) - f_0(G) \geq \mathsf{OPT}$ follows from the fact that the customer will never pay more than $f_\infty(G)$ (and will do so when the pricable edges are very expensive) and part of this will be paid to the competitor who owns the fixed cost edges; the latter is at least $f_0(G)$ (when the pricable edges are very cheap).

- **Gadgets.** There are n gadgets, each of which has (i) a fixed cost edge of length 1 from source to sink and (ii) a path alternating between two fixed-cost edges and a pricable edge where fixed cost edges have price 0 (see Fig. 4).
- **Shortcuts.** For a shortcut from gadget i to gadget j, the price is $(j-i)\cdot(k/\alpha)$ where k is chosen such that $\alpha^{k-1} \leq j-i < \alpha^k$; in this case, we additionally say that the shortcut is of *type* k. Observe that $1 \leq k \leq \alpha$.

We denote by (a_i, b_i) the pricable edge in gadget i. For the ease of referencing in the future, we let b_0 and b_{n+1} denote s and t respectively. Moreover, we add a shortcut of type α (i.e., of cost n) from b_0 to b_{n+1}.

Analysis. If all pricable edges have cost ∞ then, since all shortcuts are blocked, the shortest path will consist of all the gadgets' fixed cost edges and, thus, $f_\infty(G) = n$. If all pricable edges have cost zero then the shortest path will use all pricable edges and, hence, $f_0(G) = 0$. Therefore, $f_\infty(G) - f_0(G) = n$. We now prove that $\mathsf{OPT} = O(n/\alpha)$. This yields the theorem since $n = \alpha^\alpha$ implies that $\alpha = \Omega(\log n/\log\log n)$.

Let p be any pricing and P be the shortest path purchased by the customer given this pricing. Let $\delta_1, \ldots, \delta_r$ be the indices of gadgets that contain pricable edges on P (P-edges), so the revenue is collected from edges $(a_{\delta_1}, b_{\delta_1}), \ldots, (a_{\delta_r}, b_{\delta_r})$. Let $\delta_0 = 0$ and $\delta_{r+1} = n+1$.

The following lemma bounds the price of each pricable edge.

Lemma 1. *For any* $1 \leq i \leq r$, $p(a_{\delta_i} b_{\delta_i}) = O((\delta_{i+1} - \delta_{i-1})/\alpha)$.

The fact that $\mathsf{OPT} \leq O(n/\alpha)$ follows as an easy consequence of the lemma, since

$$\sum_{i=1}^{r} p(a_{\delta_i}, b_{\delta_i}) \leq \sum_{i=1}^{r} O\left(\frac{\delta_{i+1} - \delta_{i-1}}{\alpha}\right) \leq O(n/\alpha).$$

Proof of Lemma 1. Let P' be the subpath of P from $b_{\delta_{i-1}}$ to a_{δ_i}, P'' be the subpath from b_{δ_i} to $a_{\delta_{i+1}}$. Let k be the type of shortcut $(b_{\delta_{i-1}}, a_{\delta_{i+1}})$. Notice that $p(a_{\delta_i}, b_{\delta_i}) + c(P') + c(P'') \leq (\delta_{i+1} - \delta_{i-1})k/\alpha$, because the customer will buy shortcut $(b_{\delta_{i-1}}, a_{\delta_{i+1}})$ instead otherwise. Now proving the following claim yields the lemma.

Claim. $c(P') + c(P'') \geq \big((\delta_{i+1} - \delta_{i-1})(k-2)\big)/\alpha$.

Since both P' and P'' do not contain pricable edges, there are only two possibilities for each of them: path P' either takes the shortcut $(b_{\delta_{i-1}}, a_{\delta_i})$ or takes a sequence of fixed cost edges in gadgets $\delta_{i-1}+1, \ldots, \delta_i - 1$ (similarly for P''), and since the first option always costs less, we assume that both P' and P'' take the first option (i.e., take shortcuts).

Our first simple observation is that at least one of P' and P'' is of type at least $(k-1)$. To see this, note that assuming the contrary, we have that $(\delta_i - \delta_{i-1}) < \alpha^{k-2}$ and $(\delta_{i+1} - \delta_i) < \alpha^{k-2}$, so, adding them up, $\delta_{i+1} - \delta_{i-1} < \alpha^{k-1}$ (because $\alpha \geq 2$), contradicting the assumption that the shortcut $(b_{\delta_{i-1}}, a_{\delta_{i+1}})$ is of type k.

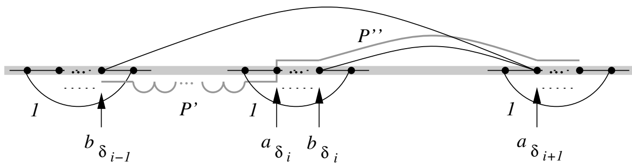


Fig. 4. Proof idea of Lemma 1

There are two cases to analyze now. The first case is when both P' and P'' are shortcuts of type at least $k-1$. In this case, we have,

$$\begin{aligned} c(P') + c(P'') &\geq (\delta_i - \delta_{i-1})(k-1)/\alpha + (\delta_{i+1} - \delta_i)(k-1)/\alpha \\ &\geq (\delta_{i+1} - \delta_{i-1})(k-1)/\alpha\,. \end{aligned}$$

In the second case, assume w.l.o.g. that P' is of type at most $k-2$. (This means that P'' is of type at least $k-1$ by our previous observation.) So, $\delta_i - \delta_{i-1} < \alpha^{k-2}$, while $\delta_{i+1} - \delta_{i-1} \geq \alpha^{k-1}$. Consequently, $\delta_i - \delta_{i-1} \leq \frac{1}{\alpha}(\delta_{i+1} - \delta_{i-1})$. Therefore, we get

$$\begin{aligned} \delta_{i+1} - \delta_i &= (\delta_{i+1} - \delta_{i-1}) - (\delta_i - \delta_{i-1}) \\ &\geq (1 - 1/\alpha)(\delta_{i+1} - \delta_{i-1})\,. \end{aligned}$$

Since P'' is of type at least $k-1$, we have

$$\begin{aligned} c(P'') &\geq \frac{(1-1/\alpha)(\delta_{i+1} - \delta_{i-1})(k-1)}{\alpha} \\ &\geq \frac{(\delta_{i+1} - \delta_{i-1})(k-2)}{\alpha}\,. \end{aligned}$$

The second inequality follows because $(k-1)(1-1/\alpha) = (k-1+1/\alpha - k/\alpha)$, and $k \leq \alpha$. □

4 Conclusions

We have proven the first explicit approximation threshold for any Stackelberg pricing problem. Still, the approximation threshold for this kind of problem in general - and the shortest path version in particular - is far from settled. The following questions seem to constitute fertile ground for future research:

- Can we prove super-constant hardness of approximation results for any kind of Stackelberg pricing problem?
- Is it possible to achieve a better than logarithmic approximation guarantee for the Stackelberg shortest path pricing problem? Is there an interesting restricted set of graphs on which constant approximation factors are possible? In the context of the first of these questions, our discussion in Section 3 points to the obvious need of coming up with novel upper-bounding techniques even

for restricted problem instances. Some progress towards answering the second of these questions has recently been made in [10], where polynomial-time algorithms are presented for the spanning-tree pricing problem in bounded-treewidth graphs.

References

1. Aggarwal, G., Feder, T., Motwani, R., Zhu, A.: Algorithms for multi-product pricing. In: Díaz, J., Karhumäki, J., Lepistö, A., Sannella, D. (eds.) ICALP 2004. LNCS, vol. 3142, pp. 72–83. Springer, Heidelberg (2004)
2. Arora, S., Lund, C.: Hardness Of Approximations. In: Approximation Algorithms for NP-hard Problems. PWS Publishing Company (1996)
3. Arora, S., Lund, C., Motwani, R., Sudan, M., Szegedy, M.: Proof Verification and Hardness of Approximation Problems. Journal of the ACM 45(3), 501–555 (1998)
4. Balcan, N., Blum, A., Mansour, Y.: Item Pricing for Revenue Maximization. In: Proc. of 9th EC (2008)
5. Briest, P.: Uniform Budgets and the Envy-Free Pricing Problem. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part I. LNCS, vol. 5125, pp. 808–819. Springer, Heidelberg (2008)
6. Briest, P., Gualà, L., Hoefer, M., Ventre, C.: On Stackelberg Pricing with Computationally Bounded Consumers. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 42–54. Springer, Heidelberg (2009)
7. Briest, P., Hoefer, M., Krysta, P.: Stackelberg Network Pricing Games. In: Proc. of 25th STACS (2008)
8. Briest, P., Krysta, P.: Buying Cheap is Expensive: Hardness of Non-Parametric Multi-Product Pricing. In: Proc. of 18th SODA (2007)
9. Cardinal, J., Demaine, E., Fiorini, S., Joret, G., Langerman, S., Newman, I., Weimann, O.: The Stackelberg Minimum Spanning Tree Game. In: Dehne, F., Sack, J.-R., Zeh, N. (eds.) WADS 2007. LNCS, vol. 4619, pp. 64–76. Springer, Heidelberg (2007)
10. Cardinal, J., Demaine, E., Fiorini, S., Joret, G., Newman, I., Weimann, O.: The Stackelberg Minimum Spanning Tree Game on Planar and Bounded-Treewidth Graphs. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 125–136. Springer, Heidelberg (2009)
11. Chuzhoy, J., Kannan, S., Khanna, S.: Network Pricing for Multicommodity Flows (2007) (unpublished manuscript)
12. Demaine, E.D., Feige, U., Hajiaghayi, M.T., Salavatipour, M.R.: Combination Can Be Hard: Approximability of the Unique Coverage Problem. In: Proc. of 17th SODA (2006)
13. Guruswami, V., Hartline, J.D., Karlin, A.R., Kempe, D., Kenyon, C., McSherry, F.: On Profit-Maximizing Envy-Free Pricing. In: Proc. of 16th SODA (2005)
14. Joret, G.: Stackelberg Network Pricing is Hard to Approximate. Networks n/a (2010), doi: 10.1002/net.20391
15. Raz, R.: A Parallel Repetition Theorem. SIAM Journal on Computing 27 (1998)
16. Roch, S., Savard, G., Marcotte, P.: An Approximation Algorithm for Stackelberg Network Pricing. Networks 46(1), 57–67 (2005)
17. von Stackelberg, H.: Marktform und Gleichgewicht (Market and Equilibrium). Verlag von Julius Springer, Vienna (1934)

The Complexity of Determining the Uniqueness of Tarski's Fixed Point under the Lexicographic Ordering*

Chuangyin Dang[1] and Yinyu Ye[2]

[1] Dept. of Manufacturing Engineering & Engineering Management, City University of Hong Kong, Kowloon, Hong Kong SAR, China
mecdang@cityu.edu.hk

[2] Dept. of Management Science & Engineering, Stanford University, Stanford, CA 94305-4026
yinyu-ye@stanford.edu

Abstract. The well-known Tarski's fixed point theorem asserts that an increasing mapping from a complete lattice into itself has a fixed point. This theorem plays an important role in the development of supermodular games for economic analysis. Let C be a finite lattice consisting of all integer points in an n-dimensional box and f be an increasing mapping from C into itself in terms of lexicographic ordering. It has been shown in the literature that, when f is given as an oracle, a fixed point of f can be found in *polynomial time*. The problem we consider in this paper is the complexity of determining whether or not f has a unique fixed point. We present a polynomial-time reduction of integer programming to an increasing mapping from C into itself. As a result of this reduction, we prove that, when f is given as an oracle, determining whether or not f has a unique fixed point is *Co-NP hard*.

Keywords: Lexicographic Ordering, Lattice, Finite Lattice, Increasing Mapping, Fixed Point, Integer Programming, Co-NP Completeness, Co-NP Hardness.

1 Introduction

There are some interesting complexity results in algorithmic game theory research on determining whether or not a game has a unique equilibrium point. For example, for the bimatrix game Gilboa and Zemel (1989) showed that it is NP-hard to determine whether or not there is a second Nash equilibrium, which became a classical result. In all of these problems, computing even one equilibrium (which is known to exist) is already difficult and no polynomial time algorithms are known. In this short paper, we consider the well-known Tarski's fixed point theorem (Tarski, 1955) that asserts that an increasing mapping from

* This work was partially supported by GRF: CityU 113308 of the Government of Hong Kong SAR.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 455–461, 2010.

a complete lattice into itself has a fixed point. In terms of lexicographic ordering, it has been shown in Chang et al. (2008) that, when the mapping and lattice are given as oracles, a fixed point can be found in *polynomial time*. Here we prove that determining whether or not the same mapping has a unique fixed point is *Co-NP hard*.

The well-known Tarski's fixed point theorem asserts that, if $(S, \preceq)$ is a complete lattice and f is an increasing from S into itself, then there exists some $x^* \in S$ such that $f(x^*) = x^*$. This theorem plays an important role in the development of supermodular games or games with strategic complementarities for economic analysis. Supermodular games were formalized in Topkis (1979) and have been extensively applied for economic analysis in the literature such as Bernstein and Federgruen (2004, 2005), Cachon (2001), Cachon and Lariviere (1999), Fudenberg and Tirole (1991), Lippman and McCardle (1997), Milgrom and Roberts (1990, 1994), Milgrom and Shannon (1994), Topkis (1998), and Vives (1990, 1999, 2005). To compute a Nash equilibrium of a supermodular game, a generic approach is to convert it into the computation of a fixed point of an increasing mapping. Recently, an algorithm was proposed in Echenique (2007) to find all pure-strategy Nash equilibria of a supermodular game. This work motivates us to study how difficult the problem is, which leads to the main results in this paper.

Let $N = \{1, 2, \ldots, n\}$. For x and y of R^n, $x \leq_l y$ if either $x = y$ or $x_i = y_i$, $i = 1, 2, \ldots, k-1$, and $x_k < y_k$ for some $k \in N$, where $\leq_l$ is called a lexicographic order on R^n. Let $C = \{x \in Z^n \mid a \leq x \leq b\}$, where a and b are two finite vectors of Z^n with $a < b$. Clearly, $(C, \leq_l)$ is a finite lattice. Let f be an increasing mapping from C into itself under the lexicographic ordering. The problem we consider in this paper is whether or not f has a unique fixed point in C. It was shown in Chang et al. (2008) that, given f as an oracle, a fixed point of f can be computed in polynomial time of $O(\log |C|)$ queries to f, where $|C|$ denotes the cardinality of C. We present in this paper a polynomial-time reduction of integer programming to an increasing mapping from C into itself. As a result of this reduction, we prove that, given f as an oracle, determining whether or not f has a unique fixed point in C is *Co-NP hard*.

2 Polynomial-Time Reduction and Main Results

For any real number α and any $x = (x_1, x_2, \ldots, x_n)^\top \in R^n$, let $\lfloor \alpha \rfloor$ denote the greatest integer less than or equal to α, $\lceil \alpha \rceil$ the smallest integer greater than or equal to α, $\lfloor x \rfloor = (\lfloor x_1 \rfloor, \lfloor x_2 \rfloor, \ldots, \lfloor x_n \rfloor)^\top$, and $\lceil x \rceil = (\lceil x_1 \rceil, \lceil x_2 \rceil, \ldots, \lceil x_n \rceil)^\top$. Let $P = \{x \in R^n \mid Ax \leq b\}$ be a full-dimensional polytope, where A is an $m \times n$ integer matrix and b an integer vector of R^m. We assume throughout this paper that $n \geq 2$. Let $x^{\max} = (x_1^{\max}, x_2^{\max}, \ldots, x_n^{\max})^\top$ with $x_j^{\max} = \max_{x \in P} x_j$, $j = 1, 2, \ldots, n$, and $x^{\min} = (x_1^{\min}, x_2^{\min}, \ldots, x_n^{\min})^\top$ with $x_j^{\min} = \min_{x \in P} x_j$, $j = 1, 2, \ldots, n$. Then, $x^{\min} \leq x \leq x^{\max}$ for all $x \in P$. Let $D(P) = \{x \in Z^n \mid x^l \leq x \leq x^u\}$, where $x^u = \lfloor x^{\max} \rfloor$ and $x^l = \lfloor x^{\min} \rfloor$. It is obvious that $x \in D(P)$ for all $x \in P \cap Z^n$. We assume without loss of generality that $x^l < x^{\min}$

(If $x_i^l = x_i^{\min}$ for some $i \in N$, let $x_i^l = x_i^{\min} - 1$). For $y \in R^n$ and $r \in N$, let $P(y, r) = \{x \in P \mid x_i = y_i,\ i = 1, 2, \ldots, r\}$.

Definition 1. *For $y \in D(P)$, $h(y) = (h_1(y), h_2(y), \ldots, h_n(y))^\top \in D(P)$ is given as follows:*

Step 1: *If $y_1 = x_1^l$, let $h(y) = x^l$. If $y \in P$, let $h(y) = y$. Otherwise, let $r = 2$ and go to* **Step 2**.

Step 2: *Solve the linear program*

$$\begin{gathered}\min x_r - v_r \\ \text{subject to } x \in P(y, r-1) \text{ and } v \in P(y, r-1)\end{gathered}$$

to obtain its optimal solution (x^*, v^*). Let $d_r^{\min}(y) = x_r^*$ and $d_r^{\max}(y) = v_r^*$. If $y_r \geq \lceil d_r^{\min}(y) \rceil$, go to **Step 3**. *Otherwise, go to* **Step 4**.

Step 3: *If $\lfloor d_r^{\max}(y) \rfloor < \lceil d_r^{\min}(y) \rceil$, go to* **Step 4**. *Otherwise, go to* **Step 5**.

Step 4: *Let $p(y) = r$ ($p(y)$ is an output parameter and is used in further discussions). If $y_{r-1} \leq x_{r-1}^l + 1$, let*

$$h_i(y) = \begin{cases} y_i & \text{if } 1 \leq i \leq r-2, \\ x_i^l & \text{if } r-1 \leq i \leq n, \end{cases}$$

$i = 1, 2, \ldots, n$. Otherwise, let

$$h_i(y) = \begin{cases} y_i & \text{if } 1 \leq i \leq r-2, \\ y_{r-1} - 1 & \text{if } i = r-1, \\ x_i^u & \text{if } r \leq i \leq n, \end{cases}$$

$i = 1, 2, \ldots, n$.

Step 5: *If $y_r > \lfloor d_r^{\max}(y) \rfloor$, let $p(y) = r$ and*

$$h_i(y) = \begin{cases} y_i & \text{if } 1 \leq i \leq r-1, \\ \lfloor d_r^{\max}(y) \rfloor & \text{if } i = r, \\ x_i^u & \text{if } r+1 \leq i \leq n, \end{cases}$$

$i = 1, 2, \ldots, n$. Otherwise, let $r = r + 1$ and go to **Step 2**.

Lemma 1. *$x^l \leq h(y) \leq_l y$ and $h(y) \neq y$ for all $y \in D(P)$ with $y \neq x^l$ and $y \notin P$.*

Proof. Clearly, the lemma holds for all $y \in D(P)$ with $y_1 = x_1^l$ and $y \neq x^l$. Let y be any given point in $D(P)$ with $y_1 \neq x_1^l$ and $y \notin P$. Let $k = p(y)$ with $p(y)$ being given in the definition. Note that $y \in P$ when $\lceil d_r^{\min}(y) \rceil \leq y_r \leq \lfloor d_r^{\max}(y) \rfloor$, $r = 1, 2, \ldots, n$. Thus, $p(y)$ is well defined, $2 \leq k \leq n$ and $x_i^l < x_i^{\min} \leq \lceil d_i^{\min}(y) \rceil \leq y_i \leq \lfloor d_i^{\max}(y) \rfloor$, $i = 1, 2, \ldots, k-1$. Therefore, one of the following five cases must occur.

Case 1: $y_k \geq \lceil d_k^{\min}(y) \rceil$, $\lfloor d_k^{\max}(y) \rfloor < \lceil d_k^{\min}(y) \rceil$ and $y_{k-1} \leq x_{k-1}^l + 1$. It follows from $y_{k-1} > x_{k-1}^l$ and **Step 4** that $x^l \leq h(y) \leq_l y$ and $h(y) \neq y$.

Case 2: $y_k \geq \lceil d_k^{\min}(y) \rceil$, $\lfloor d_k^{\max}(y) \rfloor < \lceil d_k^{\min}(y) \rceil$ and $y_{k-1} > x_{k-1}^l + 1$. It follows from $y_{k-1} - 1 < y_{k-1}$ and **Step 4** that $x^l \leq h(y) \leq_l y$ and $h(y) \neq y$.
Case 3: $y_k \geq \lceil d_k^{\min}(y) \rceil$, $\lfloor d_k^{\max}(y) \rfloor \geq \lceil d_k^{\min}(y) \rceil$ and $y_k > \lfloor d_k^{\max}(y) \rfloor$. It follows from $y_k > \lfloor d_k^{\max}(y) \rfloor$ and **Step 5** that $x^l \leq h(y) \leq_l y$ and $h(y) \neq y$.
Case 4: $y_k < \lceil d_k^{\min}(y) \rceil$ and $y_{k-1} \leq x_{k-1}^l + 1$. It follows from $y_{k-1} > x_{k-1}^l$ and **Step 4** that $x^l \leq h(y) \leq_l y$ and $h(y) \neq y$.
Case 5: $y_k < \lceil d_k^{\min}(y) \rceil$ and $y_{k-1} > x_{k-1}^l + 1$. It follows from $y_{k-1} - 1 < y_{k-1}$ and **Step 4** that $x^l \leq h(y) \leq_l y$ and $h(y) \neq y$.

For every case, one can see from the above that it always holds that $x^l \leq h(y) \leq_l y$ and $h(y) \neq y$. This completes the proof. □

As a corollary of Lemma 1, we obtain that

Corollary 1. *For any given $x^* \in D(P)$, $x^* \in P$ if and only if $h(x^*) = x^*$ and $x^* \neq x^l$.*

Theorem 1. *Under the lexicographic ordering, h is an increasing mapping from $D(P)$ into itself.*

A sketch of the proof. Let y^1 and y^2 be any given two points in $D(P)$ with $y^1 \leq_l y^2$ and $y^1 \neq y^2$. We only need to consider that $y_1^1 \neq x_1^l$ and $y^2 \notin P$. Let q be the index in N satisfying that $y_i^1 = y_i^2$, $i = 1, 2, \ldots, q-1$, and $y_q^1 < y_q^2$. Let $k_1 = p(y^1)$ and $k_2 = p(y^2)$. Clearly, k_2 is well defined and $k_2 \geq 2$. Thus, one of the following four cases must occur.

Case 1: $2 \leq k_2 \leq q-1$. Clearly, $h(y^1) = h(y^2)$.
Case 2: $2 \leq k_2 = q$.

1. Suppose that $y_q^2 \geq \lceil d_q^{\min}(y^2) \rceil$ and $\lfloor d_q^{\max}(y^2) \rfloor < \lceil d_q^{\min}(y^2) \rceil$. Then, $k_1 = k_2 = q$. Thus, $h(y^1) = h(y^2)$.
2. Suppose that $y_q^2 > \lfloor d_q^{\max}(y^2) \rfloor$ and $\lfloor d_q^{\max}(y^2) \rfloor \geq \lceil d_q^{\min}(y^2) \rceil$.
   - Consider that $y^1 \in P$. We have $h(y^1) = y^1 \leq h(y^2)$.
   - Consider that $y^1 \notin P$. We have $k_1 \geq q$.
     (a) Assume that $k_1 = q$. When $y_q^1 > \lfloor d_q^{\max}(y^1) \rfloor$, $h(y^1) = h(y^2)$. When $y_q^1 < \lceil d_q^{\min}(y^1) \rceil$, if $y_{q-1}^1 \leq x_{q-1}^l + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = x_{q-1}^l < y_{q-1}^2 = h_{q-1}(y^2)$, and if $y_{q-1}^1 > x_{q-1}^l + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = y_{q-1}^1 - 1 < y_{q-1}^1 = y_{q-1}^2 = h_{q-1}(y^2)$.
     (b) Assume that $k_1 > q$. We have $h(y^1) \leq h(y^2)$.
3. Suppose that $y_q^2 < \lceil d_q^{\min}(y^2) \rceil$. Then, $h(y^1) = h(y^2)$.

Case 3: $2 \leq k_2 = q+1$.

1. Suppose that $y_{q+1}^2 \geq \lceil d_{q+1}^{\min}(y^2) \rceil$, $\lfloor d_{q+1}^{\max}(y^2) \rfloor < \lceil d_{q+1}^{\min}(y^2) \rceil$ and $y_q^2 \leq x_q^l + 1$. Then, $y_q^1 = x_q^l$ and $k_1 = q \geq 2$. Therefore, if $y_{q-1}^1 \leq x_{q-1}^l + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = x_{q-1}^l < y_{q-1}^2 = h_{q-1}(y^2)$, and if $y_{q-1}^1 > x_{q-1}^l + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = y_{q-1}^1 - 1 < y_{q-1}^2 = h_{q-1}(y^2)$.

2. Suppose that $y^2_{q+1} \geq \lceil d^{\min}_{q+1}(y^2)\rceil$, $\lfloor d^{\max}_{q+1}(y^2)\rfloor < \lceil d^{\min}_{q+1}(y^2)\rceil$ and $y^2_q > x^l_q + 1$.
   - Assume that $y^1 \in P$. Thus, $h(y^1) = y^1 \leq h(y^2)$.
   - Assume that $y^1 \notin P$. Then, $k_1 \geq q$.
     Consider that $k_1 = q$.
     (a) Suppose that $y^1_q > \lfloor d^{\max}_q(y^1)\rfloor$. Then, $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-1$, and $h_q(y^1) < y^1_q \leq y^2_q - 1 = h_q(y^2)$.
     (b) Suppose that $y^1_q < \lceil d^{\min}_q(y^1)\rceil$. If $y^1_{q-1} \leq x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = x^l_{q-1} < y^2_{q-1} = h_{q-1}(y^2)$. If $y^1_{q-1} > x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = y^1_{q-1} - 1 < y^2_{q-1} = h_{q-1}(y^2)$.
     Consider that $k_1 > q$. We have $h(y^1) \leq h(y^2)$.
3. Suppose that $y^2_{q+1} > \lfloor d^{\max}_{q+1}(y^2)\rfloor$ and $\lfloor d^{\max}_{q+1}(y^2)\rfloor \geq \lceil d^{\min}_{q+1}(y^2)\rceil$.
   - Assume that $y^1 \in P$. Then, $h(y^1) = y^1 \leq_l h(y^2)$ follows immediately from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-1$, and $h_q(y^1) < h_q(y^2)$.
   - Assume that $y^1 \notin P$. Then, $k_1 \geq q$.
     Consider that $k_1 = q$.
     (a) Suppose that $y^1_q > \lfloor d^{\max}_q(y^1)\rfloor$. Then, $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-1$, and $h_q(y^1) < y^1_q < y^2_q = h_q(y^2)$.
     (b) Suppose that $y^1_q < \lceil d^{\min}_q(y^1)\rceil$. If $y^1_{q-1} \leq x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = x^l_{q-1} < y^1_{q-1} = y^2_{q-1} = h_{q-1}(y^2)$. If $y^1_{q-1} > x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = y^1_{q-1} - 1 < y^1_{q-1} = y^2_{q-1} = h_{q-1}(y^2)$.
     Consider that $k_1 > q$. $h(y^1) \leq_l h(y^2)$ follows immediately from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-1$, and $h_q(y^1) \leq y^1_q < y^2_q = h_q(y^2)$.
4. Suppose that $y^2_{q+1} < \lceil d^{\min}_{q+1}(y^2)\rceil$ and $y^2_q \leq x^l_q + 1$. If $y^1_{q-1} \leq x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = x^l_{q-1} < y^1_{q-1} = y^2_{q-1} = h_{q-1}(y^2)$. If $y^1_{q-1} > x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = y^1_{q-1} - 1 < y^1_{q-1} = y^2_{q-1} = h_{q-1}(y^2)$.
5. Suppose that $y^2_{q+1} < \lceil d^{\min}_{q+1}(y^2)\rceil$ and $y^2_q > x^l_{k_2-1} + 1$.
   - Assume that $y^1 \in P$. Then, $h(y^1) = y^1 \leq h(y^2)$.
   - Assume that $y^1 \notin P$. Then, $k_1 \geq q$.
     Consider that $k_1 = q$.
     (a) Suppose that $y^1_q > \lfloor d^{\max}_q(y^1)\rfloor$. Then, $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-1$, and $h_q(y^1) < y^1_q \leq y^2_q - 1 = h_q(y^2)$.
     (b) Suppose that $y^1_q < \lceil d^{\min}_q(y^1)\rceil$. If $y^1_{q-1} \leq x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = x^l_{q-1} < y^1_{q-1} = y^2_{q-1} = h_{q-1}(y^2)$. If $y^1_{q-1} > x^l_{q-1} + 1$, then $h(y^1) \leq_l h(y^2)$ follows from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-2$, and $h_{q-1}(y^1) = y^1_{q-1} - 1 < y^1_{q-1} = y^2_{q-1} = h_{q-1}(y^2)$.

Consider that $k_1 > q$. $h(y^1) \leq_l h(y^2)$ follows immediately from $h_i(y^1) = h_i(y^2)$, $i = 1, 2, \ldots, q-1$, $h_q(y^1) \leq y^1_q \leq y^2_q - 1 = h_q(y^2)$, and $h_i(y^1) \leq x^u_i = h_i(y^2)$, $i = q+1, q+2, \ldots, n$.

Case 4: $k_2 > q + 1$. It follows immediately from Lemma 1 that $h(y^1) \leq_l h(y^2)$.

The above results show that, for every case, it always holds that $h(y^1) \leq_l h(y^2)$. This completes the proof. □

From Definition 1, one can see that, for each $y \in D(P)$, it takes at most n linear programs to compute $h(y)$. Therefore, $h(y)$ is determined in polynomial time for any given $y \in D(P)$.

The following result is well known in the literature.

Theorem 2. *Determining whether there is no integer point in P is a Co-NP complete problem.*

Let $C = \{x \in Z^n \mid a \leq x \leq b\}$, where a and b are two finite vectors of Z^n with $a < b$, and f be an increasing mapping from C into itself under the lexicographic ordering. As a corollary of Corollary 1, Theorem 1 and Theorem 2, we obtain the main result of this paper:

Corollary 2. *Determining whether or not f has a unique fixed point in C is a Co-NP hard problem.*

References

[1] Bernstein, F., Federgruen, A.: Decentralized supply chains with competing retailers under demand uncertainty. Management Science 51, 18–29 (2005)

[2] Bernstein, F., Federgruen, A.: A general equilibrium model for industries with price and service competition. Operations Research 52, 868–886 (2004)

[3] Cachon, G.P.: Stcok wars: inventory competition in a two echelon supply chain. Operations Research 49, 658–674 (2001)

[4] Cachon, G.P., Lariviere, M.A.: Capacity choice and allocation: strategic behavior and supply chain performance. Management Sceince 45, 1091–1108 (1999)

[5] Chang, C.L., Lyuu, Y.D., Ti, Y.W.: The complexity of Tarski's fixed point theorem. Theoretical Computer Science 401, 228–235 (2008)

[6] Echenique, F.: Finding all equilibria in games of strategic complements. Journal of Economic Theory 135, 514–532 (2007)

[7] Fudenberg, D., Tirole, J.: Game Theory. MIT Press, Cambridge (1991)

[8] Gilboa, I., Zemmel, E.: Nash and correlated equilibria: some complexity considerations. Games and Economic Behavior 1, 80–93 (1989)

[9] Lippman, S.A., McCardle, K.F.: The competitive newsboy. Operations Research 45, 54–65 (1997)

[10] Milgrom, P., Roberts, J.: Rationalizability, learning, and equilibrium in games with strategic complementarities. Econometrica 58, 155–1277 (1990)

[11] Milgrom, P., Roberts, J.: Comparing equilibria. American Economic Review 84, 441–459 (1994)

[12] Milgrom, P., Shannon, C.: Monotone comparative statics. Econometrica 62, 157–180 (1994)

[13] Tarski, A.: A lattice-theoretical fixpoint theorem and its applications. Pacific Journal of Mathematics 5, 285–308 (1955)
[14] Topkis, D.M.: Equilibrium points in nonzero-sum n-person submodular games. SIAM Journal on Control and Optimization 17, 773–787 (1979)
[15] Topkis, D.M.: Supermodularity and Complementarity. Princeton University Press, Princeton (1998)
[16] Vives, X.: Nash equilibrium with strategic complementarities. Journal of Mathematical Economics 19, 305–321 (1990)
[17] Vives, X.: Oligopoly Pricing. MIT Press, Cambridge (1999)
[18] Vives, X.: Complemetarities and games: new developments. Journal of Economic Literature XLIII, 437–479 (2005)

Approximation Algorithms for Non-single-minded Profit-Maximization Problems with Limited Supply

Khaled Elbassioni[1], Mahmoud Fouz[2], and Chaitanya Swamy[3,⋆]

[1] Max-Planck-Institut für Informatik, Saarbrücken, Germany
elbassio@mpi-inf.mpg.de
[2] FR Informatik, Universität des Saarlandes, Saarbrücken, Germany
fmahmoud@mpi-inf.mpg.de
[3] Dept. of Combinatorics and Optimization, Univ. Waterloo, Waterloo, ON N2L 3G1
cswamy@math.uwaterloo.ca

Abstract. We consider *profit-maximization* problems for *combinatorial auctions* with *non-single minded valuation functions* and *limited supply*. We obtain fairly general results that relate the approximability of the profit-maximization problem to that of the corresponding *social-welfare-maximization* (SWM) problem, which is the problem of finding an allocation $(S_1, \ldots, S_n)$ satisfying the capacity constraints that has maximum total value $\sum_j v_j(S_j)$. Our results apply to both structured valuation classes, such as *subadditive valuations*, as well as *arbitrary valuations*. For subadditive valuations (and hence *submodular, XOS valuations*), we obtain a solution with profit $OPT_{SWM}/O(\log c_{\max})$, where OPT_{SWM} is the optimum social welfare and $c_{\max}$ is the maximum item-supply; thus, this yields an $O(\log c_{\max})$-approximation for the profit-maximization problem. Furthermore, given *any* class of valuation functions, if the SWM problem for this valuation class has an LP-relaxation (of a certain form) and an algorithm "verifying" an *integrality gap* of α for this LP, then we obtain a solution with profit $OPT_{SWM}/O(\alpha \log c_{\max})$, thus obtaining an $O(\alpha \log c_{\max})$-approximation. The latter result implies an $O(\sqrt{m} \log c_{\max})$-approximation for the profit maximization problem for combinatorial auctions with *arbitrary valuations*, and an $O(\log c_{\max})$-approximation for the non-single-minded *tollbooth problem* on trees. For the special case, when the tree is a path, we also obtain an incomparable $O(\log m)$-approximation (via a different approach) for subadditive valuations, and arbitrary valuations with unlimited supply.[1]

1 Introduction

Profit (or revenue) maximization is a classic and fundamental economic goal, and the design of computationally-efficient item-pricing schemes for various profit-maximization problems has received much recent attention [1,11,2,4,3]. We study the algorithmic problem of *item-pricing for profit-maximization* for *general* (multi unit) *combinatorial auctions* (CAs) with *limited supply*. There are n customers and m items. Each item is available in some limited supply or capacity, and each customer j has a value $v_j(S)$ for

⋆ Supported in part by NSERC grant 327620-09 and an Ontario Early Researcher Award.

[1] Omitted proofs can be found in the full version of the paper.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 462–472, 2010.

each subset S of items specifying the maximum amount she is willing to pay for that set (with $v_j(\emptyset) = 0$). Given a pricing of the items, a *feasible allocation* is an assignment of a (possibly empty) subset S_j to each customer j satisfying (i) the *budget constraints*, which require that the price of S_j (i.e., the total price of the items in S_j) is at most $v_j(S_j)$, and (ii) the *capacity constraints*, which stipulate that the number of customers who are allocated an item be at most the supply of that item. The objective is to determine item prices that maximize the total profit or revenue earned by selling items to the customers. Guruswami et al. [11] introduced the *envy-free* version of the problem, where there is the additional constraint that the set assigned to a customer must maximize her utility (defined as value−price). Item pricing has an appealing simplicity and enforces a basic notion of fairness wherein the seller does not discriminate between customers who get the same item(s). Our focus on item pricing is in keeping with the vast majority of work on algorithms for profit-maximization (for example, the above references; in fact, with unlimited supply and unit-demand valuations, our problem essentially reduces to the *Max-Buy* model in [1]). Various current trading practices are described by item pricing, and thus it becomes pertinent to understand what guarantees are obtainable via such schemes. Profit-maximization problems are typically *NP*-hard, even in various specialized settings, so we will be interested in designing approximation algorithms for these problems.

The framework of combinatorial auctions is an extremely rich framework that encapsulates a variety of applications. In fact, recognizing the generality of the envy-free profit-maximization problem for CAs, Guruswami et al. [11] proceeded to study various more-tractable special cases of the problem. In particular, they introduced the following two structured problems in the *single-minded* (SM) setting, where each customer desires a single fixed set: (a) the *tollbooth problem* where the items are edges of a graph and the customer-sets correspond to paths in this graph, which can be interpreted as the problem of pricing transportation links or network connections; (b) a further special case called the *highway problem* where the graph is a path, which can also be motivated from a scheduling perspective. The non-SM versions of such structured problems can also be used to capture various interesting scenarios.

Our results. We obtain fairly general *polytime approximation guarantees* for profit-maximization problems involving *combinatorial auctions* with *limited supply* and *non-single-minded* valuations. We obtain results for both (a) certain structured valuation classes, namely *subadditive valuations* (where $v(A) + v(B) \geq v(A \cup B)$) and hence, *submodular* valuations, which have been intensely studied recently (e.g. [14,8,9,3]; and (b) *arbitrary valuations*. Our results relate the approximability of the profit-maximization problem to that of the corresponding *social-welfare-maximization* (SWM) problem, which is the problem of finding an allocation $(S_1, \ldots, S_n)$ satisfying the capacity constraints that has maximum total value $\sum_j v_j(S_j)$. Our main theorem, stated informally below and proved in Section 3, shows that any LP-based approximation algorithm that provides an integrality-gap bound for the SWM problem with a given class of valuations, can be leveraged to obtain a corresponding approximation guarantee for the profit-maximization problem with that class of valuations. Let $c_{\max} \leq n$ denote the maximum item supply, and OPT_{SWM} denote the optimum value of the SWM problem, which is clearly an upper bound on the maximum profit achievable.

Theorem 1. *(i) For the class of subadditive (and hence submodular) valuations, one can obtain a solution with profit* $\frac{OPT_{SWM}}{O(\log c_{\max})}$*, thus achieving an* $O(\log c_{\max})$*-approximation; (ii) Given any class of valuations for which the corresponding SWM problem admits a packing-type LP relaxation with an integrality gap of* α *as "verified" by an* α*-approximation algorithm, one can obtain a solution with profit* $\frac{OPT_{SWM}}{O(\alpha \log c_{\max})}$*, thereby achieving an* $O(\alpha \log c_{\max})$*-approximation.*

(Part (ii) above does not imply part (i), because for part (ii) we require an integrality-gap guarantee which, roughly speaking, means that we require an algorithm that returns a "good" solution for *every* profile of n valuations; see Definition 1.)

A key notable aspect of our theorem is its versatility. One can simply "plug in" various known (or easily derivable) results about the SWM problem to obtain approximation algorithms for various limited-supply profit-maximization problems. For example, as *corollaries* of part (ii) of our theorem, we obtain an $O(\sqrt{m} \log c_{\max})$-approximation for profit-maximization for combinatorial auctions with *arbitrary valuations*, and an $O(\log c_{\max})$-approximation for the *non-single-minded tollbooth problem* on trees (see Section 3.1). The first result follows from the various known $O(\sqrt{m})$-approximation algorithms for the SWM problem for CAs with arbitrary valuations that also bound the integrality gap [15,12]. For the second result, we devise a suitable $O(1)$-approximation for the SWM problem corresponding to non-SM tollbooth on trees, by adapting the randomized-rounding approach of Chakrabarty et al. [6].

Notice that with bundle-pricing, which is often used in the context of mechanism design for CAs, the profit-maximization problem becomes equivalent to the SWM problem. Thus, our results provide worst-case bounds on how item-pricing (which may be viewed as a fairness constraint on the seller) diminishes the revenue of the seller versus bundle-pricing. It is also worth remarking that our algorithms for an arbitrary valuation class (i.e., part (ii) above) can be modified in a simple way to return prices and an allocation $(S_1, \ldots, S_n)$ with the following ϵ-"one-sided envy-freeness" property while diminishing the profit by a $(1-\epsilon)$-factor (for any $\epsilon \in [0,1]$): for every non-empty S_j, the utility that j obtains from S_j is at least ϵ times the maximum utility j may obtain from any set (see Remark 2).

The only previous guarantees for limited-supply CAs with a general valuation-class are those obtained via a reduction in [2], showing that an α-approximation for the SWM problem and an algorithm for the unlimited-supply SM problem that returns profit at least OPT_{SWM}/β yield an $\alpha\beta$-approximation. A simple "grouping-by-density" approach gives $\beta = O(\log m + \log n)$; using the best known bound on β [4] yields an $O\big(\alpha(\log m + \log c_{\max})\big)$ guarantee, which is significantly weaker than our guarantees. (E.g., we obtain an $O(\alpha)$-approximation for constant $c_{\max}$.) The $O(\log c_{\max})$-factor we incur is unavoidable if one compares the profit against the optimal social welfare: a well-known example with one item, $n = c_{\max}$ customers shows a gap of $H_{c_{\max}} := 1 + \frac{1}{2} + \cdots + \frac{1}{c_{\max}}$ between the optima of the SWM- and profit-maximization problems. Almost all results for profit-maximization for CAs with non-SM valuations also compare against the optimum social welfare, so they also incur this factor. Also, it is easy to see that with $c_{\max} = 1$, the profit-maximization problem reduces to the SWM problem, so an inapproximability result for the SWM problem also yields an inapproximability result for our problem. Thus, we obtain an $m^{\frac{1}{2}}$-, or n-, inapproximability for

CAs with even SM valuations (see, e.g., [10]), and *APX*-hardness for CAs with subadditive, submodular valuations, and the tollbooth problem on trees.

In Section 4, we consider an alternate approach for the non-SM highway problem that does not use OPT_{SWM} as an upper bound and achieves an (incomparable) $O(\log m)$-approximation factor. We decompose the instance via an exponential-size *configuration LP*, which is solved approximately using the ellipsoid method and rounded via randomized rounding. Here, we use LP duality to handle dependencies arising from the non-SM setting.

Theorem 2. *There is an $O(\log m)$-approximation algorithm for the non-single-minded highway problem with (i) subadditive valuations with limited supply; and (ii) arbitrary valuations with unlimited supply.*

2 Problem Definition and Preliminaries

The general setup of *profit-maximization* problems for (multi unit) *combinatorial auctions* (CAs) is as follows. There are n customers and m items. Let $[n] := \{1, \ldots, n\}$ and $[m] := \{1, \ldots, m\}$. Each item e is available in some limited supply or capacity c_e. Each customer j has a *valuation function* $v_j : 2^{[m]} \mapsto \mathbb{R}_+$, where $v_j(S)$ specifies the maximum amount that customer j is willing to pay for the set S; equivalently this is j's value for receiving the set S of items. We assume that $v_j(\emptyset) = 0$; we often assume for convenience that $v_j(S) \leq v_j(T)$ for $S \subseteq T$, but this monotonicity requirement is not crucial for our results. The objective is to find non-negative prices $p_e \geq 0$ for the items, and an allocation $(S_1, \ldots, S_n)$ of items to customers (where S_j could be empty) so as to maximize the total profit $\sum_{j\in[n]} \sum_{e\in S_j} p_e = \sum_{e\in[m]} p_e |\{j : e \in S_j\}|$ while satisfying the following two constraints: (i) *Budget constraints:* $p\big(S_j\big) := \sum_{e\in S_j} p_e \leq v_j\big(S_j\big)$; and (ii) *Capacity constraints:* Each element e is assigned to at most c_e customers: $|\{j \in [m] : e \in S_j\}| \leq c_e$.

As is standard in the literature on combinatorial auctions and profit-maximization problems (see, e.g., [13,9,3]), we assume that a valuation v is specified by a *demand oracle*, which means that given item prices $\{p_e\}$, the oracle returns a set S that maximizes the *utility* $v(S) - p(S)$. We write $c_{\max} := \max_e c_e$.

An LP relaxation. We consider a natural linear programming (LP) relaxation (P) of the SWM problem for combinatorial auctions, and its dual (D). Throughout, we use j to index customers, e to index items, and S to index sets of items.

$$\begin{aligned}
\max \quad & \sum_{j,S} v_j(S) x_{j,S} & & \text{(P)}\\
\text{s.t.} \quad & \sum_{S} x_{j,S} \leq 1 & \forall j \quad & (1)\\
& \sum_{j} \sum_{S:e\in S} x_{j,S} \leq c_e & \forall e \quad & (2)\\
& x_{j,S} \geq 0 & \forall j, S &
\end{aligned}$$

$$\begin{aligned}
\min \quad & \sum_{e} c_e y_e + \sum_{j} z_j & & \text{(D)}\\
\text{s.t.} \quad & \sum_{e\in S} y_e + z_j \geq v_j(S) & \forall j, S &\\
& y_e, z_j \geq 0 & \forall e, j. &
\end{aligned}$$

In the primal LP, we have a variable $x_{j,S}$ for each customer j and set S that indicates if j receives set S, and we relax the integrality constraints on these variables to obtain (P). The dual (D) has variables z_j and y_e for each customer j and element e respectively, which correspond to the primal constraints (1) and (2) respectively. Although (D) has an exponential number of constraints, it can be solved efficiently given demand oracles for the valuations as these oracles yield the desired separation oracle for (D). This in turn implies that (P) can be solved efficiently. We say that an algorithm $\mathcal{A}$ for the SWM problem is an *LP-based α-approximation algorithm* for a class $\mathcal{V}$ of valuations if for every instance involving valuation functions $(v_1, \ldots, v_n)$, where each $v_j \in \mathcal{V}$, $\mathcal{A}$ returns an integer solution of value at least OPT/α. For example, the algorithm in [9] is an LP-based 2-approximation algorithm for the class of subadditive valuations.

Definition 1. *We say that an algorithm $\mathcal{A}$ for the SWM problem* "verifies" an integrality gap of (at most) α *for an LP-relaxation of the SWM problem (e.g.,*(P)*), if for* every *profile of (monotonic) valuation functions $(v_1, \ldots, v_n)$, $\mathcal{A}$ returns an integer solution of value at least $(LP\text{-}optimum)/\alpha$.*

As emphasized above, an integrality-gap-verifying algorithm must "work" for every valuation-profile. In particular, an LP-based α-approx. algorithm for a given *structured class* of valuations (e.g., submodular or subadditive valuations) *does not* verify the integrality gap for the LP-relaxation. This is precisely why our guarantee for subadditive valuations (part (i) of Theorem 1) does not follow from part (ii) of Theorem 1.

In certain cases however, one may be able to encapsulate the combinatorial structure of the SWM problem with a structured valuation class by formulating a stronger LP-relaxation for the SWM problem, and thereby prove that an approximation algorithm for the structured valuation class is in fact an integrality-gap-verifying approximation algorithm with respect to this *stronger LP-relaxation*. For example, in Section 3.1 we consider the setting where items are edges of a tree and customers desire paths of the tree. This leads to the structured valuation where $v(T) = \max\{v(P) : P \text{ is a path in } T\}$ (with $v(P) \geq 0$ being the value for path P). We design an $O(1)$-approximation algorithm for such valuations, and formulate a stronger LP for the corresponding SWM problem for which our algorithm verifies a constant integrality gap.

For a given instance $\mathcal{I} = \big(m, n, \{v_j\}_{j\in[n]}, \{c(e)\}_{e\in[m]}\big)$, our algorithms will consider different capacity vectors $k \leq c$. Let (P_k) and (D_k) denote respectively (P) and (D) with capacity-vector $k = (k_e)$, and $OPT(k)$ denote their common optimal value. Let $OPT := OPT(c)$ denote the optimum value of (P) with the original capacities. We utilize the following facts, which follow from complementary slackness, and a rounding result that follows from the work of Carr and Vempala [5], and are made explicit in [13].

Claim 1. *Let $k = (k_e)$ be any capacity-vector, and let x^* and (y^*, z^*) be optimal solutions to (P_k) and (D_k) respectively: (i) If $x^*_{j,S} > 0$, then $\sum_{e\in S} y^*_e \leq v_j(S)$; (ii) If $x^*_{j,S} > 0$, and v_j is subadditive, then $\sum_{e\in T} y^*_e \leq v_j(T)$ for any $T \subseteq S$; (iii) If $y^*_e > 0$, then $\sum_{j,S:e\in S} x^*_{j,S} = k_e$.*

Remark 1. As mentioned above, we will sometimes consider a different LP-relaxation when considering the SWM problem with a structured class of valuations. Roughly speaking, the only properties we require of this LP are that it should: (a) include a constraint similar to (2) that encodes the supply constraints; and (b) be a *packing LP*, i.e., have the form $Ax \leq b,\ x \geq 0$ where A is a nonnegative matrix. Given this, parts (i) and (iii) of Claim 1 continue to hold with y_e denoting (as before) the dual variable corresponding to the supply constraint for item e, since the dual is then a covering LP.

Lemma 1 ([5,13]). *Given a fractional solution x to the LP-relaxation of an SWM problem that is a packing LP (e.g., (P$_k$)), and a polytime integrality-gap-verifying α-approx. algorithm $\mathcal{A}$ for this LP, one can express $\frac{x}{\alpha}$ as a convex combination of integer solutions to the LP in polytime. Thus, one can round x to a random integer solution $\hat{x}$ satisfying the following "rounding property": $\frac{x_{j,S}}{\alpha} \leq \Pr[\hat{x}_{j,S} = 1] \leq x_{j,S}$ for all j, S.*

3 The Main Algorithm and Its Applications

Claim 1 leads to the simple, but important observation that if $k \leq c$ and the optimal primal solution x^* is integral, then by using $\{y_e^*\}$ as the prices, one obtains a feasible solution to the profit-maximization problem with profit $\sum_e k_e y_e^*$. There are two main obstacles encountered in leveraging this observation and turning it into an approximation algorithm. First, (P$_k$) will not in general have an integral optimal solution. Second, it is not clear what capacity-vector $k \leq c$ to use, e.g., $\sum_e c_e y_e^*$ could be much smaller than OPT, and in general, $\sum_e k_e y_e^*$ could be quite small for a given capacity-vector $k \leq c$. We overcome these difficulties by taking an approach similar to the one in [7].

We tackle the second difficulty by utilizing a key lemma proved by Cheung and Swamy [7], which is stated in a slightly more general form in Lemma 3 so that it can be readily applied to various profit-maximization problems. This lemma implies that one can efficiently compute a capacity-vector $k \leq c$ and an optimal dual solution (y^*, z^*) to (D$_k$) such that $\sum_e k_e y_e^*$ is $\big(OPT - OPT(\mathbf{1})\big)/O(\log c_{\max})$, where $\mathbf{1}$ denotes the all-ones vector (Corollary 1). To handle the first difficulty, notice that part (i) of Claim 1 implies that one can still use $\{y_e^*\}$ as the prices, provided we obtain an allocation (i.e., integer solution) $\hat{x}$ that only assigns a set S to customer j (i.e., $\hat{x}_{j,S} = 1$) if $x^*_{j,S} > 0$. (In contrast, in the envy-free setting, if we use $\{y_e^*\}$ as the prices then *every* customer j with $z_j^* > 0$, and hence $\sum_S x^*_{j,S} = 1$, must be assigned a set S with $x^*_{j,S} > 0$; this may be impossible with non-single-minded valuations, whereas this is easy to accomplish with single-minded valuations (as there is only one set per customer).) Furthermore, for subadditive valuations, part (ii) of Claim 1 shows that it suffices to obtain an allocation where $\hat{x}_{j,T} = 1$ implies that there is some set $S \supseteq T$ with $x^*_{j,S} > 0$. This is precisely what our algorithms do. We show that one can round x^* into an integer solution $\hat{x}$ satisfying the above *structural* properties, and in addition ensure that the profit obtained, $\sum_{j,T} \hat{x}_{j,T}\big(\sum_{e \in T} y_e^*\big)$, is "close" to $\sum_e k_e y_e^*$ (Lemma 4).

So if $\sum_e k_e y_e^*$ is $OPT/O(\log c_{\max})$ then applying this rounding procedure to the optimal primal solution to (P$_k$) yields a "good" solution. On the other hand, Corollary 1 implies that if this is not the case, then $OPT(\mathbf{1})$ must be large compared to OPT, and then we observe that an α-approximation to the SWM problem trivially yields a solution with profit $OPT(\mathbf{1})/\alpha$ (Lemma 2). In either case we obtain the desired approximation.

The algorithm is described precisely in Algorithm 1. If we use an LP-relaxation different from (P) for the SWM problem with a given valuation class that satisfies the properties stated in Remark 1, then the only change to Algorithm 1 is that we now use this LP and its dual (with the appropriate capacity-vector) instead of (P) and (D) above.

Algorithm 1. Non-single-minded profit-maximization

Input: a profit-maximization instance $\mathcal{I} = \big(m, n, \{v_j\}, \{c_e\}\big)$ with demand oracle for each v_j

1. Define $k^1, k^2, \ldots, k^\ell$ as the following capacity-vectors. Let $k^1_e = 1\ \forall e$. For $j > 1$, let $k^j_e = \min\big\{\lceil (1+\epsilon)k^{j-1}_e \rceil, c_e\big\}$; let ℓ be the smallest index such that $k^\ell = c$.
2. For each vector $k = k^j,\ j \in [\ell]$, compute an optimal solution $(y^{(k)}, z^{(k)})$ to (D_k) maximizing $\sum_e k_e y^{(k)}_e$ among all such solutions. Select $u \in \{k^1, \ldots, k^\ell\}$ maximizing $\sum_e u_e y^{(u)}_e$.
3. Compute an optimal solution $x^{(u)}$ to (P_u). Use $\mathsf{Round}(u, x^{(u)})$ to get a feasible allocation.
4. Use an LP-based α-approx. algorithm for the SWM problem (with the given valuation class) to compute an α-approx. solution to the SWM problem with unit capacities, and a pricing scheme for this allocation that yields profit equal to the social-welfare value of the allocation.
5. Return the better of the following two solutions: (1) allocation computed in step 3 with $\{y^{(u)}_e\}$ as the prices; (2) allocation and pricing scheme computed in step 4.

Round$(\boldsymbol{\mu} = (\boldsymbol{\mu_e}), \boldsymbol{x^*})$ (x^* is an optimal solution to the SWM-LP with capacity-vector μ)
Subadditive valuations: Independently for each customer j, assign j at most one set S by choosing set S with probability $x^*_{j,S}$. If an item e gets allotted to more than μ_e customers this way, then arbitrarily select μ_e customers from among these customers and assign e to these customers. This algorithm can be derandomized via the method of conditional expectations.
General valuation class: Given an integrality-gap-verifying α-approximation algorithm (for (P_μ)), use Lemma 1 to decompose $\frac{x^*}{\alpha}$ into a convex combination $\sum_{r=1}^{\ell} \lambda_r \hat{x}^r$ of integer solutions to (P_μ). (Here $\sum_r \lambda_r = 1$ and $\lambda_r \geq 0$ for each r.) Return $\hat{x}^{(r)}$ with probability λ_r. Given item prices, this algorithm can be derandomized by choosing the solution in $\{\hat{x}^{(1)}, \ldots, \hat{x}^{(r)}\}$ achieving maximum profit.

Analysis. The analysis for both subadditive valuations and a general valuation class proceeds very similarly with the only point of difference being in the analysis of the rounding procedure (Lemma 4). First, observe that if we have an allocation $(S_1, \ldots, S_n)$ that is feasible with unit capacities, then since the sets S_j are disjoint we can charge each customer her valuation for the assigned set by pricing one of her items at this value, and hence, obtain profit equal to the social-welfare value $\sum_j v_j(S_j)$ of the allocation.

Lemma 2. *Given an LP-based α-approximation algorithm for the SWM problem with a given valuation class, one can compute a solution that achieves profit at least $\frac{OPT(\mathbf{1})}{\alpha}$.*

Lemma 3 ([7] paraphrased). *Let* (C_k)*:* $\min\ k^T y + b^T z$ *s.t.* $(y, z) \in \mathcal{P} \subseteq \mathbb{R}^{m+n}_+$, *where $k, y \in \mathbb{R}^m_+$, $b, z \in \mathbb{R}^n_+$, $\mathcal{P} \neq \emptyset$. Let $(y^{(k)}, z^{(k)})$ be an optimal solution to (C_k) that maximizes $k^T y$ among all optimal solutions, and* $\mathsf{opt}(k)$ *denote the optimal value. Let $k^1, \ldots, k^\ell$, and u be as defined in steps 1 and 2 respectively of Algorithm 1. Then,* $\sum_e u_e y^{(u)}_e \geq \big(\mathsf{opt}(c) - \mathsf{opt}(\mathbf{1})\big)/\big(2(1+\epsilon)H_{c_{\max}}\big)$.

Corollary 1. *The capacity-vector u computed in step 2 of Algorithm 1 satisfies the inequality $\sum_e u_e y^{(u)}_e \geq \big(OPT(c) - OPT(\mathbf{1})\big)/\big(2(1+\epsilon)H_{c_{\max}}\big)$.*

Lemma 4. *Let $\hat{x}$ be the integer solution returned by* Round *in step 3 of Algorithm 1. Then $\hat{x}$ combined with the pricing $y^{(u)}$ is a feasible solution to the profit-maximization problem with probability 1, which achieves expected profit at least (i) $\left(1-\frac{1}{e}\right)\sum_e u_e y_e^{(u)}$ for subadditive valuations; and (ii) $\sum_e u_e y_e^{(u)}/\alpha$ for a general valuation class.*

Theorem 3. *Algorithm 1 runs in time* $\text{poly}\big(\textit{input size}, \frac{1}{\epsilon}\big)$ *and achieves an*

(i) $O(\log c_{\max})$*-approximation for subadditive valuations, using the 2-approximation algorithm for the SWM problem with subadditive valuations in [9];*
(ii) $O(\alpha \log c_{\max})$*-approximation for a general valuation class given an integrality-gap-verifying α-approximation algorithm for the SWM problem.*

Remark 2. Note that if the allocation $(S_1, \ldots, S_n)$ returned by Algorithm 1 is obtained via Round, then S_j is always a subset of a *utility-maximizing set* of j, and with a general valuation class, if $S_j \neq \emptyset$, it is a utility-maximizing set (under the computed prices). Also, if $(S_1, \ldots, S_n)$ is obtained in step 4, then we may assume that $v_j(S_j) > v_j(S_j \setminus \{e\})$ for all $e \in S_j$; moreover, with a general valuation class, this solution can be modified to yield an approximate "one-sided envy-freeness" property. We compute $(S_1, \ldots, S_n)$ by rounding $x^{(1)}$ as described in Lemma 1. Now choose prices $\{p'_e\}$ (arbitrarily) such that $p' \geq y^{(1)}$ and $p'(S_j) = \max\{y^{(1)}(S_j), (1-\epsilon)v_j(S_j)\}$ for every j. Since any non-empty S_j is a utility-maximizing set under $y^{(1)}$, it follows that (a) p' is a valid item-pricing yielding profit at least $(1-\epsilon)\sum_j v_j(S_j)$; (b) if $S_j \neq \emptyset$, then the utility j derives from S_j under p' is at least ϵ(max utility of j under p').

3.1 Applications

Arbitrary valuation functions. The integrality gap of (P) is known to be $\Theta(\sqrt{m})$, and there are efficient (deterministic) algorithms that verify this integrality gap [15,12]. So Theorem 3 immediately yields an $O(\sqrt{m}\log c_{\max})$-approximation algorithm for the profit-maximization problem for combinatorial auctions with arbitrary valuations.

Non-single-minded tollbooth problem on trees. In this profit-maximization problem, items are *edges* of a tree and customers desire paths of the tree. More precisely, let $\mathcal{P}$ denote the set of all paths in the tree (including $\emptyset$). Each customer j has a value $v_j(S) \geq 0$ for path $S \in \mathcal{P}$, and may be assigned any (one) path of the tree. This leads to the *structured* valuation function $v_j : 2^{[m]} \mapsto \mathbb{R}_+$ where $v_j(T) = \max\{v_j(S) : S \text{ is a path in } T\}$. We use Algorithm 1 to obtain an $O(\log c_{\max})$-approximation guarantee by formulating an LP-relaxation of the SWM problem that is tailored to this setting and designing an $O(1)$-integrality-gap-verifying algorithm for this LP.

The "new" LP is almost identical to (P), except that we now *only have variables $x_{j,S}$ for $S \in \mathcal{P}$*. Correspondingly, in the dual (D), we only have a constraint for (j, S) when $S \in \mathcal{P}$. Clearly, this new LP satisfies the properties stated in Remark 1, so parts (i) and (iii) of Claim 1 hold for this new LP, and so does Lemma 1. Thus, we only need to design an $O(1)$-integrality-gap-verifying algorithm for this new LP to apply Theorem 3. Let $\{v_j : \mathcal{P} \mapsto \mathbb{R}_+\}_{j\in[n]}$ be any instance and x^* be an optimal solution to this new LP for this instance. We design a randomized algorithm that returns a (random) integer solution $\hat{x}$ of expected objective value $\Omega(\sum_{j,S\in\mathcal{P}} v_j(S)x^*_{j,S})$. This algorithm can be derandomized using the work of [16]; this yields an $O(1)$-integrality-gap-verifying algorithm for the new LP. Our algorithm is a generalization of the one proposed by [6] for unsplittable flow on a line. Root the tree at an arbitrary node. Define the *depth* of an edge (a, b) to be the minimum of the distances of a and b to the root. Define the depth of an edge-set T to be the minimum depth of any edge in T. Let $\alpha = 0.01$.

1. Independently, for every customer j, choose at most one path S, by picking S with probability $\alpha x^*_{j,S}$. Let S_j be the set assigned to j. (If j is unassigned, then $S_j = \emptyset$.)
2. Let $W = \emptyset$. Consider the sets $\{S_j\}$ in non-decreasing order of their depth (breaking ties arbitrarily). For each set $T = S_j$, if T can be added to $\{S_i : i \in W\}$ without violating any capacities, add j to W, otherwise discard T.

Let $\hat{x}$ be the (random) integer solution computed. Using a similar argument as in [6], we can prove that $\Pr[\hat{x}_{j,S} = 1] \geq 0.004 x^*_{j,S}$, so $\mathrm{E}\big[\sum_{j,S\in\mathcal{P}} v_j(S)\hat{x}_{j,S}\big] \geq 0.004 \cdot \sum_{j,S\in\mathcal{P}} v_j(S) x^*_{j,S}$. We thus obtain the following theorem as a corollary of Theorem 3.

Theorem 4. *There is an $O(1)$-integrality-gap-verifying algorithm for the above LP, and thus an $O(\log c_{\max})$-approx. algorithm for the non-SM tollbooth problem on trees.*

Since the above algorithm satisfies the rounding property in Lemma 1, we can use it to round $x^{(u)}$ (more efficiently) to a feasible allocation in step 3 of Algorithm 1, instead of using the Carr-Vempala procedure (which relies on the ellipsoid method).

4 Refinement for the Non-single-Minded Highway Problem

In this section, we describe a different approach that does not use OPT_{SWM} as an upper bound on the optimum profit. Instead our approach is based on using an exponential-size *configuration LP* to decompose the original instance into various smaller (and easier) instances. We use this to obtain an $O(\log m)$-approx. for the non-SM highway problem with subadditive valuations, and arbitrary valuations but unlimited supply (Theorem 2).

Let $\mathcal{P}$ be the set of all intervals on the line (with m edges). As with the non-SM tollbooth problem on trees, each customer j has a value for each subpath (which is now an interval). So we view v_j as a function $v_j : \mathcal{P} \mapsto \mathbb{R}_+$, and *subadditivity* means that $v_j(A \cup B) \leq v_j(A) + v_j(B)$ for any two intervals A, B, where $A \cup B$ is also an interval.

We sketch the proof of Theorem 2. First, we use a standard decomposition to partition the intervals in $\mathcal{P}$ into $O(\log m)$ disjoint sets, where each set is a union of item-disjoint "pyramids". A pyramid is a set of paths that share a common edge; two pyramids $\mathcal{P}_1$ and $\mathcal{P}_2$ are item-disjoint, if $A \cap B = \emptyset$ for all $A \in \mathcal{P}_1,\ B \in \mathcal{P}_2$. Thus, to get an $O(\log m)$-approximation algorithm, it suffices to give an $O(1)$-approximation algorithm when the intervals form a union of item-disjoint pyramids. It is unclear how to achieve a near-optimal solution even in this structured setting, as there are various *dependencies between the pyramids* in a set: a customer can only be assigned an interval in *one* of the pyramids. We solve this "union-of-pyramids" pricing problem as follows. We first trim each pyramid $\mathcal{P}_i$ in our set randomly to a one-sided half-pyramid $\mathcal{H}_i$ by (essentially) ignoring the items to the left or right of the common edge of $\mathcal{P}_i$. The details of this truncation are slightly different depending on whether we have subadditive or arbitrary valuations, but a key observation is that, in expectation, we only lose a factor of 2 by this truncation. We formulate an LP-relaxation for the pricing problem involving these half-pyramids. Let $\mathcal{R}_i$ denote the set of all possible solutions for $\mathcal{H}_i$, where a solution specifies a pricing of the intervals in $\mathcal{H}_i$ (rounded to the nearest power of 2) and an allocation of intervals to customers satisfying the budget and capacity constraints. We introduce a variable $y_{jp} \geq 0$ for each customer j and price p denoting if customer j buys a path at price p, and a variable $x_{i,R}$ for each $R \in \mathcal{R}_i$ denoting whether solution R has been chosen for $\mathcal{H}_i$. Let $p_j(R)$ be the price that j pays under the solution R, and $\mathcal{R}_{i,j,p} = \{R \in \mathcal{H}_i : p_j(R) = p\}$ be the set of solutions for $\mathcal{H}_i$ where j pays price p. We consider the following LP: $\max \sum_{j,p} p \cdot y_{jp}$ s.t. $\sum_{R \in \mathcal{R}_i} x_{i,R} = 1 \;\; \forall i, \quad \sum_p y_{jp} \leq 1 \;\; \forall j$, $\sum_{i,R: R \in \mathcal{R}_{i,j,p}} x_{i,R} \geq y_{jp} \;\; \forall j, p$, and $x_{i,R}, y_{jp} \geq 0 \;\; \forall i, R, j, p$. We solve this LP using the ellipsoid method on the dual problem; the separation oracle is provided by the solution to an easier pricing problem, where the *half-pyramids are now decoupled*. We devise an algorithm based on dynamic programming to compute a near-optimal solution to this pricing problem, which then yields a near-optimal solution to the LP. Finally, we argue that this solution can be rounded to an integer solution losing only an $O(1)$-factor. This gives us the desired $O(1)$-approx. for the "union-of-pyramids" pricing problem, which in turn yields an $O(\log m)$-approx. for our original non-SM highway problem.

Lemma 5. *There is a* $16(1 + \frac{1}{m})$*-approx. algorithm for the non-SM highway problem when intervals form a union of item-disjoint pyramids for (i) subadditive valuations with limited supply; (ii) arbitrary valuations with unlimited supply.*

References

1. Aggarwal, G., Feder, T., Motwani, R., Zhu, A.: Algorithms for multi-product pricing. In: Díaz, J., Karhumäki, J., Lepistö, A., Sannella, D. (eds.) ICALP 2004. LNCS, vol. 3142, pp. 72–83. Springer, Heidelberg (2004)
2. Aggarwal, G., Hartline, J.D.: Knapsack auctions. In: SODA, pp. 1083–1092 (2006)
3. Balcan, M.F., Blum, A., Mansour, Y.: Item pricing for revenue maximization. In: EC, pp. 50–59 (2008)
4. Briest, P., Krysta, P.: Single-minded unlimited supply pricing on sparse instances. In: SODA, pp. 1093–1102 (2006)
5. Carr, R., Vempala, S.: Randomized metarounding. Random Structures and Algorithms 20(3), 343–352 (2002)
6. Chakrabarti, A., Chekuri, C., Gupta, A., Kumar, A.: Approximation algorithms for the unsplittable flow problem. Algorithmica 47 (2007)
7. Cheung, M., Swamy, C.: Approximation algorithms for single-minded envy-free profit-maximization problems with limited supply. In: FOCS, pp. 35–44 (2008)
8. Dobzinski, S., Schapira, M.: An improved approximation algorithm for combinatorial auctions with submodular bidders. In: SODA, pp. 1064–1073 (2006)
9. Feige, U.: On maximizing welfare when utility functions are subadditive. In: STOC, pp. 41–50 (2006)
10. Grigoriev, A., van Loon, J., Sitters, R.A., Uetz, M.: How to sell a graph: Guidelines for graph retailers. In: Fomin, F.V. (ed.) WG 2006. LNCS, vol. 4271, pp. 125–136. Springer, Heidelberg (2006)

11. Guruswami, V., Hartline, J.D., Karlin, A.R., Kempe, D., Kenyon, C., McSherry, F.: On profit-maximizing envy-free pricing. In: SODA, pp. 1164–1173 (2005)
12. Kolliopoulos, S.G., Stein, C.: Approximating disjoint-path problems using packing integer programs. Math. Program. 99(1), 63–87 (2004)
13. Lavi, R., Swamy, C.: Truthful and near-optimal mechanism design via linear programming. In: FOCS (2005)
14. Lehmann, B., Lehmann, D.J., Nisan, N.: Combinatorial auctions with decreasing marginal utilities. In: EC, pp. 18–28 (2001)
15. Raghavan, P.: Probabilistic construction of deterministic algorithms: approximating packing integer programs. J. Comput. Syst. Sci. 37(2), 130–143 (1988)
16. Sivakumar, D.: Algorithmic derandomization via complexity theory. In: STOC, pp. 619–626 (2002)

Approximation Algorithms for Campaign Management

Edith Elkind[1] and Piotr Faliszewski[2]

[1] School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore
[2] Department of Computer Science, AGH Univ. of Sci. and Tech., Kraków, Poland

Abstract. We study electoral campaign management scenarios in which an external party can buy votes, i.e., pay the voters to promote its preferred candidate in their preference rankings. The external party's goal is to make its preferred candidate a winner while paying as little as possible. We describe a 2-approximation algorithm for this problem for a large class of electoral systems known as scoring rules. Our result holds even for weighted voters, and has applications for campaign management in commercial settings. We also give approximation algorithms for our problem for two Condorcet-consistent rules, namely, the Copeland rule and maximin.

1 Introduction

Elections and voting play an important role in the functioning of the modern society. In the standard model of voting, each voter's preferences are represented by a total order over the alternatives (candidates), and some voting rule is used to determine the election winner(s). However, in practice, the voters' preferences are often flexible, and it is possible to affect the outcome of the election by campaigning for or against a certain candidate. Indeed, campaign management is a multi-million dollar industry, and there is overwhelming evidence that the amount of money invested into a candidate's campaign is strongly correlated with her chances of winning the election.

The notion of *bribery* proposed by Faliszewski, Hemaspaandra, and Hemaspaandra [8] can be viewed as a formal model of electoral campaign management. In the model of [8], each (possibly weighted) voter is associated with a certain price, and, by paying the price, the briber can change that voter's vote in any way she likes. The briber's goal, then, is to get a particular candidate elected, subject to a budget constraint. To connect this description with our original campaign management scenario, observe that bribing a (weighted) voter can be interpreted as mounting an election campaign targeted at a particular group of voters with identical preferences.

However, this interpretation does not take into account that in practice it may be relatively easy to convince a voter to make small changes to his vote, but hard or impossible to convince him to adopt an entirely new preference ordering. To remedy this, several subsequent papers [7,9,6] allow the briber to modify the

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 473–482, 2010.

voters' preferences in a more fine-grained manner. However, Faliszewski [7] and Faliszewski et al. [9] depart from the assumption that the voters are represented by the preference orders. We will therefore focus on the framework of Elkind, Faliszewski, and Slinko [6], which operates in the standard model of voting, and assumes that the briber can pay each voter to swap any two candidates that are adjacent in that voter's ordering; this type of bribery is called *swap bribery*. In the context of campaign management, such a swap corresponds to an ad that compares two particular candidates. A special case of swap bribery that was also suggested in [6] is *shift bribery*, where the briber is limited to buying swaps that involve her preferred candidate; in effect, this is equivalent to allowing the briber to shift her preferred candidate up in the voters' preference orderings. The constraint that a campaign ad should involve the briber's preferred candidate is very natural from the ethics perspective; as we will see later, it also leads to more tractable computational problems.

The complexity-theoretic study of swap and shift bribery was initiated by Elkind, Faliszewski, and Slinko [6], where the authors show that the associated computational problem is hard for many voting rules (see also the parametrized-complexity study of Dorn and Schlotter [4]). However, campaign management can be naturally viewed as an optimization problem, and hence we can approach it using the framework of approximation algorithms. This line of research was first suggested in [6], where the authors give a 2-approximation algorithm for shift bribery under the Borda rule. We expand the study of approximation algorithms for shift bribery to voting rules other than Borda, and to weighted voters.

It is straightforward to show that the optimal swap bribery is hard to approximate up to an arbitrary factor for any voting rule for which the possible winner problem is NP-hard (in particular, for k-approval for $k \geq 2$ [2], Borda [2], Copeland [12], and maximin [12]): indeed, the reduction from the possible winner problem to the swap bribery problem given in [6] constructs a bribery of cost 0 for a "yes"-instance of the possible winner problem, and a bribery of non-zero cost for a "no"-instance. Therefore, in this paper we focus on shift bribery.

Our main result is a 2-approximation algorithm for shift bribery under all scoring rules (a large class of voting rules, which includes Borda); our result holds even for weighted voters. Under a scoring rule, each candidate gets a certain number of points from each voter, which is determined by that candidate's position in the voter's preferences, and the winner is the candidate with the maximum number of points. Unlike most of the existing algorithms for scoring rules (see, e.g., [8]), our algorithm does not assume that the number of candidates is constant, but rather accepts the scoring vector as an input. Our proof has an unusual structure: we first design a pseudopolynomial 2-approximation algorithm for our problem, then convert it into a $(2+\varepsilon)$-approximation scheme, and finally turn the $(2+\varepsilon)$-approximation scheme into a 2-approximation algorithm.

Interestingly, shift bribery under scoring rules provides a mathematical framework for campaign management scenarios that are not related to elections. Consider, for example, an advertiser in a sponsored search setting who wants to ensure that his ads get more clicks than those of the competitors, and is willing to make an additional investment in his campaign to achieve that. By associating the competing ads with candidates, search terms with (weighted) voters, and scores for position i with clickthrough rates for an ad in position i, we can reduce the advertiser's problem to shift bribery with weighted voters. This example suggests that our 2-approximation algorithm can be used for campaign management in a variety of settings, including—but not limited to—voting.

We complement our work on scoring rules by describing approximation algorithms for shift bribery under two voting rules that have the attractive property of *Condorcet consistency*, namely, the Copeland rule and maximin.

We omit most proofs due to space restrictions, but missing proofs can be found in our technical report [5].

2 Preliminaries

We first describe relevant notions from computational social choice and define the shift bribery problem. We take $\mathbb{Z}^+$ to be the set of all nonnegative integers.

Elections. An *election* is a pair $E = (C, V)$, where $C = \{c_1, \dots c_m\}$ is the set of *candidates* and $V = (v^1, \dots, v^n)$ is a collection of *voters*. Each voter v^i is described by her *preference order* $\succ^i$, which is a strict linear order over C: $c \succ^i c'$ means that voter v^i prefers c to c'. We will also consider settings where each voter v^i has a *weight* w_i; in this case, her vote is interpreted as w_i votes.

A *voting rule* is a function that given an election $E = (C, V)$ outputs a set $W \subseteq C$ of *election winners*. Note that we do not require the voting rule to produce a unique winner, i.e., we work in the so-called *nonunique-winner model*. This approach is standard in the computational social choice literature. In practice, a voting rule may have to be combined with a tie-breaking rule.

Voting Rules. We will now describe several well-known voting rules that will be considered in this paper. All voting rules listed below are defined for an election $E = (C, V)$ with $C = \{c_1, \dots, c_m\}$, $V = (v^1, \dots, v^n)$. For all rules defined in terms of scores (points), the winner(s) are the candidate(s) with the maximum score (highest number of points).

Scoring rules. A *scoring rule* $\mathcal{R}_{\boldsymbol{\alpha}}$ is described by a vector $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_m)$, where $\alpha_i \in \mathbb{Z}^+$ for $i = 1, \dots, n$, and $\alpha_1 \geq \dots \geq \alpha_m$. Under $\mathcal{R}_{\boldsymbol{\alpha}}$, each candidate c_i receives α_j points from each voter that ranks him in the j-th position. Note that each scoring rule is defined for a fixed number of candidates. Thus, we often consider voting rules that are defined by *families* of scoring rules $(\boldsymbol{\alpha}^m)_{m=1,2,\dots}$, with one vector for each number of candidates. In particular, *Borda* is the rule given by $\alpha^m_j = m - j$ for $j = 1, \dots, m$, and *k-approval* is the rule given by $\alpha^m_j = 1$ for $j \leq k$, $\alpha^m_j = 0$ for $j > k$; 1-approval is also known as *plurality*.

Condorcet consistent rules. For any $c_i, c_j \in C$, let $N_E(c_i, c_j)$ denote the number of voters in E who prefer c_i to c_j. If $N_E(c_i, c_j) > N_E(c_j, c_i)$, then we say that c_i wins the *pairwise election* against c_j. A candidate $c \in C$ is called the *Condorcet winner* if he wins the pairwise elections against all other candidates in C. Note that some elections may not have a Condorcet winner. We say that a voting rule $\mathcal{R}$ is *Condorcet-consistent* if for any election E that has a Condorcet winner c we have $\mathcal{R}(E) = \{c\}$. Two examples of Condorcet-consistent rules are Copeland and maximin, defined as follows. For any rational $\alpha \in [0, 1]$, *Copeland*$^\alpha$ grants one point to a candidate $c_i \in C$ for each pairwise elections that c_i wins, and α points for each pairwise election that c_i ties. The *maximin score* of c_i is the number of votes that c_i receives in her worst pairwise election, i.e., $\min_{c_j \in C \setminus \{c_i\}} N_E(c_i, c_j)$.

We denote by $Sc_E^{\mathcal{R}}(c)$ the $\mathcal{R}$-score of a candidate $c \in C$ in an election $E = (C, V)$; we omit the superscript $\mathcal{R}$ when the voting rule is clear from the context.

The Shift Bribery Problem. This section is based on the definitions from [6]. Consider an election $E = (C, V)$ with $C = \{p, c_1, \dots, c_{m-1}\}$, $V = (v^1, \dots, v^n)$. Suppose that our goal is to ensure that the designated candidate p is a winner of the election under a voting rule $\mathcal{R}$. In order to achieve this goal, we can ask each voter v^i to shift p upwards in her vote by a certain number of positions. This models the fact that we can campaign in favor of p. However, each such shift has a cost. Specifically, each voter v^i has a *cost function* $\pi^i : \mathbb{Z}^+ \to \mathbb{Z}^+$, where $\pi^i(k)$, $k \in \mathbb{Z}^+$, is the cost of shifting p upwards by k positions in $\succ^i$. We require that each π^i, $i = 1, \dots, n$, satisfies $\pi^i(0) = 0$ and $\pi^i(k) \le \pi^i(k+1)$ for $k \in \mathbb{Z}^+$. Also, when v^i ranks p in position t, we require $\pi^i(s) = \pi^i(t-1)$ for all $s \ge t$; thus, the function π^i is fully specified by its values at $1, \dots, t-1$. Note that we assume that $\pi^i(k) < \infty$ for all $i = 1, \dots, n$ and $k \in \mathbb{Z}^+$. However, all our proofs can be generalized to the case where π^i can be $+\infty$ (i.e., some voters cannot be bribed to move p by more than some given number of positions). We seek an action that makes p a winner at the minimum cost.

Definition 1. *Let $\mathcal{R}$ be a voting rule. An instance of* $\mathcal{R}$-SHIFT-BRIBERY *problem is a tuple $I = (C, V, \Pi, p)$, where $C = \{p, c_1, \dots, c_{m-1}\}$, $V = (v^1, \dots, v^n)$ is a collection of preference orders over C, $\Pi = (\pi^1, \dots, \pi^n)$ is a family of cost functions, and $p \in C$ is a designated candidate. The goal is to find a minimal value b such that there is a sequence $\mathbf{t} = (t_1, \dots, t_n) \in (\mathbb{Z}^+)^n$ with the following properties: (a) $b = \sum_{i=1}^n \pi^i(t_i)$, and (b) if for each $i = 1, \dots, n$ we shift p upwards in the i-th vote by t_i positions, then p becomes an $\mathcal{R}$-winner of E. We denote this value of b by* $\mathrm{opt}(I)$.

In WEIGHTED $\mathcal{R}$-SHIFT-BRIBERY, the description of the instance includes a vector of voters' weights $\mathbf{w} = (w_1, \dots, w_n)$, i.e., we have $I = (C, V, \Pi, p, \mathbf{w})$.

We will call the sequence $\mathbf{t} = (t_1, \dots, t_n)$ a *shift-action*. Let $shf(C, V, \mathbf{t})$ denote the election obtained from (C, V) by shifting p upwards by t_i positions in the i-th vote (or placing p on top of that vote, if v^i ranks p in position $t < t_i + 1$ before the bribery). A shift-action is *successful* if p is a winner of $shf(C, V, \mathbf{t})$. Additionally, let $\Pi(\mathbf{t}) = \sum_{i=1}^n \pi^i(t_i)$.

Let $I = (C, V, \Pi, p)$ be an instance of $\mathcal{R}$-SHIFT-BRIBERY and let $\mathbf{t} = (t_1, \dots, t_n)$. Overloading notation, we let $shf(I, \mathbf{t})$ denote an instance $\hat{I} = (C, \hat{V}, \hat{\Pi}, p)$ of $\mathcal{R}$-SHIFT-BRIBERY given by (a) $(C, \hat{V}) = shf(C, V, \mathbf{t})$, and (b) $\hat{\Pi} = (\hat{\pi}^1, \dots, \hat{\pi}^n)$ where for each $i = 1, \dots, n$ we have $\hat{\pi}^i(k) = \pi^i(k + t_i) - \pi^i(t_i)$.

That is, $shf(I, \mathbf{t})$ represents the instance of shift bribery obtained from I by applying the shift-action $\mathbf{t}$; the costs are modified to reflect the fact that some shifts have already been performed.

Given an instance I of $\mathcal{R}$-SHIFT-BRIBERY or WEIGHTED $\mathcal{R}$-SHIFT-BRIBERY, we denote by $|I|$ the representation size of I assuming that all entries of Π (and $\mathbf{w}$, for the weighted case) are given in binary. Similarly, $|\boldsymbol{\alpha}|$ denotes the number of bits in the binary encoding of a scoring vector $\boldsymbol{\alpha}$.

3 Scoring Rules

In this section, we describe a 2-approximation shift-bribery algorithm that works for all scoring rules.

Theorem 1. *There is an algorithm $\mathcal{B}$ that given a scoring rule $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_m)$ and an instance $I = (C, V, \Pi, p)$ of $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY with $|C| = m$, outputs a successful shift-action $\mathbf{t}$ for I that satisfies $\Pi(\mathbf{t}) \leq 2\mathrm{opt}(I)$, and runs in time $\mathrm{poly}(|I|, |\boldsymbol{\alpha}|)$.*

```
procedure A(C, V, Π, p)
begin
  Set m = |C|, n = |V|
  Set M = Σ_{i=1}^{n} π^i(m)
  Set b = ∞;
  for ℓ1 = 0 to M do
    for ℓ2 = 0 to M do
    begin
      I' = buy(I, ℓ1);
      I'' = buy(I', ℓ2);
      if p is an R_α-winner in I'' and
        ℓ1 + ℓ2 < b
      then set b = ℓ1 + ℓ2;
    end
  return b;
end
```

Fig. 1. Algorithm $\mathcal{A}$

We split the proof of Theorem 1 into three steps. First (Proposition 1) we describe a pseudopolynomial 2-approximation algorithm $\mathcal{A}$ for our problem. Then (Proposition 2) we use $\mathcal{A}$ to construct another algorithm $\mathcal{A}'$, which for any $\varepsilon > 0$ produces a $(2+\varepsilon)$-approximation and runs in time polynomial in the instance size and $\frac{1}{\varepsilon}$. Finally, we convert $\mathcal{A}'$ into a 2-approximation algorithm by bootstrapping. Throughout the proof, we fix a scoring rule $\mathcal{R}_{\boldsymbol{\alpha}} = (\alpha_1, \dots, \alpha_m)$. The next lemma is crucial for demonstrating the correctness of our algorithm.

Lemma 1. *Let $\mathbf{s} = (s_1, \dots, s_n)$ be a successful shift-action for (C, V, Π, p), and let $k = Sc_{shf(C,V,\mathbf{s})}(p) - Sc_{(C,V)}(p)$. Then every shift-action $\mathbf{r} = (r_1, \dots, r_n)$ such that $Sc_{shf(C,V,\mathbf{r})}(p) = Sc_{(C,V)}(p) + 2k$ is successful for (C, V, Π, p).*

Proof. When p is shifted from position $i+1$ to position i in some vote $\succ^j$, he obtains $\alpha_i - \alpha_{i+1}$ extra points, while the candidate c that was in position i in $\succ^j$ prior to the shift loses $\alpha_i - \alpha_{i+1}$ points; the scores of all other candidates remain unchanged. Since $\mathbf{s}$ increases p's score by k and p wins in $shf(C, V, \mathbf{s})$, we have $\max_{c \in C}(Sc_{(C,V)}(c) - Sc_{(C,V)}(p)) \leq 2k$. Now, $\mathbf{r}$ increases p's score by $2k$ points, and does not increase the score of any other candidate. The lemma follows. □

We are now ready to implement the first step of our plan. The algorithm $\mathcal{A}$ presented in the next proposition is inspired by the 2-approximation algorithm for the Borda rule that appears in [6]; however, its analysis is substantially different. The main idea of the correctness proof is to keep track of the overlap between, on the one hand, a greedy solution that gives as many points as possible to our preferred candidate, and, on the other hand, an optimal solution.

Proposition 1. *There exists an algorithm $\mathcal{A}$ that given a scoring rule $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_m)$ and an instance $I = (C, V, \Pi, p)$ of $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY with $|C| = m$, outputs a successful shift-action $\mathbf{t}$ for I that runs in time $\mathrm{poly}(|I|, |\boldsymbol{\alpha}|, \sum_{i=1}^{n} \pi^i(m))$ and satisfies $\Pi(\mathbf{t}) \leq 2\mathrm{opt}(I)$.*

Proof. Consider an instance $I = (C, V, \Pi, p)$ of $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY such that $C = \{p, c_1, \dots, c_{m-1}\}$ and $V = (v^1, \dots, v^n)$, and set $E = (C, V)$. For each integer $\ell \geq 0$, and each instance $J = (C, V', \Pi', p)$ of $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY, set $buy(J, \ell) = shf(C, V', \mathbf{t}^\ell)$, where $\mathbf{t}^\ell$ is chosen so as to maximize p's score subject to the constraint $\Pi'(\mathbf{t}^\ell) \leq \ell$, i.e., $\mathbf{t}^\ell \in \arg\max\{Sc_{shf(C,V',\mathbf{t})}(p) \mid \Pi'(\mathbf{t}) \leq \ell\}$. The pseudocode for algorithm $\mathcal{A}$ is given in Figure 1. The next lemma (proof omitted) implies that the running time of $\mathcal{A}$ is polynomial in $|I|$, $|\boldsymbol{\alpha}|$ and $\sum_{i=1}^{n} \pi^i(m)$.

Lemma 2. *For any $\ell \geq 0$, the instance $buy(I, \ell)$ is computable in time $\mathrm{poly}(|I|, |\boldsymbol{\alpha}|, \ell)$.*

It remains to show that $\mathcal{A}$ indeed produces a 2-approximate solution. To this end, we will show that there exist $\ell_1, \ell_2 \leq \sum_{i=1}^{n} \pi^i(m)$ such that $I' = buy(I, \ell_1)$, $I'' = buy(I', \ell_2)$, p is an $\mathcal{R}_{\boldsymbol{\alpha}}$-winner in I'', and $\ell_1 + \ell_2 \leq 2\mathrm{opt}(I)$.

Let $\mathbf{t} = (t_1, \dots, t_n)$ be an optimal shift-action that ensures p's victory, that is, $\Pi(\mathbf{t}) = \mathrm{opt}(I)$. Set $k = Sc_{shf(C,V,\mathbf{t})}(p) - Sc_E(p)$.

Consider an instance I' obtained from I by spending the total cost of the optimal shift-action greedily, i.e., so as to maximize p's score. Formally, let $\ell_1 = \Pi(\mathbf{t})$ and set $I' = buy(I, \ell_1)$. Let $\mathbf{s}' = (s'_1, \dots, s'_n)$ be the shift-action that transforms I into $I' = (C, V', \Pi', p)$, and set $E' = (C, V')$. By construction, we have $Sc_{E'}(p) \geq Sc_E(p) + k$.

Let $\mathbf{r} = (r_1, \dots, r_n)$ be the common part of shift-actions $\mathbf{t}$ and $\mathbf{s}'$, i.e., set $r_i = \min\{t_i, s'_i\}$ for $i = 1, \dots, n$. Let $I^r = shf(C, V, \mathbf{r})$, where $I^r = (C, V^r, \Pi^r, p)$, and set $E^r = (C, V^r)$.

Finally, set $\ell_2 = \Pi'(\mathbf{t} - \mathbf{r})$, $I'' = buy(I', \ell_2)$, and let $\mathbf{s}'' = (s''_1, \dots, s''_n)$ be the shift-action that transforms I' into $I'' = (C, V'', \Pi'', p)$. Let $E'' = (C, V'')$. Observe that for each $i = 1, \dots, n$ we have either $t_i - r_i = 0$, in which case $\pi'^i(t_i - r_i) = 0$, or $t_i - r_i = t_i - s'_i$, in which case $\pi'^i(t_i - r_i) = \pi'^i(t_i - s'_i) = \pi^i(t_i) - \pi^i(s'_i)$. Therefore, we have $\Pi'(\mathbf{t} - \mathbf{r}) \leq \Pi(\mathbf{t})$. Now, the total cost of $\mathbf{s}' + \mathbf{s}''$ is given by $\ell_1 + \ell_2 = \Pi(\mathbf{t}) + \Pi'(\mathbf{t} - \mathbf{r}) \leq 2\Pi(\mathbf{t})$. As $\Pi(\mathbf{t}) = \mathrm{opt}(I)$, we obtain $\ell_1 + \ell_2 \leq 2\mathrm{opt}(I)$. It remains to show that p is a winner in $shf(C, V, \mathbf{s}' + \mathbf{s}'')$.

Set $k^r = Sc_{E^r}(p) - Sc_E(p)$. The shift-actions $\mathbf{t} - \mathbf{r}$ and $\mathbf{s}' - \mathbf{r}$ satisfy

$$Sc_{shf(C,V^r,\mathbf{t}-\mathbf{r})}(p) = Sc_{E^r}(p) + (k - k^r), \tag{1}$$

$$Sc_{shf(C,V^r,\mathbf{s}'-\mathbf{r})}(p) \geq Sc_{E^r}(p) + (k - k^r). \tag{2}$$

We have $\mathit{shf}(C, V^r, \mathbf{t}-\mathbf{r}) = \mathit{shf}(C, V, \mathbf{t})$, so p is an $\mathcal{R}_{\boldsymbol{\alpha}}$-winner in $\mathit{shf}(C, V^r, \mathbf{t}-\mathbf{r})$. Thus, by Lemma 1, any shift-action that increases the score of p in E^r by $2(k-k^r)$ points ensures that p is a winner in the resulting election. We will now show that this holds for the shift-action $\mathbf{s}'' + (\mathbf{s}' - \mathbf{r})$, and hence p is a winner in $\mathit{shf}(C, V^r, \mathbf{s}'' + \mathbf{s}' - \mathbf{r}) = \mathit{shf}(C, V, \mathbf{s}'' + \mathbf{s}')$.

For each $i = 1, \ldots, n$, if $t_i - r_i \neq 0$, then $r_i = s'_i$ and the i'th voter ranks p in the same position both in V' and in V^r. Thus, $\Pi^r(\mathbf{t} - \mathbf{r}) = \Pi'(\mathbf{t} - \mathbf{r}) = \ell_2$, and applying $\mathbf{t} - \mathbf{r}$ to I' increases p's score by the same amount as applying $\mathbf{t} - \mathbf{r}$ to I^r. By equation (1), this implies

$$Sc_{\mathit{shf}(C,V',\mathbf{t}-\mathbf{r})}(p) = Sc_{E'}(p) + (k - k^r). \tag{3}$$

By definition, $\mathbf{s}''$ is a shift-action of cost at most $\ell_2 = \Pi'(\mathbf{t} - \mathbf{r})$ that applied to E' increases p's score as much as possible. Thus, equation (3) implies

$$Sc_{E''}(p) \geq Sc_{E'}(p) + (k - k^r). \tag{4}$$

Since $E'' = \mathit{shf}(E', \mathbf{s}'')$ and $E' = \mathit{shf}(E^r, \mathbf{s}' - \mathbf{r})$, by combining (2) and (4) we obtain the following inequality: $Sc_{E''}(p) \geq Sc_{E'}(p) + (k-k^r) = Sc_{\mathit{shf}(C,V^r,\mathbf{s}'-\mathbf{r})}(p) + (k - k^r) \geq Sc_{E^r}(p) + 2(k - k^r)$. Thus, p is a winner of election E''. □

We will now convert algorithm $\mathcal{A}$ into a $(2+\varepsilon)$-approximation scheme. The main idea of the proof (omitted) is to adaptively scale the bribery price functions.

Proposition 2. *There exists an algorithm $\mathcal{A}'$ that given a rational $\varepsilon > 0$, a scoring rule $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$ and an instance $I = (C, V, \Pi, p)$ of $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY with $|C| = m$, runs in time $\mathrm{poly}(|I|, |\boldsymbol{\alpha}|, \frac{1}{\varepsilon})$ and outputs a successful shift-action $\mathbf{t}$ for I that satisfies $\Pi(\mathbf{t}) \leq (2 + \varepsilon)\mathrm{opt}(I)$.*

Finally, we transform $\mathcal{A}'$ into a 2-approximation algorithm. using a bootstrapping argument.

Proof (of Theorem 1). Let $I = (C, V, \Pi, p)$ be an instance of $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY, and let $\mathbf{t} = (t_1, \ldots, t_n)$ be an optimal shift-action for I. By the pigeonhole principle, for some $i \in \{1, \ldots, n\}$ we have $\pi^i(t_i) \geq \frac{1}{n}\Pi(\mathbf{t})$. Assume for now that we know i and t_i (later, we will show how to get rid of this assumption).

Let $\mathbf{d} = (0^{i-1}, t_i, 0^{m-i})$, and set $I' = \mathit{shf}(I, \mathbf{d})$. We have $\mathrm{opt}(I') = \mathrm{opt}(I) - \pi^i(t_i)$. Let $\varepsilon = \frac{1}{n}$, and let $\mathbf{s} = (s_1, \ldots, s_n)$ be the shift-action produced by $\mathcal{A}'$ on (I', ε). Clearly, p is a winner in $\mathit{shf}(I, \mathbf{s} + \mathbf{d})$. Further, by Proposition 2, we have $\Pi(\mathbf{s}) \leq (2 + \varepsilon)(\Pi(\mathbf{t}) - \pi^i(t_i))$. Therefore, the cost of the shift-action $\mathbf{s} + \mathbf{d}$ can be estimated as $\Pi(\mathbf{s} + \mathbf{d}) = \Pi(\mathbf{s}) + \pi^i(t_i) \leq (2 + \varepsilon)(\Pi(\mathbf{t}) - \pi^i(t_i)) + \pi^i(t_i) \leq 2\Pi(\mathbf{t}) - \pi^i(t_i) + \varepsilon\Pi(\mathbf{t}) \leq 2\Pi(\mathbf{t}) + (\varepsilon\Pi(\mathbf{t}) - \frac{1}{n}\Pi(\mathbf{t})) = 2\Pi(\mathbf{t})$, where we use the fact that $\pi^i(t_i) \geq \frac{1}{n}\Pi(\mathbf{t})$. Thus, $\mathbf{s} + \mathbf{d}$ is a 2-approximate solution.

While we do not know the values of i and t_i, there are only n possibilities for the former and m possibilities for the latter, and we can try them all. □

By using algorithm $\mathcal{B}$ with $\boldsymbol{\alpha} = (m - 1, \ldots, 1, 0)$, we obtain a 2-approximation algorithm for the Borda rule. This algorithm is different from the one given by

Elkind, Faliszewski, and Slinko [6], even though they have the same approximation guarantee. Indeed, their algorithm relies on a different dynamic programming subroutine, whose running time is polynomial in the instance size and $\sum_{i=1}^{m} \alpha_i$ (rather than the instance size and $\sum_{i=1}^{n} \pi^i(m)$, as in our construction). Since for Borda the expression $\sum_{i=1}^{m} \alpha_i$ is polynomial in the size of the instance, this immediately produces a polynomial-time algorithm. It is not hard to see that the algorithm proposed by Elkind, Faliszewski and Slinko [6] can be adapted to work for any scoring rule with $\sum_{i=1}^{m} \alpha_i = \mathrm{poly}(m)$. Of course, such an algorithm would be considerably faster than the three-step procedure of Theorem 1. Indeed, one may wonder if the more complicated algorithm described above is useful at all, since the scoring vectors used in practice often have small coordinates. However, an important feature of our algorithm is that it works even if each voter v^i uses his own scoring vector $\boldsymbol{\alpha}^i$. This means that we can adapt it for *weighted* voters, by replacing a voter of weight w with a scoring vector $(\alpha_1, \ldots, \alpha_m)$ by a unit-weight voter with a scoring vector $(w\alpha_1, \ldots, w\alpha_m)$.

Corollary 1. *There is an algorithm* $\mathcal{B}^w$ *that given a scoring rule* $\boldsymbol{\alpha}$ *and an instance* $I = (C, V, \Pi, p, \mathbf{w})$ *of* WEIGHTED $\mathcal{R}_{\boldsymbol{\alpha}}$-SHIFT-BRIBERY*, outputs a successful shift-action* $\mathbf{t}$ *for* I *that satisfies* $\Pi(\mathbf{t}) \leq 2\mathrm{opt}(I)$*, and runs in time* $\mathrm{poly}(|I|, |\boldsymbol{\alpha}|)$.

Note that there does not seem to be an easy way to derive Corollary 1 from the result of [6]. Indeed, Corollary 1 is quite surprising as it is one of the first positive, nontrivial algorithmic result that applies to all scoring rules, both for the weighted case and for the unweighted case (see also the work of Conitzer, Xia, and Procaccia [13]). The weighted case is particularly important as large voter weights are ubiquitous in campaign management scenarios where a "voter" corresponds to a collection of individuals that can be "bribed" by the same promotional activity (or, in our sponsored search example, where the search terms may differ in popularity).

One may also wonder if algorithm $\mathcal{A}$ can be simplified by using a single **for**-loop, which for each value of ℓ finds the best shift-action of cost ℓ. Would such an algorithm always provide a 2-approximate solution? Similarly, would introducing further **for**-loops improve the guaranteed approximation ratio?

4 Copeland and Maximin

Let us now consider SHIFT-BRIBERY for Copeland and maximin. Elkind, Faliszewski, and Slinko [6] have shown that SHIFT-BRIBERY is NP-hard for both Copeland and maximin. In contrast, we will now provide polynomial-time approximation algorithms for both Copeland and maximin. Further, we argue that regarding the approximation ratio, our result for maximin is asymptotically optimal.

Theorem 2. *There exists a poly-time algorithm that given an instance* $I = (C, V, \Pi, p)$ *of* WEIGHTED Copeland$^\alpha$-SHIFT-BRIBERY *with* $\alpha \in [0, 1] \cap \mathbb{Q}$ *and*

$|C| = m$ outputs a shift-action $\mathbf{s}$ *such that p is a winner in shf$(C, V, \mathbf{s})$ and* $\Pi(\mathbf{s}) \leq m \cdot \mathrm{opt}(I)$.

The proof of this theorem relies on solving carefully crafted instances of the WEIGHTED Copeland$^\alpha$-MICROBRIBERY problem (see [9] for the description of the problem). This approach can be used to obtain an analogous m-approximation algorithm for WEIGHTED maximin-SHIFT-BRIBERY. However, using an algorithm of Caragiannis et al. [3] for Dodgson score, we obtain an $O(\log m)$-approximation for maximin-SHIFT=BRIBERY.

Given an election $E = (C, V)$, the *Dodgson score* of a candidate $c \in C$ is the minimum number of positions by which c needs to be shifted upwards in the preference orders of the voters in V to become the Condorcet winner. Observe that the Dodgson score of a candidate is exactly the cost of shift bribery that makes c the Condorcet winner, assuming that each unit shift has a unit cost.

It is known that determining whether a candidate is a winner in Dodgson elections is a computationally hard problem [1] (specifically, it is Θ_2^{p}-complete [10]). Caragiannis et al. [3] gave a polynomial-time $O(\log m)$-approximation algorithm for computing Dodgson scores. In fact, their algorithm is somewhat more general: using a part of this algorithm as a crucial subroutine, we provide an $O(\log m)$-approximation algorithm for maximin-SHIFT-BRIBERY.

Theorem 3. *There exists a poly-time algorithm that given an instance $I = (C, V, \Pi, p)$ of* maximin-SHIFT-BRIBERY *with $|C| = m$ outputs a shift-action* $\mathbf{s}$ *such that p is a winner in shf$(C, V, \mathbf{s})$ and $\Pi(\mathbf{s}) = O(\log m) \cdot \mathrm{opt}(I)$.*

The approximation guarantee given by Theorem 3 is asymptotically optimal. This follows from the fact that the reduction of EXACT-COVER-BY-3-SETS to maximin-SHIFT-BRIBERY given by Elkind, Faliszewski, and Slinko [6] can be modified to reduce from SET-COVER, in a way that allows maximin-SHIFT-BRIBERY to inherit the inapproximability properties of SET-COVER (see the work of Raz and Safra [11] for inapproximability results for SET-COVER).

5 Conclusions

We have presented approximation algorithms for campaign management under a number of voting rules. Most of our results hold even for weighted voters. We believe that designing algorithms for the case of weighted voters is important, since in realistic campaign management scenarios a "voter" to be bribed is usually a group of voters that can be reached by the same ad. By the same token, it would be interesting to extend our results to settings where we can reach several *non-identical* voters with the same ad; this would correspond to shift bribery with "bulk discounts". Another, more applied direction would be to identify commercial campaign management scenarios (where candidates correspond to services or products) that can be handled using our model; the sponsored search example in the introduction is the first step in that direction. Finally, a natural direction for further study is to design efficient algorithms for shift bribery with better approximation ratios, or to prove that our results are (asymptotically) optimal.

We remark that, assuming $P \neq NP$, NP-hardness proofs for shift-bribery under Borda and Copeland given in [6] preclude the existence of FPTASes for these voting rules. Closing the gap between these hardness results and the easiness results in this paper would be very interesting.

Acknowledgments. We are grateful to WINE referees for their helpful comments and to Ildikó Schlotter for detailed comments on the manuscript. Edith Elkind is supported by NRF Research Fellowship (NRF-RF2009-08). Piotr Faliszewski is supported by AGH University of Technology Grant no. 11.11.120.865, by Polish Ministry of Science and Higher Education grant N-N206-378637, and by Foundation for Polish Science's program Homing/Powroty.

References

1. Bartholdi III, J., Tovey, C., Trick, M.: Voting schemes for which it can be difficult to tell who won the election. Social Choice and Welfare 6(2), 157–165 (1989)
2. Betzler, N., Dorn, B.: Towards a dichotomy of finding possible winners in elections based on scoring rules. In: Královič, R., Niwiński, D. (eds.) MFCS 2009. LNCS, vol. 5734, pp. 124–136. Springer, Heidelberg (2009)
3. Caragiannis, I., Covey, J., Feldman, M., Homan, C., Kaklamanis, C., Karanikolas, N., Procaccia, A., Rosenschein, J.: On the approximability of Dodgson and Young elections. In: Proceedings of SODA 2009, pp. 1058–1067 (January 2009)
4. Dorn, B., Schlotter, I.: Multivariate complexity analysis of swap bribery. In: Proceedings of the 5th International Symposium on Parameterized and Exact Computation (to appear, 2010)
5. Elkind, E., Faliszewski, P.: Approximation algorithms for campaign management. Tech. Rep. arXiv:1004.0334 [cs.GT] (March 2010)
6. Elkind, E., Faliszewski, P., Slinko, A.: Swap bribery. In: Mavronicolas, M., Papadopoulou, V.G. (eds.) Algorithmic Game Theory. LNCS, vol. 5814, pp. 299–310. Springer, Heidelberg (2009)
7. Faliszewski, P.: Nonuniform bribery (short paper). In: Proceedings of AAMAS 2008, pp. 1569–1572 (2008)
8. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.: How hard is bribery in elections? Journal of Artificial Intelligence Research 35, 485–532 (2009)
9. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Llull and Copeland voting computationally resist bribery and constructive control. Journal of Artificial Intelligence Research 35, 275–341 (2009)
10. Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP. Journal of the ACM 44(6), 806–825 (1997)
11. Raz, R., Safra, S.: A sub-constant error-probability low-degree test, and sub-constant error-probability PCP characterization of NP. In: Proceedings of STOC 1997, pp. 475–484. ACM Press, New York (1997)
12. Xia, L., Conitzer, V.: Determining possible and necessary winners under common voting rules given partial orders. In: Proceedings of AAAI 2008, pp. 196–201 (July 2008)
13. Xia, L., Conitzer, V., Procaccia, A.: A scheduling approach to coalitional manipulation. In: Proceedings of EC 2010, pp. 275–284 (June 2010)

Envy-Free Pricing with General Supply Constraints

Sungjin Im[1,⋆], Pinyan Lu[2], and Yajun Wang[2]

[1] Dept. of Computer Science, University of Illinois, Urbana, IL 61801
im3@cs.illinois.edu
[2] Microsoft Research Asia
{pinyanl,yajunw}@microsoft.com

Abstract. The envy-free pricing problem can be stated as finding a pricing and allocation scheme in which each consumer is allocated a set of items that maximize her utility under the pricing. The goal is to maximize seller revenue. We study the problem with *general supply* constraints which are given as an independence system[1] defined over the items. The constraints, for example, can be a number of linear constraints or matroids. This captures the situation where items do not pre-exist, but are produced in reflection of consumer valuation of the items under the limit of resources.

This paper focuses on the case of unit-demand consumers. In the setting, there are n consumers and m items; each item may be produced in multiple copies. Each consumer $i \in [n]$ has a valuation v_{ij} on item j in the set S_i in which she is interested. She must be allocated (if any) an item which gives the maximum (non-negative) utility. Suppose we are given an α-approximation *oracle* for finding the maximum weight independent set for the given independence system (or a slightly stronger oracle); for a large number of natural and interesting supply constraints, constant approximations are available. We obtain the following results.

– $O(\alpha \log n)$-approximation for the general case.
– $O(\alpha k)$-approximation when each consumer is interested in at most k distinct types of items.
– $O(\alpha f)$-approximation when each item is interesting to at most f consumers.

Note that the final two results were previously unknown even without the independence system constraint.

1 Introduction

Every company is an entity that has the goal of maximizing revenues, and faces two fundamental problems, namely producing and pricing items. The limitations

⋆ Supported by a Samsung Fellowship. This work was done while the author was visiting Microsoft Research Asia.

[1] Given a universe of elements U and a collection $\mathcal{I}$ of subsets of elements, the pair $(U, \mathcal{I})$ is said to be an independence system if (1) $\emptyset \in \mathcal{I}$ and (2) (downward closed) if $B \in \mathcal{I}$ and $A \subseteq B$ then $A \in \mathcal{I}$.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 483–491, 2010.

of resources such as materials or human resources often restricts items that can be produced. For example, there may be a limit on the maximum number of items per group of items that can be delivered. Another possibility is that items may consume different types and amount of resources during production.

Another goal we seek to achieve, together with the maximization of revenue, when pricing items, is not to disappoint consumers by offering an insufficient supply. We assume that every consumer will buy certain items that maximize her happiness, which is defined as her valuation of the items minus their price. That is, pricing must guarantee that consumers are allocated the items that they most prefer under the pricing. When the supply of items is pre-given, such a pricing scheme is known as envy-free pricing [11].

This paper initiates a study of the problem of revenue maximization for producing items under given constraints and pricing them in an envy-free fashion. We assume that there are n consumers and that each consumer's valuation of the items is known. There are m distinct types of items that can be produced. We will use $[n]$ and $[m]$ to denote the set of consumers and the items respectively. We model the supply constraints as an *independence system* $([m], \mathcal{I})$. The downward closure property of the system is natural, i.e., if a multi-set of items can be produced, so can its subsets. The independence system captures a variety of interesting constraints such as a number of linear constraints or matroids.

The envy-free pricing problem in the absence of the general supply constraints has been studied primarily for two special cases. The first case is the unit demand consumers case (UD) in which each consumer i is allocated at most one item from among the items S_i of interest. This arises when S_i consists of similar items, so one item can fully meet consumer i's need. The other case is the single-minded consumers case (SM), in which each consumer is interested in a bundle of items B_i. The bundle B_i, only as the entire bundle, has some value to consumer i; the partial acquisition of the bundle has no value to her. This captures the situation when the items in B_i complement each other.

We will examine the unit demand consumers case which is constrained on an independence system $([m], \mathcal{I})$. We consider the general unit demand case together with two interesting special cases. The first case is where each consumer is interested in at most k distinct types of items, i.e., $|S_i| \leq k$ for all i, which we will call the unit demand with bounded set size (UD-BSS). The other case we call the unit demand with bounded frequency (UD-BF), meaning that each item is of interest to a maximum of f consumers, i.e. $|\{i \in [n] \mid j \in S_i\}| \leq f$ for all $j \in [m]$. In other words, this case is where only a small number of users compete for each item.

Our results. Assuming that we have an α-approximation oracle for finding the maximum weight independent set X in $\mathcal{I}$, we show how to deal with the envy-free pricing problem that is constrained on the independence system $([m], \mathcal{I})$. We may require a slightly stronger version of the approximation oracle that has one more matroid constraint, but which still captures many interesting cases for which good approximations are available. As regards the general unit demand case, the bounded set size case and the bounded frequency case, we give

$O(\alpha \log n)$, $O(\alpha k)$ and $O(\alpha f)$ approximations, respectively. In order to emphasize the general supply constraints under consideration, we add the suffix "-C" to each problem name.

Several results were given for pre-existing items, i.e., each item has an individual supply limit. Guruswami et al. gave an $O(\log n)$-approximation for UD by making a clever use of Walrasian equilibrium. Briest [2] showed that for a certain constant $\epsilon > 0$, it is unlikely that there exists an approximation better than $O(\log^{\epsilon} n)$ under a certain complexity assumption. Table 1 summarizes our results with and without the supply constraint $\mathcal{I}$, together with the previously known results. We note that Briest's inapproximability result for UD-BF case is is not formally stated in the paper, but is implied by the hardness instance.

Table 1. Summary of our results for the unit demand case. Our results are marked $*$

	UD-C	UD-BSS-C	UD-BF-C
Upper Bound	$O(\alpha \log n)*$	$O(\alpha k)*$	$O(\alpha f)*$

	UD	UD-BSS $(\lvert S_i\rvert \le k)$	UD-BF $(\lvert\{i \in [n] \mid j \in S_i\}\rvert \le f)$
Upper Bound	$O(\log(n))$ [11]	$O(k)$ *	$O(f)$ *
Lower Bound	$\Omega(\log^{\epsilon}(n))$ [2]	$\Omega(k^{\epsilon})$ [2]	$\Omega(f^{\epsilon})$ [2]

Our algorithms and analysis borrow some ideas from [10,11,1,3]. As is the case in [11], we make crucial use of the connection between Walrasian Equilibria and the envy-free pricing for the unit demand case. The random sampling (partitioning) technique used in [1,3] will play a role in our algorithms and analysis. However, it is non-trivial to incorporate the general independence system; the results in [1,3] are for the unlimited-supplied single-minded consumers case. Due to space limitations, we omit all proofs here, and the proofs will appear in the full version of this paper.

Related Works. The revenue that is given by optimal envy-free pricing was used as a benchmark to study the performance of truthful mechanisms where the valuations of players are not known to the mechanism [8,9]. The UD problem was shown to be APX-hard [11]; the instance assumes that each consumer is interested in a maximum of two items. Given a constant number of types of items, Hartline and Koltun [12] gave very efficient FPTASes for UD and for the unlimited supplied SM. Chen et al. [6] gave an optimal algorithm for selling one item in a network with unlimited supply when each UD consumer's valuations on the item selling in different nodes of the network are determined by an underlying metric. The unlimited supplied SM when all of the bundles have a limited size was studied in [3,1].

2 Preliminaries

2.1 Envy-Free Pricing

We provide a quick overview of the definition of envy-free pricing for the unit demand consumers case which this paper will focus on. For the general definition of envy-free pricing, see [11]. A unit-demand consumer i has a valuation v_{ij} of item j. She will be interested in acquiring only a single (copy of) item. Achieving envy-freeness requires that we allocate (if any) an item to her that maximizes (positive) her utility under the pricing scheme; here the utility of the item j to the consumer i is defined as $v_{ij} - p_j$. We want to maximize the seller's revenue, i.e., the total amount of money collected by selling items.

2.2 Supply Constraints

From the consumer point of view, given the prices $\mathbf{p}$, we need an envy-free allocation $\mathcal{A}$. On the other hand, from the seller's point of view, some allocations may not be feasible. One simple constraint may be that the seller can supply at most c_j copies of item j. In general, the constraints could be more complicated and we could express them as an independent system $([m], \mathcal{I})$, where $\mathcal{I}$ is a collection of multi-subsets of items. A collection $\mathcal{I}$ is said to be an independence system if it satisfies the conditions that (1) $\emptyset \in \mathcal{I}$ and (2) if $A \in \mathcal{I}$ and $A' \subseteq A$ then $A' \subseteq \mathcal{I}$. We say that an allocation is feasible if the multi-set of the allocated items is in $\mathcal{I}$.

In order to simplify our argument, we will sometimes use sets instead of explicitly using multi-sets. That is, we can create n copies of each item j; this is well justified because each consumer will acquire at most one copy of each item. We will use multi-sets or sets depending on which one simplifies our argument.

Our formulation of the constraints as an independent system is very general. It indeed captures a variety of scenarios, in addition to the simplest constraint that each item j can be supplied in a possible maximum quantity of c_j copies. A more complicated example is to assume there are K types of resources that can be used to produce the items and that we have a limited amount of r_k for each resource $k \in [K]$. In order to produce one copy of item j, assume that we need b_{jk} amount of resource k. Then an allocation which uses x_j copies of item j in total, is feasible iff the following constraints are satisfied: $\sum_{j \in [m]} b_{jk} x_j \leq r_k$, for all $k = 1, 2, \ldots, K$. This corresponds exactly to a multi-dimensional knapsack constraint. Our formulation can also express more complicated combinatorial constraints such as *matroid* constraints. An independence system is known as a *matroid* if it satisfies another property that if $X, Y \in \mathcal{I}$ and $|X| < |Y|$, then $\exists\ y \in Y \setminus X$ such that $X \cup \{y\} \in \mathcal{I}$. The independence system can also be the intersection of a number of matroids or linear constraints.

2.3 Envy-Free Pricing and Supply Constraints

This paper focuses on the revenue maximization problem when the pricing and allocation are envy-free and the allocation satisfies the independent system $([m], \mathcal{I})$. We will restrict our concern to the unit-demand consumers case, because the other well-studied single-minded case becomes intractable even when considered with a simple supply constraint. In general, we assume the availability of an α-approximation oracle for finding the maximum weight independent set in $\mathcal{I}$, when the weights are given. More concretely, when we assign a weight to each item, the oracle outputs an independent set in $\mathcal{I}$ whose total weight is at least $1/\alpha$ of that of the maximum weighted independent set in $\mathcal{I}$. (For different copies of the same item, we may assign different weights.)

For technical reasons, depending on the particular problem, we will require an *oracle* to compute an α-approximate maximum weighted set in $\mathcal{I} \cap \mathcal{I}'$, where $\mathcal{I}'$ is an additional matroid. Here $\mathcal{I}'$ can be a partition matroid or a transversal matroid. A partition matroid bounds the number of items that can be picked from each group of items, where the groups are a partition of the items. In a transversal matroid, a set of nodes in one part is independent when they are covered by a matching in a given bipartite graph.

We remark that introducing the approximation oracle involves generalizing the supply constraints. In many interesting cases, even after adding an additional partition matroid or transversal matroid constraint, good approximation algorithms are available. For example, in the case of the knapsack constraint defined in Sec. 2.2, a PTAS is known when K is a constant [7]. In the case of the intersection of any two matroids, an optimal algorithm exists [15]. When $\mathcal{I}$ is the intersection of y matroids, an $O(y)$-approximation exists [14]. In the case of x linear constraints and y matroids, an $O(x + y)$ approximation appears to follow from some recent results [4]; when $y = 2$, a PTAS is recently given [5].

2.4 Walrasian Equilibria

It is known that Walrasian equilibria [13] are closely related to the envy-free pricing for UD. As discussed above, we sometimes treat each copy of an item distinct for the sake of convenience. An allocation $\mathcal{A}$ can then be expressed as a matching because each consumer acquires at most one item. We will use a matching M instead of $\mathcal{A}$ in most cases. A pair of pricing and allocation $(\mathbf{p}, M)$ is said to be a Walrasian equilibrium if it is envy-free and if any unallocated item is priced at zero. As pointed out in [11], Gul and Stacchetti's result characterizes Walrasian Equilibria for UD. Let $\mathrm{MWM}(X, Y)$ denote a maximum weight matching on the subgraph of G induced on $X \cup Y$, where $X \subseteq [n]$ and $Y \subseteq [m]$. Notation-wise, if there is no confusion, we allow $\mathrm{MWM}(A, B)$ to denote the weight of the matching. Furthermore, they gave an algorithm that computes the Walrasian Equilibrium with the highest prices.

Theorem 1. *[10] Let $(\boldsymbol{p}, M)$ be a Walrasian Equilibrium. Then M is a maximum weight matching in G, i.e. $M = MWM([n], [m])$. Furthermore, for any maximum weight matching M' in G, $(\boldsymbol{p}, M')$ is also a Walrasian equilibrium.*

Algorithm MaxWEQ:
Input: $G = ([n] \cup [m], E)$ with (i, j) having weight v_{ij}.
For each item j, let $\hat{p}_j = \mathrm{MWM}([n], [m]) - \mathrm{MWM}([n], [m] \setminus \{j\})$.
Output: $\hat{\mathbf{p}}$ and $\mathrm{MWM}([n], [m])$.

We note that in any example of envy-free pricing, all copies of the same item have the same price. Note that it is the case with the output of MaxWEQ.

Theorem 2 ([10]). *The algorithm MaxWEQ outputs, in polynomial time, a Walrasian Equilibrium which maximizes the item prices. That is, for any pricing $\boldsymbol{p}$ of a Walrasian equilibrium, we have $p_j \leq \hat{p}_j$ for every item j.*

In [11], Guruswami et al. defined a Walrasian equilibrium with reserved prices, and showed how to compute it by reducing the problem to computing a Walrasian equilibrium. We will use these results to obtain an $O(\alpha \log n)$-approximation for the UD with the general supply constraints.

3 General Unit Demand Case

This section will consider the unit demand consumers problem with general supply constraints (UD-C). Our algorithm Alg-UD-C assumes that an α- approximation is available for finding the maximum size independent set constrained on the given independence system $\mathcal{I}$ and also on any transversal matroid defined over the items $[m]$. Formally, for any $E' \subseteq E$, let $\mathcal{I}'(E')$ denote the collection of subsets of items that can be covered by a matching from E'. Then the oracle (α-approximation) outputs $Y \in \mathcal{I} \cap \mathcal{I}'(E')$ such that $|Y| \geq \frac{1}{\alpha} \max_{Z \in \mathcal{I} \cap \mathcal{I}'(E')} |Z|$. Our algorithm is inspired by the $O(\log n)$-approximation for the problem without general supply constraints given in [11]. The analysis is similar to the proof of Lemma 3.1 in [11].

Algorithm: Alg-UD-C for UD-C:
Input: $[m], [n], E$, each edge $(i, j) \in E$ having weight v_{ij}.
Consider the weight of any edge, and say that the weight is λ.
let $E'(\lambda) = \{(i, j) \in E \mid v_{ij} \geq \lambda\}$.
let $\mathcal{I}'(\lambda) = \mathcal{I}'(E'(\lambda))$ denote a transversal matroid defined by $E'(\lambda)$ on $[m]$.
let $B(\lambda)$ denote a set in $\mathcal{I} \cap \mathcal{I}'(\lambda)$ of the maximum size (within a factor of α).
set the reserve prices $\mathbf{r} = (r_1, r_2, ..., r_m)$: if $j \in B(\lambda)$ then $r_j = \lambda$; or $r_j = \infty$.
let $(\mathbf{p}(\lambda), M(\lambda))$ be a Walrasian equilibrium with reserve prices $\mathbf{r}$.
Output: the pair $(\mathbf{p}(\lambda), M(\lambda))$ for any $\lambda \geq 0$ with the maximum seller profit.

Theorem 3. *Suppose that we are given an α-approximation for finding the maximum independent set that is constrained on $\mathcal{I}$ and a transversal matroid. Then Alg-UD-C is an $O(\alpha \log n)$-approximation for the UD-C problem.*

4 Bounded Set Size (UD-BSS)

This section will relax the oracle used in the previous section. That is, we will not require a transversal matroid as in the previous section. Before we present our algorithm for UD-BSS, we first consider a general problem with independent revenue functions over items, which generalizes the setting that each consumer is interested in a single item. This, together with a randomized partition as used in [1], will give the desired result for UD-BSS.

Definition 1 (maximum production vector). *Let $[m]$ be the set of items and $([m], \mathcal{I})$ an independence system. Let $f_j(\cdot) : \mathbb{N} \cup \{0\} \rightarrow R^+$ be a function defined for each $j \in [m]$ satisfying $\frac{f_j(\ell)}{\ell} \geq \frac{f_j(\ell+1)}{\ell+1}$ for all $\ell \geq 1$. For simple notation, we assume $f_j(0) = 0$. The maximum production vector asks the vector $\langle a_1, a_2, \ldots, a_m \rangle$ with $a_j \in \mathbb{N} \cup \{0\}$, such that $\sum_{j \in [m]} f_j(a_j)$ is maximized, and the production vector satisfies $\mathcal{I}$.*

The maximum production vector problem captures the setting where each consumer is interested in a single item. For item j, let $q_1, q_2, \ldots, q_\ell$ be the set of consumers interested in j with valuation $r_1 \geq r_2 \geq \ldots \geq r_\ell$. We can define $f_j(x) = xr_x$ for $x \in [\ell]$ and $f_j(x) = \ell r_\ell$ if $x > \ell$. The maximum production vector will give us the maximum revenue in this setting, since envy-freeness is automatically enforced when each consumer is interested in only a single item.

Theorem 4. *Given an α-approximation oracle for the maximum weighted independent set of $\mathcal{I}$, the maximum production vector problem can be approximated within a factor of 2α.*

Corollary 1. *Given an α-approximation oracle for the maximum weighted independent set in $\mathcal{I}$, one can obtain a 2α-approximation for UD-BSS with $k = 1$.*

The following pseudocode is our algorithm for UD-BSS. Following the randomized partition in [1], for each item, we add all of its copies to Y with probability $1/k$. We then consider only those consumers (i.e. X) who are interested in a single item in Y, in which case Corollary 1 can be applied.

Algorithm Alg-UD-BSS-C for UD-BSS-C:
Input: $[n], [m], v_{ij}$.
let $Y \leftarrow \emptyset$; for each $j \in [m]$, independently add j to Y with probability $1/k$.
let X be the set of consumers only interested in a single item in Y.
let $(\mathbf{p}', M')$ the optimal envy-free pricing for (X, Y)
let Y' be the items matched by M'.
let $(\mathbf{p}, M) = \text{MaxWEQ}([n], Y')$; for all $j \in [m] \setminus Y'$, $p_j = \infty$.
Output: $(\mathbf{p}, M)$

The pricing strategy output by Corollary 1 is an envy-free pricing scheme if only the consumers in X exist in the market. The next lemma shows that adding more customers, thus introducing more demands, does not decrease the revenue under the envy-free condition.

Lemma 1. *Consider a subset of consumers $X \subseteq [n]$. Let $\boldsymbol{p}$ be an envy-free pricing for $(X, [m])$,* where all items in $[m]$ are assigned. *One can compute a Walrasian equilibrium (thus envy-free) of pricing $\boldsymbol{p}'$ for $([n], [m])$ such that $\boldsymbol{p}' \geq \boldsymbol{p}$, i.e. $p'_j \geq p_j$ in polynomial time. Furthermore, the total revenue of the Walrasian equilibrium in $([n], [m])$ is at least the revenue of $\boldsymbol{p}$ in $(X, [m])$.*

Theorem 5. *Given an α-approximation oracle for the maximum weight independent set for $\mathcal{I}$, when $|S_i| \leq k$ for all i, there exists a $(2\alpha ek)$-approximation for UD-BSS-C.*

5 Bounded Frequency (UD-BF)

This section addresses the problem UD-BF, in which each item is of interest to a possible maximum of f consumers. Let $\{G_1, G_2, \ldots, G_\ell\}$ denote a partition of items $[m]$, and $\mu_l \geq 0$ be an integer that is associated with G_l for $l \in [\ell]$. An collection of items $\mathcal{I}' \subseteq 2^{[m]}$, with items $[m]$, is said to be a partition matroid if $X \in \mathcal{I}'$ if and only if for all $l \in [\ell]$, $|X \cap G_l| \leq \mu_l$. We assume that for any partition matroid $([m], \mathcal{I}')$ (for our purposes, we only need $\mu_l \in \{0, 1\}$ for all l), we can use an α-approximation for finding the maximum weight independence set constrained on $\mathcal{I}$ and $\mathcal{I}'$. We give the following randomized algorithm.

Algorithm Alg-UD-BF-C for UD-BF-C:
Input: $[n], [m], v_{ij}, \mathcal{I}$.
let $X \leftarrow \emptyset$; for each $i \in [n]$, (independently) add i to X with probability $1/f$.
let $Y := \{j \in [m] \mid$ there exists only one consumer $i \in X$ s.t. $j \in S_i\}$.
For each $j \in Y$, assign to j weight v_{ij}, where $i \in X$ s.t. $j \in S_i$.
let $\mathcal{I}' = \{B \subseteq [m] \mid \forall i \in X, |S_i \cap B \cap Y| \leq 1\}$.
$Y' \leftarrow$ the maximum weight independent subset of Y constrained on $\mathcal{I}$ and $\mathcal{I}'$
Output: $(\mathbf{p}, M) \leftarrow$ MaxWEQ$([n], Y')$, for any $j \notin Y'$, $p_j = \infty$.

Using a random sampling, the algorithm Alg-UD-BF-C recasts the given instance to the market (X, Y) where each item in Y is interesting only to a single consumer in X. This creates a natural partition matroid on $[m]$. With the aid of the α-approximation, the algorithm outputs the desired result.

Theorem 6. *Consider any partition matroid $([m], \mathcal{I}')$. Suppose that we have an α-approximation for the maximum weight independent set constrained on $\mathcal{I}$ and also $\mathcal{I}'$. Then the randomized algorithm Alg-UD-BF-C outputs an envy-free pricing that gives an expected revenue that is optimal within a factor of $\alpha e f$.*

References

1. Balcan, M.-F., Blum, A.: Approximation algorithms and online mechanisms for item pricing. In: EC 2006, pp. 29–35. ACM, New York (2006)
2. Briest, P.: Uniform budgets and the envy-free pricing problem. In: ICALP (1), pp. 808–819 (2008)
3. Briest, P., Krysta, P.: Single-minded unlimited supply pricing on sparse instances. In: SODA 2006, pp. 1093–1102 (2006)
4. Chekuri, C.: Personal Communication (2010)
5. Chekuri, C., Vondrák, J., Zenklusen, R.: Multi-budgeted matchings and matroid intersection via dependent rounding (2010) (manuscript)
6. Chen, N., Ghosh, A., Vassilvitskii, S.: Optimal envy-free pricing with metric substitutability. In: EC 2008, pp. 60–69 (2008)
7. Frieze, A.M., Clarke, M.R.B.: Approximation algorithms for the m-dimensional 0-1 knapsack problem: Worst-case and probabilistic analyses. European Journal of Operational Research 15(1), 100–109 (1984)
8. Goldberg, A.V., Hartline, J.D.: Competitive auctions for multiple digital goods. In: Meyer auf der Heide, F. (ed.) ESA 2001. LNCS, vol. 2161, pp. 416–427. Springer, Heidelberg (2001)
9. Goldberg, A.V., Hartline, J.D., Wright, A.: Competitive auctions and digital goods. In: SODA, pp. 735–744 (2001)

10. Gul, F., Stacchetti, E.: Walrasian equilibrium with gross substitues. Journal of Economic Theory 87, 95–124 (1999)
11. Guruswami, V., Hartline, J.D., Karlin, A.R., Kempe, D., Kenyon, C., McSherry, F.: On profit-maximizing envy-free pricing. In: SODA, pp. 1164–1173 (2005)
12. Hartline, J.D., Koltun, V.: Near-optimal pricing in near-linear time. In: Dehne, F., López-Ortiz, A., Sack, J.-R. (eds.) WADS 2005. LNCS, vol. 3608, pp. 422–431. Springer, Heidelberg (2005)
13. Koopmans, T., Beckmann, M.: Assignment problems and the location of economic activities. Econometrica 25, 53–76 (1957)
14. Lee, J., Mirrokni, V.S., Nagarajan, V., Sviridenko, M.: Non-monotone submodular maximization under matroid and knapsack constraints. In: STOC, pp. 323–332 (2009)
15. Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency, vol. 24. Springer, Heidelberg (2003)

Impersonation Strategies in Auctions

Ian A. Kash and David C. Parkes

CRCS and SEAS, Harvard University
kash@seas.harvard.edu, parkes@eecs.harvard.edu

Abstract. A common approach to analyzing repeated auctions, such as sponsored search auctions, is to treat them as complete information games, because it is assumed that, over time, players learn each other's types. This overlooks the possibility that players may impersonate another type. Many standard auctions (including generalized second price auctions and core-selecting auctions), as well as the Kelly mechanism, have profitable impersonations. We define a notion of impersonation-proofness for the auction mechanism coupled with a process by which players learn about each other's type, and show an equivalence to a problem of dominant-strategy mechanism design.

Keywords: Auctions, Impersonation, Kelly Mechanism, Ad Auctions.

1 Introduction

Most analyses of auctions emphasize uncertainty. While a bidder may know his value for an item, he is unlikely to know exactly how every other bidder values it. However, he is likely to have some beliefs about others, and the standard Bayesian analysis of auctions requires that, in equilibrium, bidders act optimally based on their beliefs about other bidders.

This approach is natural for a single, stand-alone auction. However, in some cases (for example in sponsored search auctions) the same bidders will participate in many auctions. Thus, a notion of equilibrium should take into account that, over the course of many auctions, bidders will learn about each others valuations. Unfortunately, as the folk theorem shows, the set of potential equilibria in such a repeated setting is large and complicated.

One natural class of equilibria are those where players spend some time learning until they reach an equilibrium of the "stage game," after which they use the same strategies forever. If players are no longer learning, then it seems reasonable that they have complete information about the types of other players.

However, this analysis glosses over a key point. These complete information equilibria will only be reached if players *correctly* learn each other's types. As the learning process is part of the repeated game, players may have an incentive to deviate during this process. This could be prevented by finding learning algorithms that are themselves an equilibrium of the repeated game. This is the approach taken by, for example, Brafman and Tennenholtz [3] and Ashlagi et

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 492–495, 2010.

al. [2]. However, such algorithms require that most or all of the players participate and that they learn in a particular fashion, so it seems unlikely that they will be a good predictor of real-world behavior.

Our approach, in the same spirit as complete information analysis, is to assume that players will learn and reach an equilibrium. In particular, we ignore the possibility that players will do something other than learn the types of other players. We also ignore their rewards during the learning period and assume they only care about the long-run behavior that the complete information game naturally captures. With these assumptions, should we expect to reach a complete information equilibrium? In this paper, we argue that the answer is no. In particular, players have the option to *impersonate* another type and participate in the learning algorithm as if their true type were the type they are impersonating. This causes the other players to believe they are playing a different complete information game and so a different strategy profile is reached.

Complete information equilibrium analysis has been used for many auctions, notably Kelly [7] for bandwidth allocation; Edelman, Ostrovsky and Schwartz [5] for generalized second price auctions; and Day and Milgrom [4] for core-selecting auctions. All turn out to have profitable impersonation strategies. We define a notion of *impersonation-proofness* and show that it is equivalent to selecting equilibria that implement the outcome of a dominant strategy mechanism for the incomplete information problem.

2 Model

Consider a Bayesian game G. Each of n players i has a type $\theta_i \in \Theta_i$ drawn according to the joint distribution $F(\theta_1, \ldots, \theta_n)$, which is common knowledge. Each player chooses an action $a_i \in A_i$ based on his type. Each player's utility, which may depend on the joint action and his type, is $u_i(a, \theta_i)$. A Bayesian Nash equilibrium is then defined in terms of an expectation over the types of players.

If players play this game repeatedly, we expect them to learn about their opponents. Theorems have been established regarding the Nash equilibria of the complete information game G_θ for a number of different Bayesian games G, where G_θ is G with $\theta = (\theta_1, \ldots, \theta_n)$ made common knowledge.

We model this learning process as a *mediator*: players submit a type and the mediator suggests an action for each player. To keep in mind our intuition of players learning, we require that if players report θ to the mediator, the mediator suggests a Nash equilibrium of G_θ, as a goal of most learning dynamics is to reach equilibrium[1] [8]. Formally, given a Bayesian game G, a mediator is a function $M : \Theta \rightarrow A$ such that $M(\theta)$ is a Nash equilibrium of G_θ, where $\Theta = \Theta_1 \times \ldots \times \Theta_n$ and $A = A_1 \times \ldots \times A_n$.

Given a Bayesian game G and a mediator M, we have the mediated game G_M. First, each player i learns θ_i and submits some θ_i' to M. Then i learns $M_i(\theta')$ and selects an action a_i. This formulation suggests the obvious strategy

[1] Our formalism of a mediator is inspired by that of Ashlagi et. al [1], but our motivation and definition are slightly different.

of lying to the mediator in the first stage. We call such strategies *impersonation strategies* because in practice they amount to impersonating some other type for a period of time to convince other players that the player is actually of that type. We focus on impersonation strategies in a strong sense: the player not only lies to the mediator but then follows the mediator's advice based on that lie. Thus, the player can continue this impersonation indefinitely. A player has a *profitable impersonation* when he can increase his payoff by using an impersonation strategy when all other players report truthfully and follow the mediator's advice. Formally, i has a profitable impersonation if there exists some θ_i' such that

$$u_i(M_i(\theta_i', \theta_{-i}), \theta_i) > u_i(M_i(\theta), \theta_i). \tag{1}$$

With this in mind, we say a mediated game G_M is *impersonation-proof* if no player ever has a profitable impersonation. Formally, for all i, θ, and θ_i',

$$u_i(M_i(\theta), \theta_i) \geq u_i(M_i(\theta_i', \theta_{-i}), \theta_i). \tag{2}$$

3 Example: The Kelly Mechanism

As mentioned in the introduction, many games have profitable impersonations. In this short paper, we analyze one such example. Suppose the owner of a network wants to allocate bandwidth to users of the network. Kelly [7] introduced a simple mechanism for this problem. Each player i submits a bid b_i. He then receives a $b_i / \sum_j b_j$ fraction of the bandwidth and pays a cost of b_i. This mechanism has the nice property that each player needs only submit a bid rather than describe his entire, potentially complicated, utility function. Furthermore, if all players have concave utility functions, then there is a unique complete information Nash equilibrium which can be found using a simple learning algorithm[2]. Johari and Tsitsiklis [6] showed that this mechanism has a price of anarchy of 4/3.

The following lemma (whose proof is omitted) shows that it is quite common for players to have profitable impersonations. In particular, this means that, despite having a good price of anarchy, actual performance could be poor.

Lemma 1. *Consider the Kelly mechanism with two players who have linear utility functions ($u_i(x_i) = \theta_i x_i$) with $\theta_1, \theta_2 > 0$. Unless $\theta_1 = \theta_2$, both players have a profitable impersonation.*

To illustrate Lemma 1, suppose $\theta_1 = 2$ and $\theta_2 = 1$. Then the unique equilibrium has bids $(4/9, 2/9)$ so player 1's utility is $8/9$. Now suppose player 1 impersonates $\theta_1 = 3$. Now the unique "equilibrium" has bids $(9/16, 3/16)$ and player 1's utility is $15/16$. Thus player 1 has gained by pretending to have a higher valuation.

[2] There are different models of how players optimize for this mechanism. We assume players are *price-anticipating*: they take into account how their bid affects the price they pay when determining their optimal bid.

4 Impersonation-Proofness

In this section, we examine when mediated games are impersonation-proof and thus it is plausible that players would be willing to participate as their true type. Consider a Bayesian game G in which each player i chooses an action a_i and then his utility is determined by the vector of actions a and his type θ_i. For example, in a first price auction an action is a bid and the vector of bids determines the winner and each player's payment. In problems of interest, G is induced as the result of a designed mechanism, with a set of outcomes O, a set of joint actions A, a mapping $o : A \to O$, and utility functions $u_i(a, \theta_i) = u_i(o(a), \theta_i)$. We refer to $S = (\Theta, O, u)$ as the social choice problem domain.

Any mediator M for G is a function from type vectors to action vectors, and thus when combined with the mapping $o : A \to O$ is itself a mechanism. In fact, this is a *direct revelation* mechanism. A mediator coupled with a game defines a subset of the space of possible direct revelation mechanisms, insisting that $M(\theta)$ be a complete information Nash equilibrium for all θ.

Theorem 1. *Let G be a Bayesian game that is a mechanism (not necessarily incentive compatible) for a social choice problem domain $S = (\Theta, O, u)$. There exists an impersonation-proof mediator M for G iff there exists a dominant strategy mechanism D for S such that for all θ there exists an $a(\theta)$ that is a Nash equilibrium for G_θ and $D(\theta) = o(a(\theta))$*

Theorem 1 suggests a general approach to finding impersonation-proof mediators: take a dominant strategy mechanism D for the same problem domain and find equilibria that implement $D(\theta)$ in each game G_θ. For example, the bidder-optimal locally envy-free equilibrium of a generalized second price auction implements the VCG outcome [5], so the mediator that selects this equilibrium is impersonation-proof.

References

1. Ashlagi, I., Monderer, D., Tennenholtz, M.: Mediators in position auctions. Games and Economic Behavior 67(1), 2–21 (2009)
2. Ashlagi, I., Tennenholtz, M., Monderer, D.: Robust learning equilibrium. In: Proc. of the 22nd Conference in Uncertainty in Artificial Intelligence, UAI (2006)
3. Brafman, R.I., Tennenholtz, M.: Efficient learning equilibrium. Artif. Intell. 159(1-2), 27–47 (2004)
4. Day, R., Milgom, P.: Core-selecting package auctions. Int. J. Game Theory 36, 393–407 (2008)
5. Edelman, B., Ostrovsky, M., Schwartz, M.: Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. American Economic Review 97(1), 242–259 (2007)
6. Johari, R., Tsitsiklis, J.N.: Efficiency loss in a network resource allocation game. Math. Oper. Res. 29(3), 407–435 (2004)
7. Kelly, F.P.: Charging and rate control for elastic traffic. European Transactions on Telecommunications 8, 33–37 (1997)
8. Shoham, Y., Powers, R., Grenager, T.: If multi-agent learning is the answer, what is the question? Arificial Intelligence 171(7), 365–377 (2007)

Market Equilibrium with Transaction Costs

Sourav Chakraborty[1], Nikhil R. Devanur[2], and Chinmay Karande[3,⋆]

[1] Chennai Mathematical Institute, India
sourav@cmi.ac.in
[2] Microsoft Research, Redmond WA
nikdev@microsoft.com
[3] Google Inc., Mountain View CA
chinmayk@google.com

Abstract. Identical products being sold at different prices in different locations is a common phenomenon. To model such scenarios, we supplement the classical Fisher market model by introducing *transaction costs*. For every buyer i and good j, there is a transaction cost of c_{ij}; if the price of good j is p_j, then the cost to the buyer i *per unit* of j is $p_j + c_{ij}$. The same good can thus be sold at different (effective) prices to different buyers. We provide a combinatorial algorithm that computes ϵ-approximate equilibrium prices and allocations in $O\left(\frac{1}{\epsilon}(n + \log m)mn \log(B/\epsilon)\right)$ operations - where m is the number goods, n is the number of buyers and B is the sum of the budgets of all the buyers.

1 Introduction

Identical products being sold at different prices in different locations is a common phenomenon. Price differences might occur due to different reasons such as

– Shipping costs. Oranges produced in Florida are cheaper in Florida than they are in Alaska, for example.
– Trade restrictions. A seller with access to a wider market might sustain a higher price than one that does not.
– Price discrimination. A good might be priced differently for different people based on their respective ability to pay. For example, conference registration fees are typically lower for students than for professors.

To capture such scenarios, we supplement the classical Fisher model of a market by introducing *transaction costs*. For every buyer i and every good j, there is a transaction cost of c_{ij}; if the price of good j is p_j, then the cost to the buyer i *per unit* of j is $p_j + c_{ij}$. The same good can thus be sold at different effective prices to different buyers. Apart from non-negativity, the transaction costs are not restricted in any way and in particular, do not have to satisfy the triangle inequality.

Fisher's Market Model with Transaction Costs. In Fisher's model, a market $\mathcal{M}$ has n buyers and m divisible goods. Every buyer i has budget B_i. We

⋆ Part of the work done during the author's internship at Microsoft Research.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 496–504, 2010.

consider *linear* utility functions, *i.e.*, the utility of a buyer i on obtaining a bundle of goods $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots)$ is $\sum_j u_{ij} x_{ij}$ where u_{ij} are given constants. Each good has an available supply of one unit (which is without loss of generality). In addition to its price, a buyer also pays a transaction cost c_{ij} per unit of good j. The allocation bundle for buyer i is a vector $\boldsymbol{x}_i$ such that x_{ij} denotes the amount of good j allocated to buyer i. A price vector $\boldsymbol{p}$ is an equilibrium of $\mathcal{M}$ if there exists allocations $\boldsymbol{x}_i$ such that

– $\boldsymbol{x}_i$ maximizes the utility of i among all bundles that satisfy the budget constraint, i.e. $\boldsymbol{x}_i \in \arg\max_{\boldsymbol{y}_i} \{\sum_j u_{ij} y_{ij} : \sum_j (p_j + c_{ij}) y_{ij} \leq B_i\}$
– Every good is either fully allocated or is priced at zero, i.e. $\forall j$, either $\sum_i x_{ij} = 1$ or $p_j = 0$.

Characterization of Market Equilibrium. We now characterize the equilibrium prices and allocations in our model. The ratio $u_{ij}/(p_j + c_{ij})$ denotes the amount of utility gained by buyer i through one dollar spent on good j. At given prices, a bundle of goods that maximizes the total utility of a buyer contains only goods that maximize this ratio. Let $\alpha_i = \max_j u_{ij}/(p_j + c_{ij})$ be the bang-per-buck of buyer i at given prices. We will call the set $D_i = \{\, j \mid u_{ij} = \alpha_i (p_j + c_{ij}) \,\}$ the demand set of buyer i. Hence, $x_{ij} > 0 \;\Rightarrow\; j \in D_i$. The conditions characterizing these equilibrium prices and allocations appear in table A below.

An ϵ-approximate market equilibrium is characterized by relaxing the market clearing condition (Equation (3)) and optimal allocation condition (Equation (4)). Refer to equations (7) and (8) in table B.

A: Market Equilibrium

$$\forall i \quad \sum_j (p_j + c_{ij}) x_{ij} = B_i \tag{1}$$

$$\forall j \quad \sum_i x_{ij} \leq 1 \tag{2}$$

$$\forall j \quad p_j > 0 \;\Rightarrow\; \sum_i x_{ij} = 1 \tag{3}$$

$$\forall i, j \quad x_{ij} > 0 \;\Rightarrow\; \frac{u_{ij}}{\alpha_i} = p_j + c_{ij} \tag{4}$$

B: ϵ-Approximate Market Equilibrium

$$\sum_j (p_j + c_{ij}) x_{ij} = B_i \tag{5}$$

$$\sum_i x_{ij} \leq 1 \tag{6}$$

$$p_j > \epsilon \;\Rightarrow\; \sum_i x_{ij} \geq 1/(1+\epsilon) \tag{7}$$

$$x_{ij} > 0 \;\Rightarrow\; \frac{u_{ij}}{\alpha_i} \geq \frac{p_j + c_{ij}}{1+\epsilon} \tag{8}$$

The relaxation of exact equilibrium conditions can be achieved in other ways. For example, [7] use a definition of ϵ-approximate market equilibrium that relaxes the budget constraints. Our algorithm can be easily adapted to this definition by simple modifications to the termination conditions.

Our Result

Our main result is a combinatorial algorithm that computes ϵ-approximate equilibrium prices and allocations in $O\left(\frac{1}{\epsilon}(n + \log m) mn \log(B/\epsilon)\right)$ operations - where m is the number goods, n is the number of buyers and B is the sum of the budgets of all the buyers. This algorithm is a generalization of the auction algorithm of Garg and Kapoor [7] to our model with the transaction costs. This generalization is not straight forward; the presence of transaction costs introduces new challenges. The term 'auction algorithm' is used to describe ascending price algorithms (such as the one in [7]) which maintain a feasible allocation at all times. The algorithm makes progress by revoking a portion of goods currently assigned to a buyer and reallocating it to another buyer offering a higher price. Our method of reallocating goods is similar in spirit to the path auctions used by [9]. The auction algorithms in both [7] and [9] crucially use the properties of monotonic decrease in surplus and acyclicity of the demand graph. However, these properties cease to exist when transaction costs are introduced. The main technical contribution of this paper is in dealing with the absence of these properties (and yet getting almost the same results). A more detailed discussion of the same is presented in the full version of this paper [1].

Related Work

The computation of economic and game theoretic equilibria has been an active area of research over the past decade. Hardness results and algorithmic results [10] have been delineating the boundary between what is efficiently computable and what is not.

Convex programming has been one of the main tools in designing algorithms for market equilibrium. A simple modification of the convex program introduced by [6,5] captures the equilibria of our problem as its optimal solution. (Refer to [1] for details) This proves *existence and uniqueness* of equilibria. It also implies that the ellipsoid algorithm can be used to get a polynomial time algorithm to compute the equilibrium[1]. The auction algorithm is combinatorial, runs faster and provides a simple alternative that can be implemented efficiently in practice. It is not clear if one can construct an interior point algorithm to solve the convex program. Ye [12] gave one such algorithm for the Eisenberg-Gale convex program.

A strongly polynomial time algorithm for the Fisher linear market was given by Orlin [11]; it does not seem like his ideas can be adapted directly to our setting. Chen *et al* [2] study a model similar to ours, in the context of profit-maximizing envy-free pricing (for a single commodity but at different locations).

Codenotti *et al* [3] studied transaction costs that are a fixed fraction of the price, and hence can be interpreted as taxes. The taxes could be uniform, that is, depend only on the good, or non-uniform, that is, depend on the good and the buyer. In the Fisher's model, our algorithm can also handle per-dollar taxes, with minimum modifications.

Extensions and Open Problems

All of our results can be easily extended to quasi-linear utilities, that is, the buyers have utility for money as well, which is normalized to 1. So the utility of the bundle $\boldsymbol{x}_i$ is $\sum_j (u_{ij} - p_j)x_{ij}$. Extending the results to other common utility functions is an open problem. In particular, Garg, Kapoor and Vazirani [8] extend

[1] Since the equilibrium could be irrational, the ellipsoid algorithm would compute an equilibrium with precision δ in time proportional to $\log(1/\delta)$.

the auction algorithm to separable weak gross substitute utilities. The potential function they use is the total surplus, and we don't know a combinatorial bound on the number of events in their algorithm. As mentioned earlier, this potential function cannot be used in the presence of transaction costs, and therein lies the difficulty in extending our results to this case.

The auction algorithm for the traditional models can be made to be distributed and even asynchronous, with a small increase in the running time. We show that a similar distributed/asynchronous version of the algorithm may not converge in the presence of transaction costs. An interesting open question is if there is some other asynchronous/distributed algorithm that also converges fast. In particular, is there a tattonnement process that converges fast (like in [4])?

Outline. We provide an overview, followed by the details of our algorithm in Section 2. Section 3 contains the analysis of the running time. The proofs of lemmas in Section 2 and 3 have been omitted due to space constraints and can be found in the full version of this paper [1].

2 Algorithm

Theorem 1. *We can find* ϵ*-approximate equilibrium prices and allocations in* $O\left(\frac{1}{\epsilon}(n+\log m)mn\log(B/\epsilon)\right)$ *operations where* $B=(1+\epsilon)\sum_i B_i$.

Overview. Our algorithm maintains a set of prices and allocations and modifies them progressively. To initialize, we set all the prices $p_j=\epsilon$ and all the allocations are empty. The algorithm is organized in rounds. At the end of each round, we raise the price of one good by a multiplicative factor of $1+\epsilon$. Any allocations made before the price raise continue to be charged at the earlier, lower price. Therefore at any point in the algorithm, a good may be allocated to buyers at two different prices, p_j and $p_j/(1+\epsilon)$. During a round, we take a good away from a buyer at the lower price and allocate it to a buyer (possibly the same buyer) at the current, higher price. We find a sequence of such reallocations such that we eventually find a buyer with positive surplus and a good in her demand set such that all of that good is allocated at the current price. When we find such a buyer-good pair, we increase the price of that good and end the round. The algorithm terminates when the budgets of all the buyers are exhausted.

Following invariants are maintained throughout the algorithm:

I1: Buyers have non-negative surplus i.e. no buyer exceeds her budget.

I2: All prices are at least ϵ.

I3: Every good is either priced ϵ or is fully allocated.

I4: Any good j allocated to a buyer i must be approximately most desirable. (As in Equation (8))

I5: A good j is allocated at price either p_j or $p_j/(1+\epsilon)$ where p_j is the current price.

Invariant I3 is a tighter version of equation (7). We maintain I3 and I5 until the end of the algorithm whence we merge the two price tiers. This may lead to some goods being undersold, but we prove that equation (7) still holds. Also note that invariant I4 holds for any allocations, whether at the higher or lower price tier. Unless mentioned otherwise, the statements of all the lemmas that follow are constrained to maintain these invariants.

We now present the details of our algorithm. Each round consists of roughly two parts: 1) We construct a *demand graph* G on the set of buyers and 2) We perform multiple iterations of a reallocation procedure - which we call a *transfer walk*. At the end of each round, we increment the price of some good. The sequence of rounds ends when the surplus of all the buyers reduces to zero. At the end, we readjust the allocations to merge the two price tiers. In what follows, we explain our algorithm in three parts: a) Construction and properties of the demand graph, b) Transfer walks and c) Readjustment of allocations.

Notation. We denote the allocations of good j to buyer i at prices p_j and $p_j/(1+\epsilon)$ as h_{ij} and y_{ij} respectively. We denote by $z_j = 1 - \sum_i (h_{ij} + y_{ij})$ the amount of good j unassigned at any point in the algorithm. Given any prices and allocations, the surplus r_i of buyer i is the part of her budget unspent:

$$r_i \;=\; B_i \;-\; \sum_j (p_j + c_{ij})h_{ij} \;-\; \sum_j \left(\frac{p_j}{1+\epsilon} + c_{ij}\right) y_{ij}$$

Notice that since the prices remain constant throughout a round except at the end, the demand sets of all the buyers are well defined. In each round we fix a function $\pi(i) = \min\{\, j \,|\, j \in D_i \,\}$. Intuitively, we will attempt to allocate the good $\pi(i)$ to i in this round, ignoring all the other goods in D_i for the moment. Any choice of a good from D_i suits as $\pi(i)$, but we fix a function for ease of exposition.

Construction and properties of the demand graph. We construct a directed graph G on the set of buyers. An edge exists from buyer i to k if and only if $y_{k\pi(i)} > 0$. A node i in this graph with (1) no out-edges (*i.e.* a sink), (2) $r_i > 0$ and (3) $z_{\pi(i)} = 0$ will be defined to be 'unsatisfiable'.

Lemma 1. *For an unsatisfiable node i, the price of the good $\pi(i)$ can be increased by a multiplicative factor of $1+\epsilon$.*

But the graph G may not contain an unsatisfiable node to start with. Hence we perform a series of reallocations until we create and/or find such a node.

The reallocation involves the following step: For an edge $i \to k$ in G with $r_i > 0$, we take away the lower price allocation of good $\pi(i)$ for k and allocate it to i at the current price. In short, we perform the operations $y_{k\pi(i)} \leftarrow y_{k\pi(i)} - \delta$ and $h_{i\pi(i)} \leftarrow h_{i\pi(i)} + \delta$ for a suitably chosen value of δ. This process reduces r_i, $y_{k\pi(i)}$ and increases r_k. If $y_{k\pi(i)}$ reduces to zero, we drop the edge (i,k) from the graph. When we make such a reallocation, we say that we *transfer surplus* from i to k. Note that the surplus is not conserved. This is because the price paid by i for the same amount of the good, including the transaction costs, could even be lower than the price paid by k.

Lemma 2. *If the edge from i to k exists in G with $r_i > 0$, then we can transfer surplus from i to k such that either the surplus of i becomes zero or the edge (i, k) drops out of G.*

We can repeatedly apply Lemma 2 to transfer surplus along a path in G.

Corollary 1. *If there exists a path from i to k in G and $r_i > 0$, then we can transfer surplus from i to k such that either the surplus of all the nodes on the path except k becomes zero or an edge in the path drops out of G.*

Finally, G may contain cycles. Consider the edges (i_1, i_2) and (i_2, i_3) in G and let $j_1 = \pi(i_1)$ and $j_2 = \pi(i_2)$. If the transaction costs are all zero, then it can be argued that the last price raise for j_1 must have taken place before the last price raise for j_2. Repeating this argument, one can preclude the existence of cycles in G in absence of transaction costs. This acyclicity of G forms a pivotal argument in the algorithm of Garg and Kapoor [7]. In the full version of this paper [1], we provide a sketch of how a cycle can emerge in G when transaction costs are present. We also show that the algorithm of [7] can slow down indefinitely if G contains cycles. Therefore, we need to be able to transfer surplus around a cycle.

Lemma 3. *If there exists a cycle in G and exactly one node in the cycle has positive surplus, then we can transfer surpluses in such a way that either all the node in the cycle have zero surplus or an edge in the cycle drops out.*

In a round, we use the above lemmas to perform multiple iterations of the transfer walk.

Transfer Walk

Step 1. Find a node i_0 with a positive surplus. If there are no such nodes, then terminate the round and jump to readjustment of allocations.

Step 2. Follow a path going out of i_0 in G in a depth-first-search fashion. We look at the first edge in the adjacency list of the last visited node i on the path. Let (i, k) be this edge. If node k is yet unvisited, we follow that edge to extend the path. If k is already on the path, then we have found a cycle in G. Finally if i has no out-edges, then we have found a sink. Whichever the case, we now transfer surplus along the current path from i_0 to i as in Corollary 1. If an edge along the path drops out, we trigger event 2d. Otherwise, we trigger events 2a-2c depending upon case. The transfer walk must end in a finite number of operations in one the of following events:

Event 2a - The path reaches a sink i with $z_{\pi(i)} = 0$: Let $j = \pi(i)$. By Corollary 1, we must have transferred a positive surplus to i even if r_i was zero at the begining of the walk. Hence i is an unsatisfiable node. Raise $p_j \leftarrow (1 + \epsilon)p_j$. Terminate the walk and the round.

Event 2b - The path reaches a sink i with $z_{\pi(i)} > 0$: Let $j = \pi(i)$. By invariant I3, $p_j = \epsilon$. We let $\delta = \min(\ r_i/\epsilon,\ z_j\ )$. We then assign $h_{ij} \leftarrow h_{ij} + \delta$. If $\delta = r_i/\epsilon$ then the surplus of i goes to zero otherwise z_j goes to zero. In either case we end this transfer walk.

Event 2c - The path finds a cycle: Let i be the last node visited on the path and an edge (i, k) in G reaches a node k already visited on the path. By Corollary 1, all the nodes in the cycle except i have zero surplus. Therefore, we apply Lemma 3 until the surplus of i becomes zero or an edge in the cycle drops out. We terminate the current walk.

Event 2d - An edge drops out during path transfer: In this case we terminate the current walk.

If a transfer walk ends in event 2a, we terminate the current round and start the next one. Otherwise if events 2b-2d are triggered, we start a new transfer walk. If the surplus of all buyers is found to be zero in Step 1, we move to the last phase, which is readjustment of allocations.

Readjustment of allocations. At the end of the transfer walks, all the required invariants are satisfied, but the same good may be allocated to the same or different buyers at different prices: p_j and $p_j/(1+\epsilon)$. Therefore in this phase, we merge the two tiers of allocation for every buyer-good pair to create the final allocations. For all i, j such that $y_{ij} > 0$, we assign $x_{ij} \leftarrow h_{ij} + \frac{\frac{p_j}{1+\epsilon}+c_{ij}}{p_j+c_{ij}} y_{ij}$. The final equilibrium prices are the prices at the termination of the algorithm.

Theorem 2. *The algorithm produces ϵ-approximate equilibrium prices and allocations.*

3 Analysis

Lemma 4. *If R is the number of rounds in the algorithm, then the number of transfer walks that end in an edge dropping out of G is at most nR.*

Proof of Theorem 1

Initialization and readjustment. Both the initialization and final adjustment of allocations can be performed in mn operations.

The number of rounds. The price of exactly one good is raised by multiplicative factor of $1+\epsilon$ in each round except the last round. Starting at ϵ, the maximum value to which a price may be raised is $B = (1+\epsilon)\sum_i B_i$. Therefore, there can be at most $R = 1 + \frac{m}{\epsilon}\log(\frac{B}{\epsilon})$.

Constructing the graph. For each buyer, we maintain all the goods in a balanced tree data structure that sorts the goods first by the bang-per-buck $u_{ij}/(p_j + c_{ij})$ and then by the index j. In this manner, we can compute the function $\pi(i)$ in $O(\log m)$ time. Given $\pi(i)$, every node may have an edge to every other node. Therefore, the graph G can be constructed in $O(n^2 + n\log m)$ operations. After the price increase at the end of the round, the sorted trees can be maintained in time $O(n \log m)$ while the transfer of allocations from higher to lower price tier can be completed in $O(n)$ operations.

Number of transfer walks. All the remaining computation in the algorithm takes place within the transfer walks. Since we follow the first edge going out of each vertex, the depth-first-search requires only $O(n)$ operations. The surplus transfer along a path and a cycle can similarly be performed in $O(n)$ operations. When an edge drops out, updating G involves simply incrementing a pointer. Therefore, overall a transfer walk requires $O(n)$ operations.

We will now bound the number of transfer walks that happen throughout the algorithm, including all the rounds. We will classify them by the event that ends the walk. At most R transfer walks can terminate the round. At most m walks can end with z_j going zero. Lemma 4 bounds the number of walks that end with an edge dropping out of the graph. The only remaining case is that the walk ends when the surplus of the last visited node on the path vanishes. A transfer walk ending in this case leaves one less node in G with a positive surplus. To see this, observe that a transfer walk starts with a node on the same path with positive surplus and by the time it ends in this case, all the nodes on the path have zero surplus by Corollary 1 and Lemma 3.

Let r_+ denote the number of nodes in G with positive surplus. After initialization we have $r_+ = n$. The only event which may increase r_+ is event 2d. If an edge (i, k) drops out during surplus transfer along the path, node k may be left with some positive surplus that was absent at the start of the walk. Therefore r_+ increases by at most one in this event. Combined with Lemma 4, this implies a bound of $n + nR$ on the number of times r_+ reduces.

It is clear from the above analysis that the algorithm performs at most $O(nR)$ transfer walks. Combined with the other computation bounds, this yields an upper bound of $O\left(\frac{1}{\epsilon}(n + \log m)mn \log(B/\epsilon)\right)$ on the running time of the algorithm.

References

1. Chakraborty, S., Devanur, N.R., Karande, C.: Market equilibrium with transaction costs. CoRR, abs/1001.0393 (2010)
2. Chen, N., Ghosh, A., Vassilvitskii, S.: Optimal envy-free pricing with metric substitutability. In: ACM Conference on Electronic Commerce, pp. 60–69 (2008)
3. Codenotti, B., Rademacher, L., Varadarajan, K.R.: Computing equilibrium prices in exchange economies with tax distortions. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 584–595. Springer, Heidelberg (2006)
4. Cole, R., Fleischer, L.: Fast-converging tatonnement algorithms for one-time and ongoing market problems. In: STOC, pp. 315–324 (2008)
5. Devanur, N.R.: Fisher markets and convex programs (2008) (unpublished manuscript)
6. Garg, J.: Algorithms for Market Equilibrium. PhD thesis, Dept. of Computer Science and Engineering, IIT Bombay (2008)
7. Garg, R., Kapoor, S.: Auction algorithms for market equilibrium. In: Proceedings of 36th STOC (2004)

8. Garg, R., Kapoor, S., Vazirani, V.V.: An auction-based market equilbrium algorithm for the separable gross substitutibility case. In: Jansen, K., Khanna, S., Rolim, J.D.P., Ron, D. (eds.) RANDOM 2004 and APPROX 2004. LNCS, vol. 3122, pp. 128–138. Springer, Heidelberg (2004)
9. Garg, R., Kapoor, S.: Price roll-backs and path auctions: An approximation scheme for computing the market equilibrium. In: Spirakis, P.G., Mavronicolas, M., Kontogiannis, S.C. (eds.) WINE 2006. LNCS, vol. 4286, pp. 225–238. Springer, Heidelberg (2006)
10. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V. (eds.): Algorithmic Game Theory. ch. 2,5,6. Cambridge University Press, Cambridge (2007)
11. Orlin, J.B.: Improved algorithms for computing fisher's market clearing prices: computing fisher's market clearing prices. In: STOC, pp. 291–300 (2010)
12. Ye, Y.: A path to the arrow-debreu competitive market equilibrium. Mathematical Programming 111(1-2), 315–348 (2008)

An Axiomatic Characterization of Continuous-Outcome Market Makers*

Xi Alice Gao and Yiling Chen

School or Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138
{xagao,yiling}@eecs.harvard.edu

Abstract. Most existing market maker mechanisms for prediction markets are designed for events with a finite number of outcomes. All known attempts on designing market makers for forecasting continuous-outcome events resulted in mechanisms with undesirable properties. In this paper, we take an axiomatic approach to study whether it is possible for continuous-outcome market makers to satisfy certain desirable properties simultaneously. We define a general class of continuous-outcome market makers, which allows traders to express their information on any continuous subspace of their choice. We characterize desirable properties of these market makers using formal axioms. Our main result is an impossibility theorem showing that if a market maker offers binary-payoff contracts, either the market maker has unbounded worst case loss or the contract prices will stop being responsive, making future trades no longer profitable. In addition, we analyze a mechanism that does not belong to our framework. This mechanism has a worst case loss linear in the number of submitted orders, but encourages some undesirable strategic behavior.

Keywords: Prediction markets, continuous-outcome events, combinatorial prediction markets, expressive betting.

1 Introduction

A ubiquitous need in organizations and societies is to obtain and aggregate dispersed information of uncertain events so that informed decisions can be made. Prediction markets have been designed for this goal of information aggregation and have been shown to provide remarkably accurate forecasts in practice [1,2,3,4,5,6].

A prediction market is a betting intermediary that offers contracts whose payoffs are tied to outcomes of future events. Participants reveal their information about the event through buying and selling contracts. To facilitate information aggregation, many automated market maker mechanisms [7,8,9,10,11,12] have been designed to ensure that a participant can always conduct trades with the market maker and reveal his information whenever he finds it profitable.

Many events of interests, from carbon dioxide emission level to hurricane landing location, are naturally perceived as continuous random variables with

* This work is supported by NSF under Grant No. CCF-0953516. Gao is partially supported by a NSERC PGS-D Scholarship.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 505–514, 2010.

continuous outcome spaces. However, most existing market makers, including the popular logarithmic market scoring rule (LMSR) [7,8] and the dynamic parimutuel markets (DPM) [9,13], are designed for finite-outcome random variables, and cannot handle continuous outcome spaces directly. For forecasting continuous-outcome events, these mechanisms rely on discretizing the continuous outcome space into a finite number of subsets and treat the event as a finite-outcome random variable. This approach poses the significant challenge of determining the level of discretization to be used in advance. Choosing too coarse-grained discretization could hurt information aggregation, since market participants may not be able to easily express their information with the pre-specified subsets. If the chosen discretization is too fine-grained, certain market makers like LMSR may suffer from a large worst case loss. In general, committing to an inappropriate discretization in advance may create unnecessary psychological burden for traders. In practice, Crowdcast[1], Yoopick [14], and Gates Hillman [15] prediction markets allow traders to wager on intervals through their user-friendly interfaces, although the underlying mechanisms still use some sort of discretization. Therefore, for predicting events that are naturally perceived as continuous, it is desirable to design market mechanisms that can handle the continuous-outcome spaces directly and provide sufficient expressiveness for participants to easily reveal their information on the continuous-outcome spaces.

Gao, Chen and Pennock [16] proposed the continuous-outcome LMSR and DPM. Although these continuous-outcome mechanisms offer considerable flexibility for participants to reveal their information, they suffer from some undesirable properties. In particular, the continuous-outcome LMSR can potentially lose an infinite amount of money, whereas the finite-outcome LMSR is guaranteed to have bounded worst case loss. The continuous DPM suffers from a different problem – even if a trader bets on a subspace that contains the realized outcome, he can potentially incur a loss. The intellectual quest that motivates this paper is to understand which set of desirable properties are possible or impossible to satisfy simultaneously for continuous-outcome market makers.

In this paper, we take an axiomatic approach to analyzing market makers for continuous-outcome events. We first define cost functional based market makers for continuous-outcome events. Then, we characterize desirable properties of these market makers as formal axioms. Our main contribution is an impossibility result showing that no market maker of this class can satisfy these axioms simultaneously. Specifically, for a market maker offering binary-payoff contracts, it either has unbounded worst-case loss or the contract prices will become unresponsive to trades being conducted. We also analyze a mechanism which does not fit into our axiomatic framework. This mechanism has a worst case loss linear in the number of orders, but encourages some undesirable strategic behavior.

Related Work. There have been a significant amount of efforts on designing and analyzing market maker mechanisms for finite-outcome events, including

[1] http://www.crowdcast.com

market scoring rules [7,8], dynamic parimutuel markets [9], cost function based market makers [10,17], and sequential convex parimutuel mechanisms [12]. The focus has been on analyzing the various properties of the market makers and establishing connections among them. In the context of designing combinatorial prediction markets, research has been focusing on the computational tractability of pricing expressive bets in the finite-outcome LMSR [18,19]. The work of Gao, Chen, and Pennock [16] is the closest to this paper. It generalized LMSR and DPM to handle continuous outcome spaces and analyzed the properties of the resulting mechanisms.

2 Background

In this section, we first describe a class of cost function based market makers for finite-outcome events. We then introduce and the discuss the properties of the continuous-outcome LMSR market maker.

For finite-outcome events, Chen and Pennock [10] introduced a general class of automated market maker mechanisms, called the cost function based market makers. It has been shown that this class of market makers is equivalent to most of the known finite-outcome market makers under mild conditions [10,12,17][2].

A cost function based market maker offers N contracts, each corresponding to one of N mutually exclusive and exhaustive outcomes of an event. Each contract pays off \$1 if and only if the corresponding outcome occurs. The market maker uses a differentiable *cost function* $C(\boldsymbol{q}) : \mathbb{R}^N \rightarrow \mathbb{R}$ to capture the total amount of money wagered in the market, where the vector $\boldsymbol{q}$ represents the number of shares purchased by all traders. If a trader changes the quantity vector from $\boldsymbol{q}$ to $\boldsymbol{q}'$, he pays $C(\boldsymbol{q}') - C(\boldsymbol{q})$ to the market maker and acquires $\boldsymbol{q}' - \boldsymbol{q}$ shares of contracts. The instantaneous price of the i-th contract, defined as $p_i(\boldsymbol{q}) = \partial C(\boldsymbol{q})/\partial q_i$, represents the price per share of an infinitesimal number of shares.

Chen and Vaughan [17] formalized that a cost function is *valid* if the instantaneous prices $p_i(\boldsymbol{q})$ are non-negative and form a probability distribution over the outcome space. They proved that the sufficient and necessary conditions for a cost function C to be valid are: differentiability (to ensure that prices are well-defined), increasing monotonicity (to ensure that prices are non-negative), and a translation invariance condition $C(\boldsymbol{q} + k\mathbf{1}) = C(\boldsymbol{q}) + k$, $\forall \boldsymbol{q}, k$ (to ensure that prices sum to 1 and there is no arbitrage).

It has been shown that many valid market makers based on convex cost functions have bounded worst case loss [12,17][3], where the loss of the market maker is seen as a subsidy to promote information aggregation. For instance, the popular LMSR mechanism has bounded worst-case loss given by $b \log N$.

For continuous-outcome events, Gao, Chen and Pennock [16] generalized the finite-outcome LMSR for the *interval betting* setting. Even though the resulting continuous-outcome LMSR can handle interval bets for continuous-outcome

[2] DPM is an exception to this.

[3] This is because a valid convex cost function based market maker is equivalent to a strictly proper market scoring rule under mild conditions. Any market scoring rule with a regular proper scoring rule has bounded worst case loss.

events, it suffers from unbounded loss – the market maker could potentially lose an infinite amount of money to the traders.

3 An Axiomatic Framework

In this section, we will define a general class of automated market maker mechanisms for continuous-outcome events, the *cost functional based market makers*. These market makers generalize the cost function based market makers for finite-outcome events to handle continuous-outcome spaces. We then propose three axioms to characterize desirable properties for these market makers.

3.1 Cost Functional Based Market Makers for Continuous-Outcome Events

Consider a continuous random variable X with domain $(L, U) = \{x : x \in \mathbb{R}, L \leq x \leq U, L \in \mathbb{R} \cup \{-\infty\}, U \in \mathbb{R} \cup \{+\infty\}\}$. Let $x \in (L, U)$ represent a particular outcome and let x^* denote the realized outcome in hindsight. We define a class of cost functional based market makers for predicting the realized value of X.

Cost functional based market makers are operated based on trading shares of contracts. First, we define the *quantity function* $q(x) \in \mathcal{L}^1(L, U)$, representing the number of shares purchased for outcome $x \in (L, U)$, which is analogous to the quantity vector $\boldsymbol{q}$ in the finite-outcome case[4]. The value $q(x)$ can be thought as the total number of shares purchased for contracts that will pay off when x is the realized outcome.

A cost functional based market maker uses a differentiable *cost functional*, $C[q(x)] : \mathcal{L}^1(L, U) \rightarrow \mathbb{R}$, to capture the total amount of money wagered in the market as a functional of the current quantity function $q(x)$. If a trader changes the quantity function from $q(x)$ to $q'(x)$, he obtains $q'(x) - q(x)$ shares for each outcome x and must pay $C[q'(x)] - C[q(x)]$ to the market maker. We use $p[q(x), q'(x)]$ to denote the cost of such a transaction, i.e. $p[q(x), q'(x)] = C[q'(x)] - C[q(x)]$. The market maker starts the market with some initial quantity function $q^0(x)$ such that the value of $C[q^0(x)]$ is finite.

For any $q(x)$, the *price density functional* $p[q(x)]$ is defined as the functional derivative of the cost functional with respect to $q(x)$, that is, $p[q(x)] = \delta C[q(x)]/\delta q(x)$. The functional $p[q(x)]$ maps the quantity function to a probability density function over (L, U). It is analogous to $p_i(\boldsymbol{q})$ in the finite-outcome setting. According to the calculus of functionals, we can express the cost of a transaction in terms of an integral of the price density functional, that is

$$
\begin{aligned}
p[q(x), q'(x)] &= C[q'(x)] - C[q(x)] \qquad (1)\\
&= \int_0^1 \int_L^U p[q(x) + k(q'(x) - q(x))](q'(x) - q(x))\, dx\, dk
\end{aligned}
$$

[4] $\mathcal{L}^1(L, U)$ denotes the space of Lebesgue integrable functions on (L, U) with norm $\|q(x)\| = \int_L^U |q(x)| dx$.

If a trader changes the quantity function from $q(x)$ to $q'(x)$, then the future payoff of this transaction $o[q(x), q'(x), x^*]$ is a nonzero real number if $q'(x^*) \neq q(x^*)$ (i.e. the trader is buying or selling winning contracts), and \$0 otherwise where x^* is the realized outcome. Negative payoff encodes loss from selling the winning contracts. Other than this, we put no restriction on the value of $o[q(x), q'(x), x^*]$ and leave the definition of this value to specific mechanisms.

In our framework, a cost functional is *valid* if and only if the corresponding market maker satisfies two simple conditions:

1. For every $x \in (L, U)$, and every $q(x) \in \mathcal{L}^1(L, U)$, $p[q(x)] \geq 0$.
2. For every $q(x) \in \mathcal{L}^1(L, U)$, $\int_L^U p[q(x)]dx = 1$.

These are the minimum requirements for the price density functional to represent a valid probability distribution over the outcome space. The following theorem gives the sufficient and necessary conditions for the cost functional to be valid.

Theorem 1. *A cost functional C is valid if and only if it satisfies the following properties:*

1. *Differentiability: The functional derivative $\delta C[q(x)]/\delta q(x)$ exists for all $q(x) \in \mathcal{L}^1(L, U)$ and all $x \in (L, U)$.*
2. *Increasing Monotonicity: For any $q(x), q'(x) \in \mathcal{L}^1(L, U)$, if $q'(x) \geq q(x), \forall x \in (L, U)$, then $C[q'(x)] \geq C[q(x)]$.*
3. *Positive Translation Invariance: For any $q(x) \in \mathcal{L}^1(L, U)$ and any constant k, $C[q(x) + k] = C[q(x)] + k$.*

The above concepts define a general class of market maker mechanisms for forecasting continuous-outcome events[5]. These market makers can potentially support many different betting languages. In this paper, we focus on the simple and intuitive *interval betting* language [16]. For interval betting, traders are restricted to purchasing a constant s shares of a contract on an interval $(a, b) \subseteq (L, U)$ of their choice, where $a < b$. Such a transaction increases $q(x)$ by s for every $x \in (a, b)$. We denote the quantity function after the transaction by $q'(x) = \{q(x) + s\}_{(a,b)}$ where $q'(x)$ is defined by $q'(x) = q(x) + s, \forall x \in (a, b)$ and $q'(x) = q(x), \forall x \in (L, U)\backslash(a, b)$. For such a transaction, we define the *instantaneous contract price* $p_{(a,b)}[q(x)]$ to be the integral of the price density functional over (a, b), that is, $p_{(a,b)}[q(x)] = \int_a^b p[q(x)]dx$, This is intuitively the price per share for buying an infinitesimal share of (a, b). Note that we still do not put explicit restrictions on the transaction payoff for an interval contract (a, b), i.e. $o[q(x), q'(x), x^*]$ where $q'(x) = \{q(x) + s\}_{(a,b)}$.

In the rest of the paper, we only consider cost functional based market makers for interval betting. Below we define an *Interval Cost Continuity* condition for interval betting market makers.

[5] We note that the continuous-outcome DPM is not a valid market maker in our framework because its price density function does not correspond to a probability distribution.

Definition 2 (Interval Cost Continuity). *A cost functional* $C[q(x)]$ *satisfies the* Interval Cost Continuity *condition if for any* $x^* \in (L,U)$, $q(x) \in \mathcal{L}^1(L,U)$, $s \in \mathbb{R}$, *and* $q'(x) = \{q(x)+s\}_{(x^*-\delta,x^*+\delta)}$, $C[q'(x)]$ *is right continuous at* $\delta = 0$ *and continuous for all* $\delta > 0$.

The *Interval Cost Continuity* property specifies that, for each interval bet, the cost functional value for the final quantity function must be continuous for any change δ in the size of the betting interval. As δ approaches 0 (i.e. the size of the interval approaches 0), $C[q'(x)]$ approaches to $C[q(x)]$.

3.2 Desirable Properties

We propose three formal axioms to characterize some desirable properties of the cost functional based market makers for continuous-outcome events. We only consider interval bets.

Axiom 1 (Responsive Price). *If* $q(x)$ *and* $q'(x)$ *satisfy three conditions: (1)* $q'(x) = q(x), \forall x \in (L,U)\backslash(a,b)$, *(2)* $q'(x) \geq q(x), \forall x \in (a,b)$, *and (3)* $\exists (c,d) \subseteq (a,b)$ *s.t.* $q'(x) > q(x), \forall x \in (c,d)$, *then*

$$p_{(a,b)}[q'(x)] > p_{(a,b)}[q(x)]$$

for any contract $(a,b) \subset (L,U)$.

The *Responsive Price* axiom specifies that the instantaneous contract price is strictly monotonically increasing as the quantity over one of its subintervals strictly increases. This axiom is desirable since it guarantees that the change in the instantaneous contract prices will always *respond* to trades conducted and traders are always able to conduct trades irrespective of the current prices.

Axiom 2 (Domain Consistency). *The payoff and cost of purchasing shares of* (L,U) *are always equal, that is, for all* $q(x)$, $q'(x) = \{q(x)+s\}_{(L,U)}$, *and* $x^* \in (L,U)$, *we have* $o[q(x), q'(x), x^*] = p[q(x), q'(x)]$.

Intuitively, any bet on the entire domain (L,U) should earn zero profit as the bet is not revealing any useful information about X. This axiom is required for a cost functional based market maker to be arbitrage free. For instance, the continuous-outcome LMSR satisfies this axiom. Moreover, we will show in the next section that this axiom is a sufficient and necessary condition for the contracts offered to be exclusively binary-payoff contracts.

Axiom 3 (Bounded Loss). *There exists* $B \in \mathbb{R}$, *such that, for any sequence of* n *transactions where the quantity functions satisfy* $q_i(x) = \{q_{i-1}(x)+s_i\}_{(a_i,b_i)}$ *and* $(a_i,b_i) \subseteq (L,U)$, *we have*

$$\max_{x^* \in (L,U)} \left(\sum_{i=1}^{n} (o[q_i(x), q_{i+1}(x), x^*] - p[q_i(x), q_{i+1}(x)]) \right) \leq B.$$

This axiom gives a sufficient and necessary condition for the market maker to have bounded worst case loss. The market maker's loss is the difference between the total money he pays out and the total money he collects. The worst outcome for the market maker is when this difference is maximized.

4 Impossibility Result

In this section, we present our main impossibility theorem. We first prove conditions for a valid market maker mechanism to offer exclusively binary-payoff contracts. For these market makers, we prove in Theorem 5 that the *Responsive Price* and *Bounded Loss* axioms cannot be satisfied simultaneously.

Lemma 3 (Binary Contract Lemma). *A valid market maker mechanism satisfies the* Domain Consistency *axiom if and only if it offers binary-payoff interval contracts, that is, the future payoff of any contract is fixed to be* \$1 *per share if the realized outcome* x^* *falls within the interval and* \$0 *otherwise.*

Lemma 3 shows that if a valid market maker satisfies the *Domain Consistency* axiom, the payoff of the contract has to be binary regardless of the interval chosen. We also note that with binary-payoff contracts, the *Responsive Price* axiom implies that the price of any contract never reaches 0 or 1. Next, we present Lemma 4 to facilitate the proof of our main impossibility result.

Lemma 4 (Responsive Price Lemma). *For a valid market maker satisfying the* Interval Cost Continuity *condition, if it satisfies the* Responsive Price *axiom, then for any winning contract* (a, b)*, any number of shares* $s \in \mathbb{Z}^+$*, any quantity function* $q(x)$*, and any* $\epsilon > 0$*, there exists a winning contract* $(a', b') \subset (a, b)$*, such that*

$$C[q''(x)] - C[q(x)] \leq \epsilon(C[q'(x)] - C[q(x)])$$

where $q'(x) = \{q(x) + s\}_{(a,b)}$ *and* $q''(x) = \{q(x) + s\}_{(a',b')}$.

Based on Lemma 4, if an interval (a, b) is a winning contract, there exists a subinterval of (a, b) which is also winning such that the cost of buying a constant number of shares over the subinterval is an arbitrarily small fraction of the cost of buying the same number of shares over (a, b).

Theorem 5 (Impossibility Result). *For a valid market maker satisfying the* Interval Cost Continuity *condition, if it allows traders to bet on intervals of any nonzero size and satisfies the* Domain Consistency *axiom, then it cannot satisfy the* Responsive Price *and* Bounded Loss *axioms simultaneously.*

Proof Sketch. By Lemma 3, the contracts offered must pay off \$1 per share if they are winning, and \$0 otherwise. Consider a trader who knows x^* and has a fixed budget of $\$m$. Using the following procedure, this trader could potentially get an arbitrarily large profit.

To get $\$s$ payoff, the trader can start by calculating the cost of buying s shares of an arbitrary winning contract (a, b), denoted by T. If $T \geq m$, then by Lemma 4, the trader can choose $\epsilon = \frac{m}{T}$ and find a winning contract $(a', b') \subset (a, b)$ such that the cost of buying s shares over (a', b') is no more than $\frac{m}{T} \cdot T = m$ dollars and the corresponding profit is at least $s - m$ dollars. Because s is arbitrary, the trader's profit, hence the market maker's loss, is not bounded. □

Even though Theorem 5 allows traders to bet on intervals of any nonzero size, we now show that even if we restrict the size of the smallest betting interval to be at least $z > 0$, with certain assumptions, the trader could still bet on arbitrarily small intervals of their choice.

Corollary 6. *For a valid market maker satisfying the* Interval Cost Continuity *condition and restricting the size of the smallest betting interval to be* $z \in \mathbb{R}$ *where* $0 < z < (U - L)/2$[6]*, if it satisfies the* Domain Consistency *axiom, then it cannot simultaneously satisfy the* Responsive Price *and* Bounded Loss *axioms.*

The key insight for proving the above corollary is that a trader can perform a sequence of transactions which is equivalent to purchasing shares of an arbitrarily small interval even with the restriction on the size of the smallest betting interval. It is worth noting that these transactions can be potentially completed in any order, and multiple traders can collude to complete them. Thus, it would be very challenging in general to detect such trading patterns in practice.

The unbounded worst case loss of the continuous-outcome LMSR is a special case of our impossibility result. However, compared with finite-outcome market makers, this impossibility result is rather surprising since the finite-outcome LMSR essentially satisfies the finite-outcome versions of all three axioms.

We could possibly relax the *Responsive Price* axiom to derive mechanisms with bounded worst case loss, although the resulting market maker may be trivial and less interesting. For example, a market maker can quickly increase the price of contracts to 1 once the quantity for the contracts increases beyond a certain value. Beyond this point, purchasing more shares will not earn the trader any more profit and bounded worst case loss can be achieved.

5 Discussion and Conclusion

While the class of market makers we considered is quite general, there exist other continuous-outcome mechanisms outside of this class that can achieve bounded worst case loss and responsive price simultaneously. In particular, we can operate the finite-outcome LMSR over the continuous-outcome space by discretizing the outcome space on the fly given the submitted orders. By violating our definition of *instantaneous contract price* and the *Interval Cost Continuity* condition, this mechanism achieves the worst case loss linear in the number of orders submitted, but also encourages certain undesirable strategic behaviors.

To operate the finite-outcome LMSR over a continuous-outcome space, we split the existing intervals in the state space for every submitted order on (c, d) whenever c or d falls within the existing intervals. Then the order is traded via LMSR with the state space after splitting. For every splitting of this kind, the outstanding quantities and instantaneous prices need to satisfy the following consistency constraints for the mechanism to remain arbitrage free.

- The sum of the instantaneous prices of the subintervals must be equal to the instantaneous price of the original interval.

[6] This assumption is reasonable since the size of the smallest betting interval should be much smaller than the size of the domain of the random variable.

- The number of shares of each subinterval held by all traders must be equal to the number of shares of the original interval held by all traders.

The first consistency constraint allows considerable freedom in splitting the probability estimates among the subintervals. If the probability estimates are split equally among the subintervals, then the resulting mechanism violates our definition of *instantaneous contract price* and the *Interval Cost Continuity* condition. However, this market maker has worst case loss given by $M \log 3$, where M is the number of orders submitted. However, this mechanism does not provide the incentive for traders to reveal their information truthfully. Given several subintervals with equal prices, a trader could maximize his probability of winning by betting on the largest interval regardless of his subject probability estimates for these intervals.

If the market maker splits the probability estimates in proportion to the lengths of the subintervals, it satisfies all the axioms proposed and the worst case loss becomes unbounded according to Theorem 5. Intuitively, the unbounded loss is due to the market maker assigning arbitrarily small initial probability to the smallest interval containing the realized outcome. If the traders drive the price of this interval to be $1, then the market maker is destined to lose an infinite amount of money to the traders.

In conclusion, we take an axiomatic approach to study automated market maker mechanisms for forecasting continuous-outcome events. In our axiomatic framework, we consider a general class of cost functional based market makers and define formal axioms to characterize desirable properties of these mechanisms. We then prove that it is impossible for a valid cost functional based market maker mechanism to satisfy a certain set of properties simultaneously. Our results suggest that future efforts on designing continuous-outcome market makers should focus on finding the right tradeoffs among the desirable properties. In particular, it may be fruitful to investigate whether a market maker can be designed such that the *Interval Cost Continuity* condition or the *Responsive Price* axiom is relaxed to a reasonable degree so that other desirable properties can be achieved for practical applications.

References

1. Berg, J.E., Forsythe, R., Nelson, F.D., Rietz, T.A.: Results from a dozen years of election futures markets research. In: Plott, C.A., Smith, V. (eds.) Handbook of Experimental Economic Results (2001)
2. Wolfers, J., Zitzewitz, E.: Prediction markets. Journal of Economic Perspective 18(2), 107–126 (2004)
3. Forsythe, R., Nelson, F., Neumann, G.R., Wright, J.: Anatomy of an experimental political stock market. American Economic Review 82(5), 1142–1161 (1992)
4. Forsythe, R., Rietz, T.A., Ross, T.W.: Wishes, expectations and actions: a survey on price formation in election stock markets. Journal of Economic Behavior & Organization 39(1), 83–110 (1999)
5. Debnath, S., Pennock, D.M., Giles, C.L., Lawrence, S.: Information incorporation in online in-game sports betting markets. In: EC 2003: Proceedings of the 4th ACM Conference on Electronic Commerce, pp. 258–259. ACM, New York (2003)

6. Chen, K.Y., Plott, C.R.: Information aggregation mechanisms: Concept, design and implementation for a sales forecasting problem. Working Paper No.1131, California Institute of Technology (2002)
7. Hanson, R.D.: Combinatorial information market design. Information Systems Frontiers 5(1), 107–119 (2003)
8. Hanson, R.D.: Logarithmic market scoring rules for modular combinatorial information aggregation. Journal of Prediction Markets 1(1), 1–15 (2007)
9. Pennock, D.M.: A dynamic pari-mutuel market for hedging, wagering, and information aggregation. In: EC 2004: Proceedings of the 5th ACM Conference on Electronic Commerce, pp. 170–179. ACM, New York (2004)
10. Chen, Y., Pennock, D.M.: A utility framework for bounded-loss market makers. In: UAI 2007: Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence, pp. 49–56 (2007)
11. Peters, M., So, A.M.C., Ye, Y.: Pari-mutuel markets: Mechanisms and performance. In: Deng, X., Graham, F.C. (eds.) WINE 2007. LNCS, vol. 4858, pp. 82–95. Springer, Heidelberg (2007)
12. Agrawal, S., Delage, E., Peters, M., Wang, Z., Ye, Y.: A unified framework for dynamic pari-mutuel information market design. In: EC 2009: Proceedings of the Tenth ACM Conference on Electronic Commerce, pp. 255–264. ACM, New York (2009)
13. Chen, Y., Pennock, D.M., Kasturi, T.: An empirical study of dynamic pari-mutuel markets: Evidence from the Tech Buzz Game. In: The Workshop on Web Mining and Web Usage Analysis, WebKDD. LNCS, Springer, Heidelberg (2008)
14. Goel, S., Pennock, D., Reeves, D.M., Yu, C.: Yoopick: A combinatorial sports prediction market. In: AAAI, pp. 1880–1881 (2008)
15. Othman, A., Sandholm, T.: Automated market-making in the large: the gates hillman prediction market. In: EC 2010: Proceedings of the 11th ACM Conference on Electronic Commerce, pp. 367–376. ACM, New York (2010)
16. Gao, X., Chen, Y., Pennock, D.M.: Betting on the real line. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 553–560. Springer, Heidelberg (2009)
17. Chen, Y., Vaughan, J.W.: A new understanding of prediction markets via no-regret learning. In: EC 2010: Proceedings of the 11th ACM Conference on Electronic Commerce, pp. 189–198. ACM, New York (2010)
18. Chen, Y., Fortnow, L., Lambert, N., Pennock, D.M., Wortman, J.: Complexity of combinatorial market makers. In: EC 2008: Proceedings of the 9th ACM Conference on Electronic Commerce, pp. 190–199. ACM, New York (2008)
19. Chen, Y., Goel, S., Pennock, D.M.: Pricing combinatorial markets for tournaments. In: STOC 2008: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, pp. 305–314. ACM, New York (2008)

Online Labor Markets

John J. Horton

Harvard University, Cambridge, MA 02138, USA

Abstract. In recent years, a number of online labor markets have emerged that allow workers from around the world to sell their labor to an equally global pool of buyers. The creators of these markets play the role of labor market intermediary by providing institutional support and remedying informational asymmetries. In this paper, I explore market creators' choices of price structure, price level and investment in platforms. I also discuss competition among markets and the business strategies employed by market creators. The paper concludes with a discussion of the productivity and welfare effects of online labor.

1 Introduction

In the late 1990s, a number researchers began studying the effects that the Internet was having—or might yet have—on the labor market. One question examined was whether we might see the emergence of entirely online labor markets, where geographically dispersed workers and employers could make contracts for work sent "down a wire." Such markets would be an unprecedented development, as labor markets have always been geographically segmented.

Researchers were of mixed opinions: Malone predicted the emergence of such an "E-lance" market [9], while Autor was skeptical, arguing that informational asymmetries would make such markets unlikely [3]. Instead, Autor predicted the emergence of third-party intermediaries that could use their own reputation to convey "high bandwidth" information about workers—such as ability, skills, reliability and work ethic—to buyers who would be unwilling to hire workers based solely on demographic characteristics and self-reports.

In the approximately 10 years since, we have witnessed the emergence of a number of truly global online labor markets, as Malone predicted. By 2009, over 2 million worker accounts had been created across different markets, with over $700 million in gross wages paid to workers [7]. However, consistent with Autor's position, these markets have emerged not "in the wild," but within the context of highly structured platforms created by for-profit intermediaries.

The ultimate success and trajectory of these markets remains to be seen. If they become more important, they will raise policy questions, particularly about labor laws and taxation. They might spur the already large shift towards part-time employment [10] and have implications for inequality and development.

The purpose of this paper is to describe the key economic features of online labor markets. In addition to a positive examination, this paper highlights features of the markets likely to be relevant for welfare and productivity. Special

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 515–522, 2010.

attention is given to the ability of these markets to give workers in developing countries access to buyers in rich countries.

2 Overview

Online labor markets (OLMs) fall into two broad categories: "spot" and "contest." No labor market is truly "spot" in the sense of a commodity market, but certain OLMs feature buyer/seller agreements to trade at agreed prices for certain durations of time. Examples of spot markets include oDesk, Elance, iFreelance and Guru. Workers create online profiles and buyers post jobs and wait for workers to apply and/or actively solicit applicants.

In contest markets, buyers propose contests for informational goods such as logos (e.g., 99Designs and CrowdSPRING), solutions to engineering problems (e.g., InnoCentive) and legal research (e.g., Article One Partners). Workers create their own versions of the good and the buyer selects a winner from a pool of competitors. In some markets, the buyer must agree to select (and pay) a winner before they can post a contest; in other high-stakes markets where a solution may be unlikely, the buyer is under no obligation to select a winner.

2.1 Definition

Not all people working online do so through markets: some work is unpaid (e.g., open-source software and Wikipedia) and other work products are transferred within a firm, such as through conventional off-shoring. Even within clearly identifiable markets, there is great diversity. I propose a definition of OLMs that captures the essential common features of all markets and yet distinguishes the markets from other examples of online work: a market where (1) labor is exchanged for money, (2) the product of that labor is delivered "over a wire" and (3) the allocation of labor and money is determined by a collection of buyers and sellers operating within a price system.

2.2 Nature of Labor Markets and Role for Intermediation

Labor markets are fundamentally different from other kinds of markets in at least two ways. First, there is no single "commodity" of labor with an immediately observable quality and single prevailing price—both jobs and workers are idiosyncratic. This makes it difficult for firms and workers to find a good match, and even when matches are formed, it is difficult for either party to know precisely what they are getting when they enter into contracts. Buyer/seller information asymmetries, when combined with opportunities for strategic behavior, can impede markets; if sufficiently severe, they can prevent markets from existing [2][13]. Second, labor is a service that is delivered over time, often accompanied by relationship-specific investments in human capital (e.g., learning a particular skill for a particular job), which creates a number of the incentive issues that make it hard for parties to fully cooperate [14].

In traditional labor markets, third-party intermediaries such as temp agencies, unions and testing services profit from supplying information [4]. The creators of online labor markets do the same thing, though their scope is wider and more comprehensive. They also provide infrastructure like payment and record-keeping systems, communications infrastructures and search technology—functions typically provided by a government or by parties themselves.

2.3 What the Market Creators Provide

In order to increase the information on the demand side, OLMs often offer worker skills tests, manage reputation systems and provide worker data from prior within-OLM employment, such as hours worked and wages. Making buyer feedback public not only prevents adverse selection, but also serves to reduce moral hazard, as workers make decisions about effort "in the shadow" of the evaluations that they will likely receive. To increase supply-side information, OLM creators verify buyers' abilities to pay and reports on their past behavior in the market. For example, oDesk guarantees that workers will be paid for hourly work, putting the impetus on the buyer to interrupt an unprofitable relationship.

The influence of the market creator is so pervasive that their role in the market is closer to that of a government: they determine the space of permissible actions within market, such as what contractual forms are allowed and who is allocated decision rights.[1] Presumably they design their "institutions" to maximize expected profits. For example, they design rules to reduce the probability of disputes (subject to the constraint imposed by reducing flexibility). If disputes do arise, the market creators are likely to be able to settle them quickly using clear rules or unambiguous assignments of decision rights, such as making buyers the arbiters of contract compliance.[2]

3 Price and Price Structure

Market creators have at least three ways to earn revenue: they can charge membership fees, levy ad valorem charges on payments and charge buyers and sellers for using the market (e.g., for listing a job, taking a skills test or applying for a job).[3] These different structures are not mutually exclusive and many market creators use a hybrid structure.

A market creator has to attract both buyers and sellers to a market and facilitate valuable interactions. There is a growing literature on "two-sided" markets [12] that tries to understand price structure in the presence of membership externalities. This research focuses on scenarios where the identity of the party that

[1] Their software even serves a weights-and-measures function traditionally performed by governments by keeping universal time for logging worker hours.

[2] Although there are obvious drawbacks to such an assignment of rights, it radically reduces the space for disputes. This is in fact the precise rule used by Amazon Mechanical Turk.

[3] Typical usage fees appear to be modest and may serve as a kind of Pigouvian "tax," since some of the activities seem to be over-supplied in the markets. The costs of applying for jobs are so low that there is a good deal of application "spam."

pays the fees (or receives subsidies) matters. In online labor markets, buyers and sellers independently arrive at prices after negotiation, strongly suggesting that the Coase theorem applies, which permits a conventional one-sided analysis.[4]

Suppose that potential buyer/seller pairs would get value v from completing a project and would pay a cost c, not including any fees, if the work were intermediated. The outside option is 0. The market creator's marginal intermediation costs are assumed to be zero. If an ad valorem charge γ is leveled, the project goes forward if $v - (1+\gamma)c > 0$, whereas if a lump sump fee τ is leveled, $v - c - \tau > 0$. With the lump sum fee, the buyer/seller pair makes use of the market so long as the fee is less than the surplus: $\tau < v - c$. With the ad valorem charge, the pair makes use of the market so long as $\gamma < 1 - \frac{c}{v}$. In the lump sum case, absolute surplus matters, whereas in the ad valorem case, project efficiency matters.

Depending on the distributions of c and v, either price structure might be optimal or some hybrid might be best, but the ad valorem charge appears to have several practical advantages. First, it short-circuits the chicken-and-egg dynamics of any platform with a two-sided nature [5]. No OLM sprang forth fully formed with contingents of buyers and sellers. To be useful, the markets needed members; to attract members, they needed to be useful. An ad valorem charge avoids this problem. Second, setting an optimal lump sum charge requires knowledge of project surplus, and surplus could change dramatically as different kinds of work become more or less popular, or as firms shift more work onto the market. A firm can change membership fees, but this introduces menu costs. Finally, groups of buyers can bundle their projects under a single account and amortize their membership costs over many transactions, but this strategy offers no benefits when using usage fees. While membership fees can be important and are used in some markets, the rest of this analysis focuses on the ad valorem price structure.

3.1 Setting the Optimal ad Valorem Price Level

Perhaps because of the advantages enumerated above, ad valorem charges seem to be nearly universally applied, even in the contest markets. Assume that the buyers are purchasing efficiency units of labor from homogeneous workers and that there is a single market clearing price. The market clearing price is p and the quantity of units bought and sold is Q. Figure 1 depicts the problem in terms of intersecting supply and demand curves determining the market clearing price and quantity for a given γ. The market creator's revenue is indicated by the box with height $p\gamma$ (the side runs from p to $p(1+\gamma)$) and width Q_0. As γ grows larger, the quantity is lowered, but the height of the rectangle increases. The nested revenue box shows that as supply and demand become more elastic (S' and D'), the same size γ, even if it leads to the same market clearing price, would lead to a decrease in the quantity (and hence revenue), which is now at Q_0'.

[4] For example, if buyers have to pay a membership fee, there will be fewer buyers. This lowers the demand and therefore the price of labor, transferring some of the cost of the membership fee to sellers.

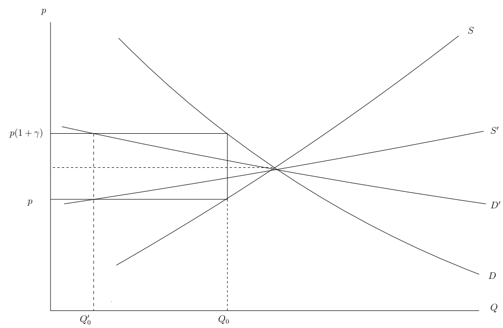


Fig. 1. Market creator profits and market clearing price

The market creator's profit maximization problem is:

$$\max_{\gamma} p(\gamma)Q(\gamma)\gamma \qquad (1)$$

where γ is the ad valorem charge and $p(\gamma)$ and $Q(\gamma)$ are the resultant prices and quantities in the market. The profit-maximizing first-order condition is $(p'Q + Q'p) + pQ = 0$, which implies that profits are a maximum when $\epsilon^p_\gamma + \epsilon^Q_\gamma = -1$. The market creator increases the ad valorem charge until a small change in γ, say $x\%$ is offset by a combined $x\%$ decrease in some percentage combination of market price and quantity. Of course, the quantities ϵ^Q_γ and ϵ^p_γ are not known and are not parameters that usually receive attention in economics. However, if we make some assumptions about the functional form of the supply and demand curves, we can solve for γ^* as a function of the relevant elasticities. Assume that both the supply and demand curves have constant elasticity of substitution, $s(p) = Q_s p^\alpha$ and $d(p) = Q_d p^\beta$. In the absence of a market creator, assuming the market could function, the efficient price for labor would be $p_e = e^{\frac{\log Q_d - \log Q_s}{\alpha - \beta}}$. Under intermediation, for the market to clear, $Q_s p_I^\alpha = Q_d(p_I(1+\gamma))^\beta$. We can solve for the intermediation market clearing price, p_I, and write it in terms of the efficient market price: $p_I = p_e(\gamma+1)^{\frac{\beta}{\alpha-\beta}}$. Because $\gamma > 0$, $\beta < 0$ (downward sloping demand curve) and $\alpha > 0$ (upward sloping supply curve), in order for the market to still clear with the ad valorem charge, the price received by sellers must be lower than in the efficient market case.[5] The market creator's profits are:

$$\pi = \left[\underbrace{(Q_s p_e^\alpha) p_e}_{\text{efficient wage bill}} \right] \times \left[\gamma \left(\gamma + 1\right)^{\frac{\beta(\alpha+1)}{\alpha-\beta}} \right] \qquad (2)$$

[5] Note, however that this "distortion" from the efficient market price is not necessarily an inefficiency, as it is the actions of the market creator that make the market possible.

Solving for the optimal charge, we have:

$$\gamma^* = \frac{\beta - \alpha}{\alpha(\beta + 1)} \tag{3}$$

If we assume that the supply and demand elasticities have the same magnitude, i.e., $\alpha = |\beta|$, then in order to give the highest observed ad valorem charge of 25% employed by BitWine[6], $\alpha = |\beta| = 9$; in order to give the more standard $\approx$ 10% used by oDesk, Amazon Mechanical Turk and others, $\alpha = |\beta| = 21$. These are remarkably high elasticities. It is not clear whether constant elasticity of substitution is a reasonable assumption, but if it is, and assuming that the market creators know their business and are not radically undercharging, it seems likely that implied elasticities are large for the simple reason that workers and buyers have ready and close substitutes for their intermediated transactions: they can make use of other online labor markets or traditional labor markets, or they can take their chances and disintermediate.

4 Competition and Specialization

It is well beyond the scope of this paper to try to model the market of intermediation markets, never mind make predictions about the likely market structure, product types, prices, etc. However, it is possible to discuss some of the key economic factors and sketch out areas for future research. The factors likely to affect ultimate market structure include whether there are economies or diseconomies of scale in providing intermediation services, barriers to entry and the potential for product differentiation.

5 Market Creator Strategy

Even after picking a price structure and level, the market creator can still increase revenues by increasing the size of the wage bill. This can be done by increasing the extent of the market, such as by recruiting more buyers and sellers, increasing worker productivity or preventing buyers and sellers from working outside the market.

5.1 Recruitment that Affects Supply and Demand

Let the market creator's initial revenues be $r_0 = p_0 Q_0$. The market creator is considering changing supply and demand in the market via recruitment of more buyers and sellers, such as through advertising. After the change, the new revenue will be $r_1 = p_1 Q_1$. Define Δx as a percentage change in x. We can write $r_1 = p_0 Q_0 (1 + \Delta p)(1 + \Delta Q)$. It is worth making the change from r_0 to r_1 if

$$\Delta Q + \Delta p + \Delta p \Delta Q > 0 \tag{4}$$

[6] BitWine is a network of freelance advisers who charge clients per-minute rates for consultations. Advisers are self-styled experts in fields such as nutrition, travel, coaching, technology and psychic prediction.

We can see that increasing within-market demand unambiguously raises profits because as ΔQ increases, so does Δp, as positive demand shocks raise both price and quantity. For supply increases, price and quantity will move in opposite directions. For small changes in supply or demand, two elasticity formulas must hold: $\Delta Q = \Delta S + \epsilon^S \Delta p$ and $\Delta Q = \Delta D + \epsilon^D \Delta p$. Because we are considering only a supply increase, $\Delta Q = \epsilon^D \Delta p$, which allows us to re-write the profit-maximizing condition as $\epsilon^D \Delta p + \Delta p + \Delta p \Delta Q > 0$. Dividing through by Δp (which is negative) and reversing the sign, we have: $\epsilon^D + \Delta Q < -1$, and since $(\epsilon^D - \epsilon^S)\Delta p = \Delta S$, the market creator finds it revenue-maximizing to increase supply so long as:

$$\epsilon_D \left(1 + \frac{\Delta S}{\epsilon^D - \epsilon^S}\right) < -1 \tag{5}$$

If supply and demand are highly elastic, $|\epsilon^D - \epsilon^S|$ is large, meaning that small positive changes in supply are likely to increase revenue.

6 Productivity and Welfare Implications

Online work offers the cost-saving benefits of telecommuting, such as reduced congestion and increased flexibility, as well as some advantages unique to the way such markets appear to be structured. First, global labor markets permit greater specialization in human capital. Second, the rapid mixture of workers across and between firms might speed up innovation spillovers, creating a kind of pseudo geographic co-location, which has been shown to increase productivity in other contexts [8]. Third, OLMs allow firms to buy small amounts of labor as needed, lowering the barriers to entrepreneurship.

OLMs also permit a kind of virtual migration that offers many of the benefits of physical migration. Assuming that increased virtual labor mobility will generate effects similar to those of increased real labor mobility, the potential gains to welfare are enormous [6]. Further, these markets create incentives for people otherwise disconnected from the global labor market to invest in their human capital [11].

Given the central role that a country's institutions play in its economic development [1], it is remarkable how little these markets demand from the institutions of the worker's home country. Prospective workers need only to be able to get online and have some way of receiving remittances. Workers do not need functioning courts, developed finance sectors, work visas, information about commodity prices, local reputations or race, class or social backgrounds required for employment in local labor markets.

Acknowledgments

Thanks to the NSF-IGERT Multidisciplinary Program in Inequality & Social Policy. Thanks to Robin Yerkes Horton, Richard Zeckhauser, Olga Rostopshova and Dana Chandler for helpful comments and suggestions.

References

1. Acemoglu, D., Johnson, S., Robinson, J.A.: Institutions as the fundamental cause of long-run growth. In: Handbook of Economic Growth,
2. Akerlof, G.: The market for "lemons": Quality uncertainty and the market mechanism. In: The Quarterly Journal of Economics, pp. 488–500 (1970)
3. Autor, D.: Wiring the labor market. Journal of Economics Perspectives 15 (2001)
4. Autor, D.H.: The economics of labor market intermediation: An analytic framework. Working Paper 14348, National Bureau of Economic Research (September 2008)
5. Caillaud, B., Jullien, B.: Chicken & egg: Competition among intermediation service providers. RAND Journal of Economics, 309–328 (2003)
6. Clemens, M., Montenegro, C., Pritchett, L., Paraguay, D., Floor, T.: The place premium: Wage differences for identical workers across the US border. World Bank Policy Research Working Paper No. 4671 (2008)
7. Frei, B.: Paid crowdsourcing: Current state & progress toward mainstream business use. Produced by Smartsheet.com (2009)
8. Greenstone, M., Hornbeck, R., Moretti, E.: Identifying agglomeration spillovers: Evidence from million dollar plants. Journal of Political Economy 118
9. Malone, T.W., Laubacher, R.J.: The dawn of the e-lance economy. Harvard Business Review 76(5), 144–152 (1998)
10. OECD. OECD Employment Outlook 2010 (2010)
11. Oster, E., Millett, B.: Do call centers promote school enrollment? Evidence from India. NBER Working Paper (2010)
12. Rochet, J., Tirole, J.: Two-sided markets: An overview. In: Institut dEconomie Industrielle Working Paper (2004)
13. Rothschild, M., Stiglitz, J.: Equilibrium in competitive insurance markets: An essay on the economics of imperfect information. The Quarterly Journal of Economics 90(4), 629–649 (1976)
14. Williamson, O.E.: Transaction-cost economics: The governance of contractual relations. Journal of Law and Economics 22(2), 233–261 (1979)

Discrete Strategies in Keyword Auctions and Their Inefficiency for Locally Aware Bidders*

Evangelos Markakis[1] and Orestis Telelis[2,**]

[1] Department of Informatics, Athens University of Economics and Business, Greece
markakis@gmail.com
[2] Department of Computer Science, University of Liverpool, UK
telelis@gmail.com

Abstract. We study formally discrete bidding strategies for the game induced by the Generalized Second Price keyword auction mechanism. Such strategies have seen experimental evaluation in the recent literature as parts of iterative best response procedures, which have been shown not to converge. We give a detailed definition of iterative best response under these strategies and, under appropriate discretization of the players' strategy spaces we find that the discretized configurations space *contains* socially optimal pure Nash equilibria. We cast the strategies under a new light, by studying their performance for bidders that act based on local information; we prove bounds for the worst-case ratio of the social cost of locally stable configurations, relative to the socially optimum cost.

1 Introduction

We study discrete bidding strategies for repeated keyword auction games, induced by the Generalized Second Price (GSP) mechanism. Sponsored search auctions have received considerable attention in the recent literature, as the premiere source of income for search engines that allocate advertisement slots. The GSP mechanism is implemented in different forms by `Google`, `Yahoo!`, and `Bing`. Other online enterprises also use flavors of GSP; e.g. `Google` exports its slot allocation and pricing system as a service. In the simplest form of the mechanism, advertisers are ranked in order of non-increasing bids and each of the first k is matched to one of k available slots, paying the next highest bid to his. In the current version bids are weighted by relevance parameters of advertisers. For one slot the GSP mechanism coincides with the VCG mechanism. For at least two slots however, the GSP auction *does not* retain the features of VCG, e.g., truthful reporting of valuations, and encourages strategic behavior.

Strategic behavior in GSP auctions raises the question of how should an advertiser decide on his bidding. A best response of a player i under a current

* This work was partly funded by the EU FP7 Network of Excellence Euro-NF, through the specific joint research project AMESA. A full version with all omitted proofs is available at the authors' homepages.

** This work was carried out while the author was at the Center for Mathematics and Computer Science (CWI) Amsterdam, The Netherlands.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 523–530, 2010.

bidding configuration is any bid value within the interval defined by the bids of at most two other players, that grants i his desired slot; but how should the exact value be decided? In practice bidders may hire consultants to design bidding strategies for them. Phenomena of competition have been observed in the adopted strategies, ranging from modest budget investment to aggressive bidding, inducing large prices for competitors. These issues have received attention in the recent literature [3,11]. Most of the existing works concern *iterative best response* procedures, viewing a GSP auction as a repeated game. Cary *et al.* [3] studied strategies where players adjust their bid iteratively, synchronously or asynchronously – in a randomly chosen order – always targeting the slot that maximizes their profits. They introduced 3 bidding strategies and proved convergence for one of them to a single fixed point, the equilibrium described in [4].

We focus on the other two simple strategies introduced in [3], that have seen less theoretical treatment, but have been used in experimental comparisons [3,9,11]. The first is *Altruistic Bidding* (AB), where every player takes a slot by minimally outbidding the player who currently owns it. The second is *Competitor Busting* (CB), where a player minimally underbids the player who owns the slot above the one aimed for. Both require discretization of the players' strategy spaces by a *bidding unit* ϵ. This may change the original game entirely. Iterative AB and CB procedures have been observed not to converge for fixed ϵ [3,9]; can we expect the best response state space to even have pure Nash equilibria (PNE)? How should ϵ be tuned so that the game in discrete strategies retains properties of the original game? The relevance of AB and CB is amplified for bidders that, due to lack of complete information, perform local best responses.

Contribution. We study iterative AB and CB best response procedures that differ from previous work [3] in that bidders only update their bid when they have incentive to target a *different* slot. We provide a detailed description of AB and redefine CB differently than it has appeared previously, to ensure its consistency with developments to follow (Section 3). We decide an upper bound on the discretization parameter ϵ to ensure that the induced discretized configurations space has a socially optimum locally envy-free PNE, analogous to the one identified in [4], that is also a PNE for the game in continuous strategies. We ensure that if iterative AB or CB converge to a socially optimum configuration, then this is a PNE even in continuous strategies. Subsequently we examine the case of bidders that take only local steps upwards or downwards due to incompleteness of available information (Section 4). We study the social inefficiency of locally stable configurations of the GSP auction and produce upper bounds on the inefficiency of configurations reached by *local* iterative AB/CB (L-AB/L-CB).

Related Work. A considerable amount of work in sponsored search auctions concerns the strategic behavior of the bidders. As mentioned above, Cary *et al.* [3] defined and studied three bidding strategies, *Altruistic Bidding* (AB), *Competitor Busting* (CB) and *Balanced Bidding* (BB). CB has been observed often in practice [2,13]. Using CB advertisers try to exhaust the budget of their competitors by placing the highest possible bid that will guarantee them the slot they decide to target. Altruistic bidding is the opposite of CB, whereas BB balances between these two extremes. For BB the authors showed that, under some conditions, it converges to the efficient locally envy-free equilibrium characterized in [4]. For AB and CB it was shown that they do not generally converge. Experimental analysis of AB and CB revealed low and high revenue respectively.

The performance of these strategies is analyzed in Bayesian settings in [12,10]. In [9], vindictive strategies are studied for games where bidders are either vindictive or cooperative. Regarding efficiency of equilibria, the first upper bounds on the Price of Anarchy with respect to the social welfare in GSP Auctions were obtained by Lahaie [7]. Tighter upper bounds were obtained for *conservative* bidders (that do not outbid their valuation) by Leme and Tardos in [8]. It was shown that the price of anarchy is at most equal to the golden ratio for the complete information game and at most 8 for the Bayesian setting.

2 Definitions and Preliminaries

An instance of the GSP Auction game has a set of n players (bidders), a set of k slots and a tuple $\langle\{\theta_j\}_{j=1}^k, \{\rho_i\}_{i=1}^n, \{v_i\}_{i=1}^n\rangle$. $\theta_j \in [0,1]$ is the probability that a link displayed in slot j is clicked (*Click-Through Rate* - CTR), $\rho_i \in [0,1]$ is the probability that an advertisement by player i is clicked (*relevance* of i) and v_i is the valuation of i. We use $\hat{v}_i$ for $\rho_i v_i$, the *expected revenue* of i. Assume $\theta_1 \geq \cdots \geq \theta_k > 0$, $\hat{v}_1 \geq \cdots \geq \hat{v}_n$ and define $\gamma_j = \theta_j/\theta_{j-1}$, $\gamma = \max_j \gamma_j$ for $j \geq 2$.

The GSP Mechanism. Players issue collectively a bid vector $\mathbf{b} = (b_1, \ldots, b_n)$; they are ranked in order of non-increasing *declared expected revenue* $\hat{b}_i = \rho_i b_i$ and matched to slots in order of non-increasing CTR. This is the *Rank-By-Revenue* (RBR) rule. When all bidders' relevances are equal, the players are ranked by non-increasing bid b_i (*Rank-By-Bid* rule - RBB). Under RBB, a player i receiving a slot j pays the bid of the $(j+1)$-th player. Under RBR, i pays the declared expected revenue of the bidder i' receiving slot $j+1$ divided by ρ_i, i.e. $\rho_{i'} b_{i'}/\rho_i$.

Given a bid configuration $\mathbf{b}$, we denote by $b_{(j)}, \rho_{(j)}, v_{(j)}$, the bid, relevance and valuation of the player occupying slot j. $\mathbf{b}_{-i}$ is the strategy profile $\mathbf{b}$ without the bid of player i and $\mathbf{b}_{-(j)}$ denotes exclusion of the bid of the player occupying slot j. Define $\mathbf{b}(j) = b_{(j)}$, and $\mathbf{b}_{-i}(j)$, $\mathbf{b}_{-(i)}(j)$ will be the bid of the player occupying slot j in $\mathbf{b}_{-i}$ and $\mathbf{b}_{-(i)}$ respectively. We use $\hat{\mathbf{b}}$ for the vector of declared expected revenues as above. The utility of a player occupying slot j under $\mathbf{b}$ is:

$$u_{(j)}(\mathbf{b}) = \theta_j \rho_{(j)} \left(v_{(j)} - \frac{\rho_{(j+1)} b_{(j+1)}}{\rho_{(j)}} \right) = \theta_j (\hat{v}_{(j)} - \hat{b}_{(j+1)}).$$

The *social welfare* $SW(\mathbf{b})$ of $\mathbf{b}$ is $SW(\mathbf{b}) = \sum_{j=1}^k \theta_j \hat{v}_{(j)} = \sum_{j=1}^k \theta_j \rho_{(j)} v_{(j)}$. We assume a deterministic tie-breaking rule in case there are ties in the ranking. Edelman *et al.* [4] identified a PNE configuration $\mathbf{b}^*$ for the GSP auction game with optimum social welfare $SW(\mathbf{b}^*) = \sum_j \theta_j \rho_j v_j$ and payments equal to the ones in the efficient dominant strategy equilibrium of the VCG mechanism. This equilibrium is also *locally envy-free*, i.e. every bidder i under $\mathbf{b}^*$ is indifferent of receiving at price $\rho_i b_i^*$ the slot right above the one he occupies under $\mathbf{b}^*$.

Local Stability. In Section 4, motivated by the costs incurred to players for learning the competitors' bids, we assume that a player only learns the price of the slots right above/below the slot he currently occupies and only considers these local deviations. In case of ties, i.e., other players above/below him bidding the same, we assume that he learns the price of the first slot below the ties. This inspires a definition of *local stability*, which is a relaxation of Nash equilibrium.

Definition 1. *Let* $\mathbf{b}$ *be a bid configuration of the Generalized Second Price Auction game with* k *slots and* $n \geq k$ *players. Fix any slot* $j_0 \in \{1, \ldots, k\}$ *and let* $j_1 = j_0 + 1$, $j_2 = j_0 - 1$. *Define* $j_1' = \min\left(\{n\} \cup \{j|\hat{b}_{(j)} < \hat{b}_{(j_1)}\}\right)$ *and* $j_2' = \max\left(\{1\} \cup \{j|\hat{b}_{(j)} > \hat{b}_{(j_2)}\}\right)$. *The bid configuration* $\mathbf{b}$ *is locally stable if:*

1. *For any slot* j_0

$$\text{if } j_0 \neq k \text{ and } j_1' \leq k+1, \ \theta_{j_0}(\hat{v}_{(j_0)} - \hat{b}_{(j_0+1)}) \geq \theta_{j_1'-1}(\hat{v}_{(j_0)} - \hat{b}_{(j_1')}), \quad (1)$$
$$\text{if } j_0 \neq 1, \quad \theta_{j_0}(\hat{v}_{(j_0)} - \hat{b}_{(j_0+1)}) \geq \theta_{j_2'+1}(\hat{v}_{(j_0)} - \hat{b}_{(j_2'+2)}), \quad (2)$$

2. *For any player* i *who does not win a slot under* $\mathbf{b}$, $\hat{v}_i \leq \hat{b}_{(k)}$.

The definition states that no player has an incentive to move to the next feasible slot upwards or downwards under $\mathbf{b}$. j_1' and j_2' determine the slot that the bidder at slot j_0 can target, in case that due to ties he cannot aim for the one right above/below him. The condition $j_1' \leq k+1$ in (1) states that a bidder may not be able to deviate downwards if all the remaining bidders have equal score. For non-winning players, we assume they know the *bidding entry level* to competition, $\hat{b}_{(k)} = \rho_{(k)} b_{(k)}$. The last constraint prescribes that no such bidder has incentive to target slot k. In analogy to the *Price of Anarchy* [6], we quantify the inefficiency of locally stable configurations by the following worst-case ratio:

Definition 2. *The Local Stability Ratio of a GSP Auction game is defined as* $\mathbf{\Lambda} = \sup_{\mathbf{b}} \frac{\sum_j \theta_j \hat{v}_j}{SW(\mathbf{b})}$, *where the supremum is over all locally stable configurations.*

We note that the notion of a locally stable configuration and hence the notion of the Local Stability Ratio can be defined for a much wider context. They are applicable to any game where the outcome is a ranking, and for every action profile b any player is allowed, in a well defined manner, to deviate upwards or downwards in the ranking and determine his new payoff. *Ranking Games* [1] constitute one such interesting class of games. (GSP Auctions differ from games studied in [1] in that a player's payoff does not depend only on his rank).

3 Discrete Bidding Strategies

We focus on *conservative* bidders [8] that never outbid their valuation v_i in fear of receiving a negative payoff. Our discussion throughout the paper is in terms of equal relevances and the RBB ranking rule. All results extend for RBR. We assume a discretization of the continuous strategy space $[0, v_i]$ of player i, in multiples of *bidding step* $\epsilon > 0$; i.e., the strategy space of i is $\Sigma_i = \{0, \epsilon, 2\epsilon, \ldots, \lfloor v_i \rfloor_\epsilon\}$, where $\lfloor x \rfloor_\epsilon$ will henceforth denote *the maximum multiple of* ϵ *that is at most* x.

We view sponsored search auctions as repeated games, and we study the bidding strategies AB and CB in the context of iterative best response procedures. In each iteration, given a current configuration $\mathbf{b} = (b_1, \ldots, b_n)$, a player i is chosen at random to respond to $\mathbf{b}_{-i}$ by choosing a bid b'_i, so as to optimize his utility $u_i(\mathbf{b}_{-i}, b'_i)$. To do so, player i aims for the most profitable slot, $j^*(i)$, which he may win by a bid $b'_i \in (\mathbf{b}_{-i}(j^*(i)), \mathbf{b}_{-i}(j^*(i) - 1)]$; i.e., b'_i *strictly beats* $\mathbf{b}_{-i}(j^*(i))$ and equals at most $\mathbf{b}_{-i}(j^*(i)-1)$, the bid issued by a player occupying slot $j^*(i)-1$. Due to discretization and possible ties, it may occur that no $b'_i \in \Sigma_i$ grants the desired slot to i. Hence we define $j^*(i) = \arg\max_j [\theta_j(v_i - \mathbf{b}_{-i}(j))]$, where the max is taken over slots j for which $\Sigma_i \cap (\mathbf{b}_{-i}(j(i)), \mathbf{b}_{-i}(j(i) - 1)] \neq \emptyset$. If there is no such slot, then the bidder does not alter his bid. If bidder i is not occupying any slot under the current configuration $\mathbf{b}$, it may be the case that there is no slot giving him positive utility, in which case the bidder does not alter his bid either. Finally, if $j^*(i)$ equals the currently occupied slot by i, then i does not alter his bid. We consider two simple ways of selecting an extremal bid in this range, namely *Altruistic Bidding* (AB) and *Competitor Busting* (CB).

Altruistic Bidding. AB [3] dictates that player i first computes his optimal slot $j^*(i)$ and then submits the most altruistic bid that is feasible and beats $\mathbf{b}_{-i}(j^*(i))$. Hence if $j^*(i) = 1$, he issues the bid $\mathbf{b}_{-i}(j^*(i)) + \epsilon$, otherwise he bids:

$$b'_i = \min[(\Sigma_i \cap \{\mathbf{b}_{-i}(j^*(i)) + \epsilon, \ldots, \mathbf{b}_{-i}(j^*(i) - 1)\}) \setminus \{b_i\}]$$

Competitor Busting. CB expresses competitive behavior of player i, in that i incurs the highest possible payment to the player receiving the slot right above $j^*(i)$. We define the bid b'_i issued by i to be the maximum feasible bid that grants i slot $j^*(i)$, except if $j^*(i) = 1$. In this case set $b'_i = \mathbf{b}_{-i}(1) + \epsilon$, otherwise:

$$b'_i = \max[(\Sigma_i \cap \{\mathbf{b}_{-i}(j^*(i)) + \epsilon, \ldots, \mathbf{b}_{-i}(j^*(i) - 1)\}) \setminus \{b_i\}]$$

Generally, b'_i *equals* (if feasible) $\mathbf{b}_{-i}(j^*(i)-1)$, except for when $\mathbf{b}_{-i}(j^*(i)-1) = b_i$, in which case $b'_i = \mathbf{b}_{-i}(j^*(i) - 1) - \epsilon$. This definition of CB differs from the one in [3], where $b'_i = \mathbf{b}_{-i}(j^*(i)-1) - \epsilon$ *always.* Note that, assuming that $j^*(i)$ differs from currently occupied slot by i under $\mathbf{b}$, we forbid $b'_i = b_i$.

We need a *tie-breaking* rule, for when a newly submitted bid ties with an existing bid of another player. If bidder i best-responds by $b'_i = \mathbf{b}_{-i}(j')$ for slot j' then bidding b'_i grants i slot $j'+1$ (or lower if there are more ties). For iterative best response this rule facilitates *dynamic temporal tie-breaking*, i.e. bidding the same bid as some player i', but later than i', may only grant a lower slot than i'.

Discretization of the players' strategy spaces in multiples of ϵ may introduce stable configurations that are not PNE in continuous strategies. Although AB and CB have seen experimental study in the recent literature [3], it is not known whether their induced state spaces maintain any PNE of the original game in continuous strategies. By conditioning on ϵ, we establish existence of a socially optimum locally envy-free PNE, which is a discretized version of the PNE identified by Edelman *et al.* in [4]. Our result is additionally strengthened by the fact that, if our iterative best response procedures converge to a socially optimum configuration $\mathbf{b}$, then $\mathbf{b}$ is a PNE of the game even with continuous strategies[1]. Let Δv denote the minimum among the distances between two valuations or the distance of a valuation from 0: $\Delta v = \min\{\{|v_i - v_j| : i, j \in N\} \cup \{|v_i| : i \in N\}\}$.

Theorem 1. *For any bidding step $\epsilon \leq \epsilon^* = (\gamma^{-1}-1)\Delta v$, the configuration space of the GSP Auction game with discrete strategies contains at least one configuration $\mathbf{b}$, that is socially optimum and locally envy-free pure Nash equilibrium for the GSP Auction game even with continuous strategies, given by:*

$$b_j = \begin{cases} b_2 + \epsilon, & \text{if } j = 1 \\ \lfloor (1-\gamma_j)v_j + \gamma_j b_{j+1} \rfloor_\epsilon, & \text{if } 2 \leq j \leq k \\ \lfloor v_j \rfloor_\epsilon, & \text{if } j \geq k+1 \end{cases}$$

Also, if iterative `AB` or `CB` converges to a socially optimum configuration, then this is a pure Nash equilibrium of the GSP Auction game in continuous strategies.

Regarding the convergence of iterative `AB`/`CB`, we found examples showing that `AB` does not always converge, even for bidding step $\epsilon \leq \epsilon^*$ and geometrically decreasing (well separated) CTRs. We were not able to prove or disprove convergence of `CB`, despite extensive experimentation (reported in the full version). Resolving convergence for `CB` is therefore an interesting open problem. Convergence of local versions of these strategies – discussed next – also remains open.

4 Locally Aware Bidders and Local Stability

It is commonly assumed in the literature that bids of other players are observable. In principle one could apply learning techniques to estimate all the other bids as shown in [2]. Such a practice incurs however costs in time and money and, given the dynamic nature of these games, the game may have switched to a different bid vector by the time one estimates all remaining bids. Modeling the uncertainty about other bidders' offers is one approach to this issue [11]. Here we take a different approach and assume that bidders have only local knowledge about the bid vector and make only local moves, adhering to the following rules:

1. They estimate the prices only for the slots right above or below their current slot and – in the absence of ties – will only move one slot upwards or downwards. In case of ties, a bidder learns the price of the first slot above or below him that he can actually target. If none of these moves are beneficial, no deviation occurs.

2. Bidders not receiving a slot only learn the price of the last slot or – in case of ties – the price of the first slot from the end that they can target.

[1] More accurately, *there is a tie-breaking rule* for the one-shot game in continuous strategies that renders $\mathbf{b}$ a PNE. However, the socially optimum locally envy-free PNE described in Theorem 1 is independent of choice of tie-breaking rules.

The restrictions of `AB`/`CB` for such *locally aware* bidders (`L-AB`/`L-CB`) are natural strategies in this setting. If iterative `L-AB` or `L-CB` converge, they will converge to a *locally stable* configuration (in ϵ-discrete strategies), as in Definition 1. We analyze first the inefficiency of locally stable configurations *in continuous strategies.* Subsequently, we consider the performance of iterative `L-AB` and `L-CB`.

Theorem 2. *The GSP Auction game in continuous strategies with conservative bidders has Local Stability Ratio at least $\Omega(\sqrt{\alpha^k})$, for any constant $\alpha > 1$.*

In the proof of this result we used a game instance with $\gamma = 1$. However, fitting of real data in previous works [5] has shown that CTRs are well separated ($\gamma < 1$), by following a power law distribution. Geometrically decreasing CTRs $\theta_j \propto \alpha^{1-j}$ for $\alpha = 1.428$, were observed in [5]. Other authors [10] have used a Zipf distribution, where $\theta_j = j^{-\alpha}$, for $\alpha \geq 1$. For such cases with $\gamma < 1$ we obtain:

Theorem 3. *The GSP Auction game in continuous strategies with conservative bidders has Local Stability Ratio at most $(1-\gamma)^{-1}$, assuming $\gamma < 1$.*

Corollary 1. *For geometrically decreasing click through rates with decay factor $\alpha > 1$ and conservative bidders, $\mathbf{\Lambda} \leq \frac{\alpha}{\alpha-1}$. For click-through rates following the Zipf distribution with $\theta_j = j^{-\alpha}$, for $\alpha \geq 1$, $\mathbf{\Lambda} \leq [1-(1-1/k)^{\alpha}]^{-1} \leq k$.*

Corollary 1 and empirical observations [5] imply a constant upper bound on $\mathbf{\Lambda}$ for geometrically decreasing CTRs. We were not able to find matching lower bounds for Theorem 3 or Corollary 1. We give experimental results in figure 1, for the inefficiency of "reverse" assignments of players to slots, in games with $k = n$ slots, $n = 2, 3, \ldots, 20$. The depicted results were found by solving non-linear programs (one for each curve), that express local stability of the reverse assignment and have $\mathbf{\Lambda}$ as objective function. Tightness of $\mathbf{\Lambda} \leq k$ is evident for Zipf-distributed CTRs. Finally, our analysis for Theorem 3 can be used in bounding the inefficiency of stable configurations of iterative `L-AB` and `L-CB`:

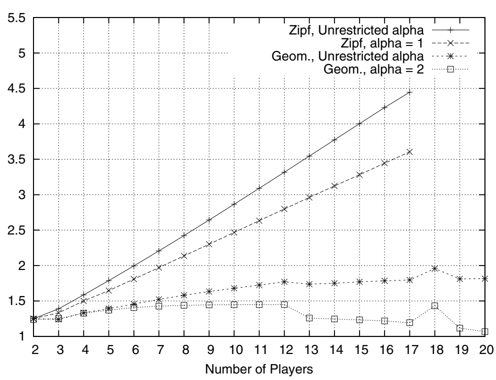


Fig. 1. Local Stability Ratio $\mathbf{\Lambda}$ of Reversed Assignments

Theorem 4. *For $\gamma < 1$ and $\epsilon \leq \epsilon^*$, the Local Stability Ratio of stable configurations with respect to iterative* `L-AB` *and* `L-CB` *is at most $(1-\gamma)^{-1} + \gamma^{-1}$. Moreover, this bound applies to stable configurations with respect to* `AB` *and* `CB`.

References

1. Brandt, F., Fischer, F.A., Harrenstein, P., Shoham, Y.: Ranking Games. Artificial Intelligence 173(2), 221–239 (2009)
2. Brandt, F., Weiß, G.: Antisocial Agents and Vickrey Auctions. In: Meyer, J.-J.C., Tambe, M. (eds.) ATAL 2001. LNCS (LNAI), vol. 2333, pp. 335–347. Springer, Heidelberg (2002)
3. Cary, M., Das, A., Edelman, B., Giotis, I., Heimerl, K., Karlin, A.R., Mathieu, C., Schwarz, M.: Greedy bidding strategies for keyword auctions. In: MacKie-Mason, J.K., Parkes, D.C., Resnick, P. (eds.) Proceedings of the 8th ACM Conference on Electronic Commerce (EC), pp. 262–271. ACM Press, New York (2007)
4. Edelman, B., Ostrovsky, M., Schwartz, M.: Internet Advertising and the Generalized Second Price Auction: Selling Billions of Dollars Worth of Keywords. American Economic Review 97(1), 242–259 (2007)
5. Feng, J., Bhargava, H.K., Pennock, D.M.: Implementing sponsored search in web search engines: Computational evaluation of alternative mechanisms. INFORMS Journal on Computing 19(1), 137–148 (2007)
6. Koutsoupias, E., Papadimitriou, C.H.: Worst-case equilibria. Computer Science Review 3(2), 65–69 (2009)
7. Lahaie, S.: An analysis of alternative slot auction designs for sponsored search. In: Feigenbaum, J., Chuang, J.C.I., Pennock, D.M. (eds.) Proceedings of the 7th ACM Conference on Electronic Commerce (EC), pp. 218–227. ACM, New York (2006)
8. Leme, R.P., Tardos, E.: Pure and Bayes-Nash price of anarchy for generalized second price auction. In: Proceedings of the 51st IEEE Annual Symposium on Foundations of Computer Science, FOCS (to appear, 2010)
9. Liang, L., Qi, Q.: Cooperative or Vindictive: Bidding strategies in Sponsored Search Auction. In: Deng, X., Graham, F.C. (eds.) WINE 2007. LNCS, vol. 4858, pp. 167–178. Springer, Heidelberg (2007)
10. Naldi, M., D'Acquisto, G., Italiano, G.F.: The value of location in keyword auctions. Electronic Commerce Research and Applications 9(2), 160–170 (2010)
11. Vorobeychik, Y.: Simulation-based Game Theoretic Analysis of Keyword Auctions with Low Dimensional Bidding Strategies. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence, UAI (2009), http://www.cs.mcgill.ca/uai2009/proceedings.html
12. Vorobeychik, Y., Reeves, D.M.: Equilibrium analysis of dynamic bidding in sponsored search auctions. In: Deng, X., Graham, F.C. (eds.) WINE 2007. LNCS, vol. 4858, pp. 155–166. Springer, Heidelberg (2007)
13. Zhou, Y., Lukose, R.M.: Vindictive Bidding in Keyword Auctions. In: Gini, M.L., Kauffman, R.J., Sarppo, D., Dellarocas, C., Dignum, F. (eds.) Proceedings of the International Conference on Electronic Commerce (ICEC). ACM International Conference Proceeding Series, vol. 258, pp. 141–146. ACM, New York (2007)

On (Group) Strategy-Proof Mechanisms without Payment for Facility Location Games*

Nguyen Kim Thang

LAMSADE, CNRS FRE 3234, Université de Paris-Dauphine, 75775 Paris, France
Department of Computer Science, University of Aarhus, Denmark

Abstract. We characterize the performance of strategyproof and group-strategyproof social choice rules, for placing a facility on the nodes of a metric network inhabited by N autonomous self-interested agents. Every agent owns a set of locations and caters to minimization of its cost which is the total distance from the facility to its locations. Agents may misreport their locations, so as to manipulate the outcome. A central authority has a set of allowable locations where the facility could be opened. The authority must devise a mechanism that, given the agents reports, places the facility in an allowable location that minimizes the utilitarian social cost — the sum of agents costs. A mechanism is strategyproof (SP) if no agent may misreport its locations and be better off; it is group-strategyproof (GSP) if no coalition of agents benefits by jointly misreporting their locations The requirement for (G)SP in this setting makes optimum placement of the facility impossible and, therefore, we consider approximation (G)SP mechanisms.

For SP mechanisms, we give a simple 3-approximation randomized mechanism and also provide asymptotic lower bounds for different variants. For GSP mechanisms, a $(2N+1)$-approximation deterministic GSP mechanism is devised. Although the mechanism is simple, we showed that it is asymptotically optimal up to a constant. Our $\Omega(N^{1-\epsilon})$ lower bound that randomization cannot improve over the approximation factor achieved by the deterministic mechanism, when GSP is required.

1 Introduction

In a metric space inhabited by N agents, we consider the problem of using agents' reports for their positions to select a facility location in order to minimize aggregatively the agents' distances from their locations to the facility. Each agent owns a set of locations. Agents are self-interested and each one aims at minimizing its individual cost, i.e. total distance from its locations to the facility. To this end, agents may manipulate by strategically misreporting their locations. We study the power and limitations of *strategyproof (SP)* and *group-strategyproof mechanisms (GSP)*, that approximate the optimum aggregate cost

* This work is supported by French National Agency (ANR), project COCA ANR-09-JCJC-0066-01 and Danish Center for Algorithmic Game Theory, funded by the Carlsberg Foundation.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 531–538, 2010.

over all agents within a bounded factor. In the paper, we consider the aggregate cost function as the *utilitarian* social cost, i.e. the *sum* of all agents' costs. A SP mechanism ensures that no agent may misreport its locations and be strictly better off. A GSP mechanism is resilient to coalitional misreports. In effect, the mechanism constitutes a rule for placing the facility, that renders truthful report of the agents' positions a dominant strategy for each agent regardless of the other agents' actions.

Contribution. Our two main results in this paper concern performance characterization of SP and GSP mechanisms for placing a single facility on arbitrary metric graphs. It is known that deterministic SP mechanisms are $\Omega(N)$-approximation due to the characterization of [4]. Hence for SP mechanisms, we turn our attention to randomized ones. We give a simple 3-approximation mechanism that is randomized over a set of preferential locations of agents. We also provide an asymptotic lower bound of 2 for any randomized SP mechanism. For GSP mechanisms, a deterministic GSP mechanism which is inspired from the dictatorship mechanism is devised, which is $(2N + 1)$-approximation. Although that mechanism is simple, we showed that it is asymptotically optimal up to a constant. Our $\Omega(N^{1-\epsilon})$ lower bound that randomization cannot improve over the approximation factor achieved by the deterministic mechanism, when group-strategyproofness is required. Our result also confirms affirmatively a conjecture posed by [1].

Related Work. There has been some early work on characterizing strategyproof facility location mechanisms without payments on lines [2] and on circles [4]. However, [3] initiated in the study of approximating the optimum social cost under the constraint of (group) strategyproofness. The authors considered facility location problems on metric spaces that are lines. A classical result by [4] dictates that the only deterministic SP mechanism that one may hope for a circle is a *dictatorship* (i.e. a mechanism that fixes an agent and always opens the facility as the preference of the agent, while totally ignore the reported locations of other agents). Such a mechanism has approximation ratio $\Omega(N)$. The model in [1] is closest to ours. [1] considered the problem on circles and on general metric networks in which each agent owns only one location and the set of allowable locations where the facility could be opened is the set of all vertices of the graph. They show that the randomized dictatorship mechanism that selects a location uniformly at random, is SP and approximates the social cost within a factor of $2 - \frac{2}{N}$ for any metric network. The authors conjectured that no GSP randomized mechanism may achieve $o(N)$ approximation ratio. This conjectured is settled in the present paper. Our model can be considered as a generalization of the one in [1].

2 Preliminaries

We consider a metric space (Ω, d), where $d : \Omega \times \Omega \to \mathbb{R}$ is the metric function. Let $\mathcal{N}$ be the set of N agents and each agent $i \in \mathcal{N}$ owns a set x_i of w_i locations

$\{x_{i1}, \ldots, x_{iw_i}\}$ where $x_{ij} \in \Omega$ for $1 \leq j \leq w_i$. A *location profile (or strategy profile)* is a vector $\mathbf{x} = (x_1, \ldots, x_N)$ where x_i is a set of locations of agents i for $i = 1, \ldots, N$. Let $\mathcal{F} \subseteq \Omega$ be a set of allowable locations where the facility can be opened. A *deterministic* mechanism is a mapping f from the set of location profiles to a location in $\mathcal{F}$. Given a reported location profile $\mathbf{x}$ the mechanism's output is $f(\mathbf{x})$ and the individual cost of agent $i \in \mathcal{N}$ under mechanism f and profile $\mathbf{x}$ is the total distance from its locations to the facility, denoted by $c_i(f, \mathbf{x})$, $c_i(f, \mathbf{x}) = \sum_{j=1}^{w_i} d(f(\mathbf{x}), x_{ij})$. A *randomized* mechanism is a function f from the set of location profiles to $\Delta(\mathcal{F})$ where $\Delta(\mathcal{F})$ is the set of distribution over $\mathcal{F}$. The cost of agent i now is the expected total distance from its locations to the facility over such distribution: $c_i(f, \mathbf{x}) = \mathbb{E}\left[\sum_{j=1}^{w_i} d(f(\mathbf{x}), x_{ij})\right]$.

The *social cost* of a mechanism f is the sum of individual costs of agents: $C(f, \mathbf{x}) = \sum_{i \in N} c_i(f, \mathbf{x})$. We say that a mechanism f is *r-approximation* if for any profile $\mathbf{x}$, $C(f, \mathbf{x}) \leq r \cdot OPT(\mathbf{x})$ where $OPT(\mathbf{x})$ is the optimal social cost. We will be concerned with *strategyproof* (SP) and *group-strategyproof* (GSP) mechanisms, which render truthful revelation of the agents' location a dominant strategy for the agents.

Definition 1. (Group-Strategyproofness) *Let $\mathbf{x}$ denote the location profile of a set $\mathcal{N}$ of N agents, over the metric space (Ω, d). A mechanism f is group-strategyproof if for every non-empty subset of agents $\mathcal{I} \subseteq \mathcal{N}$ and for every location profile $\mathbf{x}'$ with $x'_j = x_j$ for $j \notin \mathcal{I}$, there is an agent $i \in \mathcal{I}$ with $c_i(f, \mathbf{x}') > c_i(f, \mathbf{x})$.*

Similarly, a mechanism f is *strategyproof* if it satisfies the definition above in which every subset of agents $\mathcal{I}$ in the definition is restricted to be singleton. Given a subset $U \subset \Omega$ in the metric space, we define $\mathsf{med}(U)$ as $\mathsf{med}(U) := \arg\min\{v \in \mathcal{F} : \sum_{u \in U} d(v, u)\}$, break tie arbitrarily.

3 Strategy-Proof Mechanisms

Randomized mechanism. Given a location profile $\mathbf{x} = (x_1, \ldots, x_N)$ where $x_i = \{x_{i1}, \ldots, x_{iw_i}\}$ for $1 \leq i \leq N$. Let $y_i = \mathtt{med}(x_i)$. Open the facility at y_i with probability w_i/W where $W = \sum_{i=1}^{N} w_i$.

Theorem 1. *The mechanism is strategy-proof and that yields 3-approximation. Moreover, no randomized SP mechanism has approximation ratio better than $(2 - \epsilon)$ even if either each agent possesses only one location or agents have many locations and $\mathcal{F} = \Omega$.*

4 Group Strategy-Proof Mechanisms

Deterministic mechanism. Given a location profile $\mathbf{x} = (x_1, \ldots, x_N)$ where $x_i = \{x_{i1}, \ldots, x_{iw_i}\}$ for $1 \leq i \leq N$. Let $y_i = \mathtt{med}(x_i)$. Let $i^* := \arg\max_{1 \leq j \leq N} w_j$, break tie according to agents' index. Open the facility at y_{i^*}.

Theorem 2. *The deterministic mechanism is GSP that yields* $(2N+1)$*-approximation. Moreover, no randomized GSP mechanism has approximation ratio better than* $N^{1-3\epsilon}$ *for arbitrarily small* $\epsilon > 0$ *even if each agent possesses one location and the facility could be opened everywhere in the metric space, i.e* $\mathcal{F} = \Omega$*.*

The theorem follows Lemma 1 and Lemma 4.

Lemma 1. *The mechanism is GSP that yields* $(2N+1)$*-approximation.*

In the remaining of section, we prove the tight bound for any randomized GSP mechanism. Our lower bound works even in a restricted variant in which each agent owns only one location and the set of allowable facility locations $\mathcal{F}$ (where the facility could be opened) is the set of all vertices in a given network. Hence, until the end of the section, we consider and prove lower bound on this restricted variant.

Starting point. The lower bound $\Omega(N)$ for deterministic GSP mechanisms is devised from the characterization of [4]. As dictatorship is the only deterministic GSP mechanism for cycle graph, the lower bound is straightforwardly deduced. However, there is no similar characterization for randomized GSP mechanisms. In our approach, we start looking for a game which induces the same lower bound for deterministic GSP mechanisms without using the characterization in [4]. Consider the following instance. A network graph $G(U \cup V, E)$ consists of $2N$ vertices, where $U = \{u_1, \ldots, u_N\}$ and $V = \{v_1, \ldots, v_N\}$. Vertices in U form a complete graph with edge of distance 1. Each vertex v_i connects to all vertices in $U \setminus \{u_i\}$ by edge of cost $1 - \epsilon$. Consider an initial location profile $\mathbf{x}^0$ in which there are N agents, agent i locates on vertex u_i for $1 \leq i \leq N$. We study two cases.

Suppose that in profile $\mathbf{x}^0$ the facility is opened in U, w.l.o.g in u_1. Consider location profile $\mathbf{x}^1$ in which agents $2, \ldots, N$ locate at vertex v_1 and agent 1 locates at u_1. In this profile, the facility is not opened at v_1 since otherwise, agents $2, \ldots, N$ have incentive to collaborate and move to vertex v_1 in the initial profile (they decrease their cost from 1 to $(1 - \epsilon)$). Hence, the social cost is at least $(N-1)(1-\epsilon)$ while the OPT is $2 - \epsilon$ by opening the facility at v_1.

Suppose that in profile $\mathbf{x}^0$ the facility is open in V, w.l.o.g in v_1. The cost of agent 1 is $2 - \epsilon$ and the cost of the other is $1 - \epsilon$. Consider location profile $\mathbf{x}^2$ in which agents $1, \ldots, N-1$ report vertex v_N and agent N reports u_N. In this profile, the facility is not opened at v_N since otherwise in profile $\mathbf{x}^0$ agents $1, \ldots, N-1$ have incentive to collaborate and report vertex v_N (agent 1 decreases strictly her cost while the cost of the other in the cooperation remains unchanged). Again, the approximation ratio is larger than $(N-1)/2$ in this profile.

Idea for lower bound of randomized mechanisms. The previous argument does not carry for randomized mechanisms. Consider the second case of the analysis in the previous paragraph. In profile $\mathbf{x}^2$, a randomized mechanism may open a facility with probability arbitrarily close to 1 at vertex v_N and with the remaining probability (small but positive), open a facility in other vertex, for example v_2, in order to increase the cost of one agent in the cooperation and so prevent agents from collaborating. An idea to circumvent is the following. We modify the edge costs between U and V to break the symmetry. Then, argue that in some profiles with a bunch of agents in a vertex, any randomized GSP mechanism will open the facility at that vertex with large probability but there is still gap between this probability and 1. Using the gap, we amplify the approximation ratio.

Let $0 < \epsilon < 1$ be an arbitrarily small constant. Let n be a large integer and m be also an integer such that $(2m+1)^m = n^{\epsilon/2}$, i.e $m = \Theta(\epsilon \log n / \log\log n)$. We can choose n, m such that $\ell = n/m$ is integer. We define a sequence with useful property.

Lemma 2. *Consider a sequence $(\beta_i)_{i=1}^m$ defined as follows:*

$$\beta_m = n^{-3\epsilon/2}, \qquad \beta_i = m\beta_{i+1} + (m+1)\beta_m \ \forall 1 \le i \le m-1.$$

Then, sequence $(\beta_i)_{i=1}^m$ is a decreasing, $\beta_1 \le n^{-\epsilon}$ and $\beta_{i+1} + 2\beta_m = \frac{m-1}{m}\beta_m + \frac{1}{m}\beta_i$

Consider a graph $G(U \cup V, E)$ consisting of $(n+m)$ vertices $U = U_1 \cup \ldots \cup U_m$ where $U_i = \{u_{i1}, \ldots, u_{i\ell}\}$, for $1 \le i \le m$, and $V = \{v_1, \ldots, v_m\}$. Vertices in U form an independent set. Each vertex v_j is connected with all vertices in U such that the distances from v_j to any vertex in U_i are the same. Denote this distance as $d(v_j, U_i)$ (i.e $d(v_j, U_i) = d(v_j, u_{i1}) = \ldots = d(v_j, u_{i\ell})$). We define the distances between vertices in U and V as follow. For all $1 \le i, j \le m$, $d(v_j, U_i) = 1 + \beta_{t(i,j)}$ where $t(i,j) = 1 + (i - j \mod m)$ and $(\beta_i)_{i=1}^{m-1}$ is defined in Lemma 2. As the distance from any vertex in U_i to any one in U_j is the same, we also denote such distance as $d(U_i, U_j)$. Note that $d(U_i, U_i)$ means the distance between two different vertices in U_i. By definition, the diameter of the graph is at most $2 + 2\beta_1$ and $2 + 2\beta_m \le d(U_i, U_j) \le 2 + 2\beta_1 \ \forall i, j$.

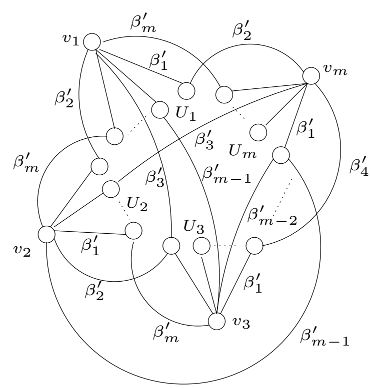


Fig. 1. A part of graph G where $\beta'_i = 1 + \beta_i \ \forall 1 \le i \le m$

Let f be a randomized GSP mechanism. Let $\mathbf{x}^0$ be a location profile in which there is one agent on each vertex in U. We prove the following main lemma.

Lemma 3. *There exists a location profile $\mathbf{x}$ in which at least $(n - \ell - 1)$ agents locate on a vertex $v_i \in V$ for some i, and the others' locations are the same as in $\mathbf{x}^0$ such that $\mathbb{P}[f(\mathbf{x}) = v_i] < 1 - \beta_m$. Moreover, among all agents whose locations are the same as in $\mathbf{x}^0$, at most three agents do not stay in U_i (in other words, almost such agents have locations in U_i).*

Proof. In profile $\mathbf{x}^0$, let p_i be the probability that the facility is opened at agent i's location for $1 \le i \le n$. Let q be the total probability that the facility is opened in U. We consider two cases where $q \ge 3\beta_1$ and $q < 3\beta_1$.

Case 1: $q \geq 3\beta_1$. Without loss of generality assume that $p_1 \leq \ldots \leq p_n$. So $p_k \leq q/(n-k+1)$. In profile $\mathbf{x}^0$, the expected cost of agent k is at least $(1-q) \cdot (1+\beta_m) + q \cdot (1 - \frac{1}{n-k+1}) \cdot (2+2\beta_m)$. Consider the profile $\mathbf{x}^1$ in which agents $1, \ldots, n-3$ locate at vertex v_i for some arbitrary i, and the others' locations are the same as in profile $\mathbf{x}^0$. Let z_k be the location of agent k in profile $\mathbf{x}^0$. By group-strategyproofness, the mechanism f must guarantee the existence of $k \in [1, n-3]$ such that $\mathbb{E}[d(f(\mathbf{x}^1), z_k)] > \mathbb{E}[d(f(\mathbf{x}^0), z_k)]$ since otherwise agents $1, 2, \ldots, n-3$ may collaborate, report together their locations as v_i and all get better off. Denote $\alpha_1 := \mathbb{P}[f(\mathbf{x}^1) = v_i]$. We have:

$$\mathbb{E}[d(f(\mathbf{x}^1), z_k)] \leq \alpha_1 \cdot (1+\beta_1) + (1-\alpha_1) \cdot (2+2\beta_1) \quad \forall\, 1 \leq k \leq n-3$$

Hence, if there exists $1 \leq k \leq n-3$ such that $\mathbb{E}[d(f(\mathbf{x}^1), z_k)] > \mathbb{E}[d(f(\mathbf{x}^0), z_k)]$ then:

$$\alpha_1 \cdot (1+\beta_1) + (1-\alpha_1) \cdot (2+2\beta_1) > (1-q) \cdot (1+\beta_m) + q \cdot (1 - \frac{1}{n-k+1}) \cdot (2+2\beta_m)$$

As $k \leq n-3$ and $q \geq 3\beta_1$, we deduce $\alpha_1 < 1 - \frac{q - \frac{2q}{(n-k+1)} - \beta_1 + \beta_m}{1+\beta_1} < 1 - \beta_m$. Thus, $\mathbf{x}^1$ is a profile that satisfies conditions of the lemma.

Case 2: $q < 3\beta_1$. Without loss of generality assume that in profile $\mathbf{x}^0$, the probability that the facility is opened at v_1 is largest among all vertices in V. So $\mathbb{P}[f(\mathbf{x}^0) = v_1] \geq (1-q)/m$. Let $a_1, \ldots, a_{n-\ell}$ be agents in $U_1 \cup \ldots \cup U_{m-1}$ such that $p_{a_1} \leq p_{a_2} \leq \ldots \leq p_{a_{n-\ell}}$. Remark that $p_{a_k} \leq q/(n-\ell-k+1)$. First, we bound the cost of agent a_k in profile $\mathbf{x}^0$. Let z_k be the location of agent a_k and let $i(k)$ be an index such that $z_k \in U_{i(k)}$. The cost of agent a_k is:

$$\mathbb{P}[f(\mathbf{x}^0) = v_1] \cdot d(U_{i(k)}, v_1) + \sum_{j=2}^{m} \mathbb{P}[f(\mathbf{x}^0) = v_j] \cdot d(U_{i(k)}, v_j) \; + \sum_{j=1}^{m} \mathbb{P}[f(\mathbf{x}^0) \in U_j] \cdot$$

$$\cdot\, d(U_{i(k)}, U_j) > \left(1 + \frac{m-1}{m}\beta_m + \frac{1}{m}\beta_{i(k)}\right)(1-q) + 2q - \frac{2q}{n-\ell-k+1}$$

Consider location profile $\mathbf{x}^2$ in which agents $a_1, \ldots, a_{n-\ell-1}$ locate at v_m and the other agents' locations are the same as in profile $\mathbf{x}^0$. Denote $\alpha_2 := \mathbb{P}[f(\mathbf{x}^2) = v_m]$. By group-strategyproofness, there exists k such that $\mathbb{E}[d(f(\mathbf{x}^2), z_k)] > \mathbb{E}[d(f(\mathbf{x}^0), z_k)]$ since otherwise in profile $\mathbf{x}^0$, agents $a_1, \ldots, a_{n-\ell-1}$ have incentive to move together to v_m and all get better off. Note that distance $d(U_{i(k)}, v_m) = 1 + \beta_{i(k)+1}$. Then in profile $\mathbf{x}^2$, we have $\mathbb{E}[d(f(\mathbf{x}^2), z_k)] < \alpha_2(1+\beta_{i(k)+1}) + (1-\alpha_2)(2+2\beta_1)$. As $\mathbb{E}[d(f(\mathbf{x}^2), z_k)] > \mathbb{E}[d(f(\mathbf{x}^0), z_k)]$ holds for some $k \in [1, n-\ell-1]$, we have:

$$\alpha_2\left(1+\beta_{i(k)+1}\right) + (1-\alpha_2)(2+2\beta_1) > \left(1 + \frac{m-1}{m}\beta_m + \frac{1}{m}\beta_{i(k)}\right)(1-q) + q$$

Therefore, $\alpha_2 < 1 - \frac{\left(\frac{m-1}{m}\beta_m + \frac{1}{m}\beta_{i(k)}\right)(1-q) - \beta_{i(k)+1}}{1+2\beta_1-\beta_{i(k)+1}} < 1 - \beta_m$, where the last inequality is due Lemma 2, the case assumption $q < 3\beta_1$ and $\beta_{i(k)} \cdot \beta_1$ is dominated by β_m for any $i(k)$. Thus, profile $\mathbf{x}^2$ satisfies conditions of the lemma. □

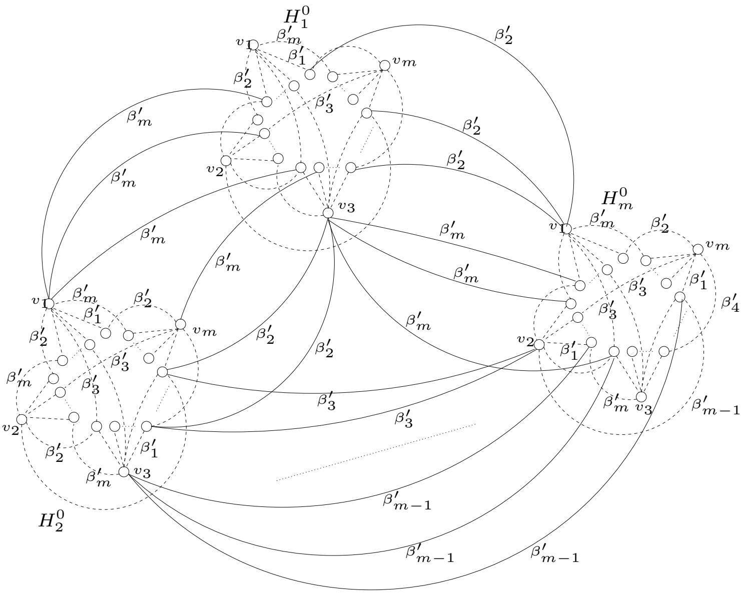


Fig. 2. A part of graph $H^1(\beta_1^1, \ldots, \beta_m^1)$ where in the picture $\beta'_i = 1 + \beta_i^1 \ \forall 1 \le i \le m$

Lemma 4. *There exits an instance with N agents in which any randomized GSP has approximation ratio at least $N^{1-3\epsilon}$ for $\epsilon > 0$ arbitrarily small.*

Proof. First, we construct recursively a family of graphs $H^j(\gamma_1, \ldots, \gamma_m)$ for $j \ge 0$ where vertices are $(U^j \cup V^j)$ and $\gamma_1, \ldots, \gamma_m$ are variables. In graph $H^j(\gamma_1, \ldots, \gamma_m)$, the lengths of edges are taken from the set $\{1+\gamma_1, \ldots, 1+\gamma_m\}$. Denote n_j be the number of vertices in graph H^j. Let n, m be large constant that are defined in the construction of graph G previously. Define graph $H^0(\gamma_1, \ldots, \gamma_m)$ is the same as graph G described previously where $U^0 = U$ and $V^0 = V$ except that now the lengths of edges are taken from the set $1+\gamma_1, \ldots, 1+\gamma_m$. For example, if we assign variable $\gamma_i = \beta_i$ for $1 \le i \le m$ (where β_i is defined in Lemma 2) then $H^0(\gamma_1, \ldots, \gamma_m) = H^0(\beta_1, \ldots, \beta_m) = G$.

Intuitively, graph H^j contains m copies of graphs H^{j-1} and each of such copies plays similar role as vertices $U_i \cup v_i$ in the description of G. Formally, graph $H^j(\gamma_1, \ldots, \gamma_m)$ consists of $n_j = mn_{j-1}$ vertices that we can partition the vertices as $U^j = U_1^j \cup \ldots \cup U_m^j$ and $V^j = V_1^j \cup \ldots \cup V_m^j$. For each $1 \le i \le m$, the restricted graph of $H^j(\gamma_1, \ldots, \gamma_m)$ on $U_i^j \cup V_i^j$ is the same graph as $H^{j-1}(\gamma_1, \ldots, \gamma_m)$. Moreover, each vertex in V_i^j connects with a vertex in $U_{i'}^j$ by an edge of length $1 + \gamma_{t(i',i)}$ (where $t(i', i) = 1 + (i' - i \mod m)$) for $1 \le i, i' \le m$. Note that in graph $H^j(\gamma_1, \ldots, \gamma_m)$, all edges have length in $\{1+\gamma_1, \ldots, 1+\gamma_m\}$ and the diameter of the graph is at most $2+2\max\{\gamma_1, \ldots, \gamma_m\}$. Additionally, an invariant in any graphs is $|U^j| > |V^j|$.

Let t be a large constant to be defined later. Let $\beta_1^t, \ldots, \beta_m^t$ be a sequence defined in Lemma 2 in which parameter n in the lemma is replaced by $n_t/2$.

Consider graph $H^t(\beta_1^t, \ldots, \beta_m^t)$ (see Figure 2 for an illustration) and initial location profile $\mathbf{x}$ in which there is one agent on each vertex in U^t. Let N be the number of agents ($N = |U^t|$). We have $N > n_t/2$ as $|U^t| > |V^t|$. Therefore, $\beta_m^t > N^{-2\epsilon}$. By Lemma 3, there exists a profile $\mathbf{x}^1$ in which at least $(N - N/m - 1)$ agents locate on a vertex $v \in V_{i_1}^t$ for some i_1 and the others' locations are the same as in $\mathbf{x}^0$ such that $\mathbb{P}[f(\mathbf{x}^1) = v] < 1 - \beta_m^t < 1 - N^{-2\epsilon}$. By the symmetry of the graph H^t, the statement is valid for any vertex $v \in V_{i_1}^t$. We denote A_1 the set of agents whose locations in $\mathbf{x}$ and $\mathbf{x}^1$ are different.

Now consider graph H^t and profile $\mathbf{x}$ restricted on vertices $U_{i_1}^t \cup V_{i_1}^t$. By construction, this is a graph H^{t-1} and we denote $\mathbf{x}'$ the profile restricted on this graph. Apply again Lemma 3 on the graph H^{t-1} and profile $\mathbf{x}'$, there exists there exists a profile $\mathbf{x}^2$ in which at least $(N/m - N/m^2 - 1)$ agents locate on a vertex $v \in V_{i_2}^{t-1}$ for some i_2 and the others' locations are the same as in $\mathbf{x}'$ such that $\mathbb{P}[f(\mathbf{x}^2) = v] < 1 - \beta_m^t < 1 - N^{-2\epsilon}$. By the symmetry of the graph H^{t-1}, the statement is valid for any vertex $v \in V_{i_2}^{t-1}$. Remark that $V_{i_1}^t \supset V_{i_2}^{t-1}$. We denote A_2 the set of agents whose locations in $\mathbf{x}'$ and $\mathbf{x}^2$ are different.

We apply the same argument by considering graph H^t and profile $\mathbf{x}$ restricted on vertices $U_{i_2}^{t-1} \cup V_{i_2}^{t-1}$ and so on. In the last round, we end up with a profile in which at least $(N/m^{t-1} - N/m^t - 1)$ agents locate on a vertex $v \in V_{i_{t+1}}^0$.

Let $v^* \in V_{i_1}^t \cap V_{i_2}^{t-1} \cap \ldots \cap V_{i_{t+1}}^0$. Consider graph H^t and a location profile $\mathbf{x}^*$ in which agents in $(A_1 \cup \ldots \cup A_t)$ locate at v^* and the others have the same locations as in $\mathbf{x}$. We have $\mathbb{P}[f(\mathbf{x}^*) = v^*] < 1 - \beta_m^t < 1 - N^{-2\epsilon}$ since otherwise, in $\mathbf{x}$, all agents in $(A_1 \cup \ldots \cup A_t)$ will get better off by reporting together their location as v^*. Hence, the social cost is at least $N^{-2\epsilon}(N - N/m^{t+1} - t)$. The optimal solution opens facility at v^* with cost at most $2N/m^t + 3t$ where $3t$ comes from the fact that at each round r for $1 \leq r \leq t$, in the considered profile of round r, there are at most three agents neither in A_r nor located in $V_{i_{t-r}}^{t-r-1}$ (Lemma 3). Choose t large enough, say $t = (1-\epsilon)\log_m N - 1$, the approximation ratio is at least $N^{1-3\epsilon}$. □

Acknowledgment. We warmly thank Orestis A. Telelis for many helpful, stimulating discussions and comments on the earlier version of this paper.

References

1. Alon, N., Feldman, M., Procaccia, A.D., Tennenholtz, M.: Strategyproof approximation mechanisms for location on networks (2009)
2. Moulin, H.: On strategy-proofness and single peakedness. Public Choice 35(4), 437–455 (1980)
3. Procaccia, A.D., Tennenholtz, M.: Approximate mechanism design without money. In: ACM Conference on Electronic Commerce, pp. 177–186 (2009)
4. Schummer, J., Vohra, R.V.: Strategy-proof location on a network. Journal of Economic Theory 104 (2001)

Threshold Models for Competitive Influence in Social Networks

Allan Borodin, Yuval Filmus, and Joel Oren

Department of Computer Science, University of Toronto, Canada
{bor,yuvalf,oren}@cs.toronto.edu

Abstract. The problem of influence maximization deals with choosing the optimal set of nodes in a social network so as to maximize the resulting spread of a technology (opinion, product-ownership, etc.), given a model of diffusion of influence in a network. A natural extension is a competitive setting, in which the goal is to maximize the spread of our technology in the presence of one or more competitors.

We suggest several natural extensions to the well-studied linear-threshold model, showing that the original greedy approach cannot be used.

Furthermore, we show that for a broad family of competitive influence models, it is NP-hard to achieve an approximation that is better than a square root of the optimal solution; the same proof can also be applied to give a negative result for a conjecture in [2] about a general cascade model for competitive diffusion.

Finally, we suggest a natural model that is amenable to the greedy approach.

1 Introduction

The problem of influence maximization has long been the focus of study in social science (e.g. [5]). It has been formally defined and addressed in [6,3] as follows: given a social-network as a directed graph, and a prescribed number k, pick the k most "influential" nodes that will function as early adopters of a particular influence, so as to maximize the final number number of *infected*, or *activated* nodes (the two terms are used in this paper interchangeably), subject to a specified model of influence diffusion.

This problem begs the natural extension of a competitive version: given the competitor's choice of early adopters of technology B, maximize the spread of technology A by choosing a set of early adopters such that the expected spread of technology A will be maximal. Indeed, this problem has been the subject of interest in subsequent papers ([2,7,1]) which present competitive extensions for the independent cascade model presented in [6].

In this paper we suggest several natural extensions to a well-studied model that was also given in [6] for the diffusion of social influence in a social network — the linear threshold model. Formally, an instance of the problem would be composed of a directed, edge-weighted graph $G = (V, E)$, a set of technology

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 539–550, 2010.

B's initial adopters $I_B \subseteq V$, and an integer k. The computational problem is how to choose a set $I_A \subseteq V - I_B$ of k nodes such that the *expected* number of A-active nodes at the end of the process, $\sigma(I_A, I_B)$, is maximized, given the specific model for competitive diffusion of technologies (when I_B is known from context we omit it and simply write $\sigma(I_A)$). It is important to note that all of the presented models can be motivated by natural processes. Our models reduce to the original linear threshold model formalized in [6] whenever I_B is the empty set. For simplicity of notation, the models presented are stated in terms of only two competing technologies. However, all of the models and results can be easily extended for when there are several competing technologies.

A well-known greedy $(1 - e^{-1})$-approximation given in [4] is used extensively for problems of maximizing set-functions, and in particular has been applied to both the original problem and competitive extensions ([2,1]). The approximation algorithm requires that the set function $\sigma(\cdot)$ at hand, which assigns a real-value to subsets of a ground set U, uphold two basic properties.

– **Monotonicity:** the value of the function increases as more items are added to the set: $\sigma(S) \leq \sigma(T)$ for any two sets $S \subseteq T$;
– **Submodularity:** the impact of adding an element to a set decreases as the set is extended (diminishing returns): $\sigma(S \cup \{x\}) - \sigma(S) \geq \sigma(T \cup \{x\}) - \sigma(T)$, for any $S \subseteq T \subseteq U$ and $x \in U - T$;

Except for the last model, described in section 6 — the *OR* model, all of the models do not satisfy submodularity. In fact, one of them is not even monotone.

Outline. The remainder of this document is organized as follows. Sections 2 and 3 describe two competitive threshold models. Section 4 shows that even when applying a final step that A-activates more nodes, the process remains non-submodular. Section 5 shows that the last two models are in general hard to approximate. On a more positive note, in section 6 we suggest a fairly natural and simple model for which the approximation algorithm given in [4] is applicable. Finally, section 7 summarizes our main results along with a few open problems and possible directions for future research.

2 The Weight-Proportional Competitive Linear Threshold Model

As in the non-competitive case, the process unfolds in discrete steps, during which new nodes become "activated" for a single technology[1]. The infection of a node by a technology represents an individual in the social network that has assumed the influence of that technology. The process is *progressive*: a node that is infected by a technology remains infected by it. As in the non-competitive case, every edge (u, v) is assigned a weight $w_{u,v} \in [0, 1]$ which roughly characterizes

[1] The term "technology" stands for any concept or influence that spreads in the social-network (car ownership, club membership, voting preference, etc.).

the weight of influence that u has over v (i.e the impact that u's infection will have over v's likelihood to be infected with the same technology as u). Also, the total weight of incoming edges to every node is bounded: for every $v \in V$ we have: $\sum_u w_{u,v} \in [0,1]$. Each node u initially chooses a threshold θ_u which represents the minimum fraction of active neighbours necessary for u's activation. As in [6], in order to make up for our lack of knowledge about each node we assume that $\theta_u \in_R [0,1]$ (uniformly at random), or $\theta_u \in_R [a,a']$ for $0 \leq a \leq a' \leq 1$ to reflect partial knowledge about a node.

In order to describe the process itself, we will use the following notation:

Definition 1. *For a given step t in the process, let Φ^t denote the set of active nodes at the beginning of step t. Furthermore, let Φ^t_A and Φ^t_B be the sets of A-active and B-active nodes in step t, respectively.*

Given the sets I_A, I_B of early technology adopters, the process unfolds as follows. First, each node chooses its threshold value at step 0. Then, in each step t, every inactive node v checks the set of edges incoming from its active neighbours. If their collective weight exceeds the threshold values by having $\sum_{u\in\Phi^t} w_{u,v} \geq \theta_v$, the node becomes active. In that case, the node will adopt technology A with probability equal to the ratio between the collective weight of edges outgoing from A-active neighbours and the total collective weight of edges outgoing from all active neighbours; that is,

$$Pr[v \in \Phi^t_A | v \in \Phi^t \backslash \Phi^{t-1}] = \frac{\sum_{u\in\Phi^t_A} w_{u,v}}{\sum_{u\in\Phi^t} w_{u,v}} \tag{1}$$

Otherwise, it will adopt technology B. Since this problem can be reduced to the single-technology linear threshold model whenever I_B is the empty set, we notice that this problem is NP-hard — as proved in [6].

Intuitively, it appears that by adding a new node to the set of initial A-adopters, the spread of technology A in the social network can only increase (or remain unchanged). However, this is in fact not always the case, even for some binary rooted trees. We will formalize this somewhat counter-intuitive behaviour.

Theorem 1. *There exists an instance of the weight-proportional competitive linear threshold problem for which monotonicity does not hold.*

Also, it can be shown that submodularity fails to hold in some cases, as the following theorem shows:

Theorem 2. *There exists a graph G, for which the expected influence of technology A is not submodular.*

The proof of the above two theorems is given in appendix A.

3 The Separated-Threshold Model for Competing Technologies

In the previous model, the node activation step regarded active nodes as equal, so that a given node is activated by its active neighbours regardless of their technologies. That is, the sum of generally active nodes was used for activating a node. However, one could model the following notion of a spread process. Each individual has separate criteria for becoming active for each technology. A node can be activated by either its A-active or B-active neighbours whenever the sums of their respective edge-weights exceed the required thresholds specified for their technologies.

Formally, consider the following model. For a given network $G = (V, E)$, every edge $(u, v) \in E$ is assigned a real-valued weight corresponding to each technology $w^A_{u,v}, w^B_{u,v} \in [0, 1]$ such that $\sum_u w^A_{u,v}, \sum_u w^B_{u,v} \in [0, 1]$, which reflects node u's impact on v. Two *disjoint* sets $I^0_A, I^0_B \subseteq V$ denote the sets of initially A-active and B-active nodes, respectively. At step 0, each node $v \in V$ picks two threshold values $\theta^A_v, \theta^B_v \in_R [0, 1]$. For step t, denote I^{t-1}_A, I^{t-1}_B as the sets of A-active and B-active nodes. During every step t, an inactive node v will become A-active if $\sum_{u \in I^{t-1}_A} w^A_{u,v} \geq \theta^A_v$, and will become B-active if $\sum_{u \in I^{t-1}_B} w^B_{u,v} \geq \theta^B_v$. If for the node v both thresholds are exceeded during the same step t, then a technology would be picked uniformly at random (we can either use a simple coin-flip or employ a more involved tie-breaking function).

In contrast to the previous model, this model *is* monotone. Its key property, which distinguishes it from the previous model, is that the probability that technology B will propagate cannot increase as a result of A-activating additional nodes. This stems from the definition of the model, in which each set of technology specific neighbours relate to a separate threshold value.

Let us use the following notation:

Definition 2. *Given the sets I_A and I_B, and a node $x \notin I_B$, define $\alpha^t_v, \hat{\alpha}^t_v$ as the probabilities that a given node v will adopt technology A by step t for the initial sets of early adopters (I_A, I_B) and $(I_A \cup \{x\}, I_B)$, respectively. Similarly, define similar probabilities $\beta^t_v, \hat{\beta}^t_v$ for technology B.*

Theorem 3. *For a given instance of the problem and a choice of early adopters: I_A, I_B and node x, $\hat{\alpha}^t_v \geq \alpha^t_v$ for any node v and for any step $t \geq 0$.*

The proof of theorem 3 is fairly straightforward, and is given in appendix B for completeness. The process is not submodular in general.

Theorem 4. *There exist instances of the competitive influence problem where the separated-threshold competitive model is not submodular.*

A corresponding counter-example for this theorem is fairly easy to construct. It appears in the full version of this paper located on the authors' personal websites.

4 Competitive Threshold Model with Forcing

We now introduce a modification which changes the concept of influence in a network: *forcing*. Specifically, at the end of the previous model, each inactive node v will choose a technology randomly (say, it will choose technology A with probability δ). This step is natural for cases where individuals have to eventually decide which influence to adopt (e.g. voting when abstentions are not allowed). For convenience we will assume that the "forcing" step occurs at step n (the spread can take up to $n-1$ steps), whether or not the spread took $n-1$ steps. Clearly this does not have any effect on the outcome of the process. We show that regardless of the forcing step, this variant does not help us achieve submodularity. However, the process remains monotone as the following theorem can be proven by extending lemma 3 (in appendix B) with an additional case analysis for the forcing step.

Theorem 5. *For a given instance of the competitive influence with forcing problem, a choice of early adopters I_A, I_B and node x, $\hat{\alpha}_v^t \geq \alpha_v^t, \hat{\beta}_v^t \leq \beta_v^t$ for any node v and for any t.*

The following theorem shows that not only is the given model non-submodular, but also regardless of the tie-breaking rule and the forcing rule (if any is used), the model remains non-submodular.

Theorem 6. *For any tie-breaking rule, and any forcing rule, the separated-threshold competitive model is non-submodular.*

A corresponding counter-example is given in appendix C.

5 Hardness of Approximation

We show that in any model with separate edge-weights and separate threshold values for each technology the problem is hard to approximate.

Theorem 7. *It is NP-hard to give an approximation with a ratio better than $\Omega(N^{\frac{1}{2}-\epsilon})$, for all $\epsilon > 0$, for the Separated-Threshold Competitive Influence problem, where N is the number of nodes in the graph.*

The proof is supplied in appendix D. It is important to note the proof of theorem can be applied to similar competitive cascade models as well. Namely, in [2] it was conjectured that when allowing 2 sets of edge weights for each edge — one for each technology, the process will remain monotone and submodular. The above hardness of approximation result can be modified in order to apply for the separate edge-weights case of the Wave Propagation model suggested by Carnes et al., thereby giving a negative answer to their conjecture.

Theorem 8. *It is NP-hard to give an approximation with a ratio better than $\Omega(N^{\frac{1}{2}-\epsilon})$, for all $\epsilon > 0$, for the Wave Propagation Competitive Influence problem given by Carnes when edges are allowed to have technology-specific probabilities.*

6 The OR Model

We now introduce a different way of extending the original threshold model, in which each technology diffuses unhindered by the competing technology. Here, the tie-breaking stage will take place after all technologies finish spreading. This model can be considered natural for cases in which individuals have the liberty of being influenced separately and independently by different technologies, but have to choose a single technology eventually.

We will define the *OR* model as follows. As before, an instance of the model is a graph $G = (V, E)$, a set of edge weights for each technology: $W_A = \{w^A_{u,v}\}_{(u,v)\in E}, W_B = \{w^B_{u,v}\}_{(u,v)\in E}$ (with the same constraints as before), and initial adopters: $I_A, I_B \subseteq V$. Additionally, for each node $v \in V$ two "decision" functions $f^A_v : 2^V \times 2^V \to [0,1]$, $f^B_v : 2^V \times 2^V \to [0,1]$ are assigned. Let each technology propagate separately throughout the graph w.r.t the original non-competitive linear threshold propagation process, and let $R_A, R_B \subseteq V$ be the sets of nodes reached by the technologies (independently). As a final step, a node $v \notin I_A \cup I_B$ will assume technology A with probability $f^A_v(R_A, R_B)$, technology B with probability $f^B_v(R_A, R_B)$, and no technology with probability $1 - f^A_v(R_A, R_B) - f^B_v(R_A, R_B)$, respectively ($f^A_v(R_A, R_B) + f^B_v(R_A, R_B) \leq 1$). We only require the functions $f^A_v(\cdot,\cdot)$, for every $v \in V$, to be monotone and submodular with respect to the set of initial A nodes.

The following theorem shows that one can efficiently find an approximation for the set that maximizes the spread of one's own technology, given a competitor[s] choice of initial adopters:

Theorem 9. *Given technology B's early adopters I_B, one can find an $(1 - e^{-1} - \epsilon)$-approximation for the competitive OR process in a polynomial number of steps, for any $\epsilon > 0$.*

The proof follows immediately from the following two lemmas which prove the properties required in [4]. We will show that this process is monotone and submodular whenever the function $f_v(\cdot,\cdot)$ is monotone and submodular w.r.t. technology A, for all $v \in V$.

Lemma 1. *The OR model is monotone with respect to the number of nodes influenced by technology A.*

Proof. Let $r_A(I_A), r_B(I_B)$ denote an outcome for a run of the independent propagation processes of the two technologies. Monotonicity w.r.t technology A is satisified if for any two sets $S \subseteq T \subseteq V - I_B$:

$$\mathbb{E}[f^A_v(r_A(S), r_B(I_B))] \leq \mathbb{E}[f^A_v(r_A(T), r_B(I_B))] \tag{2}$$

Since until the decision step the two technologies' propagations are independent, we can fix the outcome of technology B, and show that the expected propagation of technology A is monotone. This is immediate since first, the propagation of technology A until the decision step is clearly monotone (follows from the non-competitive threshold model in [6]). Second, the decision functions $f^A_v(\cdot,\cdot)$ and $f^B_v(\cdot,\cdot)$ are monotone with respect to technologies A and B, which along with the previous argument yields monotonicity.

Lemma 2. *The OR model is submodular with respect to the number of nodes influenced by technology A.*

Proof. In order to prove this, we will use a technique given in [6] that suggests an alternative and equivalent model for the propagation of a single technology. For each node $v \in V$, instead of choosing a threshold in $[0, 1]$, choose an incoming edge (u, v) with respective probability $w_{u,v}$, and no incoming edge with probability $\sum_u w_{u,v}$. A node will become infected if and only if there is a path from the initially infected nodes that consists strictly of such chosen edges.

Fix an instantiation R_B of the outcome of the propagation of technology B (independent of the propagation of technology A) and a set of chosen edges for the propagation process of technology A. For a set I_A of initial A nodes, as before, let $R_A(I_A)$ denote the set of nodes reachable from I_A in the sub-graph induced by the set of chosen edges. In order to show that the process is submodular, we need to show that for all $S \subseteq T \subseteq V - I_B$:

$$f_v^A(R_A(S \cup \{x\}), R_B) - f_v^A(R_A(S), R_B) \geq f_v^A(R_A(T \cup \{x\}), R_B) - f_v^A(R_A(T), R_B), \tag{3}$$

for all $v \in V$. We will simply use the monotonicity property of the independent propagation process and the submodularity of $f_v^A(\cdot, \cdot)$. Let $R_A(S \cup \{x\}) = R_A(S) \cup \Delta_1$, and similarly, $R_A(T \cup \{x\}) = R_A(T) \cup \Delta_2$. From the monotonicity and submodularity we get that $R_A(S) \subseteq R_A(T)$ and $\Delta_2 \subseteq \Delta_1$. Therefore:

$$\begin{aligned} & f_v^A(R_A(S) \cup \Delta_1, R_B) - f_v^A(R_A(S), R_B) \\ \geq\ & f_v^A(R_A(T) \cup \Delta_1, R_B) - f_v^A(R_A(T), R_B) \\ \geq\ & f_v^A(R_A(T) \cup \Delta_2, R_B) - f_v^A(R_A(T), R_B) \end{aligned} \tag{4}$$

The first inequality and second inequality follow from the submodularity and the monotonicity of $f_v^A(\cdot, \cdot)$, respectively. Taking all possible instantiations gives submodularity since a positive linear combination of submodular functions is submodular.

Mossel et al. [8] show that if we generalize the propagation process by replacing the linear sum (used to decide whether an uninfected node exceeds its threshold) with an arbitrary monotone submodular function, then the resulting process (under any monotone submodular objective function) is again monotone and submodular. This result generalizes to the corresponding competitive process, which we call the *generalized OR process.*

Theorem 10. *Given technology B's early adopters* I_B*, one can find an* $(1 - e^{-1} - \epsilon)$*-approximation for the generalized competitive OR process in a polynomial number of steps, for any* $\epsilon > 0$.

Proof. Use the objective function $\varphi(R_A) = \mathbb{E}_{R_B} \sum_v f_v^A(R_A, R_B)$ in the main result of [8]. The function φ counts the expected number of A-adopters at the end of the process. It is monotone and submodular because the f_v^A are.

6.1 Repeating OR Processes

Finally, we give a natural extension of the *OR* model. There are cases in which the independent propagation process will repeat several times (e.g. every day, for ℓ days). The process can be thought of as being run iteratively, where during each iteration i the previous iteration's turnouts R_A and R_B are used as the initial adopters for each technology. At the end of the ℓ'th iteration, *and only then*, the decision step is invoked by using the functions $f_v^A(\cdot,\cdot)$ and $f_v^B(\cdot,\cdot)$, for all $v \in V$. One may notice that this formulation simply defines a composition of ℓ *OR* processes (with a single execution of the decision step at the end).

We can give a natural motivation for such a process: during the course of an election race, voters will spread the word each day. However, once in while, an unaffected voter may change her mind (her threshold value) and thus the process of "rumor spread" and social-based recommendation will run again, infecting additional voters as a result.

With this in mind, the following general theorem follows from a simple generalization of the proof in [8].

Theorem 11. *A process based on the repetitive execution of the generalized OR process with a single decision step at the end is monotone and submodular.*

Note that theorem 11 holds even if the edge weights are modified between each iteration.

7 Conclusions

We have presented a number of fairly natural and general approaches for modelling competitive diffusion of influence in a social network, extending the known threshold model for the spread of a single technology. However, most of our suggested approaches have been shown to be unfit for the Nemhauser et al. [4] approximation technique. For some models, we can show NP-hardness of approximation, while for others we only show that they are not submodular (and not even monotone in one case), leaving open the question to whether an efficient approximation algorithm can be found.

We emphasize that all of the suggested models in this paper have reasonable, natural motivations, which implies that there is no single "true" model. Also, as suggested in [2], we believe that these models can be used in a more game theoretic setting, where players are the competing companies, who place bids on strategic nodes in hope for maximizing their outcome. We suggest the following directions for future research:

– Can the hardness-of-approximation result be extended to other models?
– Are there any other natural competitive models which are approximable in polynomial time?
– Study some natural game-theoretic setting for the competitive models.
– Suggest models for cases where nodes may adopt more than one technology.

References

1. Bharathi, S., Kempe, D., Salek, M.: Competitive influence maximization in social networks. In: WINE 2007. LNCS, vol. 4858, pp. 306–311. Springer, Heidelberg (2007)
2. Carnes, T., Nagarajan, C., Wild, S.M., van Zuylen, A.: Maximizing influence in a competitive social network: a follower's perspective. In: ICEC 2007: Proceedings of the Ninth International Conference on Electronic Commerce, pp. 351–360. ACM, New York (2007)
3. Domingos, P., Richardson, M.: Mining the network value of customers. In: KDD 2001: Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 57–66. ACM Press, New York (2001)
4. Nemhauser, M.F.G., Wolsey, L.: An analysis of the approximations for maximizing submodular set functions. Mathematical Programming, 14, 265–294 (1978)
5. Brown, P.R.J.: Social ties and word-of-mouth referral behavior. Journal of Consumer Research 14(3), 350–362 (1987)
6. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: KDD 2003: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146. ACM, New York (2003)
7. Kostka, J., Oswald, Y.A., Wattenhofer, R.: Word of mouth: Rumor dissemination in social networks. In: Shvartsman, A.A., Felber, P. (eds.) SIROCCO 2008. LNCS, vol. 5058, pp. 185–196. Springer, Heidelberg (2008)
8. Mossel, E., Roch, S.: Submodularity of influence in social networks: From local to global. SIAM J. Comput. 39(6), 2176–2188 (2010)

A Counter Examples for the Weight-Proportional Competitive Linear Threshold Model

In section 2 we gave two theorems concerning the monotonicity and submodularity of the model described. These theorems will be proven in this appendix.

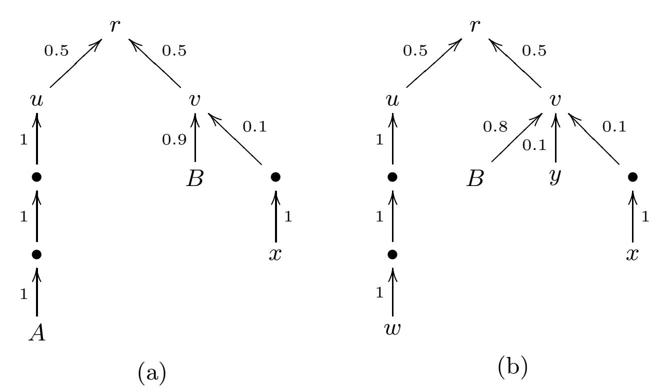


Fig. 1. Counter-examples for (a) monotonicity (b) submodularity

Theorem 12. *There exists an instance of the weight-proportional competitive linear threshold problem for which monotonicity does not hold.*

Proof. Consider the tree in figure 1(a). One can verify that $\alpha_r^4 = \frac{11}{40}$, whereas $\hat{\alpha}_r^4 = \frac{103}{400}$, which violates monotonicity.

Theorem 13. *There exists a graph G, for which the expected influence of technology A is not submodular.*

Proof. Consider the tree depicted in figure 1(b). It can be easily shown that for $S = \{w\}, T = \{w, y\}$ (the set of early adopters of technology B is denoted in the diagram) submodularity does not hold as $\alpha_r^4(S) = \frac{3}{10}, \hat{\alpha}_r^4(S) = \frac{17}{60}, \alpha_r^4(T) = \frac{7}{10}, \hat{\alpha}_r^4(T) = \frac{17}{50}$.

B Proof of Monotonicity of Separated Threshold Model

Theorem 14. *For a given instance of the problem and a choice of early adopters: I_A, I_B and node x, $\hat{\alpha}_v^t \geq \alpha_v^t$ for any node v and for any step $t \geq 0$.*

In order to prove the theorem, we will fix the set I_B of early technology B adopters and consider a set of early technology A adopters I_A and a node x not in I_B.

We prove the monotonicity by fixing an arbitrary instantiation of the thresholds, and by choosing for every node technology A or B with equal probability; these choices will be revealed in cases where the two thresholds chosen for a particular node are exceeded simultaneously. Notice that this defines a deterministic instantiation of the process.

Denote by π_1, π_2 the deterministic processes using the same instantiations of the threshold values and coin-flips, and using (I_A, I_B) and $(I_A \cup \{x\}, I_B)$, respectively. Furthermore, let $N_A^t(\pi), N_B^t(\pi)$ denote the set of A and B active nodes at step t in process π, respectively. The following lemma implies theorem 3.

Lemma 3. *The following holds for each node $v \in V$ and every step $t \geq 0$:*

1. *If v is not B-active at step t in π_1, then it isn't B-active at any step $t' \leq t$ in π_2.*
2. *If v is A-active at step t of π_1, then v is activated in some step $t' \leq t$ in π_2.*

Proof. The straightforward proof by induction is omitted for lack of space. It can be found in the full version.

C Counter-Examples for the Competitive Threshold Model with Forcing

Theorem 15. *For any tie-breaking rule, and any forcing rule, the separated-threshold competitive model is non-submodular.*

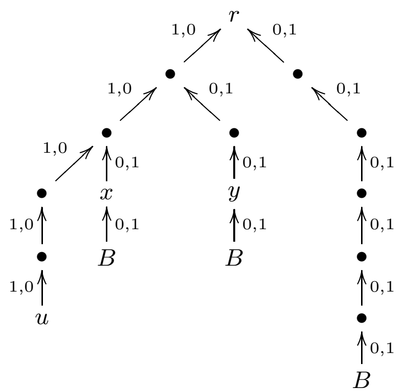


Fig. 2. A Counter-example for submodularity when applying a forcing step

Proof. We will give a counter-example in which there are no ties, and the node in question does not remain inactive. Consider the rooted tree in figure 2. Let $S = \{u\}, T = \{u, y\}$. The initially B-activated nodes are given in the diagram.

One can check that there are no ties, and that forcing never applies to the root. Also, when using S, $\alpha_r^6 = \hat{\alpha}_r^6 = 0$. On the other hand, when using T, we get $\alpha_r^6(T) = 0$ and $\hat{\alpha}_r^6(T) = 1$, contradicting submodularity.

D Proof of the Hardness of Approximation Result

Theorem 16. *It is NP-hard to approximate the Seperated-Threshold Competitive Influence problem with a ratio better than $\Omega(N^{\frac{1}{2}-\epsilon})$, for all $\epsilon > 0$, where N is the number of nodes in the graph.*

Proof. We are motivated by the counter-example in theorem 6, constructing a reduction from Vertex Cover.

The reduction. We are given an instance of Vertex Cover, a graph $G = (V, E)$ and a number k. Let α, β be constants defined later. Our new graph contains a special vertex A_0, a vertex A_v for each node $v \in V$, n^α vertices $B_0^{e,t}, X_0^{e,t}, X_1^{e,t}, M^{e,t}$ for each edge $e \in E$, and an extra n^α vertices B_1^t, P_0^t, P_1^t; here $1 \leq t \leq n^\alpha$. The rest of the graph appears in figure 3, where

- Dotted edges have A-weight 1 and B-weight 0.
- Dashed edges have A-weight 0 and B-weight 1.
- Plain edges have both weights set to 1.
- Edges with a length annotation are paths of that length of the given type.

Finally, I_B is composed of the set of nodes $B_0^{e,t}$ and B_1^t for every $e \in E$ and t.

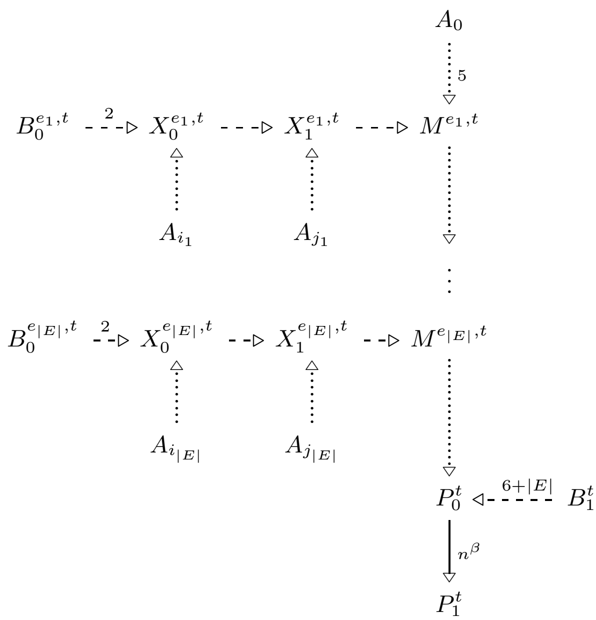


Fig. 3. The reduction. Diagram repeated for each t (except for the A_x).

Claim. If there exists a k-cover for the original graph G there exists a set $I_A \subseteq V - I_B$ of size $k+1$ that yields $\sigma(I_A) \geq n^{\alpha+\beta}$. Otherwise, for every $I_A \subseteq V - I_B$, $\sigma(I_A) = O(\max\{n^{\alpha+3}, n^{\beta+1}\})$.

Proof. Assume first that there is a k-cover S for G. Let $I_A = \{A_v | v \in S\} \cup \{A_0\}$. Since S is a vertex-cover, the spread of technology B emanating from the vertices B_0^t is completely blocked. Thus, every node on the path from A_0 to P_0^t, for all t, will be A-infected. Hence every node on the path from P_0^t to P_1^t will be A-infected. Thus, we have at least $n^{\alpha+\beta}$ A-active nodes, as required.

For the second part of the claim, for any set I_A of $k+1$ initial A-adopters, either $A_0 \notin I_A$ or $I_A \cap \{A_v\}_{v \in V}$ is not a vertex cover. Therefore the best choices for vertices in I_A are: choosing A_0, which contributes at most $(|E|+5)n^\alpha$ nodes; and choosing P_0^t, which contributes n^β nodes. The contribution of vertices of the first type is at most $O(s \cdot n^{\alpha+2}) = O(n^{\alpha+3})$, and the vertices of the second type contribute at most $O(s \cdot n^\beta) = O(n^{\beta+1})$.

Set $\beta = \alpha + 2$. The total number of vertices in the reduced graph is $N = O(n^{\alpha+\beta} + |E| \cdot n^\alpha) = O(n^{2\alpha+2})$. Thus we get that if there is a k-cover for G then the optimal I_A yields $\sigma(I_A) = \Omega(N)$, whereas any I_A that does not correspond to a k-cover yields $\sigma(I_A) = O(N^{(\alpha+3)/(2\alpha+2)})$. Hence, any algorithm that gives an approximation ratio of $o(N^{1-(\alpha+3)/(2\alpha+2)})$ can solve the NP-complete vertex cover problem. Therefore the approximation ratio of any poly-time algorithm is $\Omega(N^{1/2-\epsilon})$, for all $\epsilon > 0$, unless $P = NP$.

Course Allocation by Proxy Auction*

Scott Duke Kominers[1,**], Mike Ruberry[2,***], and Jonathan Ullman[2,†]

[1] Department of Economics, Harvard University, and Harvard Business School
skominers@hbs.edu

[2] Harvard University School of Engineering and Applied Sciences
{mruberry,jullman}@seas.harvard.edu

Abstract. We propose a new *proxy bidding mechanism* to allocate courses to students given students' reported preferences. Our mechanism is motivated by a specific strategic *downgrading* manipulation observed in the course allocation mechanism currently used at Harvard Business School (HBS). The proxy bidding mechanism simplifes students' decisions by directly incorporating downgrading into the mechanism.

Our proxy bidding mechanism is Pareto efficient with respect to lexicographic preferences and can be extended to allow for a more expressive preference language which allows complementarity, substitutability, and indifference. Simulations suggest that the proxy bidding mechanism is robust to the manipulations observed in the HBS mechanism and may yield welfare improvements.

1 Introduction

Course allocation is a *combinatorial assignment problem* that assigns students to courses, given students' preferences over course schedules. Unfortunately, any strategyproof and efficient mechanism for this problem must be dictatorial ([6]), with poor outcomes ([3]).[1]

One course allocation system is the *Bidding Points mechanism*, in which students "bid" for courses using an artificial currency ([4] [7]). Although these mechanisms are commonly used in practice, they require strategic play by students and have meager welfare guarantees.

An alternative course allocation mechanism is the *draft*, in which students take turns selecting individual courses from those with available seats, following in a "draft order." Such a mechanism is used by Harvard Business School

* We thank Susan Athey, Eric Budish, Yiling Chen, John William Hatfield, David Parkes, and Alvin Roth for helpful conversations. We thank Utku Ünver for graciously providing us with data.

** Supported by an NSF Graduate Research Fellowship and a Yahoo! Key Scientific Challenges Program Fellowship.

*** Corresponding author.

† Supported by NSF grant CNS-0831289.

[1] It is known that on restricted preference domains, better solutions may be achieved. For example, under unit-demand (the *school choice problem*), non-dictatorial strategyproof mechanisms are known (see [1] [5]).

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 551–558, 2010.

(HBS). The draft mechanism offers strong welfare guarantees under truthful play, but is easily manipulable: Budish and Cantillon [3] demonstrated that students successfully and substantially manipulate the HBS draft, to the benefit of sophisticated students at the expense of social welfare.[2]

We reinterpret the HBS draft as a game amongst naïve proxy agents who act on behalf of students. We develop a *proxy bidding mechanism* for course assignment in which students' proxies pick strategically, unlike in the draft, obviating the need for certain types of manipulations. Our proxies' behaviors are inspired by the desire to have a mechanism play optimal draft strategies on students' behalves.

We prove that the proxy bidding mechanism is Pareto efficient when students have lexicographic preferences over schedules. We present simulations that show that the proxy bidding mechanism performs favorably relative to the HBS draft in the presence of the manipulations identified by Budish and Cantillon [3]. Finally, we extend our mechanism to allow a bidding language in which complementarity, substitutability and indifference can be expressed. For brevity, most proofs are deferred to the Appendix, which is available on the authors' websites.

2 The Proxy Bidding Mechanism

Our proxy bidding mechanism provides a course allocation for a problem consisting of a set of courses $\mathcal{C}$, a vector of course capacities $(q_c)_{c\in\mathcal{C}}$, a set of players $\mathcal{N}$, as well as a set of strict ordinal preferences over courses, $(\prec_i)_{i\in\mathcal{N}}$, specified by the players.

The mechanism takes as input a set of bidding priorities B_i for each player i; these will typically be allocated randomly for reasons of fairness. The bid sets $\{B_i\}_{i\in\mathcal{N}}$ form a partition of a finite global bid set $\mathcal{B} \subset \mathbb{R}$. The maximum number of courses a player can be allocated is $|B_i|$. In principle the choice of bid sets may be arbitrary, but for our purposes we only consider bids in correspondence with turns of the HBS draft, choosing

$$B_i = \{-i, -(2\mathcal{N} - i), -(2\mathcal{N} + i + 1), \ldots\}.$$

The mechanism maintains a set of multi-unit auctions $(A_c)_{c\in\mathcal{C}}$ (interpreted as sets of winning bids, for convenience) for course seats.

Note that agents' bids are indivisible. We think of the set $\mathcal{B}$ as representing a sequence of course selection opportunities; the set B_i is the set of opportunities in $\mathcal{B}$ at which i may select a course. In this notation the HBS draft is the mechanism in which students choose courses in the sequence $\mathcal{B}$, with each student i selecting a course at each opportunity $b \in B_i$.

---

[2] In a separate work, Budish [2] introduces the effective but complex *Approximate Competitive Equilibrium from Equal Incomes* mechanism which ameliorates many of the issues discussed above.

Our proxy bidding mechanism proceeds in rounds, with players' bids being reallocated in response to the current profile of course "prices" $(p(A_c))_{c\in\mathcal{C}}$, where[3]

$$p(A_c) := \begin{cases} 0 & |A_c| < q_c, \\ \min A_c & |A_c| = q_c. \end{cases}$$

The formal proxy bidding mechanism specification is given in Algorithm 1. We take this opportunity to explain where our mechanism diverges from the draft mechanism. The inner loop of our mechanism (lines 9- 29) allows players' proxies to place the *lowest sufficient bid* into an auction to avoid overpaying for a course. In the language of the draft mechanism, overpaying for a course means selecting that course too early. If we were to replace this inner loop with a simpler procedure that places the *maximum available bid* into each successful course, then we would recover something closer to the draft mechanism.

We now give an example of the proxy bidding mechanism on simple input. Consider the following input: $\mathcal{N} = \{1, 2, 3\}$, $\mathcal{C} = \{c_1, c_2, c_3\}$, $q_{c_1} = q_{c_2} = q_{c_3} = 2$,

$$\begin{aligned} c_3 \prec_1 c_2 \prec_1 c_1, &\quad B_1 = \{6, 1\}, \\ c_3 \prec_2 c_2 \prec_2 c_1, &\quad B_2 = \{5, 2\}, \\ c_1 \prec_3 c_3 \prec_3 c_2, &\quad B_3 = \{4, 3\}. \end{aligned}$$

The mechanism will run as illustrated below.[4]

Round	i	State
1	1	$A_{c_1} = \{1\}, A_{c_2} = \{1\}, A_{c_3} = \emptyset$
2	2	$A_{c_1} = \{2, 1\}, A_{c_2} = \{2, 1\}, A_{c_3} = \emptyset$
3	3	$A_{c_1} = \{2, 1\}, A_{c_2} = \{3, 2\}, A_{c_3} = \{3\}$
4	1	$A_{c_1} = \{2, 1\}, A_{c_2} = \{6, 3\}, A_{c_3} = \{3\}$
5	2	$A_{c_1} = \{2, 1\}, A_{c_2} = \{6, 5\}, A_{c_3} = \{3\}$
6	3	$A_{c_1} = \{3, 2\}, A_{c_2} = \{6, 5\}, A_{c_3} = \{3\}$
7	1	$A_{c_1} = \{6, 3\}, A_{c_2} = \{5, 1\}, A_{c_3} = \{3\}$
8	2	$A_{c_1} = \{6, 5\}, A_{c_2} = \{2, 1\}, A_{c_3} = \{3\}$
9	3	$A_{c_1} = \{6, 5\}, A_{c_2} = \{3, 2\}, A_{c_3} = \{3\}$
10	1	$A_{c_1} = \{6, 5\}, A_{c_2} = \{3, 2\}, A_{c_3} = \{3, 1\}$

In the first round, the price of every course is 0, so player 1 bids for his two most-preferred courses. Notice that he uses his *lowest sufficient bid* in both auctions but that his biding is consistent with his actual bid set. That is, he only needs to bid 1 to win each course, but *could* win both courses as he has bids of 6 and 1 available. In the third round, player 3 sees that his most-preferred course has a price of 1 and his second most-preferred course has a price of 0; he therefore

[3] Our auction procedure has a cleanup step to guarantee that $0 \le |A_c| \le q_c$ for every $c \in \mathcal{C}$, hence it is only necessary to define the price function on this range.

[4] Each row of the right column represents the states of the auctions *after* player i has had a chance to update his bids.

bids 3 on each. But then, player 1 no longer has a winning bid in A_{c_2}. In the fourth round, player 1 observes that the price of c_2 is now 2, so he bids 6 in A_{c_2}. The mechanism continues in this fashion until it terminates six rounds later.[5] The final allocation C is given by $C_1 = \{c_1, c_3\}$, $C_2 = \{c_1, c_2\}$, $C_3 = \{c_2, c_3\}$.

Our mechanism preserves some of the draft's positive properties. In particular our simultaneous auction is guaranteed to converge and results in a Pareto efficient allocation.

Proposition 1 (Convergence). *Algorithm 1 terminates.*

If bids fail to converge there must be a "cycle," that is, a series of auction states which is repeated over the course of the mechanism. Since bids are discrete and unique, there is for any such cycle a highest bid b^* cast in any stage of the bid cycle. Moreover, at some point in the cycle the student $i \in \mathcal{N}$ holding bid b^* must cast b^* for some course c which i ranks most highly among all courses i bids for during the course of the cycle. But since b^* is maximal among all cast during the cycle, it cannot be displaced once made in the course-c auction A_c; this would contradicts the involvement of b^* in the cycle. Note that this argument directly uses the fact that the bidding mechanism disallows the withdrawl of undisplaced bids.

Proposition 2 (Pareto Efficiency). *The allocation produced by Algorithm 1 is Pareto efficient with respect to the lexicographic preferences induced by input preferences* $(\prec_i)_{i \in \mathcal{N}}$.

For any Pareto inefficient allocation, there is (by definition) a sequence of course trades which consitute a Pareto improvement. Of the students participating in these trades, one student i^* buys the course c_{i^*} he wishes to trade away with the largest bid $b_{c_{i^*}}$ used to buy a course in the Pareto-improving trade. But under the bidding mechanism, i^* should instead have bid $b_{c_{i^*}}$ in the course for which i seeks to trade. Thus, the outcome of the proxy bidding mechanism must be Pareto efficient.

3 Welfare Properties

In this section we use simulations to analyze the welfare properties of the proxy bidding mechanism. Our simulation environment produces correlated preferences for 1000 students over a set of 110 courses. Each student demands 10 courses and each course offers 100 seats. This problem is roughly the size of that of Harvard Business School. Full details of our simulation environment are deferred to the Appendix.

We use two welfare measures averaged over the population to evaluate assignments: *average rank*, the average preference rank of the courses allocated to each student; *lexicographic rank*, the highest-ranked course received. Note that

[5] Technically, the auction will not terminate until all players have declined the chance to change their bids.

Algorithm 1
Input: $\mathcal{C}$, $(q_c)_{c\in\mathcal{C}}$, $\mathcal{N}$, $(\prec_i)_{i\in\mathcal{N}}$, $\{B_i\}_{i\in\mathcal{N}}$
Output: $(C_i)_{i\in\mathcal{N}}$

```
for c ∈ C do
    A_c := ∅
end for

repeat
    active := false
    for i ∈ N do
        B'_i = ∅
        for c in order of ≺_i do
            if B_i ∩ A_c = ∅ then
                b* := min_{b∈B_i\B'_i} {b > p(A_c)}
                if b* exists then
                    add b* to B'_i
                    add min_{b∈B_i} {b > p(A_c)} to A_c
                    if |A_c| > q_c then
                        remove min_{b∈A_c} from A_c
                    end if
                    active := true
                end if
            else
                b* := B_i ∩ A_c
                b** := min_{b∈B_i\B'_i} {b ≥ b*}
                if b** exists then
                    add b** to B'_i
                else
                    remove b* from A_c
                end if
            end if
        end for
    end for
until active = false

for i ∈ N do
    C_i := {c | B_i ∩ A_c ≠ ∅}
end for
```

since we interpret preferences as rank-order lists from 0-th to $(|B_i|-1)$-st in the example presented in the previous section, the average rank of the allocation is 2/3 and the lexicographic rank is 0.[6]

[6] In the example, one student received his first- and third-most-preferred courses and two received their first- and second-most-preferred courses. Considering the most-preferred course to have rank 0, the lexicographic rank is 0 (all students received their most-preferred courses). The average rank is $((0+2)/2+(0+1)/2+(0+1)/2)/3 = 2/3$.

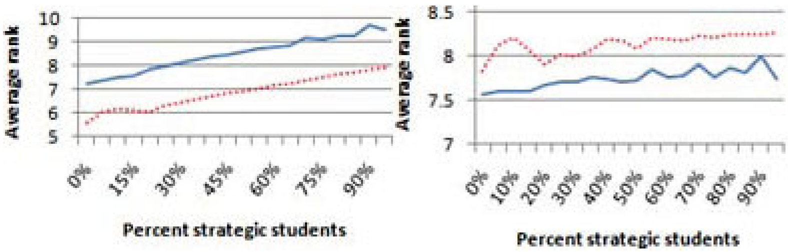


Fig. 1. A simulation of downgrading in the HBS draft (left) and proxy bidding (right) mechanisms. The solid and dotted lines respectively represent the average welfare levels of truthful and downgrading students.

3.1 Comparison with the HBS Draft

First we demonstrate that the HBS draft mechanism is susceptible to strategic play in our environment by simulating the *downgrading* manipulation highlighted by Budish and Cantillon [3], in which (some) students partially reorder their preferences in correspondence with course popularity levels. In Figure 1, average rank outcomes are plotted for both the HBS draft and the proxy bidding mechanism. The dotted and solid line respectively plot the outcome to strategic and non-strategic students. In Figure 1 (and in Figure 1, below), the horizontal axis indicates the fraction of students playing the manipulative strategy and the vertical axis indicates the outcome.

As expected, students who play the downgrading manipulation in the HBS draft receive substantially lower-ranked courses on average than students who report their preferences straightforwardly. In the proxy bidding mechanism, however, the opposite result obtains, demonstrating resilience of the proxy bidding mechanism to this manipulation. These results hold regardless of the fraction of strategic students.

Figure 2 plots cross-population welfare statistics in the presence of downgrading by part of the population. The solid line charts the performance of the proxy bidding mechanism; the dashed line charts the HBS draft. While both populations' welfare decreases as more students manipulate and at any point the welfare of proxy bidding is lower, the downgrading manipulation only benefits students in the draft, as shown in Figure 1. Once over 30% of students play downgrading strategies, average welfare in the HBS draft is worse than would be achieved in the proxy bidding mechanism under truthful play.

3.2 Strategic Play in the Proxy Bidding Mechanism

Although our proxy bidding mechanism is apparently robust to the downgrading manipulation, it is not strategyproof.

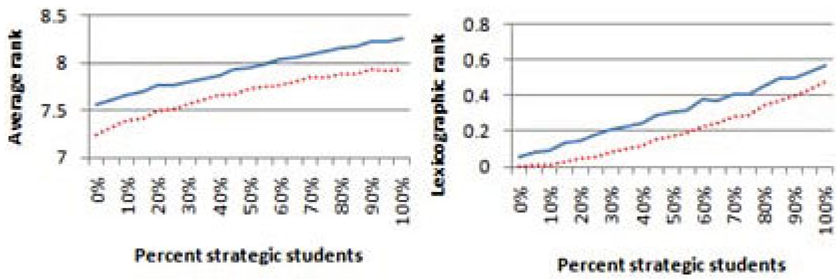


Fig. 2. Simulated average welfare levels of the entire population when part of the population plays the downgrading strategy. The dotted line is the HBS draft and the solid line proxy bidding. While at any point on the curve proxy bidding lags the draft, its resilience to this manipulation, as seen in Figure 1, suggests it will be played truthfully

Consider the following input: $\mathcal{N} = \{1,2\}$, $\mathcal{C} = \{c_1, c_2, c_3, c_4\}$, $q_{c_1} = q_{c_2} = q_{c_3} = q_{c_4} = 1$,

$$c_4 \prec_1 c_3 \prec_1 c_2 \prec_1 c_1, \quad B_1 = \{4,1\},$$
$$c_1 \prec_2 c_4 \prec_2 c_3 \prec_2 c_2, \quad B_2 = \{3,2\}.$$

The final allocation will be C given by $C_1 = \{c_1, c_2\}$, $C_2 = \{c_3, c_4\}$. If player 2 reports the preferences $c_4 \prec'_2 c_3 \prec'_2 c_2 \prec'_2 c_1$, then the final allocation C' given by $C'_1 = \{c_1, c_4\}$, $C'_2 = \{c_2, c_3\}$ is obtained. Thus a player with preferences $\prec_2$ receives a more-preferred allocation by reporting $\prec'_2$ than by reporting honestly.

4 Extended Preference Support

Lexicographic preferences over single courses provide an elegant model and allow quick computation but are unlikely to represent students' true preferences. To partially address this problem, we extend the input space of the proxy bidding mechanism to allow players to express conditional demand for courses. Specifically, we introduce `ANY`, `IF`, and `NOTIF` statements, which we illustrate in Figure 3 and define formally in the Appendix.

The `ANY` statement is an exclusive-or over a set of courses. This feature allows players to indicate indifference over a set of courses, such as identical sections of a course, or equally-preferred courses that meet at the same time. Such a feature seems especially important, as in current bidding systems it is sometimes possible ex-post for students to obtain a section of every course offered.[7] The `IF` (respectively, `NOTIF`) statement is a conditional which allows a player to demand a course c if he holds (respectively, does not hold) a more-preferred course c'.

[7] This ironic situation occurs as an outcome of the University of Michigan bidding points mechanism, which has been studied by Krishna and Unver [4].

Order	Modifier	Course name
0		Modern Art
1	`IF` Modern Art	Modern Art Criticism
2	`NOTIF` Modern Art Criticism	Renaissance Art
3	`ANY`	{Ancient History, Modern History Classical History}

Fig. 3. A student's extended preferences. The student's most-preferred courses are Modern Art and Modern Art Criticism, but she can only take the latter if she takes the former. If she does not receive Modern Art Criticism, she would like Renaissance Art instead. She also must take one art history course but is indifferent between three choices.

Our proxy bidding mechanism suitably extends to accommodate this more-expressive preference language, however the outcome produced may not be Pareto efficient if `ANY` statements are used (see the Appendix).

5 Conclusion and Future Work

We have introduced a new proxy bidding mechanism for course allocation which offers attractive welfare possibilities. It is relatively simple and extends naturally to allow for a more expressive preference language than is typically used in course allocation. Although our mechanism is resistant to the strategic manipulations that have been observed in the HBS draft, a full analysis of strategic play under the proxy bidding mechanism requires further study.

It seems likely that our approach of replacing agents by proxies who strategize on the agents' behalves would find applications in other domains. A general theory of proxy mechanisms seems appropriate for future work.

References

1. Abdulkadiroglu, A., Pathak, P.A., Roth, A.E., Sonmez, T.: Changing the Boston school choice mechanism: Strategy-proofness as equal access (2006)
2. Budish, E.: The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes (2010)
3. Budish, E., Cantillon, E.: Strategic behavior in multi-unit assignment problems: Lessons for market design (2009)
4. Krishna, A., Unver, M.U.: Improving the efficiency of course bidding at business schools: Field and laboratory studies. Marketing Science 27(262), 262–282 (2008)
5. Pápai, S.: Strategyproof assignment by hierarchical exchange. Econometrica 68(6), 1403–1433 (2000)
6. Pápai, S.: Strategyproof and nonbossy multiple assignments. Journal of Public Economic Theory 3(3), 257–271 (2001)
7. Sonmez, T., Unver, U.: Course bidding at business schools. International Economic Review 51(1), 99–123 (2010)

False-Name-Proofness in Facility Location Problem on the Real Line

Taiki Todo, Atsushi Iwasaki, and Makoto Yokoo

Department of Informatics,
Kyushu University
Fukuoka, Japan
todo@agent.is.kyushu-u.ac.jp, iwasaki@is.kyushu-u.ac.jp,
yokoo@is.kyushu-u.ac.jp

Abstract. Recently, mechanism design without monetary transfers is attracting much attention, since in many application domains on Internet, introducing monetary transfers is impossible or undesirable. Mechanism design studies how to design mechanisms that result in good outcomes even when agents strategically report their preferences. However, in highly anonymous settings such as the Internet, declaring preferences dishonestly is not the only way to manipulate the mechanism. Often, it is possible for an agent to pretend to be multiple agents, and submit multiple reports using different identifiers, e.g., different e-mail addresses. Such *false-name manipulations* are more likely to occur in a mechanism without monetary transfers, since submitting multiple reports would be less risky in such a mechanism. In this paper, we formalize false-name manipulations in facility location problems on the real line and discuss the effect of such manipulations.

1 Introduction

Facility location problems have traditionally been studied in economics and operations research. In facility location problems, a mechanism designer plans to locate facilities, while agents report their locations. A *facility location mechanism*, or a *social choice rule*, is a function that maps a reported location profile into the locations of facilities. The goal is to design facility location mechanisms that satisfy the well-known incentive property of *strategy-proofness*; for each agent, reporting his/her location truthfully is a dominant strategy regardless of the strategies of other agents.

In social choice theory, there have been a lot of works on facility location problems on the real line. Moulin [5] characterized strategy-proof facility location mechanisms on the real line under the natural assumptions of Pareto efficiency and anonymity. Furthermore, he proposed a *generalized median voter scheme* to characterize Pareto efficient and strategy-proof mechanisms on the real line. Schummer and Vohra [8] extended the generalized median voter scheme to facility location problems on any graphs.

Recently, facility location problems have also been discussed in the field of *mechanism design without money*, in which an mechanism designer plans to develop mechanisms that do not involve monetary transfers. In several domains such as the Internet, introducing monetary transfers is impossible or undesirable, mainly due to security/banking or

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 559–562, 2010.

ethical/legal issues. Thus, mechanism design without money has attracted considerable attention of computer scientists.

Procaccia and Tennenholtz [7] presented a case study in approximate mechanism design without money and established tight bounds for the approximation ratio achieved by strategy-proof facility location mechanisms on the real line. They also proposed two extension of facility location problems: a domain where two facilities must be located and a domain where each agent owns multiple locations. Lu et al. [4] improved the approximation ratio for the social cost in both domains.

However, in highly anonymous settings such as the Internet, reporting location insincerely is not the only way to manipulate a facility location mechanism. Often, it is possible for an agent to pretend to be multiple agents and report multiple locations using different identifiers, e.g., by creating different e-mail accounts. Such *false-name manipulations* are more likely to occur in a mechanism without monetary transfers, since submitting multiple reports is less risky in such a mechanism. To the best of our knowledge, there has been virtually no work on *false-name-proofness* in facility location problems.

False-name manipulations have also been widely studied in combinatorial auctions. Yokoo et al. [9] pointed out the effects of false-name manipulations in combinatorial auctions and showed that the Vickrey-Clarke-Groves (VCG) mechanism is vulnerable against false-name manipulations. Besides combinatorial auctions, false-name-proofness and its relatives have been discussed in other mechanism design fields, such as voting [3], coalitional games [1], and cost sharing [6]. In particular, Conitzer [3] proposed an extended property called *anonymity-proofness* in voting and characterized anonymity-proof voting rules.

2 False-Name-Proofness

In this paper, we deal with a facility location problem in which a mechanism locates one facility on the real line. Let n denote the number of agents (identifiers) joining a mechanism and $N(|N| = n)$ the set of agents. Note that the number of agents n is defined to be variable in $\mathbb{N}$ to discuss the change of the number of agent joining a mechanism. Each agent $i \in N$ has a true location x_i on $\mathbb{R}$. The cost of an agent is defined by the distance between her true location and the location of a facility. If a facility is located at y, the cost of agent i who has a location x_i is $\mathrm{cost}(x_i, y) = |x_i - y|$. This cost function is a special case of *single-peaked preferences* [5].

A (direct revelation, deterministic) *facility location mechanism* is a function that maps a reported location profile $x = (x_1, \ldots, x_n)$ by the set of agents to a location of a facility y on the real line. A mechanism must locate a facility with respect to any number of agents n, since we consider an environment where each agent may use multiple identifiers. For this reason, we define a facility location mechanism f as a set of functions, where each function f^n is a mapping from a set of location profiles reported by n identifiers to the real line. We assume that a mechanism is *anonymous*; i.e., the obtained results are invariant under permutation of identifiers. With this assumption, we assume without loss of generality that $x_1 \leq x_2 \leq \ldots \leq x_n$.

Definition 1 (Facility Location Mechanism). *For any natural number* $n \in \mathbb{N}$*, a* facility location mechanism f *assigns an outcome* $f^n(x)$ *to any reported location profile* $x = (x_1, \ldots, x_n) : f = \{f^n | n \in \mathbb{N}, f^n : \mathbb{R}^n \to \mathbb{R}\}$

In facility location problems, each agent reports her location x'_i, which is not necessarily her true location x_i, to the mechanism. However, in a *strategy-proof* mechanism, it is guaranteed that each agent reports her true location x_i to the mechanism if she behaves to minimize her cost. A facility location mechanism f is said to be strategy-proof if $\forall n \in \mathbb{N}, \forall i \in N, \forall x_{-i}, \forall x_i, \forall x'_i, \text{cost}(x_i, f^n(x_i, x_{-i})) \leq \text{cost}(x_i, f^n(x'_i, x_{-i}))$. Here let x_{-i} denote the reported location profile by agents except i. That is, $f^n(x'_i, x_{-i})$ is the location of a facility when agent i reports x'_i and other agents report x_{-i}. In other words, strategy-proofness requires that for each agent, reporting her true location is a dominant strategy.

In the history of facility location problems, several strategy-proof facility location mechanisms have been developed. One of the well-known strategy-proof mechanism is the *median* mechanism, which locates a facility at the median location among the reported locations (if the number of agents n is even, locates at the $n/2$-th smallest location). Furthermore, it has been known that the median mechanism always locates a facility at the oprimal location with respect to the social cost [7].

Next we formalize false-name-proofness in the facility location problem. First, let us introduce some notations for discussing false-name manipulations. Let ϕ_i denote the set of identifiers used by agent i. This is also the private information of agent i. Let x_{ϕ_i} denote a location profile reported by a set of identifiers ϕ_i and $x_{-\phi_i}$ a location profile reported by identifiers except for ϕ_i. In this definition, x_{ϕ_i} is considered as a false-name manipulation by i.

Definition 2 (False-name-proofness). *A facility location mechanism* f *is* false-name-proof *if* $\forall n \in \mathbb{N}$*,* $\forall i \in N$*,* $\forall x_{-\phi_i}$*,* $\forall x_i$*,* $\forall \phi_i$*,* $\forall x_{\phi_i}$*, the following holds:*

$$\text{cost}(x_i, f^n(x_i, x_{-\phi_i})) \leq \text{cost}(x_i, f^{n-1+|\phi_i|}(x_{\phi_i}, x_{-\phi_i})) \quad (1)$$

In other words, a mechanism is false-name-proof if for each agent, reporting her true location by using a single identifier is a dominant strategy, although she can use multiple identifiers. This is an extension of strategy-proofness to open, anonymous environments such as the Internet. The following example shows that the median mechanism does not satisfy false-name-proofness; an agent can reduce her cost by using multiple identifiers.

Example 1. Consider the median mechanism and $N = \{1, 2, 3\}$. Assume that $x_1 = 1$, $x_2 = 2$, and $x_3 = 3$. If they report their locations truthfully, the mechanism locates a facility at 2. However, if agent 1 adds two false identifiers and reports $x_{\phi_1} = (1, 1, 1)$, the mechanism locates a facility at 1. By this false-name manipulation, agent 1 can strictly reduce her cost.

One trivial solution to prevent such false-name manipulations is to use the *leftmost* mechanism, which locates a facility at the smallest location among the reported locations. Obviously, the leftmost agent (agent 1 in Example 1) has no incentive to manipulate in the leftmost mechanism, since the facility is located at her true location. Furthermore, the other agents (agents 2 and 3 in Example 1) cannot move the location closer to their true locations by any false-name manipulations.

3 Future Works

In this paper, we first formalized false-name manipulations in the facility location problem on the real line. Now let us summarize the open questions about false-name manipulations in facility location problems.

– As we stated in this paper, the leftmost mechanism is false-name-proof, while the median mechanism is not. One open question is to characterize false-name-proof facility location mechanisms. More precisely, to obtain a necessary and sufficient condition for a facility location mechanism to be false-name-proof. Since false-name-proofness is a generalization of strategy-proofness, the condition must be a stronger condition than the necessary and sufficient condition for strategy-proofness proposed by Moulin [5].
– In facility location problems, designing mechanisms which achive good approximation ratios with respect to the optimal location is one of the important works, especially from the viewpoint of computer scientists. To the best of our knowledge, no bound of the approximation ratios has been obtained for false-name-proof facility location mechanisms. We would like to obtain a bound of approximation ratios and design an optimal mechanism which achieves the bound.
– In the literatures of social choice and mechanism design, several properties related to false-name-proofness have been introduced so far, e.g., anonymity-proofness, *group-strategyproofness*, *rename-proofness* [6], and *population monotonicity* [2]. It would also be interesting to clarify the connections between false-name-proofness and these properties.

References

1. Bachrach, Y., Elkind, E.: Divide and conquer: false-name manipulations in weighted voting games. In: Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 975–982 (2008)
2. Ching, S., Thomson, W.: Population-monotonic solutions in public good economies with single-peaked preferences. RCER Working Papers 362, University of Rochester - Center for Economic Research, RCER (1993)
3. Conitzer, V.: Anonymity-proof voting rules. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 295–306. Springer, Heidelberg (2008)
4. Lu, P., Wang, Y., Zhou, Y.: Tighter bounds for facility games. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 137–148. Springer, Heidelberg (2009)
5. Moulin, H.: On strategy-proofness and single peakedness. Public Choice 35(4), 437–455 (1980)
6. Penna, P., Schoppmann, F., Silvestri, R., Widmayer, P.: Pseudonyms in cost-sharing games. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 256–267. Springer, Heidelberg (2009)
7. Procaccia, A.D., Tennenholtz, M.: Approximate mechanism design without money. In: Proceedings of the Tenth ACM Conference on Electronic Commerce (EC), pp. 177–186 (2009)
8. Schummer, J., Vohra, R.V.: Strategy-proof location on a network. Journal of Economic Theory 104(2), 405–428 (2004)
9. Yokoo, M., Sakurai, Y., Matsubara, S.: Robust combinatorial auction protocol against false-name bids. Artificial Intelligence 130(2), 167–181 (2001)

Non-separable, Quasiconcave Utilities are Easy – in a Perfect Price Discrimination Market Model (Extended Abstract)*

Vijay V. Vazirani

Georgia Institute of Technology, Atlanta, GA 30332–0280
vazirani@cc.gatech.edu

Abstract. Recent results, establishing evidence of intractability for such restrictive utility functions as additively separable, piecewise-linear and concave, under both Fisher and Arrow-Debreu market models, have prompted the question of whether we have failed to capture some essential elements of real markets, which seem to do a good job of finding prices that maintain parity between supply and demand.

The main point of this paper is to show that even non-separable, quasiconcave utility functions can be handled efficiently in a suitably chosen, though natural, realistic and useful, market model; our model allows for perfect price discrimination. Our model supports unique equilibrium prices and, for the restriction to concave utilities, satisfies both welfare theorems.

1 Introduction

The celebrated Arrow-Debreu theorem [AD54], which establishes the existence of equilibria in a very general model of the economy, has been deemed to be "highly non-constructive" since it crucially uses Kakutani's fixed point theorem; as shown by Uzawa [Uza62], the existence of general equilibrium is equivalent to fixed point theorems. The conditions imposed on utility functions of buyers in the Arrow-Debreu theorem are very weak: continuity, quasiconcavity, and non-satiation.

Over the last decade, there has been a surge of interest within theoretical computer science on studying the question of efficient computability of market equilibria – not only to provide an algorithmic ratification of Adam Smith's "invisible hand of the market" but also because of potential applications to new markets on the Internet. This study started with highly restricted utility functions, i.e., linear [DPSV08, Jai04], and gradually moved to more general functions. However, in terms of positive results, this did not go very far – the most notable case being Fisher's model under Leontief utilities [CV04, Ye07]). Recently it was shown that computing equilibria under additively separable, piecewise-linear, concave utility functions (*plc utilities*) is PPAD-complete [CDDT09, CT09, VY10], thereby

* Research supported by NSF Grants CCF-0728640 and CCF-0914732, ONR Grant N000140910755, and a Google Research Grant.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 563–570, 2010.

dealing this program a serious blow – assuming P $\neq$ PPAD, this effectively rules out the existence of efficient algorithms for almost all general and interesting classes of "traditional" market models.

On the other hand, markets in the West, based on Adam Smith's free market principle, seem to do a good job of finding prices that maintain parity between supply and demand[1]. This has prompted the question of whether we have failed to capture some essential elements of real markets in our models, see [Vaz10]. Some progress has been made on this latter question: polynomial time algorithms were given for spending constraint utilities [Vaz10] and for plc utilities in the Fisher model, provided perfect price discrimination is introduced in the model [GV10]. Both these works deal with additively separable utility functions.

Clearly, the gap between these "positive" algorithmic results in the traditional market models and the generality of the Arrow-Debreu Theorem is rather large. The main point of this paper is to show that even non-separable, quasiconcave utility functions, with the additional restrictions of continuous differentiability and non-satiation, can be handled efficiently in a suitably chosen, though natural, realistic and useful, market model; our model allows for perfect price discrimination.

Additionally, our work provides insights into the widely used practice of price discrimination. [GV10] give an application of market model to online display advertising marketplaces. We note that extending their market model from separable to non-separable utilities makes it even more relevant to this application, since the utility to an advertiser from placing ads on multiple web sites would typically be an involved, non-separable function because the web sites may be substitutes, complements, etc.

1.1 Price Discrimination and Our Results

Most businesses today charge different prices from different consumers for essentially the same goods or services in order to maximize their revenues. This practice, called *price discrimination*, is not only good for businesses but also customers – without it, some customers will simply not be able to avail of certain goods or services. It is not only widespread [Var85] but is also essential for the survival of certain businesses, e.g., in the airline industry.

Price discrimination is particularly important in new industries, such as telecommunications and information services and digital goods. Traditional economic analysis, which assumes decreasing returns to scale on production, recommends pricing goods at marginal cost. However, this is not relevant to the new industries, since they have very high fixed costs and low marginal costs, and hence such prices will not even recover the fixed costs. In these situations, product differentiation and price discrimination are an important recourse. Motivated by these considerations, price discrimination has been extensively studied in economics from many different angles; see [WMT88, Var85, Var96, Sun04, Edl98, EEH94, BT04] for just a small sampling of papers on this topic.

[1] For example, in the West, it is hard to see a sight that was commonplace in the Soviet Union, with massive surpluses of some goods and empty shelves of others.

A monopolistic situation in which the business separates the market into individual consumers and charges each one prices that they are *willing and able to pay* is called *perfect price discrimination*, sometimes also called first degree price discrimination [Var96]. More formally, a consumer's marginal willing to pay is made equal to the marginal cost of the good. Of course, to do this, the business needs to have complete information about each consumer's preferences.

For the restriction to concave utilities, we give a convex program, a generalization of the classic Eisenberg-Gale convex program, that captures equilibrium for this model. For this case, we prove both welfare theorems.

For quasiconcave utilities, we give a nonlinear program that captures equilibria. Similar to the convex program mentioned above, an optimal solution to this program also satisfies KKT conditions; moreover, this program also lends itself to a polynomial time solution using the ellipsoid algorithm. For this case, the first welfare theorem holds but the second welfare theorem fails; an example with 2 buyers and one good is very easy to construct, and will be provided in the final paper.

2 The Market Model

Our market model is based on the Fisher setting and consists of a seller with a set G of divisible goods, a set B of buyers each with money and a middleman. Assume that $|G| = g$ and $|B| = n$, and the goods are numbered from 1 to g and the buyers are numbered from 1 to n. Let $m_i \in \mathbf{Q}^+$ dollars be the money of buyer i. For each buyer i we are specified a function $f_i : \mathbf{R}_+^g \rightarrow \mathbf{R}_+$ which gives the utility derived by i as a function of allocation of goods. We will assume that f_i is polynomial time computable.

The middleman buys goods from the seller, who charges the middleman in the usual manner, i.e., depending on the prices of goods and the amounts bought. However, in selling goods, the middleman charges buyers on the basis of the utility they accrue rather than the amount of goods they receive, i.e., he price-discriminates. The rate r_i at which buyer i should get utility per dollar charged from her, at any given prices $\boldsymbol{p}$, *is determined by buyer i herself.* Each buyer has no utility for money but wants to maximize the utility she accrues. The only restriction is that the middleman refuses to sell any part of a good at a loss – the fact that the middleman knows buyers' utility functions enables him to do this (we will specify in Section 2.1 what this restriction means mathematically). We show in Lemma 2 that under these circumstances, there is a rate, as a function of prices, at which buyer i is able to maximize her utility. This is also the rate at which each buyer's marginal willingness to pay equals the marginal prices of goods she gets, as required under perfect price discrimination. At this rate r_i, the total utility buyer i is able to get will be $r_i \cdot m_i$.

In our model, the elasticity among consumers leads to profit for the middleman; in particular, if the utility functions of all buyers are linear, then the middleman will make no profit. We will study the following two cases of utility functions; clearly, the first is a special case of the second, but it has stronger properties.

- **Case 1 utility functions:** Non-separable, continuously differentiable, concave functions satisfying non-satiation.
- **Case 2 utility functions:** Non-separable, continuously differentiable, quasiconcave functions satisfying non-satiation.

We will use the following notation and definitions throughout. $\boldsymbol{x}$ will denote allocations made of all goods to all buyers. $\boldsymbol{x}_i$ will denote the restriction of $\boldsymbol{x}$ to allocations made to buyer i only, and x_{ij} will denote the amount of good j allocated to buyer i. For the sake of ease of notation, let us introduce the following w.r.t. a generic buyer: $\boldsymbol{y}$, a vector of length g, will denote allocation and its jth component, y_j, will denote the allocation of good j. $f : \mathbf{R}_+^g \rightarrow \mathbf{R}_+$ will denote her utility function. Function f is *concave* if for any allocations $\boldsymbol{y}$ and $\boldsymbol{y}^{'}$,

$$f\left(\frac{\boldsymbol{y}+\boldsymbol{y}^{'}}{2}\right) \geq \frac{f(\boldsymbol{y})+f(\boldsymbol{y}^{'})}{2}.$$

Function f is *quasiconcave* if each of its *upper level sets* is convex, i.e., $\forall a \geq 0$, the set $S_a = \{\boldsymbol{y} \in \mathbf{R}_+^g \mid f(\boldsymbol{y}) \geq a\}$ is convex. We will say that f *satisfies non-satiation* if for any allocation $\boldsymbol{y}$, there is an allocation $\boldsymbol{y}^{'}$ that weakly dominates $\boldsymbol{y}$ component wise and such that $f(\boldsymbol{y}^{'}) > f(\boldsymbol{y})$.

The overall objective is to find prices for goods such that under these transactions, the market clears, i.e., there is no surplus or deficiency of any good. These will be called *equilibrium prices*. More formally, let $\boldsymbol{p}$ be prices of goods and $\boldsymbol{r}$ be the corresponding rates of buyers, as given by Lemma 2. Assume that each buyer i is charged at rate r_i and is allocated a bundle of goods. We will say that prices $\boldsymbol{p}$ are equilibrium prices if they satisfy the following **conditions:**

1. Each good having positive price is completely sold.
2. The money spent by each buyer i equals m_i.
3. The middleman never allocates any portion of a good at a loss (the implication of this condition on allocations is given in Section 2.1).

2.1 Determining Buyers' Rates

W.r.t. any prices, we will give a closed-form definition of each buyer i's rate, r_i; for ease of notation, we will do this for the generic buyer, i.e., we will define her rate r^*. For this section, assume that prices of goods are set to $\boldsymbol{p}$. In Lemma 2 we will show r^* is indeed her optimal rate, i.e., it maximizes her utility. In Section 3 we will show that the solution of the convex (nonlinear) program will assign utilities to a buyer at precisely this rate w.r.t. equilibrium prices. Hence, there is no need to explicitly compute buyers' rates.

Assume that the middleman is charging the buyer at the rate of r units of utility per dollar. We now formally state the restriction imposed by Condition 3 of equilibrium. Conceptually, assume that the middleman is making an allocation to the buyer gradually and continuously. Clearly, the effective price at which he is selling her good j depends on the allocation made already. If the latter is $\boldsymbol{y}$, then the *marginal price of good j at allocation $\boldsymbol{y}$* is

$$\frac{\partial f}{\partial y_j}(\boldsymbol{y}) \div r.$$

Therefore, at this point the middleman is selling good j at a loss iff the above-stated quantity is less than p_j, since then he is charging the buyer less for good j than he what the seller charged him. Given two allocations $\boldsymbol{y}$ and $\boldsymbol{y}'$, we will say that $\boldsymbol{y}'$ *weakly dominates* $\boldsymbol{y}$ if for each good j, $y'_j \geq y_j$. The next lemma gives the mathematical condition that an allocation needs to satisfy in order to satisfy Condition 3.

Lemma 1. *An allocation* $\boldsymbol{y}$ *made by the middleman at rate* r *satisfies Condition 3 iff*

$$\forall \boldsymbol{y}' \text{ s.t. } \boldsymbol{y} \text{ weakly dominates } \boldsymbol{y}', \forall j : \quad \left(\frac{\partial f}{\partial y_j}(\boldsymbol{y}') \div r\right) \geq p_j.$$

Let us say that an allocation $\boldsymbol{y}$ is *feasible for rate* r if it satisfies the condition given in Lemma 1. We next define the set of maximal, under the relation "weakly dominates", allocations that are feasible. For $r > 0$, this set is:

$$S(r) = \left\{\boldsymbol{y} \mid \left(\frac{\partial f}{\partial y_j}(\boldsymbol{y}) \div p_j\right) = r \text{ if } y_j > 0 \text{ and } \leq r \text{ otherwise}\right\}.$$

Observe that if f is strictly concave, $S(r)$ will be a singleton for each r. The function $U : \mathbf{R}_+ \rightarrow \mathbf{R}_+$ gives the largest utility attained by a feasible allocation at rate r:

$$U(r) = \sup\{f(\boldsymbol{y}) \mid \boldsymbol{y} \in S(r)\}.$$

Clearly, U is a decreasing function of r. Observe that because of the non-satiation condition, $\lim_{r \to 0} U(r)$ is unbounded.

Finally, we define rate r^* as follows

$$r^* = \arg\max_r \{U(r) \geq m \cdot r\},$$

where m is the money of the generic buyer. Since function $U(r)$ is unbounded as $r \rightarrow 0$, r^* is well defined for all m.

Lemma 2. r^* *maximizes the utility accrued by the generic buyer.*

3 The Convex/Nonlinear Program

The program (1) given below is a convex program for Case 1 utility functions and simply a nonlinear program for Case 2 utility functions. Besides non-negativity, the only constraint is that at most 1 unit of each good is sold. We will denote the Lagrange variables corresponding to these constraints as p_j's and will show that at optimality, they will be equilibrium prices of the corresponding market.

$$\begin{aligned} \text{maximize} \quad & \sum_{i \in B} m_i \log(f_i(\boldsymbol{x}_i)) \\ \text{subject to} \quad & \forall j \in G: \sum_{i \in B} x_{ij} \le 1 \\ & \forall i \in B,\ \forall j \in G:\ x_{ij} \ge 0 \end{aligned} \tag{1}$$

The KKT conditions for this program are:

1. $\forall j \in G: \quad p_j \ge 0.$
2. $\forall j \in G: \quad p_j > 0 \quad \Longrightarrow \quad \sum_{i \in B} x_{ij} = 1.$
3. $\forall i \in B,\ \forall j \in G: \quad p_j \ge \frac{m_i}{f_i(\boldsymbol{x}_i)} \cdot \frac{\partial f_i}{\partial x_{ij}}(\boldsymbol{x}_i).$
4. $\forall i \in B,\ \forall j \in G: \quad x_{ij} > 0 \quad \Longrightarrow \quad p_j = \frac{m_i}{f_i(\boldsymbol{x}_i)} \cdot \frac{\partial f_i}{\partial x_{ij}}(\boldsymbol{x}_i).$

Theorem 1. *For both cases of utility functions, the optimal primal and dual solutions to program (1) give equilibrium allocations and prices, and the latter are unique. Moreover, both can be computed in polynomial time.*

Proof. Because utility functions are assumed to be continuously differentiable, for an optimal solution to program (1) there is a unique dual, i.e., prices, that satisfies the KKT conditions stated above. From these, we will derive the 3 conditions defining equilibrium. For Case 1 utility functions, (1) is a convex program and for Case 2 utility functions, the upper level sets are convex, and for both cases a separation oracle can be implemented in polynomial time. Moreover, since the constraints are all linear, by [GLS88], the optimal solutions can be computed in polynomial time to any required degree of accuracy.

The first equilibrium condition is implied by the KKT conditions 1 and 2. Consider buyer i. Because of non-satiation, $x_{ij} > 0$ for some j. For this j, let

$$r_i = \left(\frac{\partial f_i}{\partial x_{ij}}(\boldsymbol{x}_i) \ \div \ p_j \right).$$

By KKT condition 4, any good j with $x_{ij} > 0$ must satisfy this equality, and if for some good j, $x_{ij} = 0$, then by KKT condition 3,

$$r_i \ge \left(\frac{\partial f_i}{\partial x_{ij}}(\boldsymbol{x}_i) \ \div \ p_j \right).$$

This proves that the middleman does not sell any part of a good at a loss. Substituting r_i back in KKT condition 4, we get $m_i = r_i \cdot f_i(\boldsymbol{x}_i)$, thereby proving that all money of buyer i is spent and r_i is the rate whose existence is established in Lemma 2.

4 The Welfare Theorems

The *first welfare theorem* states that allocations made at equilibrium prices are Pareto optimal and the *second welfare theorem* states that for any Pareto optimal utilities $\boldsymbol{u}^*$, there is a way of setting the initial moneys of buyers in such a way that an equilibrium obtained for this instance gives precisely $\boldsymbol{u}^*$ utilities to buyers.

Theorem 2. *The first welfare theorem is satisfied by both cases of utility functions and the second welfare theorem is satisfied by Case 1 utility functions.*

References

[AD54] Arrow, K., Debreu, G.: Existence of an equilibrium for a competitive economy. Econometrica 22, 265–290 (1954)

[BT04] Bhaskar, V., To, T.: Is perfect price discrimination really efficient? An analysis of free entry. The RAND journal of economics 35(4), 762–776 (2004)

[CDDT09] Chen, X., Dai, D., Du, Y., Teng, S.-H.: Settling the complexity of Arrow-Debreu equilibria in markets with additively separable utilities. In: FOCS (2009)

[CT09] Chen, X., Teng, S.-H.: Spending is not easier than trading: on the computational equivalence of Fisher and Arrow-Debreu equilibria. Journal of the ACM 56(3) (2009)

[CV04] Codenotti, B., Varadarajan, K.R.: Efficient computation of equilibrium prices for markets with leontief utilities. In: Díaz, J., Karhumäki, J., Lepistö, A., Sannella, D. (eds.) ICALP 2004. LNCS, vol. 3142, pp. 371–382. Springer, Heidelberg (2004)

[DPSV08] Devanur, N., Papadimitriou, C.H., Saberi, A., Vazirani, V.V.: Market equilibrium via a primal-dual-type algorithm. JACM 55(5) (2008)

[Edl98] Heller, M., Edlin, W.P., Epelbaum, A.S.: Is perfect price discrimination really efficient?: Welfare and existence in general equilibrium. Econometrica 66(4), 897–922 (1998)

[EEH94] Edlin, A., Epelbaum, M., Heller, W.: Surplus maximization and price discrimination in general equilibrium: Part I. Technical Report 9405, Centro de Investigacion Economica, ITAM (1994)

[GLS88] Grotschel, M., Lovasz, L., Schirjver, A.: Geometric Algorithms and Combinatorial Optimization. Springer, Heidelberg (1988)

[GV10] Goel, G., Vazirani, V.V.: A perfect price discrimination market model with production, and a (rational) convex program for it. In: Kontogiannis, S., Koutsoupias, E., Spirakis, P.G. (eds.) SAGT 2010. LNCS, vol. 6386, Springer, Heidelberg (2010)

[Jai04] Jain, K.: A polynomial time algorithm for computing the Arrow-Debreu market equilibrium for linear utilities. In: FOCS (2004)

[Sun04] Sundararajan, A.: Nonlinear pricing of information goods. Management Science 50(12), 1660–1673 (2004)

[Uza62] Uzawa, H.: Walras' existence theorem and brower's fixpoint theorem. In: Econ. Stud. Quart., pp. 59–62 (1962)

[Var85] Varian, H.R.: Price discrimination and social welfare. American Economic Review 75(4), 870–875 (1985)

[Var96] Varian, H.: Differential pricing and efficiency. First Monday 1(2) (1996)

[Vaz10] Vazirani, V.V.: Spending constraint utilities, with applications to the Adwords market. Mathematics of Operations Research 35(2) (2010)

[VY10] Vazirani, V.V., Yannakakis, M.: Market equilibria under separable, piecewise-linear, concave utilities. In: Proceedings of The First Symposium on Innovations in Computer Science (2010)

[WMT88] Norman, G., MacLeod, W.B., Thisse, J.-F.: Price discrimination and equilibrium in monopolistic competition. International Journal of Industrial Organization 6(4), 429–446 (1988)

[Ye07] Ye, Y.: Exchange market equilibria with Leontief's utility: Freedom of pricing leads to rationality. Theoretical Computer Science 378, 134–142 (2007)

Average-Case Analysis of Mechanism Design with Approximate Resource Allocation Algorithms

Yevgeniy Vorobeychik[1] and Yagil Engel[2]

[1] Sandia National Laboratories, Livermore, CA
eug.vorobey@gmail.com
[2] IBM Research, Haifa, Israel
yagile@il.ibm.com

Abstract. Mechanism design provides a useful practical paradigm for competitive resource allocation when agent preferences are uncertain. Vickrey-Clarke-Groves (VCG) mechanism offers a general technique for resource allocation with payments, ensuring allocative efficiency while eliciting truthful information about preferences. However, VCG relies on exact computation of optimal allocation of resources, a problem which is often computationally intractable. Using approximate allocation algorithms in place of exact algorithms gives rise to a *VCG*-based mechanism, which, unfortunately, no longer guarantees truthful revelation of preferences. Our main result is an *average-case* bound, which uses information about *average*, rather than *worst-case*, performance of an algorithm. We show how to combine the resulting bound with simulations to obtain probabilistic confidence bounds on agent incentives to misreport their preferences and illustrate the technique using combinatorial auction data. One important consequence of our analysis is an argument that using state-of-the-art algorithms for solving combinatorial allocation problems essentially eliminates agent incentives to misreport their preferences.

1 Introduction

Mechanism design provides a useful practical paradigm for competitive resource allocation when agent preferences are uncertain. The field of mechanism design has received considerable attention in academic literature in the last several decades. Great technological advances, coupled with a rather mature understanding of the field, have recently brought much of this theory to bear on real resource allocation problems faced by the government and industry. Perhaps of greatest practical significance has been the field of auction theory [8], and, in particular, the design of *combinatorial auctions* [2]. In a combinatorial auction, bidders are allowed to submit bids on all subsets of a given set of items.[1] The auctioneer must then solve the *winner determination problem (WDP)*, computing

[1] Items could be actual goods for sale, slots on a schedule, locations and times of banner ads displayed on a website, etc.

A. Saberi (Ed.): WINE 2010, LNCS 6484, pp. 571–578, 2010.

which subsets of the goods will be allocated to which bidders, with the objective of maximizing allocative efficiency.

The field of mechanism design has historically occupied itself primarily with the issue of incentives, while mostly ignoring the computational aspects of the problem. As it turns out, computational impediments can be devastating for incentives. For example, while *VCG* [13] is the central mechanism used to incentivize bidders to report their true valuations, using *VCG*-based payment schemes together with an approximate algorithm for the *WDP* nearly universally fails to incentivize truthful revelation of values [15]. However, it is well known that the combinatorial auction *WDP* is NP-Hard [10]—indeed, even hard to approximate [16]—and, consequently, an approximation algorithm must, in general, be used.

Promptly, Computer Scientists went to work to fix the incentive problem in combinatorial auctions with approximate allocation algorithms. Sanghvi and Parkes [18] demonstrate that computing an improving deviation in *VCG*-based combinatorial auctions is NP-Hard, although this worst-case result is difficult to rely on in practice. Lavi and Swamy [9] present a truthful (in expectation) mechanism when the approximation algorithm bounds the integrality gap of LP relaxation, while Lehmann, O'Callaghan, and Shoham [11] and Mu'alem and Nisan [14] obtain general truthful mechanisms for combinatorial auctions when bidders are "single-minded" (i.e., each has positive value for exactly one bundle of items). Dobzinski, Nisan, and Shapira [5] present a framework for designing truthful approximation algorithms, and demonstrate instances with an asymptotically optimal worst-case bound for the general WDP. Nisan and Ronen [15] develop a second-chance mechanism in which players are not capable of computing a beneficial lie.

This extensive literature addressing the incentive problems of approximate *WDP* implicitly suggests that such problems are critical. Field practitioners of combinatorial auctions, however, seem to rarely, if ever have come up against the worst-case complexity issues [4,3]. Furthermore, the majority of combinatorial auction problems that have been studied in simulation can be solved very fast using modern algorithms [12,16,17], and, indeed, the general-purpose CPLEX integer programming tool is usually very effective [17]. How can we explain this gap between theory, which views the incentive problem of *VCG*-based mechanisms as severe, and practice, which ignores it almost entirely? We believe that the reason for this gap is that theoretical literature tends to offer worst-case analyses, whereas practitioners (be it designers of combinatorial auctions or bidders) are usually most concerned about typical cases.

We present a framework for average-case incentive analysis results as an attempt to bridge the gap between theory and practice, which provide evidence that the incentive problem with *VCG*-based mechanisms is not very severe. Specifically, we offer general techniques to empirically assess incentive effects of specific algorithms based on *average-case* bounds. For example, if an algorithm can solve the allocation problem exactly in almost every instance, there are no incentives to deviate from truthfulness in the Bayes-Nash sense. We operationalize this bound in combinatorial auctions, illustrating how a use of simulation-based model of bidder valuation distribution allows the designer to obtain precise probabilistic confidence bounds on agent incentives to lie. One side-effect of this approach is some qualitative evidence that the incentives to misreport decrease with increasing problem size—yet another piece of evidence arguing that in practice incentive compatibility of *VCG* with approximate allocation algorithms is often not very severe.

2 Preliminaries

In our setting, each player $i \in I$ submits to a central designer his utility function, as indexed by his type $t_i \in T_i$. Let O be the set of outcomes (e.g., feasible allocations), I be a set of n players and let $T = T_1 \times \cdots \times T_n$ be the joint type set. Let $F(\cdot)$ be a probability distribution over joint player types and $u_i(t_i, o)$ player utility functions where $o \in O$ typically depends on joint player report t. While we may hope that all players submit their types honestly, they may choose to lie, submitting some t_i' instead of t_i, and these lies could, in general, be a function of true type t_i.

We allow players to submit and accept payments p_i, and assume that their utility functions are quasi-linear in these, that is $u_i(t_i, o, p_i) = v_i(t_i, o) + p_i$, where $v_i(t_i, o)$ is the underlying value that player i with type t_i has for outcome o, and p_i is his payment (which is negative when the agent is paying the designer). A mechanism is a function that chooses an outcome o and assigns the payments p_i for all players i given a joint report of types $t \in T$. Thus, we use $o(t)$ and $p_i(t)$ to indicate such choices as made by some specified mechanism.

A central aspect of mechanism design is the prediction of agent play for a given choice of a mechanism. Typically the role of such predictions is played by equilibrium concepts. We appeal to two such concepts below (defined with respect to direct revelation mechanisms, that is, mechanisms which attempt to truthfully elicit player preferences). Under a *dominant strategy equilibrium* each player is (weakly) best off reporting his true type *no matter what other players do*. Under a *Bayes-Nash equilibrium*, on the other hand, each player maximizes his expected utility by reporting his true type t_i, *assuming that all other players are honest*. Both equilibrium concepts admit natural notions of approximation: in an ϵ-dominant strategy equilibrium, a player can gain no more than ϵ by deviating, no matter what the opponents do, whereas an ϵ-Bayes-Nash equilibrium guarantees that the expected gain to any player from deviation is at most ϵ, with expectation taken with respect to the joint type distribution.

A useful measure of strategic stability is that of *game-theoretic regret*. While in general this measure can be defined for any joint strategy profile, we use it only to gauge the regret of truthful reporting. Hence, we use a simpler definition, with $\tilde{\epsilon} = E_F[\epsilon(t)] = E_F[\max_i \epsilon_i(t_i)]$, where

$$\epsilon_i(t_i) = \max_{t_i' \in T_i} E_F[u_i(t_i, o(t_i', t_{-i}), p_i(t_i', t_{-i})) - u_i(t_i, o(t), p(t)) | t_i].$$

In words, it is the maximum expected benefit any player can obtain from reporting untruthfully.

A widely studied goal of mechanism design, and one we focus on here, is that of maximizing *social welfare*, or the sum of player valuations. Formally, define *social welfare* to be $V(t,o) = \sum_{i\in I} v_i(t_i,o)$, where o is an outcome and t is a joint type profile. Let $o^* : T \to O$ denote the welfare optimal (efficient) outcome (allocation) and let

$$V^*(t) = \sum_{i\in I} v_i(t_i, o^*(t)) = \max_{o\in O} \sum_{i\in I} v_i(t_i, o)$$

be the maximum welfare achieved for a type profile t. Let $V^* = \sup_{t\in T} V^*(t)$. It is well known that optimal allocation can be achieved as a truthful dominant strategy equilibrium by using Groves payments [13], with $p_i(t) = \sum_{j\neq i} v_j(t_j, o^*(t)) + h_i(t_{-i})$. Here h_i is an arbitrary function of the types reported by other players; for simplicity of exposition, we set it to 0.[2]

The algorithmic problem in mechanism design has been explored extensively in Computer Science literature, particularly in the context of combinatorial auctions [2]. In general, it is recognized that computing optimal welfare is a hard problem. However, the literature has generated a plethora of algorithms for computing approximately optimal allocation. Let $g : T \to O$ be an algorithm for computing approximately efficient allocation.[3] Since g may compute only a suboptimal allocation, we let $V_g(t)$ be the welfare at the allocation $g(t)$, that is $V_g(t) = \sum_{i\in I} v_i(t_i, g(t))$. Define *VCG*-based payments by $p_i^g(t) = \sum_{j\neq i} v_j(t_j, g(t)) + h_i(t_{-i})$. Hence, the *VCG*-based mechanism will select an outcome according to g, and the players will receive payments $p_i^g(t)$.

Approximation algorithms typically include a guarantee with respect to the quality of approximation they provide. We say that $g(\cdot)$ is an α-approximation if $V^*(t) \leq \alpha V_g(t)$ for any $t \in T$.

3 Connecting Approximation and Incentives

To begin, suppose that, somehow, we have an approximation bound for g that is a *known function* of $\alpha(t)$ for all $t \in T$. In the most trivial case, it could be just a fixed α, reducing the setup to the worst-case analysis above. Alternatively, we may be able to split the set of type profiles into subsets $T^1, T^2, \ldots$, and obtain much better uniform bounds on some of these subsets than the worst case analysis would allow; for example, perhaps we know that for some large subset of combinatorial auction problems we can compute optimal allocation fast *exactly*, or nearly so. In any case, presently we will see that we need not

[2] *VCG* extends the Groves scheme by specifying $h_i(t_{-i})$ to guarantee *individual rationality*, that is, that every player obtains positive net value from participation. Since the subject of our inquiry is the incentive structure, setting $h_i(t_{-i})$ to 0 has no impact on any of the arguments we make.

[3] Our results can be extended rather directly to randomized allocation algorithms.

even construct $\alpha(t)$ for all possible type profiles, but can obtain probabilistic bounds based on a sample of a finite subset of these.

Our first key result presents an average-case bound on incentives of players to deviate, measuring the incentives in terms of approximate Bayes-Nash equilibria.

Theorem 1. *Suppose that the algorithm g is an $\alpha(t)$-approximation. Then a player i can gain at most $\epsilon_i(t_i)$ when others are playing truthfully, where $\epsilon_i(t_i) = E_{t_{-i}}\left[\frac{\alpha(t)-1}{\alpha(t)}V^*(t)|t_i\right]$.*

The proofs of this and other results are in the appendix of the extended version of the paper.

Corollary 1. *Suppose that the algorithm g is an $\alpha(t)$-approximation. Then truthful reporting constitutes an ϵ-Bayes-Nash equilibrium for*

$$\epsilon = nE_t\left[\frac{\alpha(t)-1}{\alpha(t)}V^*(t)\right].$$

To illustrate an application of these results, suppose that the space of joint types T can be partitioned into "easy" and "hard" type profiles, that is, $T = \underline{T} \cup \overline{T}$. Let $\underline{\alpha} = \sup_{t\in\underline{T}} \alpha(t)$ and $\overline{\alpha} = \sup_{t\in\overline{T}} \alpha(t)$ and assume that $\underline{\alpha} \leq \overline{\alpha}$. Then, after some algebraic manipulation we obtain

$$E_t\left[\frac{\alpha(t)-1}{\alpha(t)}V^*(t)\right] \leq \frac{\underline{\alpha}-1}{\underline{\alpha}}E_{t\sim F|\underline{T}}[V^*(t)] + \left(\frac{1}{\underline{\alpha}} - \frac{1}{\overline{\alpha}}\right)V^*F(\overline{T}).$$

Note that since $\left(\frac{1}{\underline{\alpha}} - \frac{1}{\overline{\alpha}}\right)V^*$ is just a constant, as the probability measure of "hard" instances becomes small, the incentives for players to deviate approach $\frac{\underline{\alpha}-1}{\underline{\alpha}}E_{t\sim F|\underline{T}}[V^*(t)]$. Hence the following corollary.

Corollary 2. *Suppose that $F(\overline{T}) = 0$. Then truthful reporting constitutes an ϵ-Bayes-Nash equilibrium for $\epsilon = n\frac{\underline{\alpha}-1}{\underline{\alpha}}E_{t\sim F|\underline{T}}[V^*(t)]$.*

In the special case when $\underline{\alpha} = 1$ (that is, easy instances can be solved fast *exactly*) as is the case in many combinatorial auction settings, and when $F(\overline{T}) = 0$, that is, when the probability of drawing a hard problem is 0, truthful reporting is a Bayes-Nash equilibrium. Hence the following direct corollary.

Corollary 3. *Suppose that $\underline{\alpha} = 1$ and $F(\overline{T}) = 0$. Then the strategy $s_i(t_i) = t_i$—that is, truthfully reporting actual preferences—is a Bayes-Nash equilibrium under the allocation algorithm g.*

4 Applying the Non-uniform Incentive Bound

A key question that stems from the above analysis is how a mechanism designer would determine an incentive bound for his algorithm in practice. We would not, for example, want to require the designer to obtain a non-trivial $\alpha(t)$ for every $t \in T$. Rather, we offer the following empirical approach.

1. Obtain or construct a simulator that allows one to sample joint player types $t \in T$ according to F
2. Collect a set of K joint type samples $t^1, \dots, t^K$
3. For each t^k, compute $V_g(t^k)$ and $V^*(t^k)$ (or an upper bound on $V^*(t^k)$)
4. Compute $\alpha(t^k) = \frac{V^*(t^k)}{V_g(t^k)}$, let $\hat{Z}(t^k) = \frac{\alpha(t^k)-1}{\alpha(t^k)} V^*(t^k)$, and define
$$\hat{Z} = \frac{1}{K} \sum_{k=1}^{K} \hat{Z}(t^k)$$
5. Compute a probabilistic bound based on $\hat{Z}$

The first step requires a designer to either obtain or construct a simulator. This seems rather demanding, but may be necessary to do for a high-stakes problem anyway. Moreover, in the case of combinatorial auctions, a state-of-the-art simulator to generate realistic problem instances is already publicly available [12].

For the last step, we have a few options. A most general option would be to use a distribution-free bound (e.g., Hoeffding inequality), but these tend to be very loose. Instead, we assume that $\hat{Z}$ is Normally distributed (an assumption that is justified by the Central Limit Theorem when K is large), using $s^2(\hat{Z}(t^k))/K$ (where $s^2(\cdot)$ is the sample variance) as an estimate of the variance of $\hat{Z}$. Then,

$$E_t \left[\frac{\alpha(t) - 1}{\alpha(t)} V^*(t) \right] \leq \hat{Z} + z_\delta \sqrt{\frac{s^2(\hat{Z}(t^k))}{K}} \quad (1)$$

with probability at least $1 - \delta$, where z_δ is the value of Normal distribution at $1 - \delta$.

4.1 Example: Combinatorial Auctions

To illustrate a concrete example applying the techniques introduced above, we now offer an incentive analysis of combinatorial auctions based on auction instances (in our notation, t^k) generated by CATS [12]. Since the absolute values of the bounds are not very meaningful, we give them as fraction of V^*. While V^* is actually unknown, note that $\hat{Z}/V^* \leq \hat{Z}/\max_k V^*(t^k)$, so below we report $\hat{Z}' = \hat{Z}/\max_k V^*(t^k)$. Additionally, CATS generates a set of bids, but does not specify the number of players (which could therefore be arbitrary). Consequently, we ultimately report bounds as multiples of nV^*.

The data set we used is composed of (a) a set of samples with 1000 bids on 144 goods ($1K - 144$), (b) a set with 1000 bids on 256 goods ($1K - 256$), (c) a set with 2000 bids on 64 goods ($2K - 64$), and (d) a set with varying problem sizes (varsize). Each set contains 5000 samples, 500 for each of 10 different distributions. The data includes the result obtained by CPLEX which ran to optimality, the results obtained by CASS [6] after about 7500 seconds for $1K - 144$ and $1K - 256$, or 44000 seconds for the other datasets, and, for the dataset $1K - 256$, also the result obtained by the Gonen-Lehmann (GL) algorithm [7].[4]

[4] A small fraction of the problems for which CPLEX reported an optimal result are not in fact optimal, because the results by CASS and GL are higher. On these problems we use the maximum value of the three algorithms as the "true" optimum.

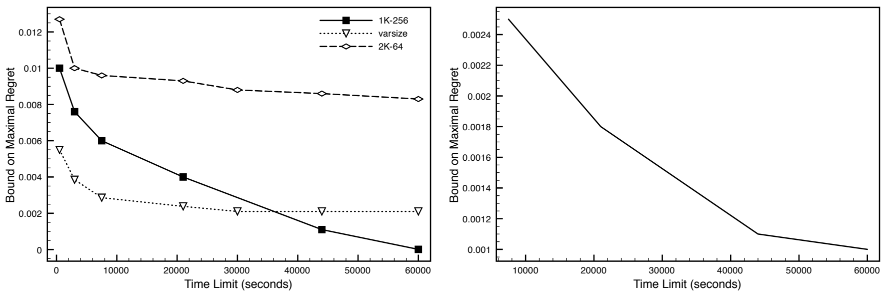


Fig. 1. Upper bound on regret, as a fraction of nV^*, *left*: for several data set sizes, and *right*: for the union of all data

We computed the bound on regret for each dataset, as well as for the union set (named *all*). For each one we include the data for all CATS distributions except "arbitrary". For $g(t)$ we used the following combination: we used the result returned by CPLEX for a sampled profile t^k if it was obtained in at most S seconds; otherwise the result returned by CASS was used. We varied the time limit S is between 500 and 60000 seconds (about 16.6 hours). The longer time limits are reasonable for high volume auctions in which a lot of money is at stake.[5]

In Figure 1 (left), we show the resulting bound for each dataset as a function of the time limit. It also quantifies the tradeoff between the amount of time given to the algorithm and regret (incentives for players to lie). The bounds are computed as explained above, with confidence level $1 - \delta = 0.95$. The results are particularly encouraging for the union set (Figure 1, right), and $1K - 256$, for which the regret approaches zero when the time limit increases. Note that the chart for the dataset of $1K - 144$ is omitted because the $\hat{Z}$ is zero. In this case, we can obtain an upper bound of 0.0006 on the proportion of suboptimally solved instances (giving a regret bound of $0.0006nV^*$) with 0.95 confidence using the Clopper-Pearson bound [1].

5 Conclusion

We presented results that allow construction of average-case bounds on agent incentives to lie about their preferences for *VCG*-based mechanisms. Conceptually, this deviates from the more traditional worst-case analysis which often fails to provide meaningful bounds. Practically, we introduce a simple method for assessing incentive properties of specific approximation algorithms in the context of economic resource allocation problems. We illustrate the resulting empirical incentive analysis for a specific approximation algorithm in the context of several combinatorial auction problems. Our results here suggest that

[5] Since the data is relatively old, our bounds are likely excessively pessimistic.

using state-of-the-art algorithms for solving combinatorial allocation problems essentially eliminates agent incentives to misreport their preferences.

References

1. Clopper, C., Pearson, E.S.: The use of confidence or fiducial limits illustrated in the case of the binomial. Biometrika 26, 404–413 (1934)
2. Cramton, P., Shoham, Y., Steinberg, R. (eds.): Combinatorial Auctions. MIT Press, Cambridge (2006)
3. Day, R.W., Cramton, P.: The quadratic core-selecting payment rule for combinatorial auctions. Working paper (2010)
4. Day, R.W., Raghavan, S.: Fair payments for efficient allocations in public sector combinatorial auctions. Management Science 53(9), 1389–1406 (2007)
5. Dobzinski, S., Nisan, N., Shapira, M.: Truthful randomized mechanisms for combinatorial auctions. In: 38h ACM Symposium on Theory of Computing, pp. 644–652 (2006)
6. Fujishjima, Y., Leyton-Brown, K., Shoham, Y.: Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In: International Joint Conference on Artificial Intelligence, pp. 548–553 (1999)
7. Gonen, R., Lehmann, D.: Linear programming helps solving large multi-unit combinatorial auctions. In: Electronic Market Design Workshop (2001)
8. Krishna, V.: Auction Theory. Academic Press, London (2002)
9. Lavi, R., Swamy, C.: Truthful and near-optimal mechanism design via linear programming. In: 46th IEEE Symp. on Foundations of Comp. Sc., pp. 595–604 (2005)
10. Lehmann, D., Müller, R., Sandholm, T.: The winner determination problem. In: Cramton et al [2], ch. 2, pp. 297–318
11. Lehmann, D., O'Callaghan, L.I., Shoham, Y.: Truth revelation in approximately efficient combinatorial auctions. J. of the ACM 49(5), 1–26 (2002)
12. Leyton-Brown, K., Shoham, Y.: A test suite for combinatorial auction. In: Cramton et al [2], ch. 18, pp. 451–478
13. Mas-Colell, A., Whinston, M.D., Green, J.R.: Microeconomic Theory. Oxford Univ. Press, Oxford (1995)
14. Mu'alem, A., Nisan, N.: Truthful approximation mechanisms for restricted combinatorial auctions. In: AAAI, pp. 379–384 (2002)
15. Nisan, N., Ronen, A.: Computationally feasible VCG mechanisms. Journal of Artificial Intelligence Research 29, 19–47 (2007)
16. Sandholm, T.: Algorithm for optimal winner determination in combinatorial auctions. Artificial Intelligence 135(1-2), 1–54 (2002)
17. Sandholm, T., Suri, S., Gilpin, A., Levine, D.: CABOB: A fast optimal algorithm for winner determination in combinatorial auctions. Management Science 51(3), 374–390 (2005)
18. Sanghvi, S., Parkes, D.C.: Hard-to-manipulate combinatorial auctions. Technical report, Harvard University (2004)

Author Index