An Equivalent Version of the Caccetta-Häggkvist Conjecture in an Online Load Balancing Problem

Angelo Monti[1], Paolo Penna[2,⋆], and Riccardo Silvestri[1]

[1] Dipartimento di Informatica, Università degli Studi di Roma "La Sapienza", via Salaria 113, Roma, Italy
{monti,silver}@di.uniroma1.it
[2] Dipartimento di Informatica ed Applicazioni "R.M. Capocelli", Università di Salerno, via S. Allende 2, Baronissi (SA), Italy
penna@dia.unisa.it

Abstract. We study the competitive ratio of certain *online* algorithms for a well-studied class of *load balancing* problems. These algorithms are obtained and analyzed according to a method by Crescenzi *et al* (2004). We show that an exact analysis of their competitive ratio on certain "uniform" instances would resolve a fundamental conjecture by Caccetta and Häggkvist (1978). The conjecture is that any digraph on n nodes and minimum outdegree d must contain a directed cycle involving at most $\lceil n/d \rceil$ nodes. Our results are the first relating this conjecture to the competitive analysis of certain algorithms, thus suggesting a new approach to the conjecture itself. We also prove that, on "uniform" instances, the analysis by Crescenzi *et al* (2004) gives only trivial upper bounds, unless we find a counterexample to the conjecture. This is in contrast with other (notable) examples where the same analysis yields optimal (non-trivial) bounds.

Keywords: Caccetta-Häggkvist conjecture, online load balancing, competitive analysis.

1 Introduction

We consider a combinatorial problem which has applications to the construction of *competitive*[1] algorithms for the well-studied class of *online load balancing* problems considered in e.g. [4,3,2,5] (see Section 1.2 for a formal definition). Our work is motivated by a technique from Crescenzi *et al.* [8] in which the simple greedy algorithm is "tuned" on the problem at hand. A rather informal

⋆ Fully supported by the European Union under the Project IST-15964, Algorithmic Principles for Building Efficient Overlay Computers (AEOLUS).

[1] Intuitively speaking, an online algorithm is c-competitive if there exists a constant b such that the algorithm outputs a solution whose cost is at most $c \cdot opt + b$ where opt is the optimum for the instance considered up to the current time step. In this case, c is the competitive ratio of the algorithm.

A. Brandstädt, D. Kratsch, and H. Müller (Eds.): WG 2007, LNCS 4769, pp. 154–165, 2007.

description of this technique is as follows (see Section 1.2 for a more formal description):

> Each online load balancing problem specifies a set of "feasible modifications" of the greedy algorithm and an "easy-to-compute" upper bound $c(\cdot)$ on the competitive ratio. In particular, every such feasible modification M describes a modified version of the greedy algorithm whose competitive ratio on this problem is at most $c(\mathrm{M})$.

This approach has been applied to the *linear* and to the *hierarchical* server topologies studied in [5] where it is rather easy to find an M such that $c(\mathrm{M})$ results in a dramatic improvement over the competitive ratio of the greedy algorithm and matches the lower bound for the problem considered [8]. It is thus natural to try to apply the same technique to more problems.

1.1 Our Contribution

In this work we consider a natural class of *s-uniform* online load balancing problems in which every task can be assigned to some *s-subset* of the n processors (this subset can vary arbitrarily from task to task). The resulting combinatorial problem is to determine (exactly) the *minimum competitive bound* $C(n, s)$ which is the smallest value that the above function $c(\cdot)$ can assume for s-uniform instances. Our major contribution is to show that the minimum competitive bound $C(n, s)$ leads to an *equivalent* version of one of the most fundamental and intriguing conjectures in graph theory (which also accounts for dozens of connections to other basic questions in combinatorics and number theory [14]):

Conjecture 1 (Caccetta-Häggkvist 1978 [7]). *Any digraph on n nodes with minimum outdegree at least d contains a directed cycle of length at most $\lceil n/d \rceil$.*

We indeed prove that, if the above conjecture is true, then $C(n, s) = n/s$. Observe that, there is a trivial upper bound $C(n, s) \leq n/s$ (see Section 1.2). Thus any improvement on the trivial bound would give a counterexample to the conjecture. At the heart of this result is another interesting number associated to the analysis of s-uniform instances which we call the *blind competitive bound* $B(n, s)$. This number is "tightly coupled" with the Caccetta-Häggkvist conjecture since we prove that, for $s \leq \sqrt{n}$,

$$B(n, s) = 1 + n - \lceil n/s \rceil$$

if and only if the conjecture holds. The number $B(n, s)$ is the minimum for $c(\cdot)$ when considering certain modifications M which result in "blind" algorithms that assign tasks without even "looking at the processors": tasks which can be potentially allocated to the same subset of processors are *all* assigned to a predetermined and fixed processor.

Our results can be seen as the hardness of obtaining any non-trivial bound with the method of [8] in the case of s-uniform instances (this is in contrast with other instances considered in [8]). These hardness results are in some sense of

a "new type" since they do *not* rely on computational assumptions and they are obtained by relating two (apparently) different problems. We feel one of the main contributions of this work is to connect the analysis of online algorithms to a fundamental conjecture in graph theory and to show that such an analysis is as difficult as solving the latter.

From another point of view, our results suggest a possible way for proving the conjecture by showing a lower bound on the competitive ratio of the online algorithms yielded by certain modifications of greedy. Such bounds have also a practical interest since these algorithms use only *local information* (namely, each task can decide its own allocation by considering only the current load of the processors in its associated subset). Blind algorithms are a notable example since lower bounds are probably easier to prove, while any tight result on the competitive ratio of the best blind algorithm for s-uniform instances would either prove or disprove the conjecture. We stress that the Caccetta-Häggkvist conjecture is considered a central and important problem in combinatorics, graph, and number theory. Thirty years of significant efforts culminated in a large number of deep connections among these areas. They have been the main subject of a recent workshop held at the American Institute of Mathematics dedicated to this conjecture (see Sullivan's paper surveying these results [14]).

Roadmap. In Section 1.2 we introduce online load balancing, the technique in [8], and the related combinatorial problems. In Section 2 we introduce and study blind algorithms, and we relate the blind competitive bound to the Caccetta-Häggkvist conjecture. We apply these results to the minimum competitive bound in Section 3. Finally, we further discuss our results and their implications in Section 4.

1.2 Online Load Balancing, Modified Greedy Algorithms, and Their Analysis

In this section we go back to our initial application that is *online load balancing* of *temporary weighted tasks* in the case of *restricted assignment* with *no preemption.* Here each task t is specified by a *subset* S_t *of processors* that can execute that task, a *weight* W_t, and a *duration* D_t. Tasks arrive one by one, each task t needs to be allocated upon its arrival to one of the processors in S_t. No task can be reallocated. The duration D_t is *unknown* and the task simply disappears without any prior notice after D_t time units from its arrival. At every time step, a processor has a *load* equal to the sum of the weights of those tasks currently in the system and which have been assigned to it. The goal is to keep, over time, the maximum processor load as low as possible. We are interested in designing online algorithms with a small *competitive ratio* c, that is, the algorithm must guarantee that the load of each processor never exceeds $c \cdot opt + b$, where opt is the optimum for the instance and b is a fixed constant.

In general, online algorithms with a "good" competitive ratio are designed "ad-hoc" for a family $\mathcal{F}$ containing all possible subsets of processors that can be associated to any task. A notable example is the hierarchical server topologies

by Bar-Noy *et al* [5] where the "combinatorial structure" of $\mathcal{F}$ impacts significantly on the competitive ratio of the algorithms. Moreover, "general purpose" algorithms, such as the greedy one, are in general "far" from the optimal [3,4,5]. The approach in [8] constructs a "modified" version of the greedy algorithm for the problem '$\mathcal{F}$' as follows:

- In an *offline phase*, each $S \in \mathcal{F}$ is mapped into a *non-empty* subset $\mathrm{M}(S) \subseteq S$, for some function $\mathrm{M}(\cdot)$.
- In the *online phase*, each task t is allocated to the currently *least-loaded* processor in $\mathrm{M}(S_t)$.

Notice that we limit ourselves to a subset of available processors. As shown in [8], by carefully choosing M, the modified greedy algorithm avoids allocations that are "too far" from the optimum. The main result in [8] is that the competitive ratio of this algorithm is at most $1 + c_{\mathcal{F}}(\mathrm{M})$, with

$$c_{\mathcal{F}}(\mathrm{M}) := \max_{S \in \mathcal{F}} \frac{|\mathrm{Adversary}_{\mathcal{F}}(S, \mathrm{M})|}{|\mathrm{M}(S)|}, \tag{1}$$

where $\mathrm{Adversary}_{\mathcal{F}}(S, \mathrm{M})$ consists of the *union* of all subsets S' in $\mathcal{F}$ such that $\mathrm{M}(S')$ intersects $\mathrm{M}(S)$. Intuitively, the tasks allocated to $\mathrm{M}(S)$ could have been assigned only to processors in $\mathrm{Adversary}_{\mathcal{F}}(S, \mathrm{M})$.

In this work, we focus on *s-uniform instances*, that is, the case in which $\mathcal{F}$ contains *all s-subsets* of the n processors. This is a natural restriction modeling problems where each task is guaranteed (only) to be assignable to s out of the n processors (though this set can change arbitrarily from task to task). With the minimum competitive bound $C(n, s)$ we ask how small the bound in (1) can be depending on n and s (see Definition 2). Notice that the resulting algorithm uses only *local* information as it assigns a task t by simply considering the current load of (a subset of) the processors that can execute that task. When this subset, which is specified by M, consists of a *single* processor, the corresponding algorithm requires "no information" on the processors' loads. The blind competitive bound $B(n, s)$ is defined as the minimum competitive bound, when restricting to these "blind" algorithms (see Definition 1). This number is a *tight* bound on the competitive ratio of these algorithms and its analysis is fundamental for the minimum competitive bound too. Both numbers initiate the study of online algorithms for load balancing problems which use only *local* information. In our view, one of the main contributions of this work is a stringent connection between the competitive analysis of certain local online load balancing algorithms and the Caccetta-Häggkvist conjecture.

Preliminaries and notation. We are given a family $\mathcal{F}$ of *distinct* subsets of an n-set (the latter, representing the processors). We let $Feas(\mathcal{F})$ be the set of all functions M mapping every subset $S \in \mathcal{F}$ into a *nonempty* subset $\mathrm{M}(S) \subseteq S$. We let

$$\mathrm{Adversary}_{\mathcal{F}}(S, \mathrm{M}) := \bigcup_{S' \in \mathcal{F}:\ \mathrm{M}(S') \cap \mathrm{M}(S) \neq \emptyset} S'.$$

In the sequel, s denotes the cardinality of the sets in $\mathcal{F}$. We will always assume that s and n are positive integers satisfying $2 \leq s \leq n$ (the case $s = 1$ is trivial and not interesting for the application). Observe that $\text{Adversary}_{\mathcal{F}}(S, \text{M})$ contains at most n elements (i.e., the n processors). Thus, the identity function $\text{M}_{\text{trivial}}(S) = S$ yields a *trivial upper bound*:

$$c_{\mathcal{F}}(\text{M}_{\text{trivial}}) \leq n/s. \tag{2}$$

We typically consider families containing *all* possible s-subsets of an n-set. In this case we write $Feas(n, s)$ and omit the subscript '$\mathcal{F}$'.

2 Blind Algorithms and the Caccetta-Häggkvist Conjecture

A simple (and somewhat naive) class of (online) algorithms assign tasks in a fixed manner without "looking" at the current loads of the processors: every task t is allocated to the processor $p(S_t)$ for some function $p(\cdot)$ (thus ignoring the allocation of all other tasks). These algorithms and their analysis via the upper bound in (1) are captured by the following:

Definition 1. *A* blind algorithm *is a function* M *mapping every s-subset of the n processors into a* 1*-subset of this s-subset. The* blind competitive ratio *is*

$$B(n, s) := \min_{\text{M} \in Blind(n,s)} \{c(\text{M})\},$$

where $Blind(n, s)$ consists of all blind algorithms.

We stress that a simple argument shows that, for blind algorithms, the upper bound in (1) gives a tight analysis:

Fact 2. *The competitive ratio of any blind algorithm* M *is exactly* $c(\text{M})$*. Hence, $B(n, s)$ is the minimum competitive ratio over all blind algorithms.*

In this section, we show that $B(n, s) = 1 + n - \lceil n/s \rceil$, where the lower bound holds *if and only if* the Caccetta-Häggkvist conjecture (see Conjecture 1) is true. The upper and the lower bounds will follow from the next two lemmata.

Lemma 1. *Let G be any digraph on n nodes with minimum outdegree d and not containing any directed cycle of length at most s. Then there exists* $\text{M} \in Blind(n, s)$ *with $c(\text{M}) = n - d$, that is, $B(n, s) \leq n - d$.*

Proof. We construct $\text{M} \in Blind(n, s)$ as follows. We identify the nodes of G with the n processors. For every s-subset S we search for an $a \in S$ such that in G there is no edge from a to another element in S. Observe that such an element must exist since otherwise we have a directed cycle involving only elements in S, and thus a directed cycle of length at most s. We then set $\text{M}(S) := \{a\}$. Observe that, if (a, b) is an edge in G and an s-subset T contains b, then it cannot be the

case $\mathrm{M}(T) = \{a\}$. This implies that the set $\mathrm{Adversary}(S, \mathrm{M})$ does not contain any node in the outneighborhood of a. Since node a has outdegree at least d, this set has cardinality at most $n - d$. Since $|\mathrm{M}(S)| = 1$ for all S, from (1) we obtain $c(\mathrm{M}) = \max_S |\mathrm{Adversary}(S, \mathrm{M})| \leq n - d$. □

Lemma 2. *Let n, s, and $d \geq s$ be positive integers such that $B(n, s) \leq n - d$. Then there exists a digraph G on at most n nodes with minimum outdegree at least d and not containing a directed cycle of length at most s.*

Proof. Let $G = G(\mathrm{M})$ be the digraph on n nodes containing the edge (a, b) if and only if there exists no S such that $b \in S$ and $\mathrm{M}(S) = \{a\}$. By construction the outneighborhood of a contains all but the elements in $\mathrm{Adversary}(S, \mathrm{M})$, that is, its outdegree is $n - |\mathrm{Adversary}(S, \mathrm{M})|$. Since $|\mathrm{Adversary}(S, \mathrm{M})| \leq B(n, s)$, the outdegree of any node a is at least $n - B(n, s) \geq d$. Hence, the graph G has minimum outdegree $d_G \geq d$.

We observe that the subgraph induced by *any* subset of s nodes must contain a *sink*, that is, a node having outdegree 0 in that subgraph: Indeed, for any S, the element a such that $\mathrm{M}(S) = \{a\}$ must be a sink. In particular, there is no directed cycle of length s.

Using this fact, we iteratively remove nodes from G and obtain a subgraph G' with $n' \leq n$ nodes, without directed cycles of length at most s, and minimum outdegree equal to the minimum outdegree d of G. Towards this end, we proceed as follows. While we can pick a set C of nodes that form a directed cycle of length at most $s - 1$ in G' (recall that there is no directed cycle of length s), we add to C, one by one, nodes of G' that have an edge directed to the current set of nodes. This process must stop when reaching at most $s-1$ nodes since otherwise, when C reaches cardinality s, by construction, it does not contain a sink, thus a contradiction. Notice that there is no edge from a node in $G' - C$ to a node in C. We can thus remove the nodes in C from the graph without decreasing its minimum outdegree.

At the end of this process, the graph G' does not contain any directed cycle of length s or smaller and its minimum outdegree is at least $d_G \geq d$. Observe that G' cannot be empty since every removed set C as above must have some outgoing edge (because of $d \geq s \geq |C|$) and this edge cannot be ingoing to the previously removed components. □

Lemmata 1 and 2 will give us the upper and the lower bound:

Theorem 3. *For any n and $s \leq \sqrt{n}$, it holds that*

$$B(n, s) = 1 + n - \lceil n/s \rceil$$

where the lower bound holds unless Conjecture 1 is false.

Proof. Let us set $d = \max\{s, \lceil n/s \rceil\}$. By contradiction, assume $B(n, s) \leq n - d$. Lemma 2 implies the existence of a digraph G on $n' \leq n$ nodes with minimum outdegree $d \geq \lceil n'/s \rceil$ and not containing directed cycles of length s or smaller. However, Conjecture 1 implies that G must have a directed cycle of length at

most $\lceil n'/d \rceil \leq \lceil n'/\lceil n'/s \rceil \rceil \leq \lceil n'/(n'/s) \rceil = s$, thus a contradiction. Since $B(n,s)$ is integer, it must be $B(n,s) \geq 1 + n - d$. Since $s \leq \sqrt{n}$, we have $d = \lceil n/s \rceil$, which proves the lower bound. In order to prove the upper bound, we consider the following digraph G, first described by Behzad, Chartrand and Wall [6]. We let $[n] = \{0, \ldots, n-1\}$ be the set of nodes. For every node $x \in [n]$, we let its out-neighborhood being the $d-1$ nodes in the interval $[(x+1) \bmod n, (x+d-1) \bmod n]$. By construction, the resulting digraph G has minimum outdegree $d-1$ and, since $d - 1 = \lceil n/s \rceil - 1 < n/s$, does not have any directed cycle of length at most s. Lemma 1 thus implies $B(n,s) \leq n - (d-1) = n - \lceil n/s \rceil + 1$, that is the upper bound. □

Remark 1. Notice that the Caccetta-Häggkvist conjecture is not "interesting" for $d > n/2$ since in this case it is easy to show that a two-cycle must exist, i.e., the conjecture holds. Lemma 2 implies that $B(n,2) = n/2$, for any n. In contrast, proving a tight bound for $B(n,3)$ is the first hard case: It corresponds to the case $d = n/3$ of the conjecture which is one of the most studied [14, Section 2.2].

It is possible to settle (weaker) lower bounds on $B(n,s)$ by using some "approximate" results for the Caccetta-Häggkvist conjecture. It is known that the conjecture holds if we consider some "additive" constant α. That is, a minimum outdegree d guarantees that every digraph on n nodes must have a directed cycle of length at most $n/d + \alpha$. Currently, the best known bound is $\alpha = 73$ by Shen [13]. This type of results imply the following:

Theorem 4. *For any n and $\alpha < s \leq \sqrt{n} + \alpha/2$, it holds that $B(n,s) \geq 1 + n - \lceil n/(s-\alpha) \rceil$.*

Proof. Since $s > \alpha$, we can consider $d = \lceil n/(s-\alpha) \rceil$. By contradiction, assume $B(n,s) \leq n - \lceil n/(s-\alpha) \rceil = n - d$. From $s \leq \sqrt{n} + \alpha/2$, we have $d \geq s$ and thus Lemma 2 implies that there exists a digraph on n nodes with minimum outdegree d and not containing any directed cycle of length s or smaller. Since $n/d + \alpha = n/\lceil n/(s-\alpha) \rceil + \alpha \leq n/(n/(s-\alpha)) + \alpha = (s-\alpha) + \alpha = s$, this graph does not contain a directed cycle of length $n/d + \alpha$ or smaller. This contradicts the definition of α. Since $B(n,s)$ is integer, it must be $B(n,s) \geq 1 + n - d$ and the theorem follows. □

For $s = 3$, Shen [12] proved another approximate version of the conjecture: if the minimum outdegree is at least $\mu \cdot n$, then there is a directed triangle, where $\mu > 1/3$ is a "multiplicative" constant (see also [14, Section 2.3]). This result, combined with Lemma 2, yields the following lower bound:

Theorem 5. *For any n, it holds that $B(n,3) \geq 1 + n - \mu \cdot n$, where $\mu = 3 - \sqrt{7} = 0.3542\cdots$.*

3 The Minimum Competitive Bound

In this section, we turn our attention to "less naive" algorithms which can be obtained with the method described in Section 1.2. In particular, we study the

bound in (1), again when $\mathcal{F}$ consists of *all s-subsets* of the n processors (for the sake of readability, we omit the subscript '$\mathcal{F}$'):

Definition 2. *The* minimum competitive bound *is*

$$C(n,s) := \min_{\mathrm{M} \in Feas(n,s)} \{c(\mathrm{M})\},$$

where $Feas(n,s)$ *consists of all functions* $\mathrm{M}(\cdot)$ *mapping every s-subset of n processors into a non-empty subset* $\mathrm{M}(S) \subseteq S$.

Notice that we have a trivial upper bound $C(n,s) \leq n/s$ (see Equation 2). We prove that $C(n,s) = n/s$, unless we disprove Conjecture 1. That is, the trivial upper bound is likely the best possible. We will first prove lower bounds for some special cases (these results do not require Conjecture 1).

Lemma 3. *Let* $\mathrm{M} \in Feas(n,s)$ *such that* $|\mathrm{M}(S)| \geq 2$*, for all s-subset* S*. Then, there exists an s-subset* T *for which* $|\mathrm{Adversary}(T, \mathrm{M})| = n$.

Proof. Without loss of generality, we can assume that $|\mathrm{M}(S)| = 2$, for every S. Indeed, if we shrink all $\mathrm{M}(S)$ into a two-set $\mathrm{M}'(S) \subseteq \mathrm{M}(S)$, we obtain a function $\mathrm{M}' \in Feas(n,s)$ satisfying $\mathrm{Adversary}(S, \mathrm{M}') \subseteq \mathrm{Adversary}(S, \mathrm{M})$.

We use $\mathrm{Adversary}(S)$ as a shorthand for $\mathrm{Adversary}(S, \mathrm{M})$ and assume, by way of contradiction, that $|\mathrm{Adversary}(S)| < n$, for all S of size s. Using this fact, we give an iterative way to define a suitable sequence $B^1 \subset B^2 \subset \cdots \subset B^k$ as follows. We start from an arbitrary s-subset S^1 and let $B^1 := \mathrm{M}(S^1)$. At each iteration i, we "expand" the current B^i into a new set $B^{i+1} := B^i \cup \{b_i\} \cup \mathrm{M}(S^{i+1})$, where b_i and S^{i+1} are defined as follows. Each S^i is an s-subset and thus the hypothesis $|\mathrm{Adversary}(S^i)| < n$ implies that we can chose $b_i \notin \mathrm{Adversary}(S^i)$. We then define S^{i+1} as an s-subset such that $b_i \in \mathrm{M}(S^{i+1})$, if such a set exists; otherwise, S^{i+1} is an arbitrarily chosen s-subset containing B^i and b_i. Below we will show that the set B^{i+1} adds 2 or 3 elements to the set B^i, thus implying that we can stop when $s - 2 \leq |B^k| \leq s$.

Claim (1). $\mathrm{M}(S)$ cannot intersect B^i if S is an s-subset containing $B^i \cup \{b_i\}$.

Proof of Claim (1). We proceed by induction on i. For $i = 1$, if $\mathrm{M}(S)$ intersects $B^1 = \mathrm{M}(S^1)$, then $\mathrm{Adversary}(S^1)$ contains S. Since $b_1 \in S$, this contradicts the definition of b_1. Now assume the claim holds for $i-1$ and let S be an s-subset containing $B^i \cup \{b_i\}$. Since $B^i = B^{i-1} \cup \{b_{i-1}\} \cup \mathrm{M}(S^i)$, S contains $B^{i-1} \cup \{b_{i-1}\}$, and the inductive hypothesis implies that $\mathrm{M}(S)$ cannot intersect B^{i-1}. If $\mathrm{M}(S)$ intersects $\mathrm{M}(S^i)$ then, since $b_{i-1} \in S$, we have the contradiction $b_i \in \mathrm{Adversary}(S^i)$. If $\mathrm{M}(S)$ contains b_{i-1}, the definition of S^i implies that $b_{i-1} \in \mathrm{M}(S^i)$. (Recall that $b_{i-1} \notin \mathrm{M}(S)$ only in the case there is no s-subset S with $b_{i-1} \in \mathrm{M}(S)$.) But then $\mathrm{M}(S)$ would again intersect $\mathrm{M}(S^i)$, which leads to the same contradiction as above. The inductive step thus follows from $B^i = B^{i-1} \cup \{b_{i-1}\} \cup \mathrm{M}(S^i)$. The claim thus follows. □

Since $|\mathrm{M}(S)| = 2$, Claim (1) implies that B^{i+1} is obtained from B^i by adding at least two (and at most three) new elements not in B^i. We can thus define k as

the first integer such that $s-2 \leq |B^k| \leq s$. We next show that in each of the three cases a contradiction arises:

1. For $|B^k| = s-2$, we consider *any* s-subset $S(x) := B^k \cup \{b_k\} \cup \{x\}$, with $x \notin B^k \cup \{b_k\}$. Claim (1) implies $\mathrm{M}(S(x)) = \{b_k, x\}$, and thus Adversary$(S(x))$ contains also all elements not in $B^k \cup \{b_k\}$, that is, $|\mathrm{Adversary}(S(x))| = n$.
2. For $|B^k| = s-1$, we simply observe that for $S := B^{k-1} \cup \{b_{k-1}\}$ Claim (1) yields $\mathrm{M}(S) = \{b_{k-1}\}$, contradicting the hypothesis $|\mathrm{M}(S)| \geq 2$ for all s-subsets S.
3. For $|B^k| = s$, $B^{k-1} \cup \{b_{k-1}\}$ must have size $s-2$, since $|B^{k-1}| < s-2$. For every s-subset $S(x,y) := B^{k-1} \cup \{b_{k-1}, x, y\}$ Claim (1) implies $\mathrm{M}(S(x,y)) = \{x, y\}$. If we keep x fixed and consider all y not in this set, we obtain the contradiction $|\mathrm{Adversary}(S(x,y))| = n$.

This concludes the proof of the lemma. □

Observe that the above result says that, if $C(n,s) < n/s$, then the corresponding M must be such that $|\mathrm{M}(S)| = 1$ for at least one S. In order to prove the lower bound $C(n,s) = n/s$, we will make use of the following result showing that, without loss of generality, we can restrict ourselves to optimal modifications M having a "canonical" structure (the result applies to any family $\mathcal{F}$ of s-subsets):

Lemma 4. *For any* $\mathrm{M} \in Feas(\mathcal{F})$*, there exists an* $\mathrm{M}_c \in Feas(\mathcal{F})$ *such that* $c_{\mathcal{F}}(\mathrm{M}_c) \leq c_{\mathcal{F}}(\mathrm{M})$ *and* M_c *is* canonical*, that is,* $\mathrm{M}_c(S) \not\subset \mathrm{M}_c(T)$ *for all* $S, T \in \mathcal{F}$*.*

Proof. Consider two s-subsets S and T such that $\mathrm{M}(S) \subset \mathrm{M}(T)$. (Otherwise the lemma holds.) If we shrink $\mathrm{M}(T)$ to $\mathrm{M}(S)$, what we obtain is a new $\mathrm{M}' \in Feas(\mathcal{F})$ such that $\mathrm{M}'(T) = \mathrm{M}(S)$ and $\mathrm{M}'(U) = \mathrm{M}(U)$ for $U \neq T$. This implies $\mathrm{Adversary}_{\mathcal{F}}(U, \mathrm{M}') \subseteq \mathrm{Adversary}_{\mathcal{F}}(U, \mathrm{M})$ for all s-subsets U, and that

$$\frac{|\mathrm{Adversary}_{\mathcal{F}}(T, \mathrm{M}')|}{|\mathrm{M}'(T)|} \leq \frac{|\mathrm{Adversary}_{\mathcal{F}}(S, \mathrm{M})|}{|\mathrm{M}(S)|};$$
$$\max_{U \neq T} \frac{|\mathrm{Adversary}_{\mathcal{F}}(U, \mathrm{M}')|}{|\mathrm{M}'(U)|} \leq \max_{U \neq T} \frac{|\mathrm{Adversary}_{\mathcal{F}}(U, \mathrm{M})|}{|\mathrm{M}(U)|}.$$

This yields $c_{\mathcal{F}}(\mathrm{M}') \leq c_{\mathcal{F}}(\mathrm{M})$. To obtain the final family M_c it suffices to iterate the above transformation at most $|\mathcal{F}|$ times. (At every iteration we let M being the family obtained in the previous iteration and pick S and T as above with $\mathrm{M}(S)$ not containing another $\mathrm{M}(U)$.) The lemma thus follows. □

We first give a tight bound for some special cases for which we do not need the Caccetta-Häggkvist conjecture:

Theorem 6. *For every* n*, if* $s \geq \sqrt{n}$ *or* $s = 2$*, then it holds that* $C(n,s) = n/s$*.*

Proof. Let M be such that $c(\mathrm{M}) = C(n,s)$. We first consider $s \geq \sqrt{n}$. If there exists one S with $|\mathrm{M}(S)| = 1$, then $c(\mathrm{M}) \geq |\mathrm{Adversary}(S, \mathrm{M})| \geq s \geq n/s$, where the two inequalities follow from $S \in \mathrm{Adversary}(S, \mathrm{M})$ and from $s \geq \sqrt{n}$,

respectively. Otherwise, in the case $|\mathrm{M}(S)| \neq 1$ for every S, Lemma 3 implies that $c(\mathrm{M}) \geq n/s$. (Recall that $|\mathrm{M}(S)| \leq |S| \leq s$.)

Let us now consider the case $s = 2$. From Lemma 4 we can assume that M is canonical. This implies that the n processors are partitioned into two sets N_1 and N_2 such that the following holds. For every two-subset $S \subseteq N_i$, it holds that $|\mathrm{M}(S)| = i$, with $i = 1, 2$. Let $n_1 := |N_1|$ and $n_2 := |N_2|$, and let $\mathcal{F}_i$ denote the family of all two-subsets of N_i, for $i = 1, 2$. Let M' be the function M restricted $\mathcal{F}_1$ and observe that $\mathrm{M}' \in \mathit{Blind}(n_1, 2)$. Hence, there is one $S \subseteq N_1$ for which $|\mathrm{Adversary}_{\mathcal{F}_1}(S, \mathrm{M}')| \geq B(n_1, 2) = n_1/2$ (see Remark 1). That is, $\mathrm{Adversary}(S, \mathrm{M})$ contains at least $n_1/2$ elements from N_1. We next show that it must also contain *all* elements x in N_2. Indeed, for every two-subset $S(x)$ consisting of $x \in N_2$ and $\mathrm{M}(S)$, Lemma 4 implies that $\mathrm{M}(S(x)) = \mathrm{M}(S)$, and thus $x \in \mathrm{Adversary}(S, \mathrm{M})$. Hence, $|\mathrm{Adversary}(S, \mathrm{M})| \geq n_1/2 + n_2 = n_1/2 + (n - n_1) = n - n_1/2 \geq n/2$, where the last inequality follows from $n_1 \leq n$. □

Finally, from Lemma 3 we obtain the main result of this section:

Theorem 7. *For every n and $2 < s < \sqrt{n}$, it holds that $C(n, s) = n/s$. The lower bound holds unless Conjecture 1 is false. Hence, the trivial upper bound $C(n, s) \leq n/s$ is the best possible one.*

Proof. Let $\mathrm{M} \in \mathit{Feas}(n, s)$ with $c(\mathrm{M}) = C(n, s)$. From Lemma 4, we can assume M being canonical. Because of Lemma 3, the theorem holds if $|\mathrm{M}(S)| \geq 2$ for all s-subsets S. Otherwise, we consider the subset N_1 of those processors x such that $\{x\} = \mathrm{M}(S)$, for some S. Let N_2 be the complement of N_1, that is, the subset of processors not in N_1. From the hypothesis, we have $3 \leq s < n/s$. We consider the following two cases for $n_2 := |N_2|$:

1. $n_2 < n/s$. In this case $n_1 := |N_1| = n - n_2 > n - n/s$. Since M is canonical, for every s-subset T contained in N_1, it must be the case $|\mathrm{M}(T)| = 1$. Let $\mathcal{F}_1$ denote the family of all s-subsets of N_1 and let M' be the function obtained by restricting M to $\mathcal{F}_1$. Observe that M' is a function in $\mathit{Blind}(n_1, s)$. Hence, there exists $S \in N_1$ such that $C(n, s) \geq |\mathrm{Adversary}(S, \mathrm{M}')| \geq B(n_1, s)$. From the proof of Theorem 3, if Conjecture 1 holds, then $B(n_1, s) \geq 1 + n_1 - \max\{s, \lceil n_1/s \rceil\} > 1 + n - n/s - \max\{s, \lceil n_1/s \rceil\} \geq 1 + n - n/s - \max\{s, \lceil n/s \rceil\} = 1 + n - n/s - \lceil n/s \rceil > n - 2n/s$, where the last inequality follows from $\lceil n/s \rceil < 1 + n/s$. Since $s \geq 3$, we have $n - 2n/s \geq n/s$, thus implying $C(n, s) \geq B(n_1, s) > n/s$.
2. $n_2 \geq n/s$. By definition of N_1 and N_2, every s-subset S contained in N_2 must satisfy $|\mathrm{M}(S)| \geq 2$. Since $s < \sqrt{n}$, we have $n_2 \geq n/s > s$ and thus N_2 contains some s-subset. Let us consider the function M' obtained by restricting M to the s-subsets of N_2. Observe that $\mathrm{M}' \in \mathit{Feas}(n_2, s)$. Lemma 3 implies that there exists $S \subseteq N_2$ with $\mathrm{Adversary}(S, \mathrm{M})$ containing all the elements in N_2. If the set $\mathrm{Adversary}(S, \mathrm{M})$ contains also N_1, then we clearly have $C(n, s) \geq n/s$. Otherwise, we consider an $x \in N_1$ with $x \notin \mathrm{Adversary}(S, \mathrm{M})$. For $\{x\} = \mathrm{M}(T)$, if $\mathrm{Adversary}(T, \mathrm{M})$ contains N_2, then $C(n, s) \geq n_2 > n/s$ (recall that $|\mathrm{M}(T)| = 1$) and the theorem holds. Otherwise, we can pick a

$y \in N_2$ with $y \notin \text{Adversary}(T, \text{M})$. Observe that $\text{M}(S)$ must contain at least $s-1$ elements, unless $C(n,s) \geq n/s$. We can thus construct an s-subset $S' := \{x\} \cup R'$ with R' containing y and other $s-2$ elements from $\text{M}(S)$. Observe that $\text{M}(S')$ cannot contain x since otherwise $\text{M}(S') \cap \text{M}(T) = \{x\} \neq \emptyset$ and thus $y \in S' \subseteq \text{Adversary}(T, \text{M})$, contradicting the definition of y. Similarly, $\text{M}(S')$ cannot contain any of the elements in $R' \setminus \{y\}$ since otherwise $\text{M}(S') \cap \text{M}(S) \neq \emptyset$ and thus $x \in S' \subseteq \text{M}(S)$, contradicting the definition of x. Hence, it must be the case $\text{M}(S') = \{y\}$, contradicting $y \notin N_1$ (since $y \in N_2$). □

4 Conclusions and Open Questions

We have applied the approach by Crescenzi *et al* [8] to a natural class of s-uniform instances, which model the problem version in which the only available information is that each task is assignable to s out of the n processors, for some known constant s. We have shown that this approach is unlikely to lead to any "satisfactory" upper bound. Namely, the minimum competitive bound $C(n,s)$ is equal to the trivial n/s upper bound, unless we find a counterexample to the Caccetta-Häggkvist conjecture [14]. Even for rather limited algorithms, for which the analysis in [8] is tight, an exact answer is "equivalent" to the conjecture above. That is, the competitive ratio $B(n,s)$ of the best algorithm in this class can be determined for all s and n if and only if we resolve the conjecture.

We consider the study of these algorithms interesting by itself since they only require *local* information. Indeed, the online load balancing problem considered here arises in many practical situations (e.g., when connecting mobile devices requiring different bandwidth to one of the "geographically close" base stations). The natural greedy algorithm can have a rather unsatisfactory competitive ratio in several cases [3,5], which motivated the development of more sophisticated "ad-hoc" algorithms [4,5]. The latter are *not* local, though their competitive ratio is significantly better than greedy one. To the best of our knowledge, there is no prior study of *local* online algorithms for this problem version (apart from the tight bound $\Theta(n^{2/3})$ on the greedy [4]). Online local algorithms for a different task allocation problem have been studied by Kuhn *et al* [9]. In their problem, the goal is to maintain (roughly) the same number of tasks on each processor, and tasks can be moved only "locally", i.e., between adjacent processors.

We conclude observing that our results might be used to write a computer program to check the Caccetta-Häggkvist conjecture. Observe that, if we believe the conjecture is true, then a program which verifies it for a fixed n and d, will have to go through *all* possible digraphs on n nodes and minimum outdegree d. This is because we have to show that there is no way to avoid a directed cycle with $\lceil n/d \rceil$ nodes. Theorem 3 gives an alternative that is to come up with an (efficient) algorithm to compute $B(n,s)$. Obviously, this algorithm should not rely on the Caccetta-Häggkvist conjecture, that is, it should be possible to prove its correctness *independently* from the conjecture (e.g., the algorithm returns an optimal modification M for any given $\mathcal{F}$ containing only s-subsets). Notice that,

once again, the case $s = 3$ seems to be the "first" difficult one. Indeed, for $s = 2$ the optimal modification M for any family $\mathcal{F}$ reduces to the problem of orienting the edges of an indirected graph in order to minimize the maximum indegree (see Aichholzer *et al* [1] and Nash-Williams [11]). Such optimal orientation can be computed with standard flow techniques, thus yielding an optimal algorithm for $s = 2$ (see the full version of this work [10]). Unfortunately, the results do not apply to $s = 3$, which remains an interesting open question.

Acknowledgments. We wish to thank Noga Alon for a very useful discussion and for pointing us to the Caccetta-Häggkvist conjecture via the proxy of Janos Körner. We also thank Pilu Crescenzi and Carmine Ventre for several comments on a previous version of this work.

References

1. Aichholzer, O., Aurenhammer, F., Rote, G.: Optimal graph orientation with storage applications. SFB Report Series 51, TU-Graz (1995)
2. Azar, Y.: On-line Load Balancing. In: Fiat, A., Wöginger, G.J. (eds.) Online Algorithms. LNCS, vol. 1442, Springer, Heidelberg (1998)
3. Azar, Y., Broder, A., Karlin, A.: On-line load balancing. Theoretical Computer Science 130(1), 73–84 (1994)
4. Azar, Y., Kalyanasundaram, B., Plotkin, S., Pruhs, K., Waarts, O.: On-line load balancing of temporary tasks. Journal of Algorithms 22, 93–110 (1997)
5. Bar-Noy, A., Freund, A., Naor, J.: On-line load balancing in a hierarchical server topology. SIAM Journal on Computing 31(2), 527–549 (2001)
6. Behzad, M., Chartrand, G., Wall, C.: On minimal regular digraph with given girth. Fundamenta Mathematicae 69, 227–231 (1970)
7. Caccetta, L., Häggkvist, R.: On minimal regular digraph with given girth. In: Proc. of 9th S-E Conf. Combinatorics, Graph Theory and Computing, pp. 181–187 (1978)
8. Crescenzi, P., Gambosi, G., Nicosia, G., Penna, P., Unger, W.: On-line load balancing made simple: Greedy strikes back. Journal of Discrete Algorithms 5(1), 162–175 (2007)
9. Kuhn, F., Schmid, S., Wattenhofer, R.: A self-repairing peer-to-peer system resilient to dynamical adversarial churn. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 13–23. Springer, Heidelberg (2005)
10. Monti, A., Penna, P., Silvestri, R.: An Equivalent Version of the Caccetta-Häggkvist Conjecture in an Online Load Balancing Problem (2007), Technical report available at `http://www.dia.unisa.it/~penna`
11. Nash-Williams, C.: Edge-disjoint spanning trees of finite graphs. J. London Math. Soc. 36, 445–450 (1961)
12. Shen, J.: Directed triangles in digraphs. Journal of Combinatorial Theory 74, 405–407 (1998)
13. Shen, J.: On the Caccetta-Häggkvist conjecture. Graphs and Combinatorics 18(3), 645–654 (2003)
14. Sullivan, B.D.: A Summary of Results and Problems Related to the Caccetta-Häggkvist Conjecture. In: ARCC Workshop: The Caccetta-Haggkvist conjecture (2006), available at `http://www.aimath.org/pastworkshops/caccetta.html`