Aspects of Coordinating the Bidding Strategy in Concurrent One-to-Many Negotiation

Khalid Mansour, Ryszard Kowalczyk and Michal Wosko

Abstract Automated negotiation is an important mechanism of interaction between software agents and has been an active research area for more than a decade. When the automated negotiation process involves multiple agents, the problem of interdependency between the actions of agents during negotiation arises and consequently, a coordination mechanism becomes an essential part of the negotiation process. One of the important characteristics of a negotiating agent is its bidding strategy. This work addresses the problem of coordinating the bidding strategy of an agent negotiating concurrently with multiple agents (i.e., one-to-many negotiation) and discusses different interdependency factors affecting it.

Keywords Multiagent systems · Bidding strategy · Negotiation · Coordination

1 Introduction

Negotiation is a method of interaction between different parties that can effectively resolve conflicts [1]. This paper discusses various aspects of coordinating or managing related automated negotiations for a software agent negotiating concurrently with other software agents (i.e., one-to-many negotiation) in terms of deciding on the bidding strategy for each negotiation instance in each negotiation round.

K. Mansour (✉) · R. Kowalczyk · M. Wosko
Swinburne University of Technology, Melbourne, Australia
e-mail: mwmansour@swin.edu.au

R. Kowalczyk
e-mail: rkowalczyk@swin.edu.au

M. Wosko
e-mail: mwosko@swin.edu.au

F. Sun et al. (eds.), *Knowledge Engineering and Management*,
Advances in Intelligent Systems and Computing 214,
DOI: 10.1007/978-3-642-37832-4_10,

The key point in the coordination theory is managing interdependencies amongst related activities to achieve a common goal [2]. In other words, when the interdependencies between different related activities arise, the need for a coordination mechanism becomes essential. The related activities in the context of automated negotiation are the instances of interactions between autonomous agents. Given that an agent initiates multiple negotiation instances, there is a need to manage the bidding strategy for each instance given the behaviors of the opponents in each negotiation instance. Managing the bidding strategy involves managing the negotiation variables, e.g., a concession parameter.

The coordination as a process can be classified into two types: *centralized coordination* and *distributed coordination*. In this work, we mainly address the centralized approaches for coordinating multiple negotiations conducted by a buyer agent.

In our work, we consider the automated one-to-many multiagent systems and assume the following:

- agents are rational,
- agents are self-interested and aim to maximize their utility,
- agents do not disclose their private information such as their utility structures or deadlines, and
- agents use the alternating offers protocol [3].

In general, automation of any process reduces the time needed to do the job and produces more efficient/effective results. Automation of negotiation has similar objectives where software agents can work on behalf of the users to negotiate with each other for buying, selling, task assignment, resource allocation, etc.

Many negotiation frameworks are proposed in the literature to describe and automate the process of generation offers and counteroffers. The process of offers/counteroffers generation (i.e., bidding strategy) depends on some criteria that control the process. The most used criteria are the agent's internal resources such as time and the behavior of the opponents, e.g., [4].

Many published works address the problem of coordinating multiple related negotiations, e.g., [5–9]. Most of the previous works consider negotiation over one issue (e.g., price) for the purpose of reaching one agreement. When both the number of issues and the number of agreements increase, the coordination process becomes more complicated.

The contribution of this paper can be summarized as follows:

- investigate possible sources of interdependencies between related negotiations,
- discuss possible coordination approaches for different negotiation scenarios, and
- show a sample of empirical results which demonstrate the importance of coordination.

The rest of the paper is organized as follows: Sect. 2 discusses the negotiation model that captures the negotiation settings of our work while Sect. 3 discusses possible sources of interdependency in negotiation domain. Section 4 presents the related work. Section 5 presents the coordination approach while Sect. 6 shows an illustrative example of some empirical results and finally Sect. 7 concludes the paper.

2 Negotiation Model

We consider a buyer agent negotiating with a set of seller agents $\mathbf{S} = \{s_1, s_2, \ldots, s_n\}$ concurrently, see Fig. 1. We assume that the seller agents are independent in their actions, i.e., they do not exchange information. The buyer agent has a set of delegate negotiators $\mathbf{D} = \{d_1, d_2, \ldots, d_n\}$. It creates and destroys delegate negotiators during negotiation in response to the number of the seller agents who enter or leave negotiation. Each delegate d_i negotiates with a seller s_i. The possible negotiation issues over which **D** and **S** negotiate are included in the set $\mathbf{J} = \{j_1, j_2, \ldots, j_g\}$ and each issue $j_i \in \mathbf{J}$ must be an issue of negotiation by at least one negotiation pair, i.e., (d_i, s_i).

To make our negotiation framework more comprehensive, we introduce the notion of negotiation objects set (**O**) notion. The negotiation object is any item in which agents have interest to negotiate over. A negotiation object represents either a physical item (e.g., a printed book) or a non-physical item, e.g., a web service. Let $\mathbf{O} = \{o_1, o_2, \ldots, o_m\}$. Each $o_i \in \mathbf{O}$ represents an object of negotiation. The illustration of the idea is shown in Fig. 1.

We assume that each negotiation delegate is responsible to negotiate over one object, and many delegates can negotiate over the same object, but a delegate cannot negotiate over more than one object concurrently, see function f_d in Eq. 1.

In our model, each negotiation delegate is mapped onto an object, a deadline $t_{\max} \in \mathbb{N}^*$ and an offer generation tactic $\theta \in \Theta$. Each object is mapped onto a negotiation issue set ($J_l \in 2^{\mathbf{J}}$). Finally, each issue is mapped onto a set of constraints, e.g., the reservation values ([min,max]). The number and types of constraints vary. Equation 1 shows the formal representation of the three functions (i.e., f_d, f_o, f_j).

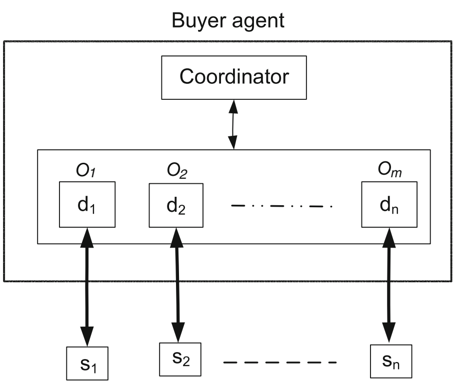

Fig. 1 One-to-many negotiation

$$
\begin{aligned}
&f_d : \mathbf{D} \xrightarrow[(]{1-1} \mathbf{O} \times \mathbb{N}^* \times \Theta) \\
&f_o : \mathbf{O} \xrightarrow[2]{1-1} \mathbf{J} \qquad (1) \\
&f_j : \mathbf{J} \xrightarrow[(]{1-1} [\text{min,max}] \times \ldots)
\end{aligned}
$$

In each negotiation round, the buyer agent may need to execute one or more of the functions in Eq. 1 to reflect some changes in the environment. At the start of a negotiation process, all the functions in Eq. 1 are executed. For example, using f_d, a delegate d_i can be assigned a currency converter web service as a negotiation object, 30 negotiation rounds as $t_{\max}$ and a linear time-dependent counteroffer generation tactic. For the currency converter web service object, the price and response time can be assigned as negotiation issues using f_o. Finally, for the price and response time issues, reservation values are assigned using f_j. Similar assignments can be done to the rest of delegates, objects, and issues.

3 Sources of Interdependency in Negotiation Domain

Interdependency between related activities is the driving force behind the need for coordination. Figure 2 shows a categorization of possible dependencies between related activities that can apply in different application domains. As in Fig. 2, we identify several types of dependencies in the automated negotiation domain.

Objects under negotiation may *share resources*, for example, an agent negotiating for buying a laptop and a camera needs to allocate a certain amount of the available budget (a resource) for buying the laptop and another amount for buying the camera. In that sense, the laptop and the camera share a resource. The agent needs to distribute/redistribute the resource in a way to achieve the negotiation goal, i.e., reach a valuable agreement.

The *task assignment* dependency appears when multiple agents negotiate over performing different tasks. It also may depend on the distribution of certain

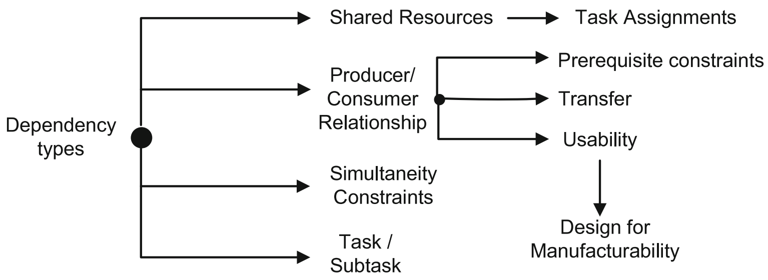

Fig. 2 Common dependency types [2]

resources, i.e., the assignment of tasks to agents is related to the amount of resources allocated to each agent.

The *prerequisite constraints dependency* exists when one activity must be completed before another activity can start or finish [2]. When an agent seeks to buy a hardware and a software given that both the hardware and software must be compatible, it can select to buy the hardware first and then buy a compatible software or vice versa since buying both simultaneously may result in buying incompatible products.

The *simultaneity constraints* dependency determines which negotiations can run concurrently and which negotiations should not run concurrently. In other words, the process of running different negotiations or taking certain actions during negotiation such as quitting a certain instance of negotiation needs to be synchronized.

The *task/subtask dependency* in a negotiation process can be illustrated when one negotiation depends on some other negotiations, i.e., the negotiations are multi-linked [10]. For example, if there are three negotiations a, b, c, but negotiation a depends on negotiations b and c (i.e., negotiation a is successful iff negotiation a is successful and both negotiations b and c are also successful), then we consider that negotiation a has two subtasks (i.e., b and c).

One of the most complicated activities during negotiation is deciding on the value of an offer/counteroffer (i.e., choosing a certain bidding strategy to generate an offer/counteroffer) in each negotiation round. Calculating the value of an offer/counteroffer in each negotiation round is a non-trivial task due to the following:

- the process can be affected by the actions of the outside options,
- the interdependency between the issues of the same object,
- the interdependency between the issues of different objects, and
- the interdependency between different negotiation objects.

In the next few subsections, we elaborate more on the interdependency in one-to-many negotiation looking at the interdependency from the point of view of a buyer agent who negotiates concurrently with multiple seller agents.

3.1 Interdependency Amongst Objects

Accepting a number of agreements by an agent in a certain order while negotiating concurrently with multiple opponents is equivalent to procuring the same number of objects in the same order while negotiating with opponents sequentially. Procuring a certain number of objects sequentially is the easiest solution to solve the problem of procuring a certain number of objects in a certain predefined order. However, using the sequential approach has a few drawbacks. First, because negotiations are conducted once at a time, it is difficult to predict the results of the future negotiations in terms of (1) whether a certain negotiation instance will be able to reach an agreement (2) the expected utility of the agreements. Second, it is difficult to allocate resources for each negotiation instance because we have no knowledge about the demand behavior of the opponents of the future negotiations. For example, we might allocate more resources for the first few negotiation instances to guarantee reaching agreements but the resources for the next negotiations may not be enough to guarantee reaching agreements over the rest of the objects. Finally, the sequential approach takes more time.

The alternative solution to the sequential negotiation approach is adopting the concurrent negotiation approach where a negotiating agent receives feedback during negotiation in terms of the opponents' offers and can act accordingly to fine-tune its strategy and resource allocation pattern. The drawback of the concurrent approach is the need for coordination whenever any type of interdependency exists between objects or between objects' issues, etc. For example, different objects may have interdependency between their attributes such as interface compatibility between two different softwares. Sometimes buying object an (o_1) before buying object an (o_2) causes a loss in utility such as confirming a hotel reservation before confirming a flight. If the flight is canceled for any reason then the buyer is obliged to fulfill his/her obligations towards the hotel reservation.

In some cases, the order of procurement is not defined before the start of negotiation, it could be determined dynamically during negotiation. For example, a person needs to book a flight and an accommodation before starting his/her vacation and at the same time, he/she does not know which one is more difficult to find. During negotiation, the agent working on behalf of that person can detect which one is more difficult to attain and decide on the order of agreements and resource allocation dynamically. The agent may find that booking a flight is way more difficult than booking an accommodation, then it decides to secure an agreement for the flight before securing an agreement for the accommodation.

3.2 Interdependency Amongst Issues

Each object under negotiation is characterized by one or more negotiation issues. Different issues can be interdependent in terms of their acceptable values. For example, an agent may accept to pay high price for a high quality product. When the negotiation issues are interdependent then the utility function is nonlinear, otherwise the utility function can be a weighted sum of the utility of each issue. Apart from dealing with the problem of searching for the best offer/counteroffer that can achieve the highest possible utility in case the utility function is nonlinear, we focus on the problem of allocating shared resources amongst different issues, e.g., price. In our work, we call the issues of different objects that share resources *common issues*.In other words, a *common issue* is an issue that is common amongst multiple objects. For example, multiple services can have the price issue as a *common issue*.

Definition 1 **A common negotiation issue** is an issue $j_i \in \mathbf{J}$ s.t. at least two subsets $J_k, J_l \in 2^{\mathbf{J}}$ exist where $j_i \in J_k \cap J_l$.

To this end, we propose managing resources shared amongst *common issues* as an approach for coordinating the bidding strategy which takes into consideration the behaviors of the opponents over the *common issues*. Managing the distribution of the available resources (which is part of the bidding strategy) is one solution for managing the interdependency problem.

4 Related Work Review and Analysis

This section addresses the coordination of automated negotiation in literature and analyzes possible negotiation scenarios where the need for some coordination mechanism is essential.

To help presenting and analyzing the related work in a systematic way, Fig. 3 shows possible negotiation scenarios taking into consideration the three main criteria that can determine a particular negotiation scenario, i.e., the *issues of negotiation*, the *number of opponents* and the *number of required agreements*. We consider that each *negotiation object* requires one *object agreement* and each issue requires one *issue agreement*. Formally, if an object o_i has k issues, then we need k *issue agreements* to make an *object agreement* for the object o_i. For the rest of this document, we call *an object agreement* an *agreement*.

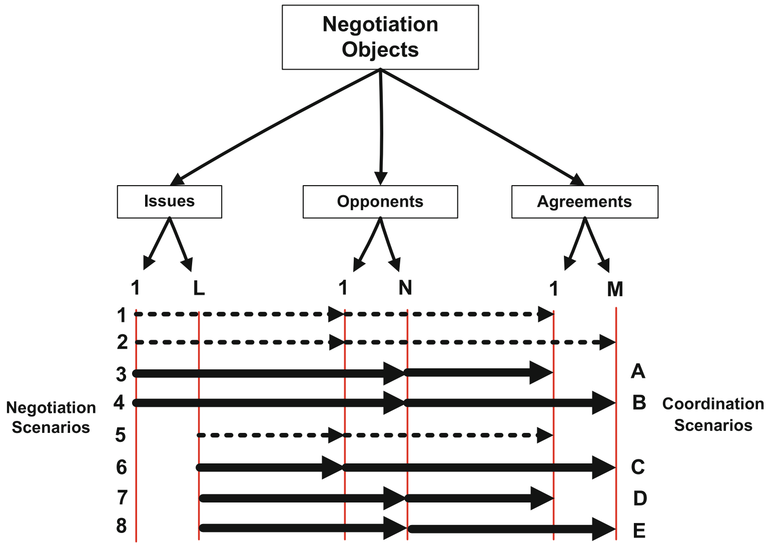

Fig. 3 Possible negotiation scenarios [11]

The *agreements node* in Fig. 3 refers to agreements over objects given that each negotiation object requires one agreement while that agreement requires multiple issue agreements in case the object has multiple issues.

For an agent interacting with other agents through negotiation, we can describe eight possible scenarios of interaction taking into consideration the main criteria of negotiation objects (i.e., the number of issues and the number of opponents) as shown in Fig. 3. The number of agreements indicates the number of negotiation objects under negotiation and vice versa.

Scenario 1 in Fig. 3 shows that an agent has one object characterized by one issue and negotiates with one opponent for the purpose of securing one agreement. As mentioned earlier, we assume that the number of objects is equal to the number of agreements, accordingly in Fig. 3, we can decide whether an agent negotiates over one object or more by looking at the arrow targeting the agreements node, if the arrow ends at 1, then the number of objects is 1, otherwise the number of objects is more than 1.

We consider Fig. 3 as our base for reviewing and analyzing the related work. Scenarios 1, 2, and 5 are basically bilateral negotiations where the number of objects/issues vary. For example scenario 1 represents the bilateral negotiation where an agent interacts with another agent over one issue while scenario 2 is also a bilateral negotiation where two agents negotiate over multiple objects given that each object has one issue.

This paper does not intend to investigate the bilateral negotiation, it rather focuses on the *one-to-many* negotiation where the coordination process is explicitly needed since there are multiple and related negotiation instances that need to be managed. In our work, we assume that the different actions of an agent during negotiation are related.

4.1 One-to-Many Negotiation

One-to-Many Negotiation Over a Single Issue During the last decade, work has been done to address the one-to-many negotiation as an alternative mechanism to the single-sided auction [5, 7, 12–15].

Adopting the one-to-many negotiation as an alternative to the single-sided auction has many advantages. Not only does the agent on the *one* side receive offers, but it also proposes counteroffers to each individual agent on the *many* side. Accordingly, the chance of reaching an agreement will improve since each agent in the negotiation process may analyze the previous offers aiming at predicting the preferences of its opponents and try to propose counteroffers that might improve the chance of reaching an agreement. For more details about the advantages of using the one-to-many negotiation over the reverse English auctions, see [16] and [17].

One of the first explicit architectures for the one-to-many negotiation was presented in [16] where the buyer agent consists of sub-negotiators and a coordinator. The sub-negotiators negotiate concurrently with a set of seller agents given that each sub-negotiator negotiates with one seller. That paper discusses four different coordination strategies: the desperate strategy in which the buyer agent accepts the first agreement and quits negotiations with all other sellers; the patient strategy where the buyer agent makes temporary agreements with some or all sellers during negotiation and holds on to these agreements until all the remaining instances of negotiations are finished, then the buyer agent selects the agreement with the highest utility; the optimized patient strategy is similar to the patient strategy except that it does not accept a new agreement with less utility than the highest accepted one; and finally the manipulation strategies in which the coordinator changes the negotiation strategies of its sub-negotiators during negotiation which were left for future work.

Other existing work [5, 17] develops coordination methods that change the negotiation strategy during negotiation. For example, the decision-making technique in changing the negotiation strategies [5] during negotiation depends on historic information about previous negotiations in terms of agreement rate and utility rate.

While the work of [7, 15] considers a decommitment penalty during negotiation, [9] assumes that the buyer agent incurs no penalty for exercising decommitment during negotiation and proposes a coordination mechanism to change the negotiation strategy during negotiation using only the current information during negotiation, i.e., the sellers's offers during negotiation. [9] argues that granting the buyer agent the privilege of reneging from an agreement without a penalty, while forcing the seller agents to honor their agreements can be a realistic scenario in situations where the number of seller agents is large and/or the seller agents are offering infinite supply, e.g., information. In such cases, a seller agent might be satisfied to make deals with many potential buyers in a hope that some of these buyers will confirm their deals later.

Some heuristic methods were proposed to estimate the expected utility in both synchronized multi-threaded negotiations and dynamic multi-threaded negotiations [18]. The synchronized multi-threaded negotiation model considers the existing outside options for each single negotiation instance, while the dynamic multi-threaded negotiation model considers also the uncertain outside options that might come in the future. In both cases, the methods assume a knowledge of the probability distribution of the reservation prices of the opponents. In many cases, this kind of information is not available.

While [14] proposes a decision-making strategy using Markov chains to decide whether to accept the best available offer or to proceed in negotiation with a hope to achieve a better deal, their work assumes that the buyer cannot make temporary deals with his opponents.

One-to-Many Negotiation Over Multiple Issues In real life, most negotiations involve more than one issue. For example, buying a laptop may involve negotiating the price of the laptop and both, the memory size and processor speed. If the agents participating in negotiation are competitive and self-interested, then the objective of each agent is to reach an agreement with the highest possible utility regardless of the opponents' needs or preferences. However, when negotiation involves multiple issues, agents usually have divergent preferences over different issues which allows reaching an efficient agreement for both parties, i.e., achieving a win–win outcome.

Our previous work investigates negotiation scenario **D** in Fig. 3 where a buyer agent negotiates with multiple seller agents over one object characterized by several issues [19]. For that scenario, we use a meta-strategy that uses two different offer generation tactics: the time-dependent tactics and the trade-off tactic. In each negotiation round, the buyer agent needs to decide on using a time-dependent tactic or the trade-off tactic depending on the behaviors of the opponents. During negotiation, the buyer agent assigns each seller agent to either a favorable group or unfavorable group. The favorable group offers more concessions than the concessions offered by the corresponding buyer agent's delegates. The meta-strategy is applied amongst the favorable group of seller agents.

One-to-Many Negotiation Over Multiple Objects To the best of our knowledge, little work has been done on that scenario. The work in [20] investigates scenario **C** in Fig. 3 where a buyer agent seeks agreements over multiple objects given that each object has several issues and a single provider. The work in [20] investigates the process of adapting the local reservation values during negotiation subject to the behaviors of the existing opponents, while the work in [21] involves adaptation of both the initially generated counteroffer values and the weight matrix of the counteroffers' issues during negotiation.

5 The Coordination Approach

During multi-bilateral concurrent negotiation, the buyer agent needs to coordinate its actions against its opponents in each negotiation round. One of the important actions is deciding on the bidding strategy that can be used to generate the next counteroffer. Part of that process is to distribute/redistribute the available resources amongst the buyer's delegates in a way to achieve the goal of the negotiation process in terms of reaching valuable agreements. Coordinating the buyer's actions in that context means managing the buyer's negotiation strategy during negotiation.

Formally, let Ω^a be the negotiation strategy of an agent a, then $\Omega^a = \langle IV^a, RV^a, T^a, \Theta^a \rangle$, where $IV^a, RV^a, T^a, \Theta^a$ stands for the initial offer value(s), the reservation value(s), the deadline(s), and the set of offer generation strategies of an agent a respectively.

Our representation of an agent's strategy Ω^a is similar to its representation in [21], the difference is that the fourth part of the strategy in [21] represents the β parameter in the time-dependent tactics [4] while the fourth part in our model (Θ^a) has a more general representation which indicates any possible offer/counteroffer generation method, e.g., trade-off, time-dependent, behavior dependent, etc., and their associated parameters. Any change to the components of Ω^a during negotiation means a change in agent a's negotiation strategy.

6 An Illustrative Example

This section shows some empirical results that compare between using a dynamic and a static strategy in coordinating the bidding strategy of different buyer's delegates. It is not in the scope of this paper to explain in detail about the specific dynamic strategy and experimental settings that produced the results shown in Fig. 4. The purpose of displaying Fig. 4 is to demonstrate the difference in performance between using a dynamic strategy that changes some of the negotiation strategy components (i.e., components of Ω) and a static strategy that initializes the strategy components and does not change them during negotiation. The results show that the buyer agent $T2$ who uses the dynamic strategy outperforms the buyer agent $T1$ who uses the static strategy in both utility rate and agreement rate.

For testing the agreement rate, we ran an experiment 500 times for each different number of seller agents, then the results were averaged. For example, when the number of seller agents per object is two, we repeated the experiment 500 times and then the results were averaged. We did the same for testing the utility rate. The dynamic strategy in this case involves assigning a new local reservation value for each issue of each object in each negotiation round depending on the behaviors of the current opponents in terms of their concessions. The experimental results shown in this section are related to scenario **B** in Fig. 3 where a buyer agent seeks to procure multiple objects given that each object has a single issue and multiple providers. The numbers of seller agents are shown on the top of Fig. 4a, b. Figure 4 shows that when the number of opponents increases, it is a favorable situation for both types of buyer agents since more seller agents means better opportunity for reaching agreements and getting better utility.

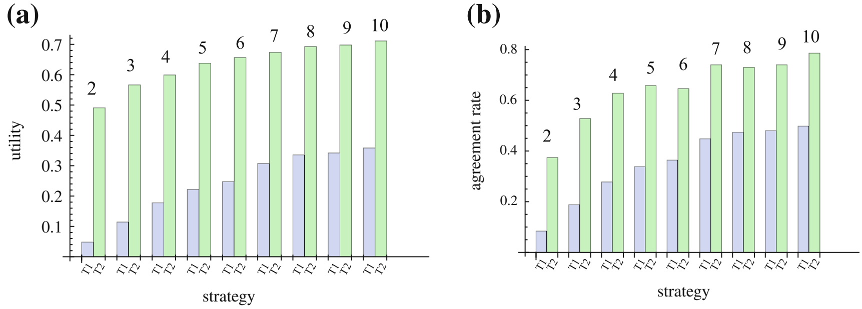

Fig. 4 Comparing between dynamic and static strategies. **a** Average utility. **b** Agreement rate

7 Conclusion

This paper addresses some aspects of coordination in concurrent one-to-many negotiation considering different sources of interdependency between the instances of multiple negotiations conducted concurrently by an agent. In our work, we consider a buyer agent negotiating concurrently with multiple seller agents for the purpose of procuring one or more objects. We propose adapting the negotiation bidding strategy in terms of distributing/redistributing of resources during concurrent negotiations subject to the behaviors of the current opponents. We further need to investigate adapting the bidding strategy of a buyer agent to generate counteroffers that improve the probability of reaching an agreement with the highest possible utility in different coordination scenarios, see Fig. 3. In other words, we need to investigate what negotiation strategy components or parameters to select for adaptation in each scenario to design an effective and robust dynamic negotiation strategy.

Acknowledgments This work was partially supported by Smart Services CRC.

References

1. Raiffa H (1982) The Art and Science of Negotiation. Harvard Univiversity Press, Cambridge
2. Malone TW, Crowston K (1994) The interdisciplinary study of coordination. ACM Comput Surv 26(1):87–119
3. Rubinstein A (1982) Perfect Equilibrium in a Bargaining Model. Econometrica 50(1): 97–109
4. Faratin P (2000) Automated service negotiation between autonomous computational agents. Ph.D thesis, University of London
5. Nguyen TD, Jennings NR (2004) Coordinating multiple concurrent negotiations. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems, New York, USA. pp 1062–1069
6. Yan J, Zhang J, Lin J, Chhetri M, Goh S, Kowalczyk R (2006) Towards autonomous service level agreement negotiation for adaptive service composition. In: Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design, pp 1–6
7. Nguyen T, Jennings N (2005) Managing commitments in multiple concurrent negotiations. Electron Commer Res Appl 4(4):362–376
8. Richter J, Chhetri MB, Kowalczyk R, Vo QB (2011) Establishing composite SLAs through concurrent QoS negotiation with surplus redistribution. Concurrency and Computation: Practice and Experience
9. Mansour K, Kowalczyk R, Vo QB (2010) Real-time coordination of concurrent multiple bilateral negotiations under time constraints. LNAI 6464:385–394
10. Zhang X, Lesser V, Abdallah S (2005) Efficient management of multi-linked negotiation based on a formalized model. Auton Agent Multi-Agent Syst 10(2):165–205
11. Mansour K (2012) Managing concurrent negotiations in multi-agent systems. In: Proceedings of the Canadian Conference on AI, pp 380–383
12. Gerding EH, Somefun DJa, La Poutré JA (2004) Automated bilateral bargaining about multiple attributes in a one-to-many setting. In: Proceedings of the 6th international conference on Electronic commerce-ICEC ’04, p 105

13. Yan J, Zhang J, Lin J, Chhetri M, Goh S, Kowalczyk R (2007) Autonomous service level agreement negotiation for service composition provision. Future Gener Comput Syst 23(6):748–759
14. An B, Sim KM, Miao CY, Shen ZQ (2008) Decision making of negotiation agents using markov chains. Multiagent Grid Syst 4:5–23
15. An B, Lesser V, Irwin D, Zink M (2010) Automated negotiation with decommitment for dynamic resource allocation in cloud computing. In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010), Toronto, pp 981–988
16. Rahwan I, Kowalczyk R, Pham HH (2002) Intelligent agents for automated one-to-many e-commerce negotiation. In: Proceedings of the Twenty-Fifth Australian Computer Science Conference, Melbourne, Australia, pp 197–204
17. Nguyen TD, Jennings NR (2003) Concurrent bi-lateral negotiation in agent systems. In: Proceedings of the Fourth DEXA Workshop on E-Negotiations
18. Cuihong L, Giampapa J, Sycara K (2006) Bilateral negotiation decisions with uncertain dynamic outside options. IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), 36(1):31–44
19. Mansour K, Kowalczyk R (2012) A meta-strategy for coordinating of one-to-many negotiation over multiple issues. In: Wang Y, Li T (eds) Foundations of Intelligent Systems, Shanghai, Springer, Heidelberg, pp 343–353
20. Mansour K, Kowalczyk R, Chhetri MB (2012) On effective quality of service negotiation. In: Proceedings of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp 684–685
21. Fatima S (2004) An agenda-based framework for multi-issue negotiation. Artificial Intelligence 152(1):1–45