J Supercomput
DOI 10.1007/s11227-012-0808-4

Negotiation strategies considering market, time and behavior functions for resource allocation in computational grid

Sepideh Adabi · Ali Movaghar · Amir Masoud Rahmani · Hamid Beigy

Abstract Providing an efficient resource allocation mechanism is a challenge to computational grid due to large-scale resource sharing and the fact that Grid Resource Owners (GROs) and Grid Resource Consumers (GRCs) may have different goals, policies, and preferences. In a real world market, various economic models exist for setting the price of grid resources, based on supply-and-demand and their value to the consumers. In this paper, we discuss the use of multiagent-based negotiation model for interaction between GROs and GRCs. For realizing this approach, we designed the Market- and Behavior-driven Negotiation Agents (MBDNAs). Negotiation strategies that adopt MBDNAs take into account the following factors: *Competition*, *Opportunity*, *Deadline* and *Negotiator's Trading Partner's Previous Concession Behavior*. In our experiments, we compare MBDNAs with MDAs (Market-Driven Agent), NDF (Negotiation Decision Function) and Kasbah in terms of the following metrics: total tasks complementation and budget spent. The results show that by taking the proposed negotiation model into account, MBDNAs outperform MDAs, NDF and Kasbah.

S. Adabi (✉) · A.M. Rahmani
Department of computer engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran
e-mail: adabi.sepideh@ieee.org

A.M. Rahmani
e-mail: rahmani@srbiau.ac.ir

A. Movaghar · H. Beigy
Sharif University of Technology, Tehran, Iran

A. Movaghar
e-mail: movaghar@sharif.edu

H. Beigy
e-mail: beigy@sharif.edu

Published online: 24 July 2012

Keywords Computational grid · Grid resource management · Grid resource allocation · Negotiation model · Software agent

1 Introduction

In the last decade, the resource sharing paradigm proposed by grid is gaining more and more importance both in the academia and in the industry [1] and many large corporations are currently using computational grids to improve their operations [2]. As grid resource owners (GROs) and grid resource consumers (GRCs) may have different goals, preferences and policies, which are characterized and specified through a *utility model* (or *utility function*), an efficient resource management, is central to its operations [3]. Resource management refers to the operations used to control how capabilities provided by grid resources and services are made available to other entities such as users, applications, or services [4].

Utilization of grid resource is not for free [5], which means that the Grid Resource Owners (GROs) charge Grid Resource Consumers (GRCs) according to the amount of resource they consume, so adapting some of the successful ideas of economical models to resource allocation in large-scale computing systems is essential for realizing the vision of grid computing environments [6]. In recent years, usage of market-based methods (i.e., A market method is the overall algorithmic structure within which a market mechanism or principle is embedded [7]) for grid resource management is one of the solutions which has received much attention [8]. Numerous economic models [9], including microeconomic and macroeconomic principles for resource management, are surveyed in [10–15]. As a negotiation-like protocol is found to be suitable when the participants cooperate to create value [16, p. 6], adopting negotiation mechanism for successfully reconciling the differences between GROs and GRCs seems to be more prudent rather than using other commonly referenced work (e.g., see [17–20]). Sim [21] pointed out some issues that should be considered in building the negotiation mechanism for grid resource management: (1) modeling devaluation of resources, (2) considering market dynamics, (3) relaxing bargaining criteria, and (4) resource co-allocation. To complete the issues of [21] we present another issue that should be considered in building the efficient negotiation mechanism for grid resource management: (5) modeling the decision criteria that are used by negotiators of real-life trading market for selecting the pattern of concession during negotiation process. The importance of such improved and extended negotiation model is when the designers of negotiation agents have to face with two opposite concepts: time of acquiring grid resources (respectively, leasing grid resources) and price of acquiring grid resources (respectively, price of leasing grid resources). It means that GRCs (respectively, GROs) should achieve lower utilities to avoid the risk of losing deals to other competitors (and vice versa). Like most of the commonly previous work in the grid environment (e.g., see [22–27]), we propose a new negotiation model for optimizing GROs' and GRCs' profit through providing software components (*Agent*). The software agents that make adjustable amounts of concession by considering *Competition*, *Opportunity*, *Time* and *Previous Concession Behavior of Negotiator's Trading Partner* factors are called *MBDNAs* (Market- and Behavior-driven Negotiation Agents).

The new features of this work are:

(a) Designing a *new* multiagent-based strategic negotiation model for both bilateral and multilateral negotiations. This is so important that not only bilateral negotiation (where resources are provided by one agent and thus an agent is negotiating with one trading partner) but also multilateral negotiation (where resources are provided by multiple agents and thus an agent is negotiating with multiple trading partners) is considered in designing negotiation model. Multilateral negotiation is more realistic in resource allocation process of computational grids where there are more than one seller that sell special type of resource.
(b) Modeling concession behavior of negotiator's trading partner which is inspired by the real-life trading market. In real-life trading market the behavior of one negotiator serves as a stimulus for the other negotiator who then screens it, selects its key elements and tries to interpret them [28]. Negotiators should view their trading partners' behavior to select suitable tactics and strategies [28]. There are few existing negotiation agents that consider behavior-dependent function to determine the amount of concession during negotiation process (e.g., [29–31]). Whereas these negotiation agents using complex techniques (like artificial intelligence) that need more computational cost for modeling the behavior function, our work proposes a simple and applicable approach to model the concession behavior of negotiator's trading partner. The importance of such an approach is when the negotiation agents have short deadline and cannot tolerate extra computational cost to make near optimal concession amount. In addition we present two new criteria to classify the behavior of negotiator's opponents: *royalty* and *hasty* which are defined based on the number of successful negotiations between a negotiator and its trading partner in all the GRNMs (grid resource negotiation markets) they both participated and the average negotiation time between a negotiator and its trading partner in all GRNMs which both participate, respectively.
(c) Determining the specific amount of concession to each negotiator's trading partner separately, instead of the same amount to all. Although there are many agent-based systems for negotiation in e-commerce (e.g., just to name a few: NDF [32], 2-phase negotiation [33], service negotiation [34], Kasbah [35], Tete-a-Tete [36], MDA and EMDA [37–40], Zhao and Li [41], SNAP [42–44] and An [45]), the strategies of most of them make the same concession amount for all negotiators' trading partners. In contrast, our work considers different concession amount for different negotiator's trading partners (by applying a multicriteria decision function) which provides more flexibility in keeping the chance of making deal (by computing rational and sufficiently minimum price) with more than one opponent.

The remainder of the paper is structured as follows. In Sect. 2, some of the most well known negotiation models for resource management are reviewed. In Sect. 3, proposed four-phase scenario for resource allocation in computational grid and the proposed multiagent-based strategic negotiation model as the heart of the scenario is explained. The simulation configuration and experimental results are analyzed in Sect. 4. Conclusions and information on future work are given in Sect. 5.

2 Related work

In this section we review and compare the existing state-of-the-art negotiation agents from the issues for making a negotiation model in [21] and our extra proposed issue for making appropriate negotiation model points of view.

Whereas the agents in NDF [32], 2-phase negotiation [33], service negotiation [34], Kasbah [35], Tete-à-Tete (extended Kasbah, which focuses on multiple-issue negotiation rather than single-issue negotiation) [36], MDA and EMDA [37–40], Zhao and Li [41] and An [45] considered the issue of time constraint, the agents in SNAP [42–44] and policy-driven negotiation [46] did not consider this issue in designing the agents.

2-phase negotiation [33], MDA and EMDA [37–40] and [45] modeled market dynamics in their concession making strategies, but NDF [32], service negotiation [34], Kasbah [35], Tete-à-Tete [36], SNAP [42–44] and policy-driven negotiation [46] and [41] did not consider the market factors in making concession amount.

Among the reviewed negotiation models, just service negotiation model [34] considered the influence of behavior-dependent functions on the negotiation results. Also, no other reviewed protocol, excepting SNAP [42–44], addresses the influence of grid resource co-allocation factor on the negotiation results in the grid resource allocation process.

While the protocol adopted by [32, 35, 46–48] is simply a bilateral exchange of messages, the protocol adopted by 2-phase negotiation [33], service negotiation [34], MDAs [39, 40], EMDAs [37, 38], Zhao and Li [41] and An [45] is concerned with alternating offers. In comparison to alternating offers protocol, bilateral exchange of messages protocol provides less flexibility in not allowing multiple messages from both GROs and GRCs to be exchanged.

3 Proposed four-phased scenario for resource allocation in computational grid

This work considers grid environment as a collection of virtual organizations (VOs), which is a group of GRCs and GROs collaborating to facilitate usage of high-end computational resources. VO is formed dynamically while the members (e.g., GRCs | GROs) of grid domain join/leave it. As both GROs and GRCs want to maximize their profit (i.e., the GROs wish to increase their revenue and the GRCs to solve their problems within a minimum possible cost), an economy-aware grid needs to support this challenge. To realize this, a *Multiagent-based Strategic Negotiation Model* for resource allocation and for regulation of supply and demand in grid computing environments is proposed. The proposed *Multiagent-based Strategic Negotiation Model* is at the heart of four-phase scenario for grid resource allocation.

The scenario of resource allocation in the economy-aware grid environment includes the following four major phases:

1. *Registering GRCs and GROs*
2. *Creating MBDNAs and providing the required information* (*that is, necessary for starting negotiation*)
3. *Starting negotiation based on proposed strategy*
4. *Terminating Negotiation process and executing task* (*if negotiation is successful*)

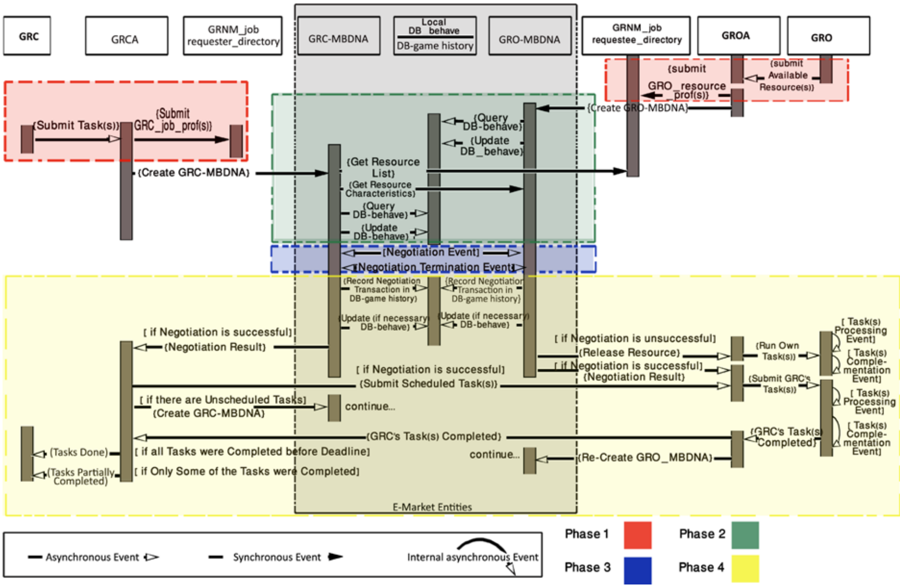


Fig. 1 Event diagram showing message-flow in the proposed four-phase scenario (for grid resource allocation)

The proposed scenario is based on synchronous and asynchronous message exchange systems. A general overview of the event diagram is shown in Fig. 1.

3.1 Registering GRCs and GROs

Each GRC that is represented by a GRC agent (e.g., *GRCA*) can have one or more jobs $\{job_1, \ldots, job_p\}$. Jobs submitted by GRCs into a cluster have varying requirements depending on GRC-specific needs and expectations. The GRC_i's pth job characteristics (e.g., $GRC\_job\_prof^i_p$) include the following: unique identifier, job length measured in MI (millions of instructions), length of input and output data, earliest start time (i.e., the job cannot start before its earliest start time), the period of resource usage, job's negotiation deadline (i.e., the latest start time of the job. Obviously, a job's finish time $\in$ [*earliest start time* + *period of resource usage*, *negotiation deadline* + *period of resource usage*]), initial price, reservation price, and the originator of the job [39].

Also, it is assumed that each GRO, which is represented by a GRO agent (e.g., *GROA*), may possess k computing machines (which is denoted by $\{M_{j1}, \ldots, M_{jk}\}$) for the grid environment. As noted in [39, p. 1384], "*Each computing machine* M_{jk} *can be a single processor, a shared memory multiprocessor, or a distributed memory cluster of computers.* M_{jk} *can be formed by one or more processing elements* $\{PE_1, \ldots, PE_l\}$, and each PE_i *can have different speeds measured in terms of MIPS (millions of instructions per second).*" The GRO_j's rth resource characteristics (e.g., $GRO\_resource\_prof^j_r$) include unique identifier, the architecture of computing resource (e.g., HPalpha server), list of computing machines (e.g., $\{M_{j1}, \ldots, M_{jk}\}$), required bandwidth length, required memory capacity, and expected and reserve prices of leasing a computing machine.

The $GRCA_i$ (respectively, $GROA_j$) should register each of its $GRC\_job\_prof^i_p(s)$ (respectively, $GRO\_resource\_prof^j_r[s]$) in *GRNM_job requester_directory* (respectively, *GRNM_job requestee_directory*).

3.2 Creating MBDNAs and providing their required information

It was noted in [21, p. 245] that "*software agents, in particular, negotiation agents, can play an essential role in realizing the grid vision*". Software Agent is a component with the capability of accomplishing its tasks on behalf of its owner [49]. In this work, MBDNAs (which are categorized into *GRC_MBDNA* and *GRO_MBDNA* entities) are expected to realize the grid vision. A GRC_MBDNA_i (respectively, GRO_MBDNA_j) is generated according to $GRCA_i$ (respectively, $GROA_j$), which is registered in GRNM to perform the negotiation process.

In the following sections, each *GRC_MBDNA* (respectively, *GRO_MBDNA*) is represented by A symbol for ease of reading. Also let us assume that kth trading partner of negotiator A is denoted by B_k.

Following are the functions performed by negotiator A of type *GRC_MBDNA* in the second phase of resource allocation scenario:

1. Start the process of resource discovery (e.g., discovering appropriate $GRO\_MBDNA(s)$ that match with the A's requirements).
2. Query its local *DB_behave* database (which is considered to store the previous concession behavior of negotiators' trading partners who participated in GRNM previously) to retrieve all records (if they exist) for which the value of their B_k_id field is equal to the identifier of one of A's trading partners. The retrieved records are used to calculate the previous concession behavior of negotiators' trading partners (details are provided in Sect. 3.3.3).
3. Increase the $\#GRNM_{B_k-A}$ field of retrieved records by one.

And the functions that are performed by negotiator A of type *GRO_MBDNA* in the second phase of resource allocation scenario are as same as the second and third functions performed by negotiator A of type *GRC_MBDNA*.

3.3 Starting negotiation, based on the proposed strategic negotiation model

The negotiation model has three parts [50]: (1) the used utility models or preference relationships for the negotiating parties, (2) the negotiation strategy applied during the negotiation process and (3) the negotiation protocol. The negotiation model in this work applies a new negotiation strategy which not only models the market conditions and time but also models concession behavior of negotiator agent's trading partner to determine the appropriate amount of concession. The *new* multicriteria negotiation strategy maximizes the negotiators' achieved utility and improves their success rate for both bilateral and multilateral negotiations.

The negotiation agents that adopt new proposed negotiation model are called MBDNAs (Market- and Behavior-driven Negotiation Agents). The following three sub-sections address the three parts of negotiation model in the proposed MBDNAs' negotiation mechanism.

3.3.1 Negotiation utility model

Utility functions are used to express grid user's Quality of Service (QoS) requirement, resource provider's benefit function and system's objectives [51]. The grid computational resource allocation mechanism in this paper is under budget and time constraints which means that a negotiator A of type *GRC_MBDNA* (respectively, *GRO_MBDNA*) makes computational resource acquiring (respectively, assigning) decisions within the budget and time constraints. That is, the negotiation objectives are the expected price that will be obtained via negotiation process and the negotiation time that will be spent in the grid resource allocation market. So, the negotiator A of type *GRC_MBDNA* tries to purchase as much computational resource as possible with the objectives of spending the least possible amount of money (minimizing its payment) and minimizing its negotiation time, also the negotiator A of type *GRO_MBDNA* tries to sell as much computational resources as possible with the objectives of maximizing its revenue and minimizing its negotiation time.

Let us assume that number of negotiator agent A's trading partners and competitors at round t are $no.trading\_partner_t^A$ and $no.competitor_t^A$ respectively. Negotiator agent A duplicates itself according to $no.trading\_partner_t^A$ and creates negotiator agent instances $A\_CHILD^t = \{A\_child_1, A\_child_2, \ldots, A\_child_{no.trading\_partner_t^A}\}$ to conduct negotiation process on behalf of it in the *GRNM*[1] *area* that are assigned to. For understanding the meaning of *GRNM area* an example will be presented. Let assume that negotiator agent A has $no.trading\_partner_t^A = 3$ trading partners and $no.competitor_t^A = 6$ competitors at round t. We consider that negotiator agent A finds $competitor_1$, $competitor_3$, $competitor_4$ and $competitor_6$ as the potential competitors against $trading\_partner_1$, $competitor_1$, $competitor_3$ and $competitor_5$ against $trading\_partner_2$ and $competitor_4$ and $competitor_6$ against $trading\_partner_3$. The *GRNM area* is composed of A_child_k (i.e., kth instance of A), one of its trading partner and competitors those are found against that trading partner. Therefore as shown in Fig. 2, three *GRNM areas* are found in the described example.

The utility of A_child_k if B_k (i.e., A_child_k's trading partner in that *GRNM area* that A_child_k is located) accepts A_child_k's proposal (i.e., $P_t^{A\_child_k}$) and the utility generated for A_child_k if A_child_k accepts the last counter-proposal of B_k (i.e., $P_{t-1}^{B_k}$) are $U_t^{A\_child_k}[P_t^{A\_child_k} \to B_k]$ and $U_t^{A\_child_k}[P_{t-1}^{B_k} \to A\_child_k]$ respectively. If the negotiation ends in disagreement, both negotiation sides (e.g., negotiator agent of type *GRC_MBDNA* and negotiator agent of type *GRO_MBDNA*) receive the worst possible utility (e.g., zero). We should highlight that by using *Rubinstein's sequential alternating offer protocol* [52], negotiators in make alternate offers rather than moving simultaneously (details are described in Sect. 3.3.3).

For ease of analysis, the utility function of A_child_k of type *GRC_MBDNA* considering $P_t^{A\_child_k}$ to B_k and $P_{t-1}^{B_k}$ to A_child_k at negotiation round t can be expressed as (1):

$$\begin{aligned} &U_t^{A\_child_k}\big[P_t^{A\_child_k} \to B_k\big] = \big(RP_A - P_t^{A\_child_k}\big)/(RP_A - IP_A) \quad \text{and} \\ &U_t^{A\_child_k}\big[P_{t-1}^{B_k} \to A\_child_k\big] = \big(RP_A - P_{t-1}^{B_k}\big)/(RP_A - IP_A) \end{aligned} \tag{1}$$

[1] Grid resource negotiation market.

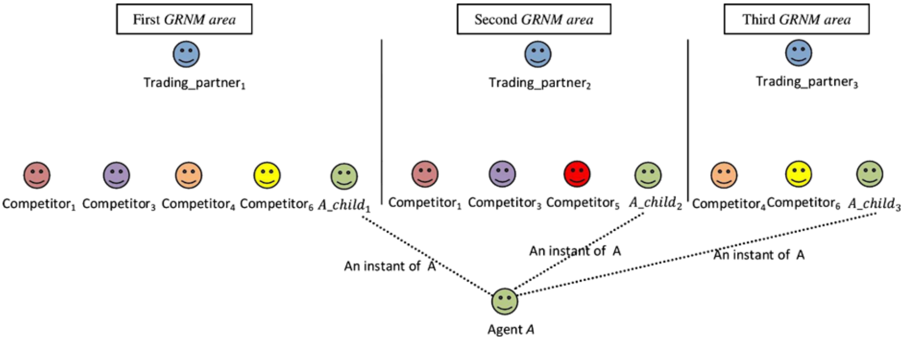


Fig. 2 An example to show *GRNM area* concept

where RP_A is A's reserve price, IP_A is A's initial price, $P_t^{A\_child_k}$ is A_child_k's potential proposal at negotiation round t and $P_{t-1}^{B_k}$ is B_k's proposal at negotiation round $t-1$. For example a *GRC_MBDNA* in name A considers 100\$ to buy a special type of resource (i.e., $RP_A = 100\$$) and starts the negotiation process with 20\$ (i.e., $IP_A = 20\$$). From A's perspective 20\$ is the best price that can be paid to buy that type of resource (as 20\$ generates the highest utility for A, $[(100\$ - 20\$)/(100\$ - 20\$)] = 1$) and saves 80\$ for him. Also from A's perspective 100\$ is the worst price that can be paid to buy that type of resource (as 100\$ generates the lowest utility for A, $[(100\$ - 100\$)/(100\$ - 20\$)] = 0$) and saves nothing for him. Furthermore, let us assume that the proposed price from B_k (i.e., kth trading partner of) at negotiation round $t-1$ is 62\$. At negotiation round t the kth child of A (i.e., A_child_k) makes its potential concession amount by considering current market situation. Let assume that the potential concession amount of A_child_k that can be proposed to B_k is equal to 50\$. Now A_child_k should decide to accept 62\$ or continue the negotiation process by proposing 50\$. This decision is made by computing the utilities generated from 62\$ and 50\$ as follows: $U_t^{A\_child_k}[P_{t-1}^{B_k} \to A\_child_k] = [(100\$ - 62\$)/(100\$ - 20\$)]$ and $U_t^{A\_child_k}[P_t^{A\_child_k} \to B_k] = [(100\$ - 50\$)/(100\$ - 20\$)]$. By comparing the generated utilities of 50\$ and 62\$, A_child_k decides to continue the negotiation process instead of accept the counter offer. Rationally, from *GRC_MBDNA*'s perspective the price that saves more money is considered as more appropriate price.

Also the utility function of A_child_k of type *GRO_MBDNA* considering $P_t^{A\_child_k}$ to B_k and $P_{t-1}^{B_k}$ to A_child_k at negotiation round t can be expressed as (2):

$$\begin{aligned} &U_t^{A\_child_k}\left[P_t^{A\_child_k} \to B_k\right] = \left(P_t^{A\_child_k} - RP_A\right)/(IP_A - RP_A) \quad \text{and} \\ &U_t^{A\_child_k}\left[P_{t-1}^{B_k} \to A\_child_k\right] = \left(P_{t-1}^{B_k} - RP_A\right)/(IP_A - RP_A) \end{aligned} \tag{2}$$

where RP_A is A's reserve price, IP_A is A's initial price, $P_t^{A\_child_k}$ is A_child_k's potential proposal at negotiation round t and $P_{t-1}^{B_k}$ is B_k's proposal at negotiation round $t-1$. For example a *GRO_MBDNA* in name A cannot sell its resource less than 20\$ (i.e., $RP_A = 20\$$) and starts the negotiation process with 100\$ (i.e., $IP_A = 100\$$). From A's perspective 100\$ is the best price that can be achieved in trading process (as 100\$ generates the highest utility for A, $[(100\$ - 20\$)/(100\$ - 20\$)] = 1$) and makes maximum revenue (i.e., 80\$) for him. Also from A's perspective 20\$ is the worst price that can be achieved in trading process (as 20\$ generates the lowest utility for A, $[(20\$-20\$)/(100\$-20\$)]=0$) and makes no profit for him. Furthermore, let us assume that the proposed price from B_k (i.e., kth trading partner of) at negotiation round $t-1$ is 50\$. At negotiation round t the kth child of A (i.e., A_child_k) makes its potential concession amount by considering current market situation. Let assume that the potential concession amount of A_child_k that can be proposed to B_k is equal to 62\$. Now A_child_k should decide to accept 50\$ or continue the negotiation process by proposing 62\$. This decision is made by computing the utilities generated from 50\$ and 62\$ as follows: $U_t^{A\_child_k}[P_{t-1}^{B_k} \rightarrow A\_child_k]=[(50\$-20\$)/(100\$-20\$)]$ and $U_t^{A\_child_k}[P_t^{A\_child_k} \rightarrow B_k]=[(62\$-20\$)/(100\$-20\$)]$. By comparing the generated utilities of 50\$ and 62\$, A_child_k decides to continue the negotiation process instead of accept the counter offer. Rationally, from *GRO_MBDNA*'s perspective the price that makes more profit is considered as more appropriate price.

If the proposed deal from A_child_k of type *GRC_MBDNA* at round t (e.g., $P_t^{A\_child_k}$) is not greater than the one at round $t+2$ (e.g., $P_{t+2}^{A\_child_k}$), then $U_t^{A\_child_k}[P_t^{A\_child_k} \rightarrow B_k] > U_{t+2}^{A\_child_k}[P_{t+2}^{A\_child_k} \rightarrow B_k]$. Also, If the proposed deal from A_child_k of type *GRO_MBDNA* at round t (e.g., $P_t^{A\_child_k}$) is greater than the one at round $t+2$ (e.g., $P_{t+2}^{A\_child_k}$), then $U_t^{A\_child_k}[P_t^{A\_child_k} \rightarrow B_k] > U_{t+2}^{A\_child_k}[P_{t+2}^{A\_child_k} \rightarrow B_k]$.

3.3.2 Negotiation strategy

In each round of the negotiation, a negotiator agent A's choice is called a *strategy*. While some other negotiation mechanisms (like [53–55]) are focused on multi-issue negotiation that aim to balance the QoS constraints, MBDNAs focus on single-issue (e.g., price-only) negotiation (like [24, 39, 56–62]). This is because our first goal is to design MBDNAs with price-oriented negotiation strategies by considering a new simple and applicable mechanism and second consider more complex techniques (like fuzzy decision making approach) that extend the suitable proposed price-oriented strategies of MBDNAs to handle more than one QoS parameter. The first goal is considered in this paper and the second goal will be considered in the future work. Hence, the amount of concession determination, at negotiation round t, is a chosen strategy by A. The following concession functions of proposed MBDNAs are described.

Sim [40] investigated the way to assess the probability of successfully reaching a consensus in different market situations by considering the difference between the payoffs generated by the proposal of negotiator A_child_k and the proposal of its trading partners at each round t. Coming to details, the proposal of A_child_k to its trading partner B_k at round $t-2$ is $P_{t-2}^{A\_child_k} \rightarrow B_k$ and the proposal of B_k to A_child_k at round $t-1$ is $P_{t-1}^{B_k} \rightarrow A\_child_k$, also, $U_t^{A\_child_k}[P_{t-2}^{A\_child_k} \rightarrow B_k]$ and $U_t^{A\_child_k}[P_{t-1}^{B_k} \rightarrow A\_child_k]$ be the utilities of A_child_k at negotiation round t if B_k accepted A_child_k's proposal which was proposed at negotiation round $t-2$ and the best utility generated for A_child_k if A_child_k accepts the last counter-proposal of

$B_k \in \{B_1, B_2, \ldots, B_{no.trading\_partner^A_{t-1}}\}$ respectively. The (best) spread in the current cycle t (before making new proposal) is

$$k_t = U_t^{A\_child_k}\left[P_{t-2}^{A\_child_k} \to B_k\right] - U_t^{A\_child_k}\left[P_{t-1}^{B_k} \to A\_child_k\right]. \tag{3}$$

Negotiation is described as a process where the parties attempt to narrow the spread in (counter-) proposals between (or among) negotiators through concession; therefore, for making a suitable concession the *expected utility* of each negotiator's next proposal is determined by itself as follows:

$$U_t^{A\_child_k}\left[P_t^{A\_child_k} \to B_k\right] = k_{t+1} + U_t^{A\_child_k}\left[P_{t-1}^{B_k} \to A\_child_k\right]. \tag{4}$$

Finally, the amount of concession at round t (e.g., Δ_t) is

$$\Delta_t = k_t - k_{t+1} \tag{5}$$

where the appropriate value of k_{t+1} is defined thus:

$$k_{t+1} = FST_t^{A\_child_k} \times k_t \tag{6}$$

$FST_t^{A\_child_k}$ is a price-oriented strategy that is taken by A_child_k to determine the amount of concession at round t and is defined through (7):

$$FST_t^{A\_child_k} = 1 - \left|IST_t^{A\_child_k} - \left(1 - PreBehave\_Depend_t^{B_k}\right) \times \kappa)\right| \tag{7}$$

where $\kappa = 1/(1 - PreBehave\_Depend_t^{B_k})$ if $IST_t^{A\_child_k} = 0$, else $\kappa = 1$. Also $PreBehave\_Depend_t^{B_k}$ is *Previous Concession Behavior of* A_child_k*'s Trading Partner* factor and $IST_t^{A\_child_k}$ is denoted by (8):

$$IST_t^{A\_child_k} = T_t^{A\_child_k} \times O_t^{A\_child_k} \times CC_t^{A\_child_k} \tag{8}$$

where $T_t^{A\_child_k}$, $O_t^{A\_child_k}$ and $CC_t^{A\_child_k}$ are *Time*, *Opportunity*, and *Competition* functions of negotiator A_child_k, respectively.

The following four sub-sections address *Time, Opportunity, Competition* and *Behavior* functions in detail.

(a) *Time function* ($T_t^{A\_child_k}$)

As noted by Binmore and Dasgupta [60] "*The passage of time has a cost in terms of both dollars and the sacrifice of utility which stems from the postponement of consumption, and it will be precisely this cost which motivates the whole bargaining process. If it did not matter when the parties agreed, it would not matter whether they agreed at all.*" The effect of time discount factor in negotiator's bargaining power can be modeled via time-dependent function. The present work focuses on time-dependent function that is given in [39] as follows:

$$T_t^{A\_child_k}\left(t, t^A_{\text{deadline}}, \lambda\right) = 1 - \left(\frac{t}{t^A_{\text{deadline}}}\right)^{\lambda} \tag{9}$$

where A's time preference is denoted by λ (e.g., concession rate with respect to time. For instance, an agent may prefer to concede less rapidly in the early rounds of negotiation and more rapidly as its deadline approaches), A's deadline (e.g., a time frame by which A needs negotiation result-in other words it corresponds to the latest start time) by t^A_{deadline} and current negotiation round by t. The two parameters λ and t^A_{deadline} are considered private information. The following are the three major classes of concession-making strategies with respect to the remaining trading time (details are discussed in [39, 40]):

(i) *Conservative* $(1 < \lambda < \infty)$—An agent A_child_k makes smaller concession in early rounds and larger concession in later rounds.
(ii) *Linear* $(\lambda = 1)$—An agent A_child_k makes a constant rate of concession.
(iii) *Conciliatory* $(0 < \lambda < 1)$—An agent A_child_k makes larger concession in the early trading rounds and smaller concessions in the later rounds.

According to (9), the concession rate that is made by A_child_k should be increased as $T_t^{A\_child_k}$ tends to become *zero* (e.g., negotiator's deadline is reached). Obviously the value of *Time* function of all negotiator A's children is the same.

(b) *Opportunity Function* (O_t^A)

Opportunity is defined as the subjective probability that the agent will obtain a certain expected utility with at least one of its trading partners. From *Opportunity* factor's perspective, the amount of concession is determined based on the number of agent's trading partner and differences in proposals and counter proposals thus [39]:

$$O_t^A\big(no.trading\_partner_t^A, \langle U_t^A\big[P_{t-2}^A \to B_k\big]\rangle, \langle U_t^A\big[P_{t-1}^{B_k} \to A\big]\rangle\big) \\ = 1 - \prod_{j=1}^{no.trading\_partner_t^A} \frac{U_t^A[P_{t-2}^A \to B_k] - U_t^A[P_{t-1}^{B_k} \to A]}{(U_t^A[P_{t-2}^A \to B_k] - c^A)} \tag{10}$$

where c^A is the worst possible utility for A (e.g., if the negotiation ends in disagreement) and $U_t^A[P_{t-2}^A \to B_k] - U_t^A[P_{t-1}^{B_k} \to A]$ measures the cost of accepting the trading partner's last proposal. Obviously the value of *opportunity function* of all negotiator A's children is the same. According to (10), the concession rate that is made by A_child_k should be increased as O_t^A tends to become *zero*.

(c) *Competition Function* $(CC_t^{A\_child_k})$

As mentioned in [40, p. 714], "Since market-driven agents are utility maximizing agents, an agent A is more likely to reach a consensus if its proposal is ranked the highest by some other agent B_i". Let an agent A_child_k has $no.competitor_t^{A\_child_k}$ competitors at round t. If the last proposal of A_child_k's competitor agent (e.g., $AC_l \in \{AC_1, AC_2, \ldots, AC_{no.competitor_t^{A\_child_k}}\}$) generates a utility $U_t^{B_k}[P_{t-1}^{AC_l} \to B_k]$ for B_k and the last proposal of A_child_k generates a utility $U_t^{B_k}[P_{t-1}^{A\_child_k} \to B_k]$ for B_k, by considering the mentioned concept, the proposal of A_child_k is ranked the highest by B_k if $U_t^{B_k}\big[P_{t-1}^{A\_child_k} \to B_k\big] > \forall U_t^{B_k}\big[P_{t-1}^{AC_i} \to B_k\big] \in \big\{U_t^{B_k}\big[P_{t-1}^{AC_1} \to B_k\big], U_t^{B_k}\big[P_{t-1}^{AC_2} \to B_k\big], \ldots, U_t^{B_k}\big[P_{t-1}^{AC_{no.competitor_t^{A\_child_k}}} \to B_k\big]\big\}$. So, the probability of A_child_k being considered the most preferred trading partner by B_k is calculated thus:

$$CC_t^{A\_child_k}\big(no.competitor_t^{A\_child_k}\big) = 1 - \big[no.competitor_t^{A\_child_k} \big/ \big(no.competitor_t^{A\_child_k} + 1\big)\big]. \tag{11}$$

According to (11), the concession rate that is made by A_child_k should be increased as $CC_t^{A\_child_k}$ tends to become *zero*.

(d) *Behavior Function* ($PreBehave\_Depend_t^{B_k}$)

Recall that in real-life trading market the behavior of one negotiator serves as a stimulus for the other negotiator who then screens it, selects its key elements and tries to interpret them [28]. Negotiators should view their trading partners' behavior to select suitable tactics and strategies [28]. By considering this concept we model the concession behavior of negotiator's trading partners to determine the pattern of concession in grid resource allocation problem. Behavior is meaningful when a pair of grid's resource allocators of the opposite type met each other previously in numbers of GRNMs, so first of all we analyze workload traces from [61] to investigate this. By analyzing work load traces from [61], which are stored in Standard Work load Format (SWF), one can observe that GROs and GRCs repeat their supplies and demands, respectively, to the grid environment and in most instances, based on their supplies and demands, GROs (respectively, GRCs) can find a number of their previous trading partners as the new trading partners in the current GRNM. To prove this claim, it is assumed that (based on the existing SWF archives [61]) $grid.name$ represents the name of observed grid and also the maximum number of potential, unique users of a grid in $grid.name$ which is called $max\_pot\_user_{grid.name}$ corresponds to the total number of requested jobs found in $grid.name$'s SWF archive. Further, the set of observed unique users in that $grid.name$'s SWF archive are called $unique\_user\_set_{grid.name}$ and the number of $unique\_user\_set_{grid.name}$'s members is called $unique\_user\_set\_mem_{grid.name}$. The percentage of $grid.name$'s users that are observed previously in $unique\_user\_set_{grid.name}$ is denoted by $repeated\_user_{grid.name}$ and defined as (12). Hence, the variety of $grid.name$'s users increased as $repeated\_user_{grid.name}$ tends to become *zero percent*. We have

$$repeated\_user_{grid.name} = \left(1 - \frac{unique\_user\_set\_mem_{grid.name}}{max\_pot\_user_{grid.name}}\right) \times 100. \tag{12}$$

The results of SWF archives' observations [61] from $repeated\_user_{grid.name}$ perspective are illustrated in Fig. 3.

To model the concession behavior of kth trading partner of negotiator agent A (i.e., B_k) a new factor in name $PreBehave\_Depend_t^{B_k}$ which is defined based on two following parameters is introduced: (1) the number of successful negotiations between A and B_k in all the GRNMs they both participated (e.g., $\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}}$) and (2) the ratio of average negotiation time between A and B_k in $\#GRNM_{B_k-A}$ (e.g., $Ave.neg.time_A^{B_k}$) to $\sum_{k=1}^{no.trading\_partner_t^A} Ave.neg.time_A^{B_k}$. This means that the B_k, whose ratio of $\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}}$ is the lowest and its $Ave.neg.time_A^{B_k}$ is too far from zero

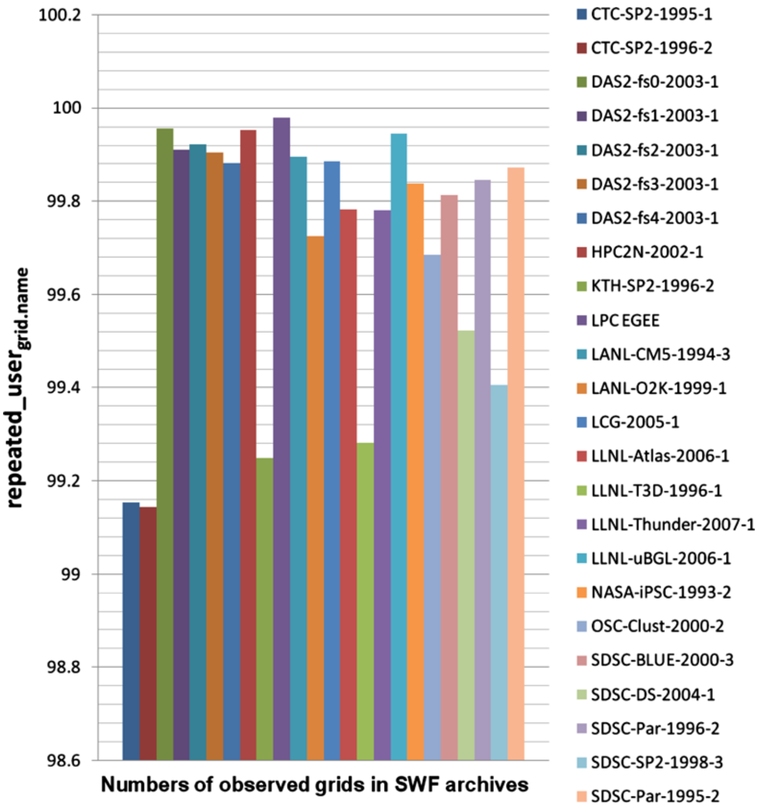


Fig. 3 $repeated\_user_{grid.name}$ in observed grids (based on work load traces from [61])

(makes a longer negotiation) is a *misbehaved* trading partner and deserves to receive more penalty.

$$PreBehave\_Depend_t^{B_k} = \frac{1}{\eta}[1 - \mu\ \rho] \tag{13}$$

- $IF\ (\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}} = 1)\ AND\ (Ave.neg.time_A^{B_k} <> 0)\ THEN\ (\mu = 1\ AND\ \rho = 1 - \frac{Ave.neg.time_A^{B_k}}{\sum_{k=1}^{no.trading\_partner_t^A} Ave.neg.time_A^{B_k}})$
- $IF\ (\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}} <> 1)\ AND\ (Ave.neg.time_A^{B_k} = 0)\ THEN\ (\mu = \frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}}\ AND\ \rho = 1)$
- $IF\ (\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}} <> 1)\ AND\ (Ave.neg.time_A^{B_k} <> 0)\ THEN\ (\mu = \frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}}\ AND\ \rho = 1 - \frac{Ave.neg.time_A^{B_k}}{\sum_{k=1}^{no.trading\_partner_t^A} Ave.neg.time_A^{B_k}})$
- $IF\ (\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}} = 1)\ AND\ (Ave.neg.time_A^{B_k} = 0)\ THEN\ (\mu = 1\ AND\ \rho = 1)$

Table 1 The data fields of an agent A's local *DB_behave* database's record and their brief description

Field name	Description
B_k_id	The identifier of B_k
$\#GRNM_{B_k-A}$	Numbers of GRNMs that both B_k and A participate in
$\#Suc.neg_{B_k-A}$	Numbers of successful negotiations between A and B_k in all GRNMs that both of them participate in
$Ave.neg.time_A^{B_k}$	Average of negotiation time between A and B_k in all GRNMs that both of them participate in

Experiment was made with $\eta = 4$ (by experiment, it is believed to be an appropriate value for tuning the amount of concession). The best value of $PreBehave\_Depend_t^{B_k}$ factor (i.e., zero) is achieved in case of $\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}} = 1$ and $Ave.neg.time_A^{B_k} = 0$. So, when the $\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}}$ is equal to one the effectiveness of the first parameter in $PreBehave\_Depend_t^{B_k}$ factor is ignored (i.e., $\mu = 1$) also when the $Ave.neg.time_A^{B_k}$ is equal to zero the effectiveness of the second parameter in $PreBehave\_Depend_t^{B_k}$ factor is ignored (i.e., $\rho = 1$). Similarly, when the $\frac{\#Suc.neg_{B_k-A}}{\#GRNM_{B_k-A}}$ is equal to one and the $Ave.neg.time_A^{B_k}$ is equal to zero the effectiveness of both parameters in $PreBehave\_Depend_t^{B_k}$ factor are ignored (i.e., $\mu = 1$ and $\rho = 1$).

A negotiator agent A has local database in name *DB_behave* (see Table 1) to store the parameters that make up the $PreBehave\_Depend_t^{B_k}$ factor.

3.3.3 Negotiation protocol

Type of Negotiation Protocol specifies the mechanism and the specific negotiation rules it uses for a particular negotiation. In designing both types of MBDNA (i.e., GRO_MBDNA and GRC_MBDNA), *Rubinstein's sequential alternating offer protocol* [52] is used. The negotiation procedure of *Rubinstein's sequential alternating offer protocol* is as follows: The players (negotiators) can take actions only at certain times in the (infinite) set $T = \{1; 2; 3; \ldots; t\}$. In each period $t \in T$, one of the players, say A, proposes an agreement, and the other player B either accepts it or rejects it. If the offer is accepted, then the negotiation ends, and the agreement is implemented. If the offer is rejected, then the process passes to period $t + 1$; in this period, player B proposes an agreement, which player A may accept or reject.

In setting the stage for specifying negotiation protocol and negotiation strategy, the following assumptions and rules apply:

1. Time is discrete and is indexed by $\{0, 1, 2, \ldots\}$—it is a logical and believable assumption, which is made in other models also [40, p. 713] and [62, p. 152].
2. Grid resource negotiation progresses in a series of rounds.
3. Multiple pairs of negotiators can negotiate deals simultaneously.
4. Negotiators do not form coalitions; the assumption is logical, because the type of game is non-cooperative (negotiators make decisions independently) with an arbitrary, finite number of negotiators.

5. Negotiation focuses on a single-issue (e.g., price-only).
6. Typically, a negotiator proposes its most preferred deal initially [63].
7. Whenever it is the A's turn to move (e.g. determine the amount of concession), it proposes a deal from its possible negotiation set (e.g., $[IP_A, RP_A]$, where IP_A and RP_A are, respectively, the initial and reserve prices of A).
8. If no agreement is reached, grid resource negotiation proceeds to the next round. At every round, the negotiator offers appropriate concession using the proposed multifactors function (see Sect. 3.3.2).
9. Negotiation between two negotiators terminates (i) when an agreement is reached, or (ii) with a conflict when one of the negotiators' deadline is reached [63].
10. When the negotiation ends, the history of negotiation is stored—this may be a good augmentation of database for future work (see Sect. 5).
11. Negotiation begins with negotiators having private information (e.g. deadline, reserve price, time preferences, strategies and payoffs according to them). So, no negotiator knows the private information of the opponent.
12. For strategic reasons, negotiators have information of only the index of the time period, and the then existing number of competitors and trading partners in grid resource negotiation market (GRNM) [40].
13. Without loss of generality, A of type *GRC_MBDNA* makes the concession first [40].
14. If the initial price of A of type *GRC_MBDNA* is not equal to or greater than the reserve price of B_k of type *GRO_MBDNA*, the negotiation process terminates with conflict.
15. Negotiation process in GRNM begins if only there are at least two negotiators of the opposite type (i.e., one negotiator of type *GRC_MBDNA* and the other of type *GRO_MBDNA*).

3.4 Terminating negotiation process and executing task (if negotiation is successful)

When the negotiation process between negotiator agent A of type *GRC_MBDNA* or *GRO_MBDNA* and its trading partner B_k of each pair reaches an agreement, the following steps are performed:

(a) *If* A is the negotiator agent who firstly accepts its trading partner's proposal, *Then* A stores the information of negotiation's transactions between itself and its opponents in *DB_game history* database.

(b) *If* a record of which its B_k*_id* field is corresponding to the identifier of B_k exists (among retrieved records), *Then* A effects the following changes in the retrieved record from *DB_behave* database:
   (1) Update the $Ave.neg.time_A^{B_k}$ field value using $\frac{\text{previous value} + \text{new negotiation time}}{2}$.
   (2) Increase the $\#Suc.neg_{B_k-A}$ field value by one.
   Otherwise:
   (1) Create a new record based on the template described in Table 1 and insert it into the *DB_behave* database.

(c) A sends negotiation results (e.g., the price for leasing the resource and the period of utilization) to corresponding *GRCA* or *GROA*.

Also successful *GRCA* and *GROA* commence executing the task of completing the resource allocation process. The *GRCA* entity submits the consumer's task(s) to *GROA*, which in turn submits the task(s) to *GRO*, which services the task(s). The sequence of messages involved in task execution is shown in Fig. 1. The *GROA*, on completing the execution of task(s), sends the result back to the *GRCA*(s). Finally, the results are announced to *GRC*.

When the negotiation process between negotiator agent A of type *GRC_MBDNA* or *GRO_MBDNA* and its trading partner B_k of each pair does not reach an agreement, the following step is performed:

(a) *If* A is the negotiator agent who firstly accepts its trading partner's proposal, *Then* A stores the information of negotiation's transactions between itself and its opponents in *DB_game history* database.

4 Simulation and experimental results

Simulation is used extensively for modeling and evaluation of real world systems. Consequently, modeling-and-simulation has emerged as an important discipline around which many standard and application-specific tools and technologies have been built. To evaluate the performance of MBDNAs against MDAs [39, 40], Kasbah agents [35] and NDFs [32], GridSim [64] is developed. The simulation environment consists of: (1) a virtual e-market; (2) a society of negotiation agents comprising MBDNAs, MDAs, Kasbah agents and NDFs; and (3) a controller agent.

(1) *Virtual e-market*
In a virtual e-market, negotiation agents have one of the following roles: grid resource consumer (GRC) or grid resource owner (GRO). In each negotiation round t, each negotiator of type GRC or GRO which its turn to move (make decision), decides to whether accept the counter-proposal or generate the next proposal according to its negotiation's model.
(2) *Society of negotiation agents*
Four kinds of negotiation agent, MBDNAs, MDAs, Kasbah and NDFs, are simulated. For each negotiation agent of type GRC_MBDNA or GRO_MBDNA a local database in name *DB_behave* is considered. At first entrance of each GRC_MBDNA or GRO_MBDNA to the e-market, the contents of data fields of its *DB_behave* are set to null.
(3) *Controller agent*
The controller agent generates negotiation agents, randomly determines their parameters (e.g., their roles as either GRC or GRO, initial prices (IP), reserve prices (RP), negotiation strategies (λ), deadlines, their competitors and trading partners), and simulate the entrance of agents to the GRNM following a uniform distribution.

4.1 Objectives

We consider that MDAs and EMDAs [37–40] are appropriate tools for comparing our proposed MBDNAs with them as: (1) MDAs and EMDAs take into consideration the issue of time constraint (by using time-dependent function which is similar to our proposed MBDNAs' time-dependent function) and market dynamics (which are also considered in designing our proposed MBDNAs) in constructing negotiation strategy, (2) a large number of commonly previous researches in the field of negotiation-based grid resource allocation reviewed, referenced or enhanced the idea of MDAs [39, 40] and/or EMDAs [37, 38] besides compared their achieved results with them (e.g., see [21, 31, 45] and [65–70]) and (3) according to [65] and [66] they can be modified to support negotiation activities in cloud computing environment. As augmenting MBDNA with additional capability of relaxing bargaining terms in face of intense market pressure to achieve both suitable utilities and suitable success rate under different market conditions (e.g., given different supplies and demands) for both GROs and GRCs is planned to be our near future research work, we do not compare the current research to EMDAs [37, 38] (which relax their bargaining terms in the face of intense market pressure), and instead leave it for future research. The focus is, therefore, on MDAs [39, 40] and on those state-of-the-art negotiation models for grid resource management, whose performance was compared by [39] with that of the MDAs (e.g., Kasbah [35] and NDF [32]).

To better understand the similarity and difference between MBDNAs, MDAs, Kasbah agents, and NDFs, the negotiation strategies of MDAs, Kasbah agents, and NDFs are discussed.

Market-Driven Agents (MDAs): An MDA determines the amounts of concessions using three negotiation decision functions: *Time* (T_t^A), Opportunity (O_t^A) and *Competition* (CC_t^A).

Time function: MDAs model devaluation of resources with passing time by using the same function as (9).

Opportunity function: Opportunity function of MDAs is as same as the opportunity function of MBDNAs (see (10)).

Competition function: the probability of negotiator agent A being considered the most preferred trading partner by at least one of $B_k \in B$ is calculated thus (we should highlight that Sim [39, 40] did not consider *GRNM area* concept):

$$\begin{aligned} &CC_t^A\left(no.competitor_t^A, no.trading\_partner_t^A\right) \\ &\quad = 1 - \left[\left(no.competitor_t^A\right)/no.competitor_t^A + 1\right]^{no.trading\_partner_t^A}. \end{aligned} \tag{14}$$

More details can be found in [39, 40].

Negotiation Decision Function (NDF): NDF uses time-dependent function to model devaluation of resource with respect to passing time. The time-dependent negotiation function of an NDF agent is given thus:

$$f^A(t) = K^A + \left(1 - K^A\right)\left(\min\left(t, t^A_{\text{deadline}}\right)/t^A_{\text{deadline}}\right)^{1/\psi} \tag{15}$$

where t is the current discrete trading time, t^A_{deadline} is negotiation agent A's deadline, ψ is A's time preference and k^A is a constant that determines the price to be offered in the first proposal of A. The strategies of NDF agents can be classified into three classes as follows: *Boulware* ($\psi < 1$), *Conceder* ($\psi > 1$) and *Linear* ($\psi = 1$). These strategies correspond, respectively, to the *Conservative, Conciliatory* and *Linear* strategies of MDA agents. The offer $p(t)$ of a GRO (respectively, GRC) NDF agent at t is

$$p(t) = \begin{cases} IP_B + (RP_B - IP_B)(\frac{t}{t^A_{\text{deadline}}})^{\frac{1}{\Psi}} & \text{for } GRC \\ IP_S - (IP_S - RP_S)(\frac{t}{t^A_{\text{deadline}}})^{\frac{1}{\Psi}} & \text{for } GRO \end{cases} \tag{16}$$

where IP_B and RP_B are the initial and reserve prices of agent B. Similarly, IP_S and RP_S are the initial and reserve prices of agent S.

Kasbah: Chavez and Maes describe the architecture of Kasbah electronic marketplace (implemented at the MIT Media Laboratory) where is the most significant example of e-commerce negotiation system. This market place simulates an environment where a user can create an autonomous agent to buy or sell a product, negotiating product price on his/her behalf. The agent configuration includes some behavior rules, such as the maximum time to reach a deal, the desired price interval and the price suggestion function. Kasbah provides a small number of facilities to the user's negotiation process. It was noted in [35] that "*Users wishing to buy and users wishing to sell certain products in that marketplace can initialize agents by specifying what they want to buy or sell, the desired price, the highest acceptable price (for buying agents) or the lowest acceptable price (for selling agents), the date they want the transaction to be completed and a negotiation strategy.*" A seller (respectively, buyer) Kasbah agent can specify the "Decay" function which is used by the agent to lower (respectively, raiser) the asking price after expiration of the fixed time. A seller (respectively, buyer) agent adopts "*Anxious*", "*Cool-headed*" and "*Greedy*" strategies which follow a *Linear* curve, *an Inverse-Quadratic* (respectively, *Quadratic*) curve and an *Inverse-Cubic* (respectively, *Cubic*) curve, respectively. Kasbah agent's "*Anxious*" strategy corresponds to the MDAs' *Linear* strategy (respectively, NDF's *Linear* strategy) and "*Cool-headed*" and "*Greedy*" strategies correspond to the MDAs' *Conservative* strategy with some values of λ (respectively, NDFs' *Boulware* strategy with some values of Ψ).

By comparing MBDNAs against MDAs one can understand that MDAs do not employ any mechanism for classifying the negotiator's opponents from their behavior point of view and make penalties for *misbehaved* opponents to put them under pressure to refine their behavior and make reward for *well-behaved* opponents to encourage them in continuing their good behavior. In addition MDAs do not make *GRNM areas* which can be helpful in determining the specific amount of concession to each negotiator's trading partner based on the situations of the *GRNM area*. Also, by comparing NDFs (respectively, Kasbah agents) and MBDNAs, one can understand that NDFs (respectively, Kasbah agents) do not consider market dynamicity, market competition and any mechanism for classifying the negotiator's opponents from their behavior point of view to put *misbehaved* opponents under pressure to refine their behavior. In addition NDFs (respectively, Kasbah agents) do not make *GRNM areas* which can be helpful in determining the specific amount of concession to each negotiator's trading partner based on the situations of the *GRNM area*. Also, while NDFs and Kasbah agents adopt bilateral negotiation protocol MBDNAs and MDAs adopt alternating offers protocol to provide more flexibility in allowing multiple messages from both GROs and GRCs to be exchanged.

The similarity between MBDNAs, MDAs, NDFs and Kasbah agents is that they all have quite similar time-dependent negotiation strategies. Intuitively, for every time-dependent negotiation strategy in MDA, NDF and Kasbah, there is a corresponding

Table 2 Summary and comparison

References	MBDNAs	MDAs	NDF	Kasbah
Negotiation protocol				
Bilateral negotiation	Yes	Yes	Yes	Yes
Multilateral negotiation	Yes	Yes	No	No
Determine the specific amount of concession to each negotiator's trading partner instead of the same amount to all	Yes	No	No	No
Negotiation strategies				
Behavior of the negotiator's trading partner_dependent	Yes	No	No	No
Change in the number of negotiator's competitors_dependent	Yes	Yes	No	No
Change in the number of negotiator's trading partners_dependent	Yes	Yes	No	No
Remaining time to deadline_ dependent	Yes	Yes	Yes	Yes
Fixed rate adjustment_oriented	No	No	No	Yes

strategy in MBDNA, so MDAs, Kasbah agents and NDFs are good choices for comparing MBDNAs against them.

In order to complete the comparison procedure of MBDNA against other mentioned negotiator agents another issue in name time consumption of the negotiation process should be considered. With respect to this issue we should highlight that in comparison to MDA, Kasbah and NDF, MBDNA has the following extra step in each decision making round: extract previous concession behavior of its opponents from the local *DB_behave* database. According to the study of real workload traces from [61], which are stored in Standard Work load Format (SWF), we investigate that the maximum number of negotiator opponents is limited to 300. Hence, with respect to time, using any kind of search algorithm is logical. In spite of this, we consider a scenario in which a negotiator agent faces with large number of opponents. This scenario can be occurred when the MBDNA is use to negotiate in the forests of interconnected grids in different parts of the world (i.e., InterGrids). In the mentioned scenario we propose to perform hash searches on the *DB_behave* database. Well-designed hash searches are more efficient that any other kinds of search and the behavior record of an opponent can be found in constant time (i.e., $O(1)$ time) [75]. Also, remember that parallel negotiation activities are performed in A's *GRNM areas* that cause no extra time complexity.

For the benefit of readers, Table 2 summarizes and compares the main features of the proposed negotiation model against the MDAs, NDFs and Kasbah agents in terms of their negotiation protocol and negotiation strategies.

Although both GRC and GRO agents are simulated, but without loss of generality it is sufficient to demonstrate the properties of MBDNAs from the perspective of GRC agents. So we conduct four types of experiment: (1) GRC agents are MDAs and GRO agents are MBDNAs, (2) GRC agents are Kasbah and GRO agents are MBDNAs, (3) GRC agents are NDFs and GRO agents are MBDNAs and (4) both GRC agents and GRO agents are MBDNAs.

The reason that in the first three experiments just GRO agents are considered as MBDNAs while GRC agents are considered as one of MDA, Kasbah or NDF is based on a common assumption in microeconomics, namely *ceteris paribus* [71]. As mentioned in [72]: "the effect of a particular factor can be analyzed by holding all other factors constant." The purpose of the first three experiments is to compare the performance of negotiation agents that are not designed with the proposed multicriteria negotiation strategy (i.e., MDA, Kasbah and NDF) against negotiation agents that are designed with the proposed multicriteria negotiation strategy from GRC perspective, so it seems prudent to avoid any possible influence on the negotiation outcomes when negotiator agents of type GRO make concession amount. Hence in our experiment GRO agents are programmed as MBDNA because MBDNAs are designed with the proposed multicriteria negotiation strategy that is believed to be the best one. Also, in the fourth experiment we programmed both GRC and GRO agents as MBDNA to investigate the performance of MBDNAs of type GRC against MBDNAs of type GRO when both of them are programmed with the same proposed multicriteria negotiation strategy.

4.2 Experimental setting

All the following input parameters are required for setting grid simulation testbed: (a) *the grid load* (*which is represented by Grid_load symbol*), (b) *the e_market type*, (c) *job size* (*measured in* (*MI*)), (d) *deadline for agents to complete their negotiation process*, (e) *the total resource capacity of a GROA* (*measured in* (*MIPS*)), (f) *Market density*, (g) *negotiation strategy* and (h) *time-dependent factor*. The values of the most mentioned parameters that are used to conduct simulation are derived from [37–40] and [45]. The input parameters and their possible values are presented in Table 3.

(a) *Grid load*

Grid load refers to the utilization status of computing resources. As the load varies continuously with time, the simulation should be carried out by considering various grid loads. Sim [39] proposes two parameters R_p and C_c to represent grid load, where R_p is defined as the expected amount of processing requested per time interval (which is measured in MI) and C_c as the total computing capacity of the grid (which is measured in MI). It was noted in [39] that "R_p *depends on both the requests* (*tasks*) *from the GRCs which depend on* P_m (*i.e., the probability of a GRC generating a task that needs computing resources at each negotiation round. This parameter is used to simulate the arrival of a task to the grid at each negotiation round*) *and the average size of each task. It is assumed that the arrival rate of tasks follows a Poisson distribution, and the average task size approximates the range between 50–400 MIs. Different levels of system utilization* (*different grid loads*) *are simulated by varying the time interval between the possible arrivals of two tasks*". As grid load tends to become one (respectively, to zero), fewer (respectively, more) computing resources in the grid are available for lease.

$$Grid\_load = \frac{R_p}{C_c} \quad \text{where } 0 < Grid\_load \le 1. \tag{17}$$

(b) *E_market types*

As the availability of grid resources varies continuously with time, the simulation should be carried out by considering different GRC-to-GRO ratios. These ratios characterize three types of e-market: *GRC_favorable*, *GRO_favorable* and *Balanced*. The *GRC-favorable* e-market addresses more GRO agents and consequently

Table 3 Input parameters for setting grid simulation testbed and their possible values

Input	Possible values			
E-market type	GRC_favorable	$P_{GRC} < 0.5$	$GRC\_to\_GRO = \{1:2, 1:5, 1:10, 1:30, 1:50, 1:100\}$	
	GRO_favorable	$P_{GRC} > 0.5$	$GRC\_to\_GRO = \{2:1, 5:1, 10:1, 30:1, 50:1, 100:1\}$	
	Balanced	$P_{GRC} = 0.5$	$GRC\_to\_GRO = \{1:1, 5:5\}$	
P_{GRC}: *Probability an agent being a GRC*				
Market density	Sparse	Moderate	Dense	
P_{gen}	0.25	0. 5	1	
P_{gen}: *Probability of generating an agent per round*				
	$0 < Grid\_load \leq 1$ $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$			
Grid_load	Low: $0 \leftarrow Grid\_load$	High: $Grid\_load \rightarrow 1$		
	Short	Moderate	Long	
Deadline (*No. of rounds*)	100	1600	3100	
Job size (*MI*)	50–400			
Resource capacity (*MIPS*)	200–3000			
Negotiation strategy	MBDNAs' negotiation strategy is described in Sect. 3	MDAs' negotiation strategy is inspired by [39, 40]	NDFS' negotiation strategy is inspired by [32]	Kasbah agents' negotiation strategy is inspired by [35]
(Amount of time preference, type of strategy: abbreviation)				
MBDNA time preferences	MDA time preferences	NDF time preferences	Kasbah time preferences	
($\lambda = 1/3$, *Conciliatory: CC*)	($\lambda = 1/3$, *Conciliatory: CC*)	($\Psi = 3$, *Conceder: C*)	No corresponding strategy	
($\lambda = 1$, *Linear: L*)	($\lambda = 1$, *Linear: L*)	($\Psi = 1$, *Linear: L*)	($\lambda = 1$, *Anxious: A*)	
($\lambda = 2$, *Conservative: CS*)	($\lambda = 2$, *Conservative: CS*)	($\Psi = 1/2$, *Boulware: B*)	($\lambda = 2$, *Cool-headed: CH*)	
($\lambda = 3$, *Conservative: CS*)	($\lambda = 3$, *Conservative: CS*)	($\Psi = 1/3$, *Boulware: B*)	($\lambda = 3$, *Greedy: G*)	
($\lambda = 20$, *Conservative: CS*)	($\lambda = 20$, *Conservative: CS*)	($\Psi = 1/20$, *Boulware: B*)	No corresponding strategy	

more opportunity for acquiring resources; the *GRO-favorable* market addresses more GRC agents and consequently more opportunity for leasing out resources; the *Balanced* market addresses normal competition among GRO agents and GRC agents. GRC_to_GRO ratio is controlled by the probability P_{GRC} of an agent being GRC agent (or GRO agent). P_{GRC} follows a uniform distribution. We conduct two types of experiment to investigate the benefit of MBDNAs from e-market type's perspective: (1) a moderate difference between the number of GRCs and GROs and (2) a large difference between the number of GRCs and GROs. The goal of having these two kinds of experiment is to show the better impact of the proposed factors by increasing the difference between the number of GRCs and GROs.

(c) *Job size*

The GRC agent's job size is measured in millions of instructions (MI).

(d) *GRC agent's deadline*

As described before, agent's deadline constraint plays a major role in choosing the appropriate strategy. According to [39], three categories can be described for the agent's deadline constraint: *Short*, *Moderate* and *Long*. Space limitation precludes all possible values of GRC agent's deadline from being included in depicting figures, and Table 3 only contains GRCA's job deadline values equal to 100, 1600 and 3100 which represent *Short*, *Moderate* and *Long* deadline, respectively.

(e) *GRO agent's resource capacity*

The GRO agent's total resource capacity is measured in millions of instructions per second (MIPS).

(f) *Market density*

Market density depends on the number of GRC agents and GRO agents participating in the GRNM. Market density is controlled by the probability P_{gen} that an agent will enter the GRNM in each round of negotiation. P_{gen} Follows a uniform distribution. Market density can be categorized into three categories: *Dense*, *Moderate* and *Sparce*. Space limitations preclude all the categories of market density from being included in depicting figures, hence, only results comparing the performance of MBDNAs with MDAs, NDFs and Kasbah agents in *Dense* markets are presented.

(g) *Negotiation strategy*

The proposed multicriteria negotiation strategy of MBDNAs is described in Sect. 3. Also the negotiation strategies of MDAs, Kasbah agents and NDFs are inspired by [35, 39, 40] and [32], respectively.

(h) *Time-dependent factor*

As mentioned before the rationale for comparing MBDNAs with MDAs, Kasbah agents and NDFs is that all these agents take into consideration the issue of time constraint, and their time-dependent strategies have quite similar to each other. Space limitation precludes all possible values of negotiator's time preference from being included in depicting figures, and Table 3 only contains some values for negotiator's time preference.

4.3 Performance measure

Because Grids are dynamic in their nature, it is difficult to benchmark and evaluate them. Moreover, there is no general consensus on which metrics to use [73, 74]. The common metrics to be studied are as follows [39]:

- *Task completion*: Task completion is defined as the percentage (P_{tc}) of a GRC's set of tasks that is accomplished by successfully negotiating and leasing grid resources; let N_{tot} denote the total number of tasks requested by a GRC and N_{suc} the number of tasks that are successfully scheduled and executed. P_{tc} is given as

$$P_{\text{tc}} = \frac{N_{\text{suc}}}{N_{\text{tot}}} \tag{18}$$

- *Budget spent*: Budget spent defines how efficiently the available budget was spent (Bud_{eff}). Let Bud_{init} denote the initial budget allocated to a GRC and Bud_{spent} the amount of budget spent in leasing computing resource(s) for processing tasks that are successfully scheduled and executed. Bud_{eff} is given as

$$Bud_{\text{eff}} = \frac{N_{\text{suc}}/Bud_{\text{spent}}}{N_{\text{tot}}/Bud_{\text{init}}} \tag{19}$$

As described in [39], "$\frac{\mathrm{N_{suc}}}{\mathrm{Bud_{spent}}}$ represents the actual number of tasks processed per currency unit measured in "Grid dollars" or "G\$" [17], and $\frac{\mathrm{N_{tot}}}{\mathrm{Bud_{init}}}$ represents the expected number of tasks processed per currency unit before they are successfully scheduled (and executed)."

4.4 Evaluation and discussion

Following are the two most important observations from the results:

Observation 1 Figures 4 and 5 are illustrated according to the two kinds of experiment (i.e., a moderate difference between the number of GRCs and GROs and a large difference between the number of GRCs and GROs). Figure 4 shows the first type of experiment while Fig. 5 shows the second type of experiment. According to a common concept in the social sciences: "having more individuals (big human society) leads to better analyzing of human kind's behavior, better decision making about goodness of their behavior in the chosen society and determining more appropriate reactions against different human behavior types" one can understand that by increasing in the difference between the number of GRCs and GROs from Fig. 4 to Fig. 5 (i.e., having bigger society of opponents) the better performance is achieved. In other words, by having more opponents the decision maker negotiator can have more appropriate and better evaluation of the goodness of its opponent's behavior among the other members of the society and tunes the concession amount based on the evaluation result. For example if a negotiator has one opponent it thinks that its opponent's behavior is the best (which it may be not) while having more opponents can tune and correct the negotiator's view about its opponents' behavior by comparing one against the others. Following the common observations in Figs. 4 and 5 are discussed. The experimental results in Figs. 4 and 5 show the following.

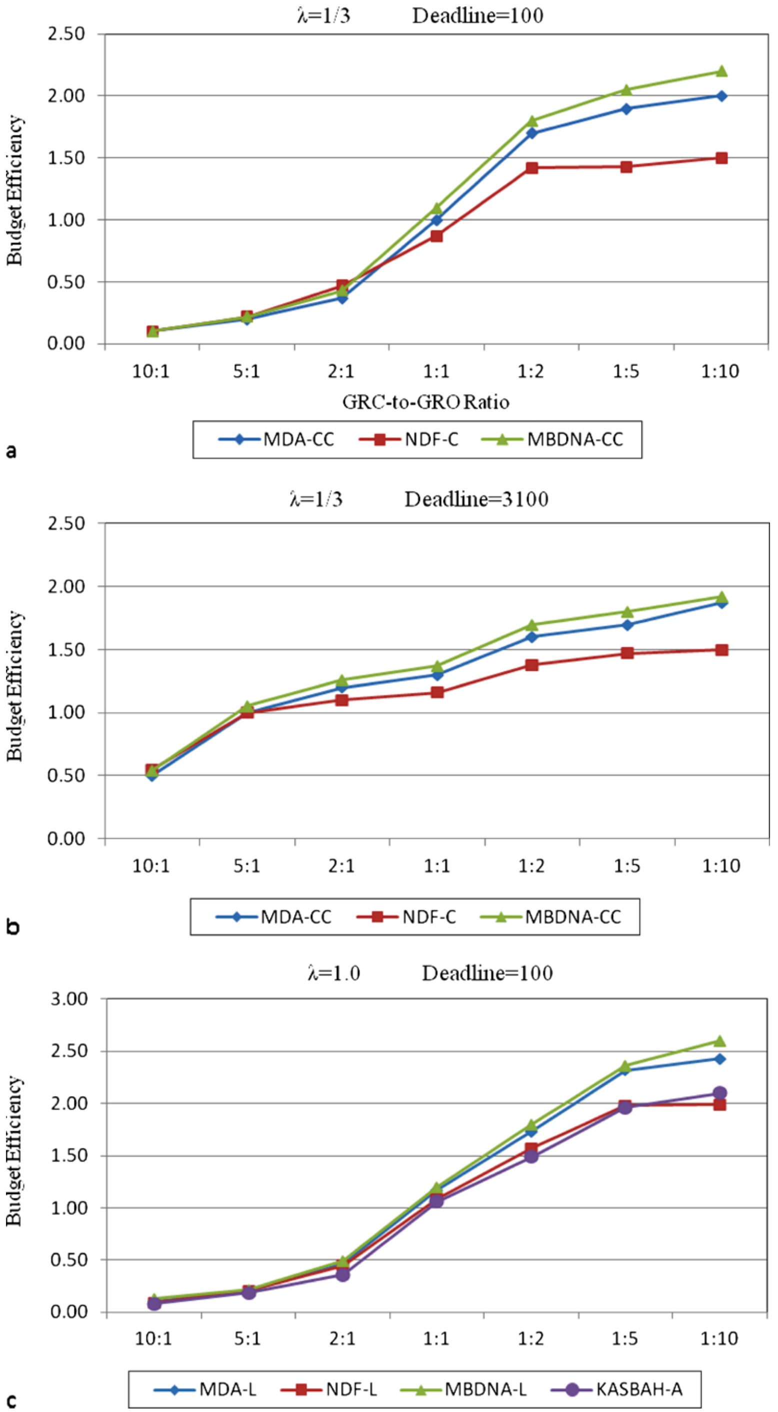


Fig. 4 Performance under different market types considering a moderate difference between the number of GRCs and GROs

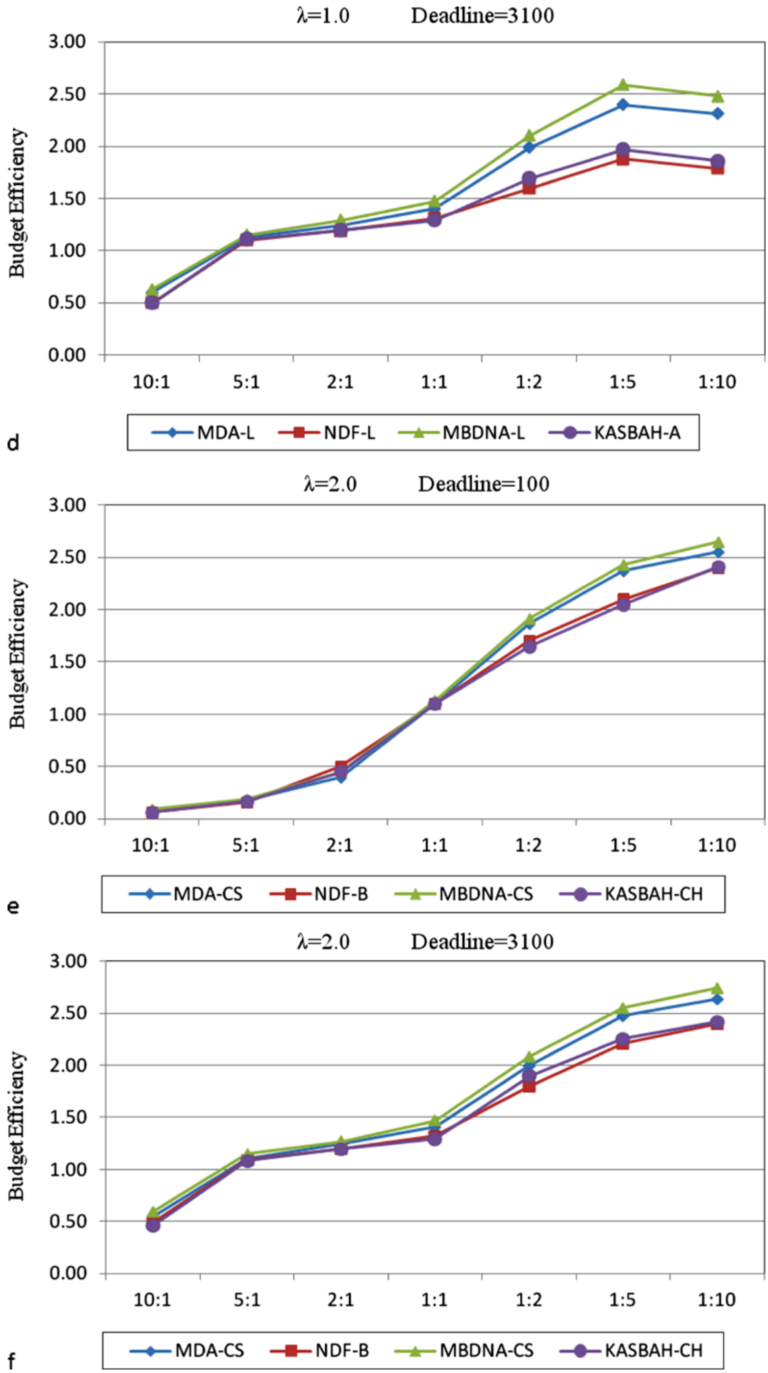


Fig. 4 (*Continued*)

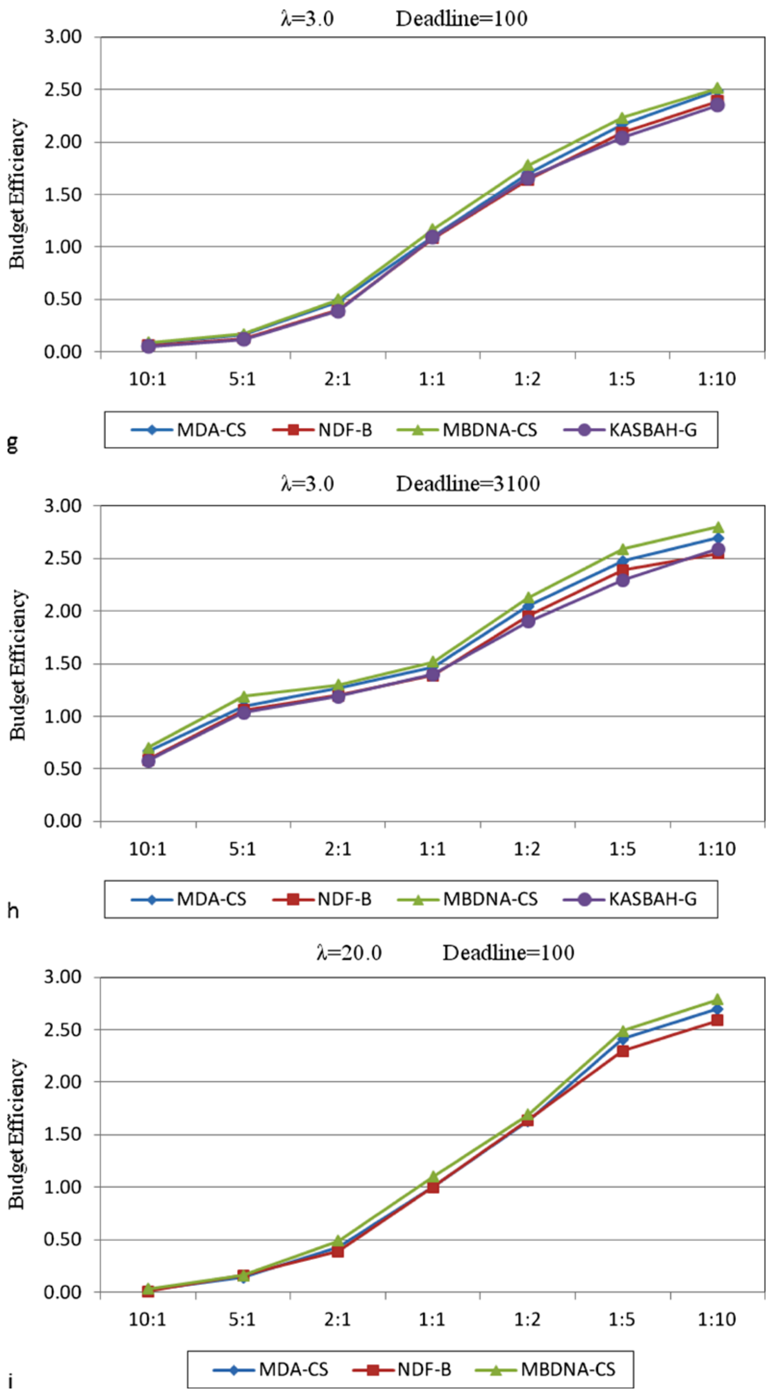


Fig. 4 (*Continued*)

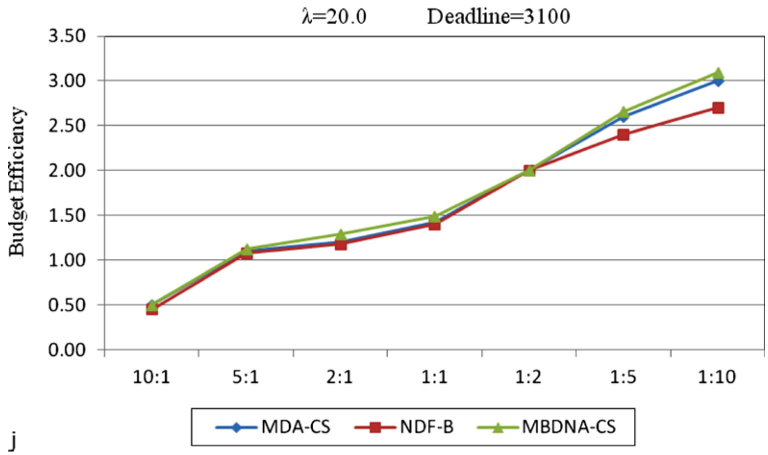


Fig. 4 (*Continued*)

(1) Given the same GRC-to-GRO ratio MBDNAs achieve higher budget efficiencies (except for *GRO-favorable* markets where the bargaining power of GRC negotiators is weak due to low probability of creating GRO agents) by using new negotiation strategy. From negotiation strategy point of view, MBDNAs' negotiation strategy is composed of an extra factor in name *Previous Concession Behavior of Negotiator's Trading Partner* in comparison to MDAs and three extra factors in names C*ompetition*, O*pportunity* and *Previous Concession Behavior of Negotiator's Trading Partner* in comparison to NDFs and Kasbah agents. *Previous Concession Behavior of Negotiator's Trading Partner* factor which is considered to make penalties for *misbehaved* trading partners not only increases the chance of reaching a consensus with *well-behaved* trading partners in different market types but also puts *misbehaved* trading partners under pressure to have better behavior in next meeting (to avoid achieving low success rate and/or losing utility). This idea is inspired from real-life trading market where the negotiators analyze their opponents' behavior and classified them into *misbehaved* and *well-behaved* opponents. Then, during the negotiation process, the negotiators consider penalties for *misbehaved* opponents to put them under pressure to refine their behavior and rewards for *well-behaved* opponents to encourage them in continuing their good behavior. Consequently, with respect to this factor the negotiators' achieved utility will be bettered by participating in more numbers of trading markets. Also negotiation strategy must takes into consideration the dynamics of a grid-computing environment because it is expected that resources and services are constantly being added/removed from a grid. This concept is modeled via two factors: *Competition* and *Opportunity*.

(2) Negotiation results become more unfavorable with the increase of the GRC-to-GRO ratio for all types of negotiator (i.e., MBDNAs, MDAs, NDFs and Kasbah agents). This is because with small number of trading alternatives (partners), a negotiator agent generally has a lower chance of reaching a consensus at its own term.

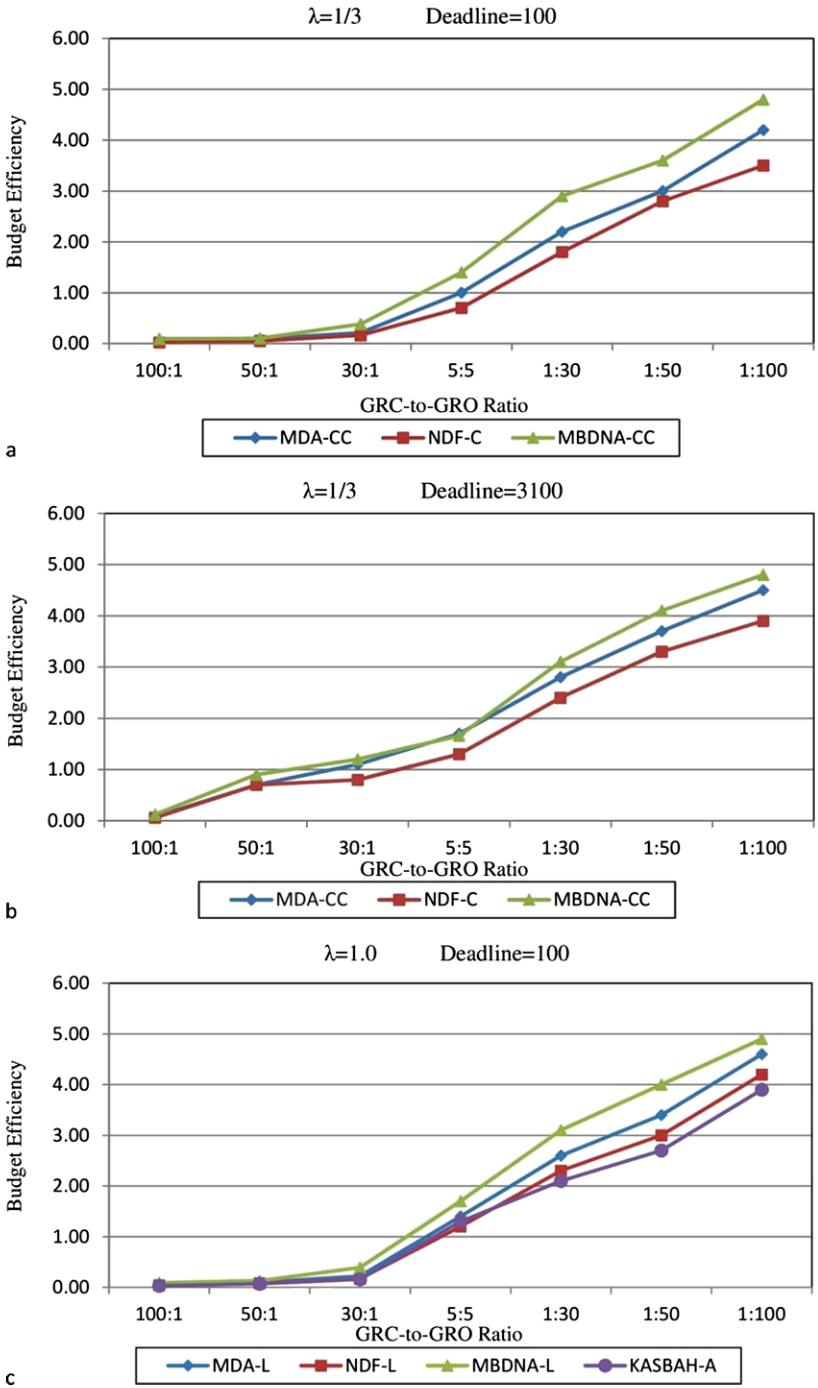


Fig. 5 Performance under different market types considering a large difference between the number of GRCs and GROs

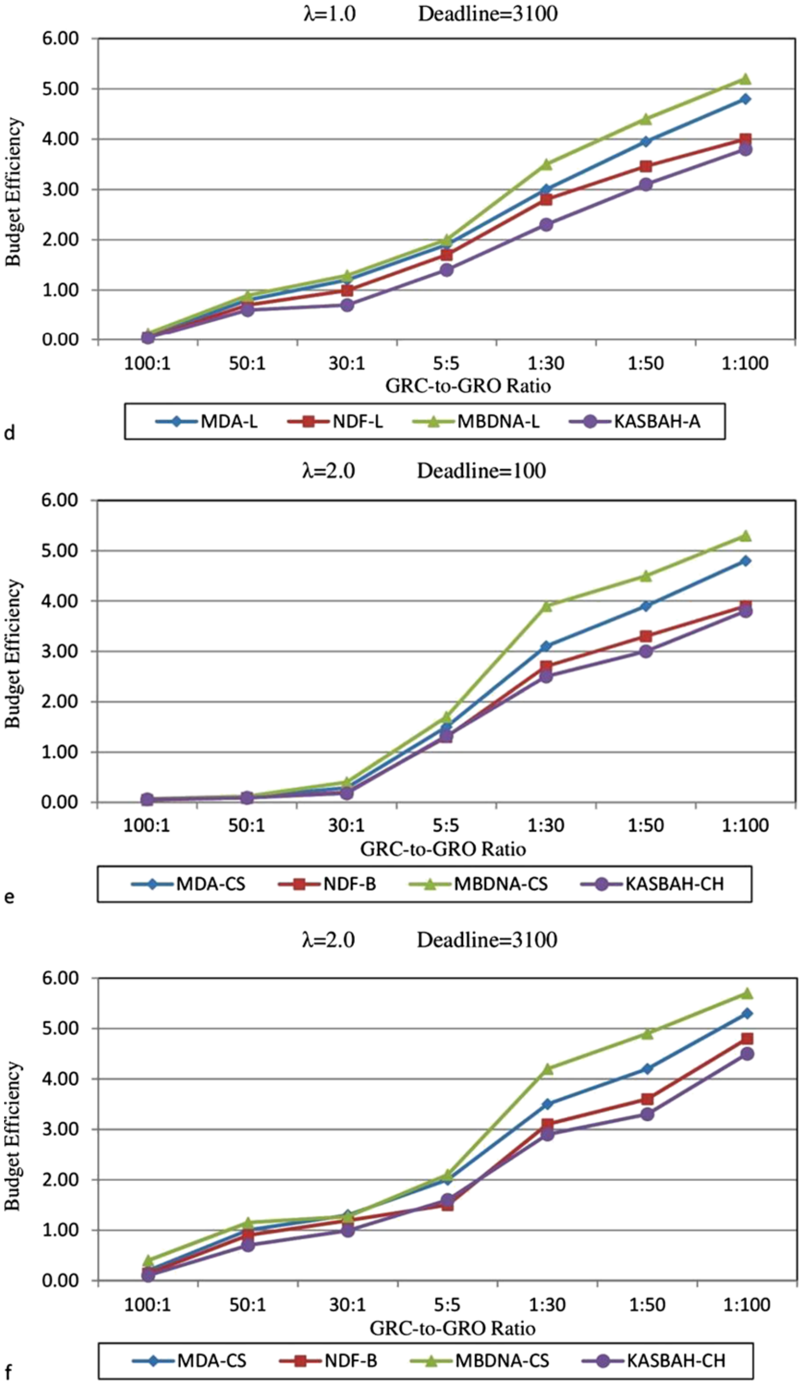


Fig. 5 (*Continued*)

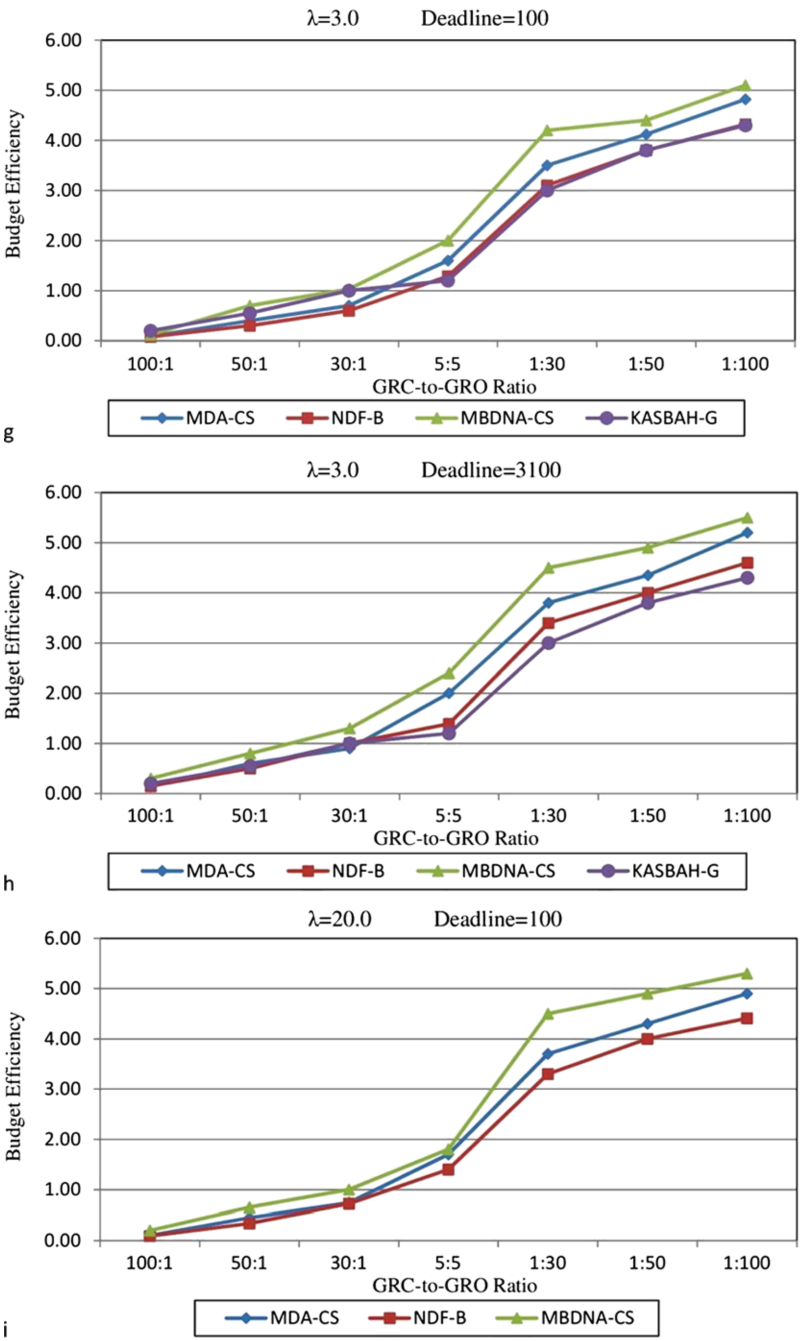


Fig. 5 (*Continued*)

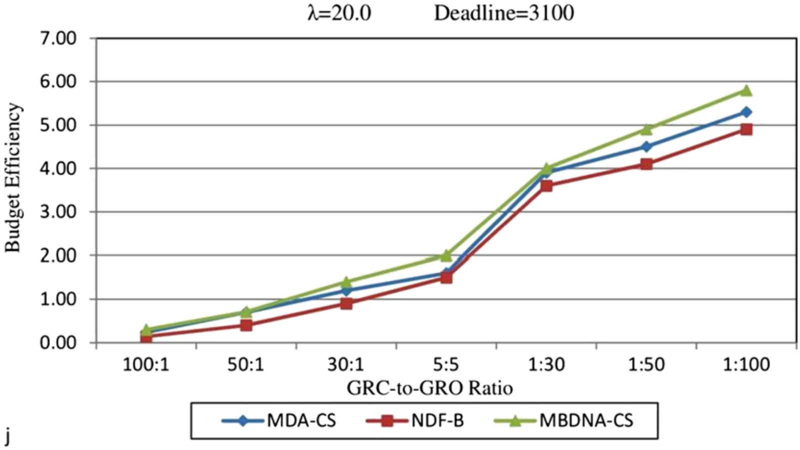


Fig. 5 (*Continued*)

(3) When the type of market tends to be *GRO-favorable* (e.g., where the GRC's competition degree is very high and probability that a GRO agent enters the market at any time is <0.5), the budget efficiency of the all types of agent (i.e., MBDNAs, MDAs, NDFs and Kasbah agents) are close especially in the *short* deadline case since under very extreme competition conditions (i.e., *GRO-favorable* market type where GRC-to-GRO ratio $= \{2:1, 5:1, 10:1, 30:1, 50:1, 100:1\}$), the bargaining power of GRCs decreases and it may be extremely difficult for all types of GRC-negotiator to reach any consensus so they have to concede more to avoid the risk of losing grid resources (which leads to lower average utility) and also with *short* deadline (in comparison to *moderate* and *long*) due to not having plenty of time to complete a deal the bargaining positions of all types of GRC agent are weaker and if final agreement is reached, all of them are likely to make relatively more concessions (which leads to lower average utility).

(4) Adopting more patient strategies can increase a negotiator's budget efficiency because a negotiator agent puts its opponent under pressure to concede more for narrowing the differences between the proposals and counter-proposals. This means that a more patient negotiator agent is less likely to reach earlier agreement and instead prefer to keep its budget efficiency at the highest acceptable level while its opponent should adopt impatient strategies in negotiation process to avoid the risk of losing deals.

Observation 2 The experimental results in Fig. 6 show the following.

(1) Given the same Grid-load, MBDNAs achieve slightly higher success rate (especially for *moderate* and *long* deadline) by using new negotiation strategy. From negotiation strategy point of view, since MBDNAs are more likely to adopt and relax their bargaining criteria in face of following pressures: (a) come close to their negotiation deadline, (b) decrease in number of negotiator's trading alternative and/or increase in differences in proposals and counter-proposals, (c) increase in number of negotiator's competitors and (d) decrease in number of *well-*

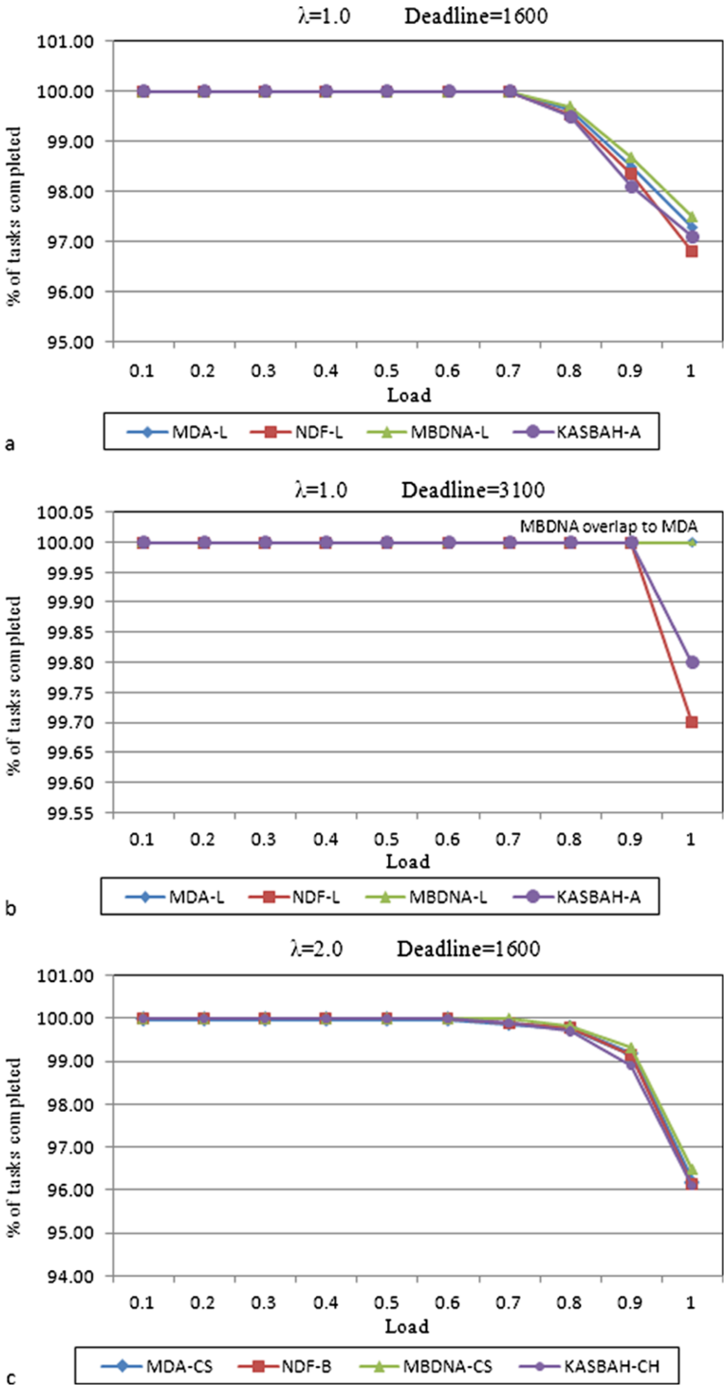


Fig. 6 Performance under different grid work loadings

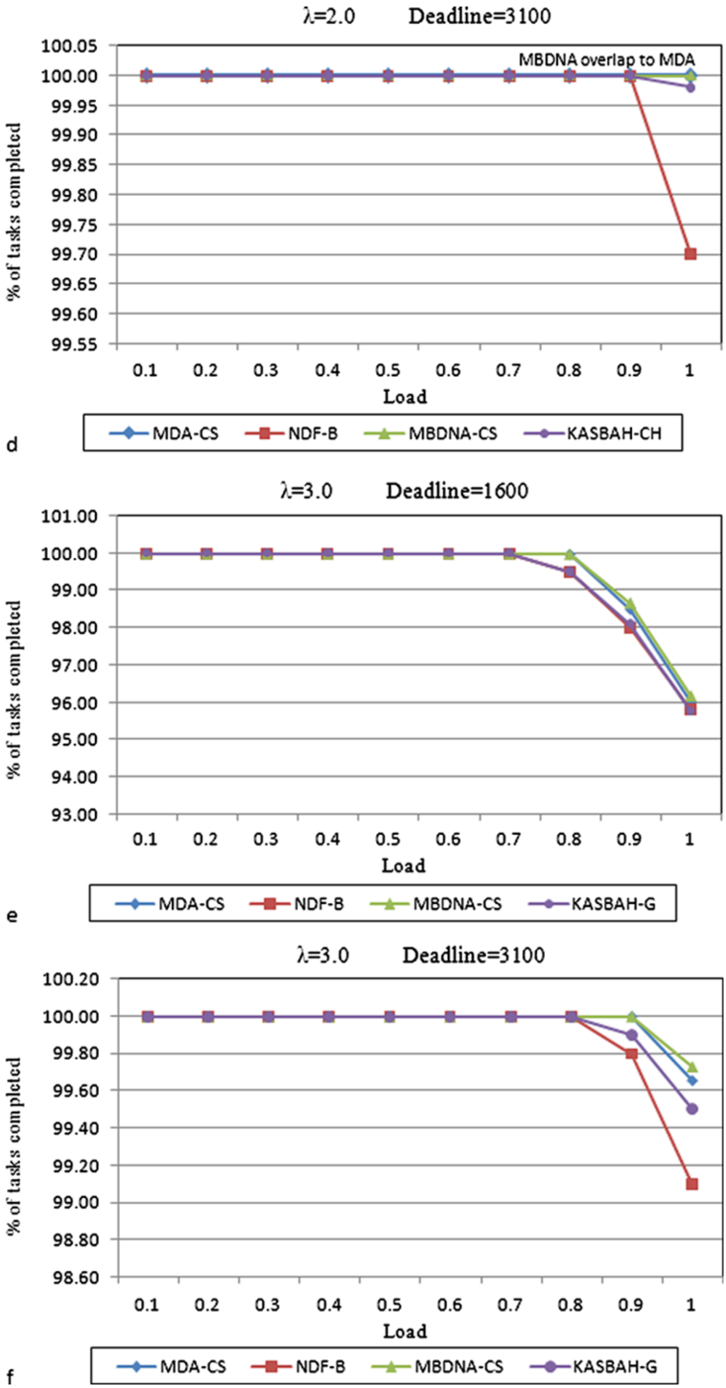
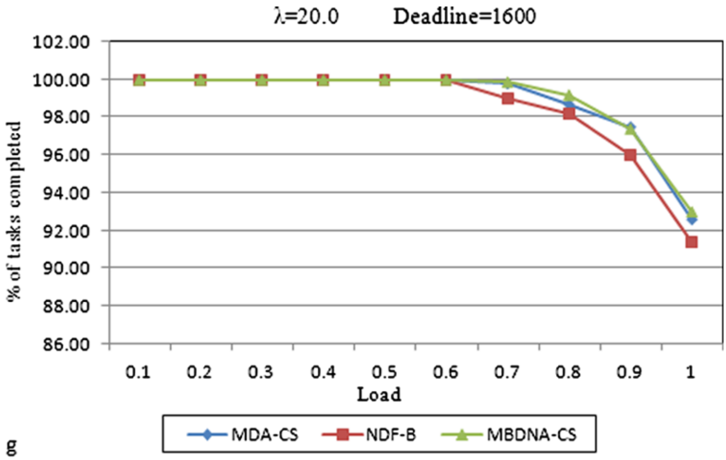
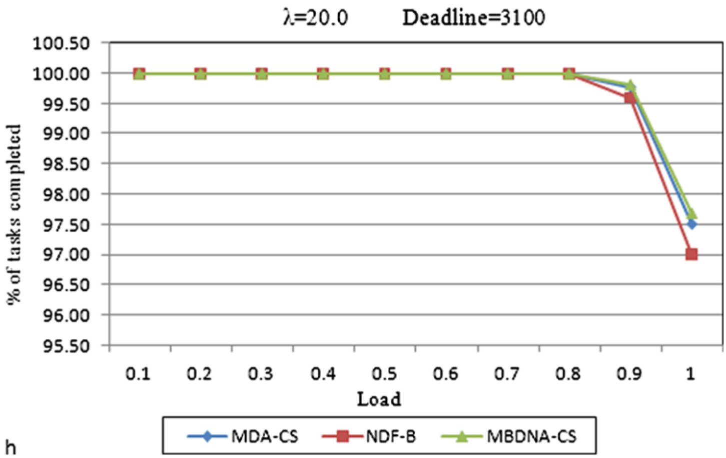


Fig. 6 (*Continued*)



g



h

Fig. 6 (*Continued*)

behaved opponents, they are more likely than MDAs, NDFs and Kasbah agents to reach agreement with their trading partners. While the punishment strategy (i.e., make small concession amount in front of misbehaved opponents and slightly narrow the difference between proposal and counter proposal) of MBDNA increases the budget efficiency but it may increase the risk of losing a deal (i.e., lower success rate) specially in stiff competition. So we should investigate if the MBDNA's strategies have any bad side effect on success rate. As illustrated in Fig. 6 one can understand that the new negotiation do not have any bad side effect on success rate and even in some cases improve it. Hence, according to the combination of two performance metrics (i.e., budget efficiency and success rate), MBDNAs outperform other negotiations agents.

(2) Negotiation results become more unfavorable with the increase of the Grid_load for all types of negotiator (i.e., MBDNAs, MDAs, NDFs and Kasbah agents). With the increase of Grid_load, there were fewer available resources in the grid,

and it became increasingly difficult for all types of agent to successfully negotiate for resources.

(3) Given the same Grid_load and time preference, GRCs of all types who have *long* deadline achieved higher success rate. With *long* deadline (in comparison to *moderate* and *short*) due to have plenty of time for trading the bargaining positions of all types of negotiator are stronger and they both likely to complete deals successfully (i.e., have higher success rate). However, as MBDNAs are designed with more appropriate negotiation strategy, they are more likely to achieve same or higher success rate than other types of negotiator especially under intense grid market pressure.

(4) Adopting more patient strategies can decrease a negotiator's success rate. This means that a more patient negotiator agent is more likely to face the risk of losing deals especially in the case of stiff competition.

5 Conclusion

This paper presents an approach to allocate resources in grid environment via negotiation between GRC_MBDNA (Grid Resource Consumer Market- and Behavior-driven Negotiation Agent) and GRO-MBDNA (Grid Resource Owner Market- and Behavior-driven Negotiation Agent) to enhance the success rate and utility of negotiation agents. The scenario of resource allocation proposed here in the economy-aware grid environment includes the following four major phases:

(1) *Registering GRCs and GROs*
(2) *Creating MBDNAs and providing the required information* (*that is, necessary for starting negotiation*)
(3) *Starting negotiation based on proposed strategy*
(4) *Terminating negotiation process and executing task* (*if negotiation is successful*)

In this approach, the authors investigated the benefit of the proposed *Behavior* factor in designing the negotiation agents (e.g., MBDNAs) so as to handle resource allocation in a computational grid environment, as also in a simulated environment. Simulation results show that MBDNAs perform better than do the MDAs [39, 40], Kasbah [35] and NDFs [32] by taking the function of the proposed negotiation factor into consideration. For our future work we will extend MBDNA to have more suitable factors (e.g., flexibility in negotiator's trading partner's proposal and negotiator's proposal deviation of the average of its trading partners' proposals) in designing negotiation agent besides considering to relax bargaining terms to achieve both suitable utilities and suitable success rate under different market conditions for both GRO group and GRC group.

Although there is good opportunity for grid applications to benefit from MBDNAs in regulating the supply (grid resources which are provided by resource owners) and demand (grid resource consumers' requirements) in grid computing environments, there are still many challenges that need to be overcome before designing more effective negotiation agents. Some of these challenges are: (1) designing negotiation agents that not only applying near optimal negotiation strategies but also have the appropriate flexibility of relaxing their bargaining criteria to quickly complete a deal in the face of intense grid market pressure and (2) designing negotiation agents that not only react to current market situations but also to future market situations. One way to deal with the first challenge is to design negotiation agents that have the flexibility of relaxing bargaining criteria using fuzzy rules and a way to deal with the second challenge is to design negotiation agents with learning and predicting capabilities by analyzing *DB_game history* database that contains information of negotiation's transactions between negotiation agent and its opponents.

Acknowledgements We want to express our gratitude to Dr. Hui Li who graciously provided us with the Standard Workload Format (http://www.cs.huji.ac.il/labs/parallel/workload/logs.html) through which the CTC SP2, DAS2 fs0, DAS2 fs1, DAS2 fs2, DAS2 fs3, DAS2 fs4, HPC2N, KTH SP2, LPC EGEE, LANL CM5, LANL O2K, LCG, LLNL Atlas, LLNL T3D, LLNL Thunder, LLNL uBGL, NASA iPSC, OSC Cluster, SDSC BLUE, SDSC DS (DataStar), SDSC Par96, SDSC Par95 and SDSC SP2 traces are made publicly available.

Appendix

For the benefit of readers, the authors summarize in Table 4 the key symbols and their definitions used in this paper.

Table 4 Notation and basic terms used in the paper (alphabetic sort)

Symbol	Basic definition
A_child_k	kth instance of negotiator A
AS	Action space
$Ave.neg.time_A^{B_k}$	The average negotiation time between A and B_k in all GRNMs which both participate
B_k	kth trading partner of negotiator A
Bud_{eff}	How efficiently the available budget was spent
Bud_{init}	Initial budget allocated to a GRC
Bud_{spent}	The amount of budget spent in leasing computing resource(s) for processing tasks that are successfully scheduled and executed
C_c	The total computing capacity of the grid
c^A	The worst possible utility for A (e.g., if the negotiation ends in disagreement)
CC_t^A	Competition function of negotiator A at negotiation round t
$FST_t^{A\_child_k}$	Final price-oriented strategy that is taken by A_child_k
GRC	Grid resource consumer
GRC_MBDNA	Grid resource consumer of type MBDNA
$GRC\_job\_prof_p^i$	GRC_i's pth job characteristics
$Grid\_load$	Utilization status of computing resources
$grid.name$	Name of observed grids in work load traces
$GRNM$	Grid resource negotiation market
GRO	Grid resource owner
GRO_MBDNA	Grid resource owner of type MBDNA

Table 4 (*Continued*)

Symbol	Basic definition
$GRO\_resource\_prof_r^j$	GRO_j's rth resource characteristics
IP_A	Initial price of negotiator A
$IST_t^{A\_child_k}$	Initial price-oriented strategy that is taken by A_child_k
K^A	The constant that determines the price to be offered in the first proposal of negotiator A of type NDF
$max\_pot\_user_{grid.name}$	Maximum number of potential unique users of a grid in $grid.name$
N_{suc}	The number of tasks that are successfully scheduled and executed
$no.competitor_t^A$	Number of negotiator A's competitors at round t
$no.trading\_partner_t^A$	Number of negotiator A's trading partners at round t
O_t^A	Opportunity function of negotiator A at negotiation round t
$P_t^{A\_child_k}$	A_child_k's proposal at round t
$P_{t-1}^{B_k}$	Proposal of B_k at round $t-1$
P_{gen}	The probability that an agent enter the GRNM in each round of negotiation
P_{GRC}	The probability of an agent being GRC agent
P_m	The probability of a GRC generating a task that needs computing resources at each negotiation round
P_{tc}	Percentage of a GRC's set of tasks that is accomplished by successful negotiation and leasing grid resources
$P_t^{consensus}$	The consensus price
$PreBehave\_Depend_t^{B_k}$	MBDNAs' behavior function (e.g., previous behavior of B_k)
R_p	The expected amount of processing requested per time interval
$repeated\_user_{grid.name}$	Represents percentage of $grid.name$'s users that are observed previously in $unique\_user\_set_{grid.name}$
RP_A	Reserve price of A
t	Negotiation round
t_{deadline}^A	A's deadline (e.g., a time frame by which A needs negotiation result)
T_t^A	Time preference function of negotiator A at negotiation round t
u_{min}	The amount that is considered to distinguish the utilities between deals and no deals
$U_t^{A\_child_k}[P_t^{A\_child_k} \rightarrow B_k]$	Utility of A_child_k's at round t if its proposal is accepted by B_k
$U_t^{A\_child_k}[P_{t-1}^{B_k} \rightarrow A\_child_k]$	Utility that is generated for A_child_k by accepting the opponent's proposal $P_{t-1}^{B_k}$
$unique\_user\_set\_mem_{grid.name}$	The set of observed unique users in the $grid.name$'s SWF archive
$\#GRNM_{B_k-A}$	Total number of GRNMs in which both B_k and A participate
$\#Suc.neg_{B_k-A}$	Total number of successful negotiations between A and B_k in all GRNMs which both participate
λ	Negotiator A (of type MBDNA/MDA)'s time preference
ψ	Negotiator A (of type NDF)'s time preference
Δ_t	The amount of concession at negotiation round t

References

1. Mazzoleni P, Crispo B, Sivasubramanian S, Bertino E (2009) Efficient integration of fine-grained access control and resource brokering in grid. J Supercomput 49(1):108–126
2. Bradley A, Curran K, Gerard P (2006) Discovering resources in computational GRID environments. J Supercomput 35(1):27–49
3. Krauter K, Buyya R, Maheswaran M (2002) A taxonomy and survey of grid resource management systems. Int J Softw Pract Exp 32(2):135–164
4. Foster I, Kesselman C (2004) The grid 2: blueprint for a new computing infrastructure, 2nd edn. Morgan Kaufmann, San Mateo. ISBN:1-55860-933-4
5. Xing L, Lan Z (2009) A grid resource allocation method based on iterative combinatorial auctions. In: ITCS, international conference on information technology and computer science, July 2009, vol 2, pp 322–325.
6. Weng C, Li M, Lu X (2007) Grid resource management based on economic mechanisms. J Supercomput 42(2):181–199
7. Tucker PA, Berman FD (1996) On market mechanisms as a software technique. Technical report CS96-513, Department of Computer Science and Engineering, University of California, San Diego
8. Izakian H, Abraham A, Tork Ladani B (2010) An auction method for resource allocation in computational grids. Future Gener Comput Syst J 26:228–235
9. Buyya R, Abramson D, Giddy J, Stockinger H (2002) Economic models for resource management and scheduling in grid computing. Concurr Comput Pract Exp 14:1507–1542. doi:10.1002/cpe.690
10. Miller M, Drexler K (1988) Markets and computation: agoric open systems. In: Huberman B (ed) The ecology of computation. North-Holland/Elsevier, Amsterdam, pp 133–176
11. Buyya R, Abramson D, Giddy J (2000) An economy driven resource management architecture for global computational power grids. In: The 2000 international conference on parallel and distributed processing techniques and applications (PDPTA 2000), June 2000, Las Vegas, USA
12. Buyya R (2002) Economic-based distributed resource management and scheduling for grid computing. PhD dissertation, Monash Univ, Melbourne, Australia
13. Electricity trading over the internet begins in six New England states (1999, May 13) [Online]. Available: http://industry.java.sun.com/javanews/stories/story2/0,1072,15093,00.html
14. Huhns M, Stephens L (2000) Multiagent systems and societies of agents. In: Weiss G (ed) Multiagent systems. MIT Press, Cambridge
15. Lazar A, Semret N (1997) Auctions for network resource sharing. Columbia Univ, TR 468-97-02
16. Kersten G, Noronha S, Teich J (2000) Are all e-commerce negotiations auctions? In: Proc 4th int conf design cooperative syst, Sophia-Antipolis, France, pp 1–11
17. Wolski R, Brevik J, Plank J, Bryan T (2003) Grid resource allocation and control using computational economies. In: Berman F, Hey A, Fox G (eds) Grid computing—making the global infrastructure a reality. Wiley, New York, pp 747–771
18. Wolski R, Plank J, Brevik J (2001) G-commerce: building computational marketplaces for the computational grid [Online]. Available: http://www.cs.ucsb.edu/~rich/publications
19. Buyya R, Vazhkudai S (2001) Compute power market: towards a market-oriented grid. In: Proc 1st IEEE/ACM int symp cluster comput grid, Brisbane, Australia, Qld, May 2001, pp 15–18
20. Wolski R, Brevik J, Plank J, Bryan T (2001) Analyzing market-based resource allocation strategies for the computational grid. Int J High Perform Comput Appl 15(3):258–281 [Online]. Available: http://www.cs.utk.edu/~rich/publications/CS-00
21. Sim KM (2010) Grid resource negotiation: survey and new directions. IEEE Trans Syst Man Cybern C Appl Rev 40(3):245–257
22. Chunlin L (2011) Two-level market solution for services composition optimization in mobile grid. J Netw Comput Appl 34(2):739–749
23. Johnston W (2002) The computing and data grid approach: infrastructure for distributed science applications. Comput Inform 21(4):293–319
24. Li C, Li L (2007) A distributed iterative algorithm for optimal scheduling in grid computing. Comput Inform 26(6):605–626
25. Srinivas VV, Varadhan VV (2011) Intelligent agent based resource sharing in grid computing. Inf Technol Mob Commun, Commun Comput Inf Sci 47(Part 1):106–110
26. Pastore S (2008) The service discovery methods issue: a web services UDDI specification framework integrated in a grid environment. J Netw Comput Appl 31:93–107
27. Foster I, Jennings NR, Kesselman C (2005) Brain meets brawn: why grid and agents need each other. In: Proc towards the learning grid, pp 28–40

28. Smolinski R (2006) Fundamentals of international negotiation. In: Paluchowski WJ (ed) Negocjacje: wsrod jawnych zagrozen i ukrytych mozliwosci. Poznan, Rebis, pp 175–189
29. Montes J, Sánchez A, Pérez MS (2011) Grid global behavior prediction. In: The 11th IEEE/ACM international symposium on cluster, cloud and grid computing (CCgrid 2011), pp 124–133
30. Mok WWH, Sundarraj RP (2005) Learning algorithms for single-instance electronic negotiations using the timedependent behavioral tactic. ACM Trans Internet Technol 5(1):195–230
31. Ren F (2010) Autonomous agent negotiation strategies in complex environments. PhD thesis, Wollongong Univ, New South Wales, Australia
32. Faratin P, Sierra C, Jennings NR (1998) Negotiation decision functions for autonomous agents. Int J Robot Auton Syst 24:159–182
33. Lang F (2005) Developing dynamic strategies for multi-issue automated contracting in the agent based commercial grid. In: International symposium on cluster computing and the grid (CCGrid 2005), Cardiff, UK. IEEE Comput Soc, Los Alamitos, pp 342–349
34. Lawley R, Luck M, Decker K, Payne T, Moreau L (2003) Automated negotiation between publishers and consumers of grid notifications. Parallel Process Lett 13:537–548
35. Chavez A, Maes P (1996) Kasbah: an agent marketplace for buying and selling goods. In: Proceedings of the first international conference on the practical application of intelligent agents and multi_agent technology, London, UK, pp 159–178
36. Guttman RH, Maes P (1998) Agent-mediated integrative negotiation for retail electronic commerce. In: Proceedings of the 2nd international workshop on cooperative information agents (CIA98), Minneapolis, Minnesota
37. Sim KM, Ng KF (2006) A relaxed-criteria bargaining protocol for grid resource management. In: Proceedings of the sixth IEEE international symposium on cluster computing and the grid workshops (CCGRIDW'06), Singapore
38. Sim KM, Ng KF (2007) Relaxed-criteria negotiation for G-commerce. Int Trans Syst Sci Appl 3:105–117. Invited Paper
39. Sim KM (2006) Grid commerce, market-driven G-negotiation, and grid resource management. IEEE Trans Syst Man Cybern, Part B 36:1381–1394
40. Sim KM (2005) Equilibria, prudent compromises, and the "waiting" game. IEEE Trans Syst Man Cybern, Part B 35:712–724
41. Zhao H, Li X (2009) Efficient grid task-bundle allocation using bargaining based self-adaptive auction. In: Proceedings of the 9th IEEE/ACM international symposium on cluster computing and the grid, CCGrid 09, Shanghai. IEEE Comput Soc, Los Alamitos, pp 4–11
42. Czajkowski K, Foster I, Kesselman C (1999) Resource co-allocation in computational grids. In: 8th IEEE international symposium on high performance distributed computing (HPDC-8'99), Redondo Beach, California. IEEE Comput Soc, Washington, DC, pp 219–228
43. Czajkowski K, Foster I, Kesselman C, Sander V, Tuecke S (2002) SNAP: a protocol for negotiating service level agreements and coordinating resource management in distributed systems. In: 8th workshop on job scheduling strategies for parallel processing (JSSPP). Lecture notes on computer science series, vol 2537, Edinburgh, Scotland. Springer, Berlin, pp 153–183
44. Czajkowski K, Foster I, Kesselman C (2005) Agreement-based resource management. Proc IEEE 93:631–643
45. An B (2011) Automated negotiation for complex multi-agent resource allocation. PhD thesis, University of Massachusetts–Amherst
46. Gimpel H, Ludwig H, Dan A, Kearney B (2003) PANDA: specifying policies for automated negotiations of service contracts. In: ICSOC 2003. Lecture notes in computer science, vol 2910. Springer, New York, pp 287–302
47. Venugopal S, Chu X, Buyya R (2008) A negotiation mechanism for advance resource reservation using the alternate offers protocol. In: Proceedings of the 16th international workshop on quality of service (IWQoS 2008), Twente, The Netherlands
48. Dang Minh Q, Jorn A (2008) Bilateral bargaining game and fuzzy logic in the system handling SLA-based workflow. In: Proceedings of the 22nd international conference on advanced information networking and applications-workshops. IEEE Comput Soc, Los Alamitos
49. Wooldridge M (2002) An introduction to multiagent systems. Wiley, New York
50. Kraus S (2001) Strategic negotiation in multi-agent environments. MIT Press, Cambridge
51. Chunlin L, Layuan L (2008) A new optimal approach for multiple optimisation objectives grid resource allocation and scheduling. Int J Syst Sci 39(12):1127–1138
52. Rubinstein A (1982) Perfect equilibrium in a bargaining model. Econometrica 50(1):97–109

53. Allenotor D, Thulasiram R (2008) Grid resources pricing: a novel financial option based quality of service-profit quasi-static equilibrium model. In: Proceedings of the 2008 9th IEEE/ACM international conference on grid computing
54. Lang F (2005) Developing dynamic strategies for multi-issue automated contracting in the agent based commercial grid. In: Fifth IEEE international symposium on cluster computing and the grid (CCGrid'05), vol 1, pp 342–349
55. Ito T, Klein M, Hattori H (2008) A multi-issue negotiation protocol among agents with nonlinear utility functions. Multiagent Grid Syst 4(1):1
56. Li C, Li L, Lu Z (2005) Utility driven dynamic resource allocation using competitive markets in computational grid. Adv Eng Softw 36:425–434
57. Li Z, Cheng C, Huang F (2009) Utility-driven solution for optimal resource allocation in computational grid computer languages. Syst Struct Elsevier 35:406–421
58. Ghosh P, Roy N, Das S, Basu K (2004) A game theory based pricing strategy for job allocation in mobile grids. In: Proc int parallel distrib process symp, Santa Fe, NM, p 8b
59. Ghosh P, Roy N, Das S, Basu K (2005) A pricing strategy for job allocation in mobile grids using a non-cooperative bargaining theory framework. J Parallel Distrib Comput 65(11):1366–1383
60. Binmore K, Dasgupta P (1987) Nash bargaining theory: an introduction. In: Binmore K, Dasgupta P (eds) The economics of bargaining. Blackwell, London
61. http://www.cs.huji.ac.il/labs/parallel/workload/logs.html
62. Osborne MJ, Rubinstein A (1990) Bargaining and markets. Academic Press, New York
63. Sim KM (2006) Relaxed-criteria G-negotiation for grid resource co-allocation (position paper). ACM SIGecom, E-Commerce Exch 6(2):37–46
64. Buyya R, Murshed M (2002) GridSim: a toolkit for the modeling and simulation of distributed management and scheduling for grid computing. The journal of Concurrency and Computation: Practice and Experience (CCPE) 14(13–15)
65. Aminul H, Hashimi SA, Parthiban R (2011) A survey of economic models in grid computing. Future Gener Comput Syst J 27(8):1056–1069
66. Chacin P, Leon X, Brunner R, Freitag F, Navarro L (2008) Core services for grid markets. In: Proceedings of the CoreGrid symposium (CGSYMP), Spain. Springer, Berlin, pp 205–215
67. Montano BR, Yoon V, Drummey K, Liebowitz J (2008) Agent learning in the multi-agent contracting system [MACS]. J Decis Support Syst 45(1):140–149
68. Shen C, Peng X, Lu Y, Liu L (2011) An adaptive many-to-many negotiation model in an open market. J Comput Inf Syst 7(4):1038–1045
69. Sim KM (2010) Towards complex negotiation for cloud economy. In: Advances in grid and pervasive computing. Lecture notes in computer science, pp 395–406
70. Yoo D, Sim KM (2010) A multilateral negotiation model for cloud service market. Grid Distrib Comput Control Autom Commun Comput Inf Sci 121:54–63
71. Salvatore D (1997) Microeconomics theory and applications. Addison-Wesley, Reading
72. Sim KM (2008) Evolving fuzzy rules for relaxed-criteria negotiation. IEEE Trans Syst Man Cybern, Part B 38(6):1486–1500
73. Nemeth Z, Gombas G, Balaton Z (2004) Performance evaluation on grids: directions, issues, and open problems. In: Proc 12th euromicro conf parallel, distrib netw-based process, pp 290–297
74. Németh Z (2003) Grid performance, grid benchmarks, grid metrics. Cracow grid workshop. In: Proc 3rd Cracow grid workshop, Cracow, October 2003, pp 34–41
75. Computer networks and distributed systems, retreat of the computer networks and distributed systems research group (2005) Institute of computer science and applied mathematics, University of Bern. http://www.iam.unibe.ch/rvs/events/