Double Auction Protocols for Resource Allocation in Grids

Umesh Kant Daniel Grosu
Dept. of Computer Science, Wayne State University
5143 Cass Avenue, Detroit, MI 48202
ap5651@wayne.edu, dgrosu@cs.wayne.edu

Abstract

In this paper we propose the double auction allocation model for grids, and three double auction protocols for resource allocation: Preston-McAfee Double Auction Protocol (PMDA)*,* Threshold Price Double Auction Protocol (TPDA) *and* Continuous Double Auction Protocol (CDA)*. We study these protocols in terms of economic efficiency and system performance. The results show that* CDA *protocol is better from both resource's and user's perspective providing high resource utilization.*

1. Introduction

Grid computing environments enable flexible, secure, and coordinated resource sharing among dynamic collections of institutions distributed across the world called virtual organizations [4]. These virtual organizations share a diverse set of resources such as computers, storage, networks or sensors. In these settings where resources are geographically distributed and users and resource owners have different objectives, the design of optimal resource allocation protocols is a challenging task. The resource management protocols used in traditional computing systems cannot be simply applied to these complex environments because they assume complete control over resources. Thus we need new resource allocation protocols that take into account the objectives of both users and resource owners. A viable solution is to consider market-based resource allocation protocols [2, 9] which are based on trading and resource brokering policies between resource owners and users. These protocols are suitable for grid environments because of their decentralized structure and the use of incentives for resource owners to contribute resources.

There exists two categories of market based models that are suitable for resource management in distributed systems. The first one is the *commodities market model* [2] in which resource owners determine the price for their services and charge users depending on the amount of resource they consume. The second one is the *auction model* in which each service provider and user acts independently and they agree privately on the selling price. The advantage of using auctions for resource allocation is that they require little global information, have decentralized structure and are easy to implement. Depending on the type of interactions between sellers and buyers auctions can be classified into two classes. In *one-sided auctions* bids are submitted by the grid users to a central auctioneer. The auctioneer decides the winner based on different auction mechanisms. Examples of one-sided auctions are: First-price auction (the highest bidder wins and pays the amount he bid) and Vickrey auction (the highest bidder wins and pays an amount equal to the second highest bid). In *two-sided auctions*, also called double auctions, both users and resource owners submit bids. To distinguish the bids we call 'asks' the bids submitted by the resource owners. The selling price and the users and the resource owners that trade are decided by the central auctioneer according to different types of double auction mechanisms. All the previous studies on auction allocation considered only one type of auction (double or Vickrey) and compared it with other economic-based and conventional models. Therefore, in this paper we investigate several types of double auctions in terms of their suitability in grid systems, economic efficiency and system performance. We define the double auction model in which *resource owners* provide services like computational power, data storage, software or computer networks and *users* consume services provided by resource owners.

Considering the double auction model we evaluate three types of double auction allocation protocols: *Preston-McAfee Double Auction (PMDA)* [7], *Threshold Price Double Auction (TPDA)* [10] and *Continuous Double Auction (CDA)*. The most simple and reliable way to evaluate these protocols is to perform experiments in a real grid environment. This approach suffers of some problems. One is the difficulty to find a real grid environment which allows changes and different configurations. Also the repeatability of the results will be compromised due to the varying nature of load on resources. The only suitable solution for investigating the effectiveness of these resource allocation protocols over a wide range of scenarios with reproducible results is to consider simulations. Thus we developed a simulator which allows us to compare the three double auction allocation protocols.

Related work. Several economic-based resource management models for distributed systems have been investigated in the past [1, 3, 5, 8, 9]. The authors of [2] provide a comprehensive survey of economic models for resource management in distributed systems. Resource allocation in grids based on commodities markets and auctions was studied by Wolski *et al.* [9]. These two models were compared in terms of price stability and market equilibrium. In [5] the authors studied the continuous double auction as a resource allocation mechanism.

Our contributions. The study in this paper is motivated by the fact that previous work studied only one type of auction (double or second-price) and compared it with other economic-based and conventional models without investigating the suitability of different auction types for resource allocation. In this paper we present the double auction allocation model for grids, and three double auction protocols for resource allocation: *PMDA*, *TPDA* and *CDA*. We study them by simulation and analyze their suitability for grids in terms of economic efficiency and system performance. The results show that the *CDA* protocol is better from both resource's and user's perspective providing high resource utilization.

Organization. The paper is structured as follows. Section 2 presents the double auction allocation model. In Section 3 we describe the three double auction protocols. In Section 4 we give a brief description of our simulation environment. Section 5 presents the experimental results obtained by simulating the double auction protocols. In Section 6 we draw conclusions and present future research directions.

2. Double Auction Allocation Model

There are three main types of participants in the double auction model (Figure 1): Grid Service Providers (GSP), User Brokers (UB) and Local Markets for Auctions (LMA). In the following we present each of these participants and describe their role in the model and their characteristics.

User broker (UB): User Brokers helps users decide the appropriate choice of resources and auction types. A User Broker is associated with a User and it is responsible for auction (resource) discovery, auction analysis and selection, bid submission, sending user jobs to resources, collecting the results and providing the user with a uniform view of grid resources. There are four component agents of the user broker. The *Job Management Agent* is responsible

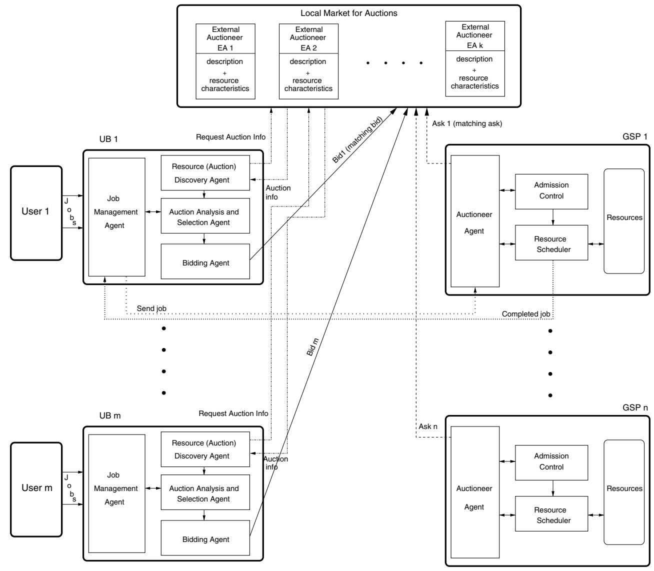

Figure 1. Double Auction Allocation Model

for user interaction, job creation, submission and monitorization. It also coordinates the auction analysis and selection, resource discovery and the bidding process. When the jobs complete it collects the results of the computation. The *Resource (Auction) Discovery Agent* is responsible for resource/auction discovery. The *Auction Analysis and Selection Agent* is responsible for analyzing the auction information submitted by the Local Market for Auctions. Based on the user requirements and on the properties of the auctions it selects an auction in which the user will participate. The *Bidding Agent* is responsible for choosing and submitting the bid to the selected External Auctioneer (EA). If it is a successful bid the Job Management Agent sends the user jobs for execution to the corresponding GSP.

In our model we assume that there are m users, $U_1, U_2, \ldots, U_m$, each having a number of tasks ready to be submitted for execution. User U_i is characterized by three parameters:

(i) *Work* (w_i): It is defined as the total amount of work in millions of instructions for all the tasks of user U_i.

(ii) *Number of tasks* (m_i): Represents the total number of tasks of user U_i.

(iii) *Budget* (B_i): It is defined as the maximum amount of 'grid dollars' (G$) a user can pay to resources for executing their tasks. The payments given to GSPs must be within the user budget.

The goal of the users is to finish their work at the earliest time and to pay as little as possible out of their budgets. The price paid is decided by the double auction protocols.

Grid Service Providers (GSP): GSPs own resources and they contribute their resources to the Grid. User are charged by the GSPs for services they provide. Each GSP decides to participate in a double auction protocol run by an External Auctioneer (EA). Each GSP has an Auctioneer Agent which is responsible for preparing the ask prices and sending them to EA. It receives the result of the auction from EA and once the users are decided it accepts jobs from these users. It is also responsible for admission control and resource scheduling.

We consider that GSP$_i$ is responsible for resource R_i's management ($i = 1, 2, \ldots, n$). Each resource R_i is characterized by the following:

(i) *Processing rate* (s_i): It is given in millions instructions per second (MIPS).

(ii) *Reservation price* (p_i): It is defined as the minimum price accepted by resource R_i for one second of job execution.

(iii) *Cost* (C_i): Represents the cost incurred by R_i for one second of task execution.

(iv) *Resource Profit* (P_i): It is the profit gained by R_i by executing user tasks. It is defined as the total payment given by the users for executing their tasks minus the total cost. The payment is given in G$.

Local Market for Auctions (LMA): It provides support for implementing and deploying double auction mechanisms, and enables the users to find the right auctions that match their requirements and preferences. It provides a set of External Auctioneers (EA) which will be responsible for running different double auctions. LMA takes a request from a user specified in an appropriate language and returns the auctions that match the request. It also accepts ask prices from GSPs in case some of them decided to participate in one of the available double auctions.

3. Double Auction Allocation Protocols

Preston-McAfee Double Auction Protocol (PMDA). The auction considered as the basis for this protocol is the double auction developed by Preston and McAfee [7]. In this type of auction the users submit bids and GSPs submit asks to an External Auctioneer. The equilibrium price is determined by matching asks (starting from the lowest price to the highest) with demand bids (starting from the highest price to the lowest). This protocol is executed by EA which is part of LMA. Once a set of GSPs decided to participate in a double auction, EA posts the auction description on LMA. We assume that only GSPs having resources of the same type participate in one double auction protocol. We also assume that the following users $\{U_1, U_2, \ldots, U_m\}$ decided to participate in the double auction.

PMDA Protocol:

Phase I: Bidding

1. UB$_j$, $j = 1, 2, \ldots, m$, sends bid b_j to EA.
2. for $j = 1$ to m, EA receives bid b_j.
3. GSP$_i$, $i = 1, 2, \ldots, n$, sends ask a_i to EA.
4. for $j = 1$ to n, EA receives ask a_i.

Phase II: Completion

1. After EA collects all the bids $\{b_1, b_2, \ldots, b_m\}$ and all the asks $\{a_1, a_2, \ldots, a_n\}$, it does the following:
   - 1.1. Sorts bids in decreasing order and asks in increasing order
     $b_{\pi(1)} \geq b_{\pi(2)} \geq \ldots \geq b_{\pi(m)}$
     $a_{\sigma(1)} \leq a_{\sigma(2)} \leq \ldots \leq a_{\sigma(n)}$
     where π and σ are the permutations defining the orders statistics above.
   - 1.2. Finds k such that $b_{\pi(k)} \geq a_{\sigma(k)}$ and $b_{\pi(k+1)} < a_{\sigma(k+1)}$.
   - 1.3. Determines the trading price t.
     $t = \frac{1}{2}(b_{\pi(k+1)} + a_{\sigma(k+1)})$
   - 1.4. If $a_{\sigma(k)} \leq t \leq b_{\pi(k)}$, notifies GSP$_{\sigma(i)}$/UB$_{\pi(i)}$, $i = 1, 2, \ldots, k$, that they can trade at price t.
   - 1.5. If $t \geq b_{\pi(k)}$ or $t < a_{\sigma(k)}$, notifies GSP$_{\sigma(i)}$/UB$_{\pi(i)}$, $i = 1, 2, \ldots, k-1$, that they can trade. Each GSP gets $a_{\sigma(k)}$, and each UB pays $b_{\pi(k)}$.
   - 1.6. Sends reject messages to GSPs and UBs that do not trade.
2. UBs that trade send jobs to the corresponding GSPs and GSPs execute them.
3. UBs send payments to the corresponding GSPs.

If the condition in (1.5) holds, GSP$_{\sigma(i)}$ receives $a_{\sigma(k)}$ and UB$_{\pi(i)}$ pays $b_{\pi(k)}$, for $i = 1, 2, \ldots, k-1$. As a result of this trade there is a surplus of $(k-1)(b_{\pi(k)} + a_{\sigma(k)})$. We assume here that this surplus is kept by EA which plays the role of a budget balancer. We also assume that $b_{\pi(m+1)}$ is the lowest possible valuation of the users, $a_{\sigma(n+1)}$ is the highest possible valuation of the resources and that $b_{\pi(m+1)} < a_{\sigma(n+1)}$ holds.

Threshold Price Double Auction Protocol (TPDA). The auction considered as the basis for this protocol is the double auction developed by Yokoo *et al.* [10]. This protocol takes into account the possibility of cheating by submitting false-name bids. One example of cheating is when a GSP may try to make profit by pretending to be a user (potential buyer) and submitting a false-name bid. The idea is to use a *threshold price r* which is determined by EA without knowing the evaluations of GSPs and users. This threshold price will be used to control the number of trades and the trading price. As in PMDA the users submit bids and GSPs submit asks to an External Auctioneer. The trading price is determined by matching asks (starting from the lowest price to the highest) with demand bids (starting from the highest price to the lowest) considering the threshold price. This protocol is executed by EA which is part of LMA. Once a set of GSPs decided to participate in a double auction, EA posts the auction description on LMA. We assume that only GSPs having resources of the same type participate in one double auction protocol. We also assume that the following users $\{U_1, U_2, \dots, U_m\}$ decided to participate in the double auction.

TPDA Protocol:
Phase I: Bidding (described in the PMDA protocol above)
Phase II: Completion

1. After EA collects all the bids $\{b_1, b_2, \dots, b_m\}$ and all the asks $\{a_1, a_2, \dots, a_n\}$, it does the following:
   1.1. Sorts bids in decreasing order and asks in increasing order
   $b_{\pi(1)} \geq b_{\pi(2)} \geq \dots \geq b_{\pi(i)} \geq r > b_{\pi(i+1)} \geq \dots \geq b_{\pi(m)}$
   $a_{\sigma(1)} \leq a_{\sigma(2)} \leq \dots \leq a_{\sigma(j)} \leq r < a_{\sigma(j+1)} \leq \dots \leq a_{\sigma(n)}$
   where π and σ are the permutations defining the orders statistics above.
   1.2. If $i = j$, it notifies GSP$_{\sigma(l)}$/UB$_{\pi(l)}$, $l = 1, 2, \dots, i$, that they can trade at price r.
   1.3. If $i > j$, it notifies GSP$_{\sigma(l)}$/UB$_{\pi(l)}$, $l = 1, 2, \dots, j$, that they can trade. Each UB that trade pays $b_{\pi(j+1)}$ and each GSP that trade gets r. EA gets the surplus amount of $j(b_{\pi(j+1)} - r)$.
   1.4. If $i < j$, it notifies GSP$_{\sigma(l)}$/UB$_{\pi(l)}$, $l = 1, 2, \dots, i$, that they can trade. Each UB that trade pays r and each GSP that trade gets $a_{\sigma(i+1)}$. EA gets the surplus amount of $i(r - a_{\sigma(i+1)})$.
   1.6. Sends reject messages to GSPs and UBs that do not trade.
2. UBs that trade send jobs to the corresponding GSPs and GSPs execute them.
3. UBs send payments to the corresponding GSPs.

Continuous Double Auction Protocol (CDA). In CDA there are no trading phases, at any time users and GSPs can submit bids and asks to EA. EA maintains a lists of the current bids and asks and matches two offers when the bid is higher or equal to the lowest ask. The trade occurs at the average of matching ask and bid prices. This protocol is executed by EA which is part of LMA. Once a set of GSPs decided to participate in a CDA, EA posts the auction description on LMA. We assume that only GSPs having resources of the same type participate in CDA.

CDA Protocol:

EA maintains the sorted lists of current asks and bids:
$b_{\pi(1)} \geq b_{\pi(2)} \geq \dots \geq b_{\pi(k)}, k \leq m$
$a_{\sigma(1)} \leq a_{\sigma(2)} \leq \dots \leq a_{\sigma(l)}, l \leq n$
where π and σ are the permutations defining the orders statistics above.

1. When EA receives a new bid b_i from UB$_i$ it does the following:
   1.1. If $b_i \geq a_{\sigma(1)}$ then it notifies UB$_i$ and GSP$_{\sigma(1)}$ that they can trade at the price $\frac{1}{2}(b_i + a_{\sigma(1)})$.
   1.2. If $b_i < a_{\sigma(1)}$ then no trade occurs and EA inserts b_i into the sorted list of current bids.
2. When EA receives a new ask a_j from GSP$_j$ it does the following:
   2.1. If $a_j \leq b_{\pi(1)}$ then it notifies UB$_i$ and GSP$_{\pi(1)}$ that they can trade at the price $\frac{1}{2}(b_{\pi(1)} + a_j)$.
   2.2. If $a_j > b_{\pi(1)}$ then no trade occurs and EA inserts a_j into the sorted list of current asks.
3. UBs that trade send jobs to the corresponding GSPs and GSPs execute them.
4. UBs send payments to the corresponding GSPs.

4. Simulation Environment

In order to study the allocation protocols presented in the previous section we developed a grid simulator using the Mercatus toolkit [6]. The simulator facilitates the evaluation of double auction resource allocation protocols in terms of their economic efficiency by reporting user spending, resource cost and resource profit. It also allows the study of system efficiency in terms of execution time and resource utilization. We simulated a grid environment consisting of fifteen resources shared by ten users. Table 1 shows the parameters of the users. The total work for a user is given in millions of instructions and the budget is given in 'grid dollars' (G$). Resources are divided into two groups based on their processing rate. Table 2 shows the parameters of the two groups of resources. Members indicate the number of resources that belong to each group. In addition to the three double auction protocols we simulated a Round-Robin type protocol. We use Round-Robin as a reference for the other protocols when we study the system utilization. In the following we give a brief description of the Round-Robin protocol (RR). In RR no pricing is involved. An arriving task is matched with the next available resource that has the reservation price equal to or less than the task budget.

User	Total work	Budget (G$)	Number of tasks
0	4.2	100	4
1	6.5	150	4
2	3.0	180	5
3	4.0	100	3
4	4.25	100	5
5	5.9	140	5
6	4.0	100	6
7	5.0	160	7
8	4.6	200	4
9	3.1	140	5

Table 1. Users.

Group	Members	Processing rate (MIPS)	Cost
0	6	50	400
1	9	100	800

Table 2. Resources.

5. Experimental Results

In this section we investigate by simulations the three double auction protocols (PMDA, TPDA, CDA) and the Round-Robin (RR) protocol. As PMDA, TPDA and CDA involve payments we compare them in terms of profit gained by the resources and the amount of G$ spent by users. We also compare PMDA, TPDA, CDA and RR in terms of resource utilization. All the simulations are run for a network latency of 10^{-5} sec. and the results are obtained as an average of five simulations. To fairly compare the various protocols we use the same resource registration sequence. Resource registration sequence is the order in which the resources register themselves with the External Auctioneer.

Resource profit. The *resource profit* is the difference between the payments received by a resource and the cost of processing the assigned user tasks. Here we present the profit as percentage of the cost. The resource profit helps us to determine which auction protocol is better from resource's perspective. Figure 2 shows the percentage profit earned by each resource in PMDA, TPDA and CDA. We observe a lot of variation in resource profits for different resources in case of PMDA and TPDA. This does not happen in case of CDA where all the resources are treated more fairly than compared to PMDA and TPDA. In case of PMDA Resource 5 gains a very high profit because it wins a task according to case (1.4) of PMDA, and the payment given by users is high. Resource 7 also wins a task according to case (1.4) of PMDA and hence has a high profit percentage. Resources which sometimes win tasks according to case (1.4) and sometimes according to case (1.5) obtain intermediate profit percentage. Resources which win tasks according to case (1.5) of PMDA have the lowest profit percentage. PMDA protocol requires that there should be at least two resources (according to case (1.5)) in order to complete a trading phase, but sometimes it may happen that there is no other resource available during consecutive auction rounds and some resources gain a very low profit. TPDA does not have such restriction and we observe that resources do not gain a very low profit. One example is Resource 4 who gains a low profit percentage in case of PMDA and high profit percentage in case of TPDA.

We can conclude that CDA and TPDA are more stable in the sense that they provide a fair distribution of profit compared to PMDA. PMDA exhibits large variations in terms of profit across all the resources. Also TPDA provides the additional advantage of being robust to false name bids.

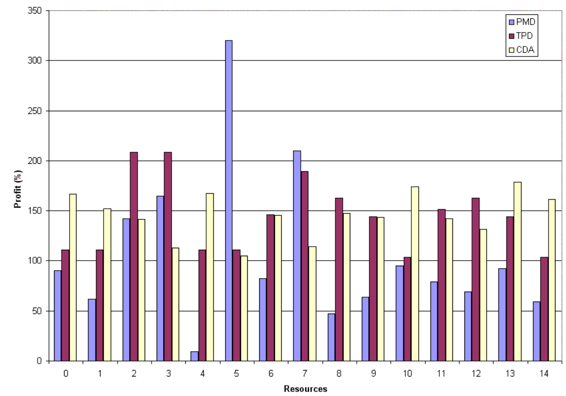

Figure 2. Resource Profit

Resource utilization. *Resource utilization* is defined as the percentage of time a resource is busy executing tasks with respect to the total simulation time. Figure 3 shows the resource utilization of each resource in PMDA, TPDA, CDA and RR. The number of messages exchanged between participants is less in case of CDA and RR compared to TPDA and PMDA thus the resource utilization is high in CDA and RR compared to TPDA and PMDA. Some resources may have low resource utilization in case of PMDA if during some auction rounds no other resource is available as PMDA protocol requires two resources to be available according to case (1.5). Because such resources wait until other resources become available, they have low utilization. The processing power of a resource also affects the resource utilization. Resources with high processing power finish their tasks early and become available to serve other tasks. So we can see that in general resources that belong to Group 1 (R6 to R14) in average have higher resource utilization compared to resources that belong to Group 0 (R0 to R5) for all protocols. Another reason that resources in Group 0 do not obtain high utilization in case of CDA and RR is that if a resource gets a big task in the beginning it spends a big fraction of time in serving it and when it starts with another task is not able to finish it before simulation time limit. According to our simulation setup we disregard the partial work done by a resource. In case of PMDA resources 0 and 2 obtain a better utilization compared to CDA. The reason is they received different tasks in case of CDA and PMDA. We can conclude that in most of the cases CDA exhibit a high utilization close to that of RR. CDA will be a better choice for resources that want to fully utilize their processing capabilities.

User's spending. The budget spent by a user is defined as the percentage of the budget a user spends with respect to

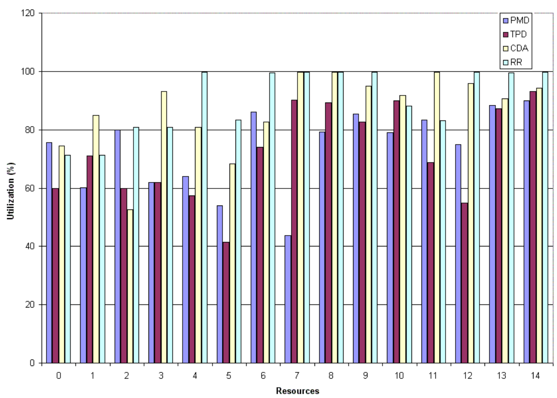

Figure 3. Resource Utilization

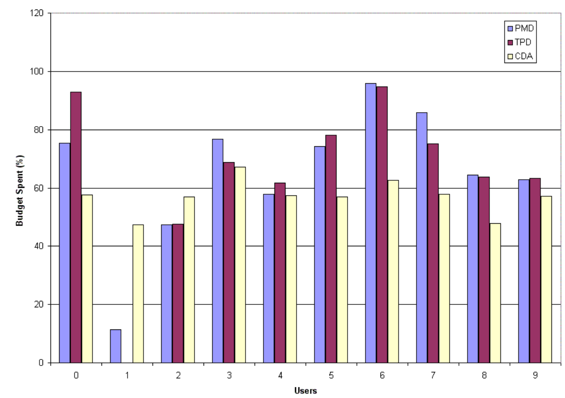

Figure 4. The budget spent by users

the original budget it had assigned at the start of the simulation. This accounting helps us determine which auction protocol is better from the user's perspective. Figure 4 shows the percentage of budget spent by each user in PMDA, TPDA and CDA. We see that User 1 spends zero G$ in case of TPDA. This is because even though the User 1 had won resources none of the resources was able to finish the work before the simulation ended. The budget spent in case of PMDA and TPDA is close but varies more from one user to another. Whereas in case of CDA the budget spent by different users do not vary much. In most of the cases the budget spent in case of CDA is less than in case of PMDA and TPDA. We can conclude that CDA is a better choice from the user's perspective.

6. Conclusion

In this paper we proposed the double auction allocation model for grids, and three double auction protocols for resource allocation: *Preston-McAfee Double Auction Protocol (PMDA)*, *Threshold Price Double Auction Protocol (TPDA)* and *Continuous Double Auction Protocol (CDA)*. We developed a simulator for these protocols and studied them in terms of economic efficiency and system performance. We also compared their performance with the performance of Round-Robin protocol. The results showed that the *Continuous Double Auction (CDA)* protocol is better from both resource's and user's perspective providing high resource utilization. Future work include an extensive simulation and the development of other auction based protocols for resource allocation in grids.

References

[1] D. Abramson, R. Buyya, and J. Giddy. A computational economy for grid computing and its implementation in the Nimrod-G resource broker. *Future Generation Computing Systems*, 18(8):1061–1074, October 2002.

[2] R. Buyya, D. Abramson, J. Giddy, and H. Stockinger. Economic models for resource allocation and scheduling in grid computing. *Concurrency and Computation: Practice and Experience*, 14(13-15):1507–1542, November 2002.

[3] K. M. Chao, R. Anane, J. H. Chen, and R. Gatward. Negotiating agents in a market-oriented grid. In *Proc. of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, pages 406–407, May 2002.

[4] I. Foster and C. Kesselman. *The Grid: a Blueprint for a New Computing Infrastructure*. 2nd edition, Morgan Kaufmann, San Francisco, CA, 2003.

[5] J. Gomoluch and M. Schroeder. Market-based resource allocation for grid computing: A model and simulation. In *Proc. of the 1st International Workshop on Middleware for Grid Computing*, pages 211–218, June 2003.

[6] D. Grosu and U. Kant. Mercatus: a toolkit for the simulation of market-based resource allocation protocols in grids. In *Proc. of the 1st International Workshop on Scientific Applications on Grid Computing (SAG'04), at the IEEE/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, September 2004.

[7] R. P. McAfee. A dominant strategy double auction. *Journal of Economic Theory*, 56:434–450, 1992.

[8] S. Vazhkudai and G. von Laszewski. A greedy grid - the grid economic engine directive. In *Proc. of the 15th IEEE International Parallel and Distributed Processing Symposium*, pages 1806–1815, April 2001.

[9] R. Wolski, J. S. Plank, J. Brevik, and T. Bryan. Analyzing market-based resource allocation strategies for the computational grid. *The International Journal of High Performance Computing Applications*, 15(3):258–281, Fall 2001.

[10] M. Yokoo, Y. Sakurai, and S. Matsubara. Robust double auction protocol against false-name bids. In *Proc. of the 21st IEEE International Conference on Distributed Computing Systems*, pages 137–145, April 2001.