AUCTION-BASED RESOURCE ALLOCATION PROTOCOLS IN GRIDS

Daniel Grosu
Department of Computer Science
Wayne State University
5143 Cass Avenue
Detroit, MI 48202
email: dgrosu@cs.wayne.edu

Anubhav Das
Department of Computer Science
Wayne State University
5143 Cass Avenue
Detroit, MI 48202
email: anubhav@cs.wayne.edu

ABSTRACT

In this paper, we study the auction model for resource management using the SimGrid simulation framework. We investigate three types of auction allocation protocols: (i) *First-Price Auction*, (ii) *Vickrey Auction* and (iii) *Double Auction*. The goal is to find which one is best suitable for the grid environment from users' perspective as well as from resources' perspective. The results showed that when we consider a mix of risk-averse and risk-neutral users *First-Price Auction* favors resources while *Vickrey Auction* favors users. On the other hand the *Double Auction* favors both users and resources.

KEY WORDS

resource allocation, grid computing, auctions, simulation.

1 Introduction

Grid systems have emerged as promising next generation computing platforms that enable the building of a wide range of collaborative problem-solving environments emerging in industry, science, and engineering. Grid environments enable flexible, secure, and coordinated resource sharing among dynamic collections of institutions distributed across the world called virtual organizations [1]. The resources shared by such virtual organizations may be computational resources, data storage or computer networks. In such scenarios where resources are geographically located worldwide and users and resource owners have different objectives, it is difficult to design optimal resource allocation mechanisms that meet the objectives of both resource owners and users as well. To address this complex problem we rely on economic-based resource allocation mechanisms [2, 3, 4]. These mechanisms are based on trading and brokering policies between the resource owners (service providers) and users (service consumers). These economic approaches are suitable for grids because of their decentralized structure and the use of incentives for resource owners to contribute resources. Also, the objectives of both users and resource owners are considered when making allocation decisions.

The most commonly studied economic models in the context of resource management in distributed systems are commodities markets and auctions. In the *commodities market model* [2] the users are charged a publicly agreed price per unit of resource consumed. In the *auction model* 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. There exists several studies on applying auction models in resource management [5, 6]. All of them studied 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 auctions in terms of their suitability in grid systems, economic efficiency and system performance. We define the auction model in which the main participants are resource owners and users. The *resource owners* provide services like computational power, data storage, software or computer networks and *users* consume services provided by resource owners. Each user has a *broker* who manages and schedules user's jobs in the Grid, bids the price that user agrees to pay in the auction and hands payments to resource providers. Each resource owner has an *auctioneer agent* who has the responsibility of setting the rules of the auction and conducting the auction for the resource. This involves collecting bids from brokers participating in the auction, deciding the winner in the auction (based on a given auction algorithm) and collecting the payment from the winner. Besides it also interacts with the local scheduler to schedule the jobs of the user who wins in the auction.

Using the auction model we evaluate three types of auction allocation protocols: *First Price Auction*, *Vickrey Auction* [7], and *Double Auction* [8]. To evaluate these resource allocation mechanisms, the most simple and reliable way would be to perform real experimentation. This involves scheduling and executing real applications on real resources. The problem with this approach is that, firstly, real applications may run for long time which is time consuming, secondly, we cannot explore a wide range of different resources by the means of experimentation on real resources. Finally, due to the varying nature of load on resources the results obtained will not be repeatable. Thus the most viable approach is to resort to simulation. In order to perform simulations we developed a simulator based on the SimGrid simulation framework [9]. The simulator allows us to compare the three auction-based allocation protocols in terms of economic efficiency and system performance.

Related Work

Economic-based resource management systems have been investigated by several researchers in [5, 6, 10, 11, 12, 13, 14, 15]. A comprehensive survey of economic models for resource management can be found in [2]. Nimrod-G, proposed in [10], is a computational economy driven resource broker that manages Grid resources. It supports several economic models such as commodities markets, spot markets and contract net. Wolski *et al.* [6] investigated the problem of resource allocation in grids under two economic models: commodities markets and auctions. They compared these two models in terms of price stability and market equilibrium. Gomoluch and Schroeder [5] investigated the performance of the double auction protocol for resource allocation. They compared it to the conventional round-robin approach and showed that the double auction protocol is superior to round-robin. Several computational markets are based on auctions, examples are Spawn [16], and CPM [17]. One of the earliest works that considered auctions as a resource allocation mechanism is [18]. The authors considered auctions to allocate processor time in a single computer. In [14] the authors investigated the allocation of one resource among tasks in which the tasks can independently offer, bid and exchange funds for resources they need. Two simulation toolkits GridSim [19] and SimGrid [9] provide core functionalities to build simulators for studying resource allocation algorithms in Grid environments.

Our contributions

The main motivation behind the study in this paper is that previous work studied only one type of auction (double or Vickrey) 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 auction model, and three auction-based protocols: *First-Price Auction*, *Vickrey Auction*, and *Double Auction*. We simulate these three protocols using the SimGrid simulation framework and study them in terms of economic efficiency and system performance. The results show that *First-Price Auction* is better from resource perspective while *Vickrey Auction* is better from user perspective. The third type of protocol, *Double auction* favors both resources and users.

Organization

The paper is structured as follows. Section 2 presents the auction allocation model and the three auction protocols. In Section 3 we give a brief description of the SimGrid simulation environment. Section 4 presents the simulation of auction allocation protocols using the SimGrid simulator and experimental results. In Section 5 we draw conclusions and present future research directions.

2 Auction Allocation: Model and Protocols

2.1 Auction Allocation Model

The main participants in the auction model (Figure 1) are: 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):

Each grid user has a User Broker. The User Broker 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 components of the user broker:

Job Management Agent: It is responsible for user interaction, job creation, submission and monitorization. It also coordinates the mechanism analysis and selection, resource discovery and the bidding process. When the jobs complete it collects the results of the computation.

Resource (Auction) Discovery Agent: It is responsible for resource/auction discovery. It sends a request for resources/auctions to the Local Market for Auctions. The Local Market for Auctions sends back the information on the auctions that match the request.

Auction Analysis and Selection Agent: It 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.

Bidding Agent: It is responsible for choosing and submitting the bid to the selected GSP auctioneer agent or to the selected External Auctioneer (EA). If it is a successful bid the Job Control 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. The tasks of user U_i are denoted by T_{ij}, $j = 1, 2, \ldots, m_i$, where m_i is the total number of tasks generated by U_i. The tasks of U_i are characterized by three parameters:

(i) *Execution time* (t_{ij}): It is defined as the execution time of task T_{ij} (in seconds) on a reference resource R_{ref}. The slowest resource in the system is considered as the reference resource.

(ii) *Resource preference* (r_{ij}): It is defined as the index of the resource on which task T_{ij} needs to be executed. If task T_{ij} needs to be executed on resource R_l then $r_{ij} = l$.

(iii) *Task budget* (g_{ij}): It is defined as the maximum amount user U_i can pay to any resource for executing T_{ij}. It is given in 'grid dollars' (G$).

User U_i has a total budget G_i in 'grid dollars' which is given by $G_i = \sum_{j=1}^{m_i} g_{ij}$. U_i's bid b_{ij} for task T_{ij}'s

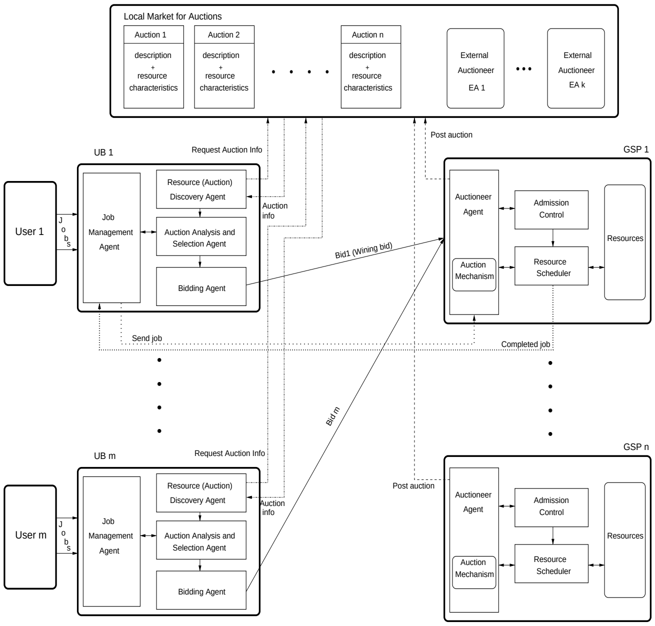

Figure 1. Auction Allocation Model

execution on a resource is within the task budget, i.e. $b_{ij} \leq g_{ij}$. If user U_i wins in an auction it pays b_{ij}, otherwise it pays nothing.

Grid Service Providers (GSP):
GSPs contribute their resources to the Grid and charge the users for services. Depending on the type of auction chosen by the GSP we have two scenarios. The first scenario is when a GSP decides to participate in a one-sided auction protocol (e.g. First Price, Vickrey). In this case GSPs create auctions mechanisms that are posted on the Local Market for Auctions. Different GSPs can deploy different auction types. The Auctioneer Agent is responsible for posting the GSP's auction mechanism on the LMA. It also runs the auction allocation mechanism, collects the bids from users and determine the winning users. The winning users are the users that send jobs for execution. Once the winning users are determined it informs the users of the result of the auction by sending success or reject messages. It also coordinates the admission control and resource scheduling.

The second scenario is when a GSP decides to participate in a two-sided auction protocol (e.g. Double Auction) run by an External Auctioneer (EA). In this case the Auctioneer Agent 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. As in the previous case 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 MIPS.

(ii) *Reservation price* (p_i): It is defined as the minimum price accepted by resource R_i for one second of job execution. We consider here the following pricing strategy: the higher the processing rate the higher the reservation price.

(iii) *Cost* (C_{ikl}): Represents the cost incurred by R_i when executing task T_{kl}. It is defined as: $C_{ikl} = p_i \, \frac{s_{ref}}{s_i} \, t_{kl}$.

(iv) *Resource Profit* (P_{ijk}): It is the profit gained by R_i by executing task T_{jk}. It is defined as $P_{ijk} = PY_{ijk} - C_{ijk}$, where PY_{ijk} is the payment given to R_i by user U_i for executing T_{jk}. The payment is given in G$. The total profit for a resource is the sum of all the profits obtained by executing all the assigned tasks.

Local Market for Auctions (LMA):
It provides support for GSPs to deploy auction mechanisms, and enables the users to find the right auctions that match their requirements and preferences. It also provide a set of External Auctioneers (EA) which will be responsible for running two-sided auctions (e.g. double auction). 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 two-sided auctions.

2.2 Auction Allocation Protocols

We present three auction allocation protocols: First Price Auction Protocol (FPA), Vickrey Auction Protocol (VA), and Double Auction Protocol (DA).

2.2.1 First Price Auction Protocol (FPA)

The auction considered as the basis for this protocol is the first price sealed-bid auction. In this type of auction, bidders don't know the bid values of other bidders. The bidder who bids the highest wins the auction and pays exactly the amount he bids. In the description of the FPA protocol we assume that the protocol is implemented by GSP$_i$. After GSP$_i$ has posted the auction description on LMA the users $\{U_1, U_2, \ldots, U_m\}$ decided to participate in the auction at GSP$_i$. In the following we give the description of the protocol. In this description, b_j is the bid submitted by UB$_j$ for a given task of U_j.

FPA Protocol:

Phase I: Bidding

1. UB$_j$, $j = 1, 2, \ldots, m$ sends bid b_j to GSP$_i$.
2. for $j = 1$ to m
   GSP$_i$ receives bid b_j.

Phase II: Completion

1. After GSP$_i$ collects all the bids $\{b_1, b_2, \ldots, b_m\}$, it does the following:
   1.1. Determines the winner U_w: $w = \{i | b_i = \max\{b_1, b_2, \ldots, b_m\}\}$
   1.2. Notifi es the winner, UB_w.

1.3. Sends reject messages to the user brokers UB_l, $l \neq w$.

2. UB_w sends the job to GSP_i and GSP_i executes it.
3. UB_w sends the payment b_w to GSP_i.

2.2.2 Vickrey Auction Protocol (VA)

The auction considered as the basis for this protocol is the Vickrey auction [7], also called the second-price auction. In this type of auction bidders don't know the bid values of other bidders. The highest bidder wins and pays the price equal to the second highest bid. This protocol is executed by GSP_i. After GSP_i has posted the auction description on LMA the users $\{U_1, U_2, \ldots, U_m\}$ decided to participate in auction at GSP_i. In the following we give the description of the protocol. In this description, b_j is the bid submitted by UB_j for a given task of U_j.

VA Protocol:

Phase I: Bidding

1. UB_j, $j = 1, 2, \ldots, m$ sends bid b_j to GSP_i.
2. for $j = 1$ to m
   GSP_i receives bid b_j.

Phase II: Completion

1. After GSP_i collects all the bids $\{b_1, b_2, \ldots, b_m\}$, it does the following:
   1.1. Determines the winner U_w: $w = \{i | b_i = \max\{b_1, b_2, \ldots, b_m\}\}$
   1.2. Determines the second-highest bid b_{w2}.
   1.3. Notifies the winner, UB_w.
   1.4. Sends reject messages to the user brokers UB_l, $l \neq w$.
2. UB_w sends the job to GSP_i and GSP_i executes it.
3. UB_w sends the payment b_{w2} to GSP_i.

2.2.3 Double Auction Protocol (DA)

The auction considered as the basis for this protocol is the double auction [8]. 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.

DA 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 $i = 1$ 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 $\text{GSP}_{\sigma(i)}$/$\text{UB}_{\pi(i)}$, $i = 1, 2, \ldots, k$, that they can trade at price t.
   1.5. If $t \geq b_{\pi(k)}$ or $t < a_{\sigma(k)}$, notifies $\text{GSP}_{\sigma(i)}$/$\text{UB}_{\pi(i)}$, $i = 1, 2, \ldots, k-1$, 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, $\text{GSP}_{\sigma(i)}$ receives $a_{\sigma(k)}$ and $\text{UB}_{\pi(i)}$ 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.

Example:

Suppose four users $\{U_1, U_2, U_3, U_4\}$ participate in DA bidding the following values {140, 50, 70, 10} for a group of four resources/GSPs. Resources $\{R_1, R_2, R_3, R_4\}$ submit the following asks {75, 40, 30, 15}. According to DA Protocol, EA arranges the bids in decreasing order {140, 70, 50, 10} and asks in increasing order {15, 30, 40, 75} then determines that users $\{U_1, U_2, U_3\}$ and resources $\{R_2, R_3, R_4\}$ trade at the price G$ 42.5.

3 Simulation Environment

To investigate the allocation mechanisms presented in this paper we use the SimGrid simulation framework proposed by Casanova in [9]. The SimGrid toolkit provides tools for developing and evaluating resource allocation algorithms in heterogeneous distributed environments. It facilitates the creation of realistic resource models with different resource configurations where each resource can have varying performance characteristics like workload, data storage capacity or network bandwidth.

In the following we describe the SimGrid implementation of our simulator. The resources characterized by

Resources	R_0-R_3	R_4-R_7	R_8-R_{11}	R_{12}-R_{15}
s_i (MIPS)	500	1000	1500	2000
p_i (G$/sec)	5	10	15	20

Table 1. Resources in the system.

processing rates and reservation prices are created in the system using the functionalities provided for hosts in the SimGrid. The users are created in the system each having some computational tasks to be executed. Computational tasks characterized by execution time, budget and resource preference are created using the functionalities provided for tasks in the SimGrid. A pthread entity is created for each resource which acts as an auctioneer. The implementation of the entity depends on the auction algorithm deployed. The auctioneers conduct multiple rounds of auction at the resources. Once multiple rounds of auction are over at the resource, the tasks scheduled at the resource are executed using the simulation functionality in SimGrid. After the simulation is over, several parameters like (resource profit, resource utilization) are measured.

4 Experimental Results

In this section we investigate by simulation the three proposed auction allocation protocols (FPA, VA and DA) using our simulator. The simulated grid environment consists of 16 resources/GSPs $\{R_0, R_1, \ldots, R_{15}\}$, having four different processing rates and reservation prices as given in Table 1. The processing rates are within the range [500, 2000] which characterizes a real grid environment. The resources with higher processing speeds have higher reservation prices compared to resources with low processing speeds because they can execute more portion of a job in one second thus incurring more cost to them.

We consider ten users $\{U_0, U_1, \ldots, U_9\}$ where each user has some computational tasks which need to be executed on the resources in the system. A total of sixty computational tasks are considered in the system. We assume that the resource preference r_{ij} for task t_{ij} is uniformly distributed over the interval [0, 15], which corresponds to the 16 resources. The budget of each task is uniformly distributed over the interval [45 G$, 630 G$]. The lower limit of tasks' budget interval is given by the product of the lowest computational time of a task and the lowest reservation price of a resource while the upper limit is given by the product of the highest computational time of a task and the highest reservation price of a resource. The execution time t_{ij} of task T_{ij} follows an exponential distribution.

To simulate a real scenario we consider two categories of users. These two categories are characterized by the users' bidding strategy as follows:

(i) Risk Averse Users: The users in this group are likely to raise their bids in the consecutive rounds of an auction. The reason is to win the auction by bidding a higher value than the value bid in the previous round. They are most likely to be in a winner's curse situation which is the situation in which a winner pays more for an item than its value. For our simulation we assume U_0, U_1, U_2, U_3, U_4 to be risk averse.

(ii) Risk Neutral Users: The users in this group bid close to their valuations and are less likely to be in a winner's curse situation. For our simulation we assume U_5, U_6, U_7, U_8, U_9 to be risk neutral.

The bid value b_{ij} of task t_{ij} is a valid bid if it satisfies the following two conditions: (i) $b_{ij} \leq g_{ij}$, the bid should be less than the budget of task T_{ij}; and (ii) $b_{ij} > C_{kij}$ the bid should be higher than the cost incurred by the resource R_k in executing task T_{ij}.

In Table 2 we present the users participating in FPA and VA at each resource and in Table 3 the users participating in DA.

Resource	Users			
R0	U0	U2	U6	U8
R1	U1	U3	U7	U9
R2	U0	U2	U4	U8
R3	U1	U3	U5	U9
R4	U0	U2	U4	U6
R5	U1	U3	U5	U7
R6	U2	U4	U6	U8
R7	U3	U5	U7	U9
R8	U0	U4	U6	U8
R9	U1	U5	U7	U9
R10	U0	U2	U6	U8
R11	U1	U3	U7	U9
R12	U2	U4	U8	
R13	U3	U5	U9	
R14	U0	U4	U6	
R15	U1	U5	U7	

Table 2. Users participating in FPA and VA.

Resources	Round	Users			
R0,R1,R2,R3	1	U0	U1	U2	U3
	2	U6	U7	U8	U9
	3	U2	U3	U4	U5
	4	U8	U9	U0	U1
R4,R5,R6,R7	1	U4	U5	U6	U7
	2	U0	U1	U2	U3
	3	U6	U7	U8	U9
	4	U2	U3	U4	U5
R8,R9,R10,R11	1	U8	U9	U0	U1
	2	U4	U5	U6	U7
	3	U0	U1	U2	U3
	4	U6	U7	U8	U9
R12,R13,R14,R15	1	U2	U3	U4	U5
	2	U8	U9	U0	U1
	3	U4	U5	U6	U7

Table 3. Users participating in DA.

4.1 User Payments

The motivation behind the study of users' payments for the three auction models is to determine which auction protocols favor the users in terms of payments. The *payment* given by a winning user to a resource is defined by the auction protocol. Figure 2 shows the payments handed by the users in the three auction protocols. The results are discussed for the two groups of users.

(i) Risk-Averse Users: User U_2 has the highest payment in FPA compared to all the other users. This is due

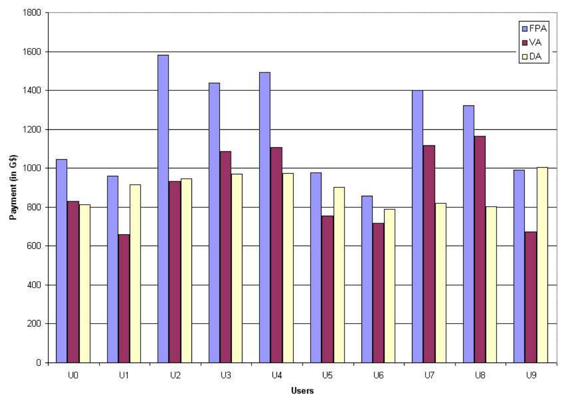

Figure 2. Users Payments

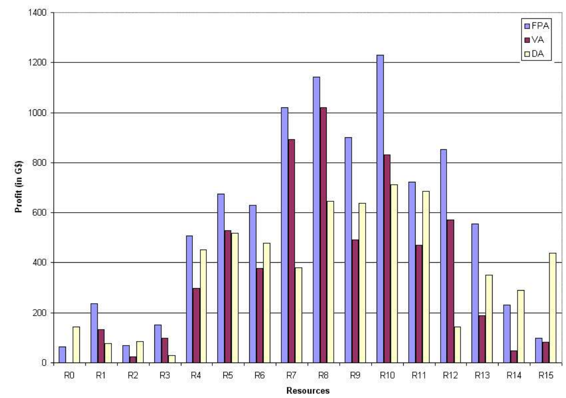

Figure 3. Resource Profits

to the bidding strategy of U_2 (a risk-averse user bids higher than the actual valuation) and the high budgets of U_2's tasks which leads to high bids, further leading to high payment. In case of VA this user has the highest decrease in payment. This is because the bids are close to the valuation price of the risk-neutral competitors. Discussing the case for user U_1, the payment in DA is close to the payment in FPA. This is because its competitors in several rounds of DA are risk-averse users U_0, U_2 and U_3 as shown in Table 3 which bid higher than their valuation but much lower than the bid of U_1. Since the bids of user U_0, U_1, U_2, U_3 are close, the equilibrium price is close to the bid of user U_1. Thus user U_1 pays an amount close to its bid value.

(ii) Risk-Neutral Users: Because of the bidding strategy of risk-neutral users their payments in the three protocols are lower compared to the payments of risk-averse users. In FPA user U_7 has the highest payment among the risk-neutral users because of the high budgets of its tasks and also because its bids are close to these budget values. In DA the payment of U_7 is lower than in FPA because its competitors are risk-neutral users U_6, U_8 and U_9 as shown in Table 2 which bid close to their valuation but less than the bid value of user U_7. The equilibrium price is much lower than the bid of U_7 which leads to low payment in DA. U_9's payment in DA is high since it competes with risk-averse users U_0 and U_1 (see Table 3) who bid values higher than their valuations but lower than the bid value of user U_9. This leads to an equilibrium price close to U_7's bid and so the payment is close to this bid value, which is the amount paid in FPA.

It can be concluded that for a mix of risk-neutral and risk-averse users in the system FPA leads to higher payments than DA and VA. The payment in DA and VA varies depending on the type of the users and on the users' bids.

4.2 Resource Profits

The *resource profit* is defined as the difference between the payment received from the winning users in an auction at one resource and the cost incurred in processing the jobs allocated to it. The resource profits are studied to determine which auction protocol is better from resource's perspective.

Figure 3 shows the profits of the resources in the three auction protocols. Due to space limitation we discuss here only the profits obtained by the resources belonging to Group 1 and Group 2.

(i) Group 1 ($R_0 - R_3$): The profit of R_0 in DA is higher compared to FPA and it is almost zero in VA. This is because users having tasks with low budgets compete for resource R_0 and they bid low values leading to low payment in FPA. In case of VA one of the risk-neutral users, U_6, lowers its bid in one of the rounds and this bid is the second highest bid in VA leading to low payment in VA compared to FPA. In DA in which resource R_0 participates, the risk-averse user U_2 bids a value higher than its valuation but much lower than the bid value of U_0, another risk-averse user. Since there is a big difference between the bid values of user U_2 and U_0 the equilibrium price is much lower than the value bid by user U_2 and higher than the ask of resource R_0. Thus R_0 obtains a high profit. The same observations hold for resource R_2. Resource R_1 has the highest profit in FPA among all the resources in its group. This is because users having low computational cost tasks compete for resource R_1, thus leading to less cost incurred by resource R_1. Moreover, risk-averse users U_1 and U_3 bid higher than their valuations leading to high payment to resource R_1.

Group 2 ($R_4 - R_7$): For each resource in this group, the profit in DA is lower than in FPA. This is because the risk-neutral users U_5, U_6, U_7 who participate in the auction bid values close to their valuation and in one of the rounds of DA for *Group 2* user U_5 bids lower than U_6. The equilibrium price which is calculated as the average of U_5's bid and the corresponding ask of the resource at the mismatch point is close to the bid value of user U_5 (since bids of U_5 and U_6 are close). The ask of the resource is before the mismatch point, thus leading to a low profit for the resource.

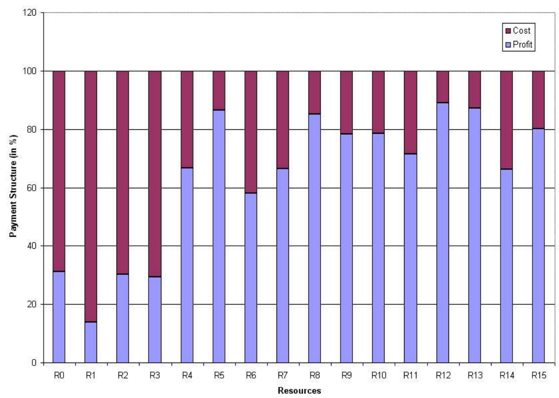

Figure 4. Payment Structure of Resources in DA

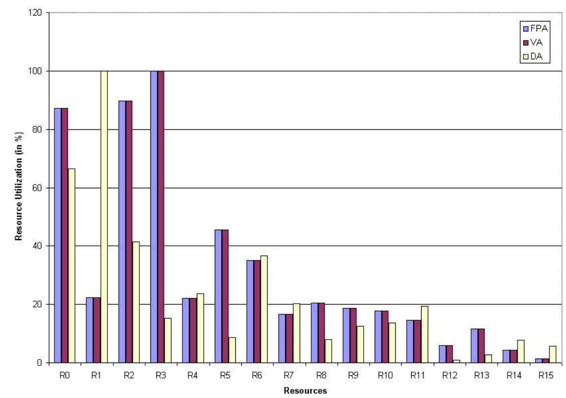

Figure 5. Resource Utilization

Similar situation occurs for U_6 and U_7 in another round where U_6 bids lower than U_7 which further decreases the profit. Resource R_7 has the highest profit in FPA among the resources in the group. This is because users having low cost computational tasks and high budgets compete for resource R_7 leading to low cost incurred by R_7. The bids are close to the valuation price because risk-neutral users U_5, U_7, U_9 participate at R_7. This leads to high payment to resource R_7.

It can be concluded that for a mix of risk-neutral and risk-averse users in the system resources gain more profit when FPA is deployed compared to other two auction protocols, VA and DA.

4.3 Payment Structure

The *payment structure* for each resource is defined as the representation of total cost and profit of the resource as fractions of the total payment received by the resource. Due to space limitations we present only the payment structure of the DA protocol. The results for the other protocols will be presented in an extended version of this paper.

DA: Figure 4 shows the payment structure for each resource in DA. The results are discussed for the first two groups of resources.

(i) Group 1 ($R_0 - R_3$): In this group R_1 has the lowest profit percentage. This is because most of the high computational cost tasks are scheduled and executed at R_1 which increases the cost incurred by R_1. Further in one of the rounds of DA for *Group 1* the participants are risk-neutral users U_6, U_7, U_8, U_9. In that round U_6 bids less than the bid of U_7 and the equilibrium price is close to U_7's bid and to R_0's ask which leads to low profit for R_1. The highest profit percentage is obtained by resource R_0. This is because the participants in one of the rounds of DA for *Group 1* are risk-averse users U_0, U_1, U_2, U_3. In the same round U_1 bids much lower than U_0 but higher than its valuation. The equilibrium price is much lower than U_0's bid and higher than R_0 ask which leads to high profit for R_0.

(ii) Group 2 ($R_4 - R_7$): In this group the profits of resources is higher than the profits of resources in *Group 1*. This is because most of the participants in DA of *Group 2* are risk-averse users U_0, U_1, U_2, U_3. One of the risk-averse users bids lower than the other risk-averse users leading to an equilibrium price lower than the ask of the resource, before the mismatch point. This leads to high profit for resources whose asks are before the mismatch point. On resource R_5 a small number of low cost computational tasks are scheduled leading to low cost. In case of resource R_6 some of the tasks scheduled and executed on it are of high computational cost, leading to high cost incurred by R_6. Similar observations can be made for *Group 3* and *Group 4*.

4.4 Resource Utilization

Resource utilization for a resource (U_i) is defined as the ratio of the total execution time ($T_{exec,i}$) at the resource to the total simulation time (T_{sim}): $U_i = \frac{T_{exec,i}}{T_{sim}}$. Figure 5 shows the resource utilization for all resources in the three auction protocols.

FPA: Resource R_3 has the highest utilization of 100%. This is because users participating in FPA at R_3 have high computational cost tasks leading to high utilization. Resources R_0 and R_2 have utilization of 80%-90%. This is because some of the users participating in auctions at R_0 and R_2 have high computational cost tasks. Thus increasing the utilization of R_0 and R_2. Resource R_1 has 20%-25% utilization because most of the tasks scheduled at R_1 have low computational cost. Resources $R_4, R_7, R_8, R_9, R_{10}$ and R_{11} have utilization of 15%-25% because they execute approximately the same number of average and low computational cost tasks. Resources R_5 and R_6 execute approximately the same number of high and low computational tasks, so the utilization is in the range 35%-45%. Resources R_{12}, R_{13}, R_{14} and R_{15} have low utilization in the range of 3%-10% because few low cost tasks are scheduled on these resources.

VA: Since tasks assignment in VA and FPA is the same, the resource utilization of resources in VA is the same as in FPA.

DA: The maximum utilization is at the resources in *Group 1*. This is because users participating in DA of *Group 1* have high computational cost tasks. Further in *Group 1* resource R_1 has the maximum utilization of 100% because most of the tasks of users participating in auction of *Group 1* are scheduled at R_1, thus increasing the utilization of R_1. In *Group 2* utilization of R_4, R_6 and R_7 is higher than in FPA because the same number of high, average and low computational cost tasks are distributed among these resources in DA while in the case of FPA the number of high, average and low computational cost tasks for each resource is not the same. Considering *Group 3*, resources R_8, R_9, R_{10} have low utilization of 5%-15%. This is because low computational cost tasks are scheduled and executed on them. Resource R_{11} has an equal number of high and low computational cost tasks due to which its utilization (20%) is highest in the group. In *Group 4* utilization of R_{12} and R_{13} is low because a small number of low computational cost tasks are scheduled on them, thus decreasing the utilization. Resource R_{14} and R_{15} have equal number of average and low cost computational tasks leading to a better utilization compared to the other resources in *Group 4*. In most of the cases the maximum utilization of resources is obtained in case of FPA and VA.

5 Conclusion

In this paper we presented the auction allocation model, and three auction-based protocols: *First-Price Auction*, *Vickrey Auction*, and *Double Auction*. We simulated these three protocols using the SimGrid simulation framework and studied them in terms of economic efficiency and system performance. The results showed that when we consider a mix of risk-averse and risk-neutral users the *First-Price Auction* is better from resource's perspective while the *Vickrey Auction* is better from user's perspective. The third type of auction, *Double Auction*, favors both resources and users.

References

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

[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] J. Nabrzyski, J. M. Schopf, and J. Weglarz. *Grid Resource Management, State of the Art and Future Trends*. Kluwer Academic Publishers, Boston, MA, 2003.

[4] R. Wolski, J. S. Plank, T. Bryan, and J. Brevik. G-commerce: market formulations controlling resource allocation on the computational grid. In *Proc. of the 15th IEEE International Parallel and Distributed Processing Symposium*, April 2001.

[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] 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.

[7] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16(1):8–37, March 1961.

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

[9] H. Casanova. Simgrid: A toolkit for the simulation of application scheduling. In *Proc. of the IEEE/ACM International Symposium on Cluster Computing and the Grid*, pages 430–437, May 2002.

[10] 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.

[11] C. Chen, M. Maheswaran, and M. Toulouse. Supporting co-allocation in an auctioning-based resource allocator for grid systems. In *Proc. of the 11th IEEE Heterogeneous Computing Workshop*, pages 89–96, April 2002.

[12] 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.

[13] D. Ferguson, Y. Yemini, and C. Nikolaou. Microeconomic algorithms for load balancing in distributed systems. In *Proc. of the 8-th IEEE Intl. Conf. on Distr. Comp. Systems*, pages 491–499, 1988.

[14] R. Gagliano, M. Fraser, and M. Schaefer. Auction allocation of computing resources. *Communications of the ACM*, 38(6):88–100, June 1995.

[15] 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.

[16] C. A. Waldspurger, T. Hogg, B. A. Huberman, J. O. Kephart, and W. S. Stornetta. Spawn: A distributed computational economy. *IEEE Trans. Software Eng.*, 18(2):103–117, February 1992.

[17] R. Buyya and S. Vazhkudai. Compute Power Market: Towards a market-oriented grid. In *Proc. of the 1st IEEE/ACM Symposium on Cluster Computing and the Grid*, pages 574 – 581, May 2001.

[18] I. Sutherland. A futures market in computer time. *Communications of the ACM*, 11(6):449–451, June 1968.

[19] R. Buyya and M. Murshed. Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. *Concurrency and Computation: Practice and Experience*, 14(13-15):1175–1220, November 2002.