A Comparative Analysis of Single-Unit Vickrey Auctions and Commodity Markets for Realizing Grid Economies with Dynamic Pricing

Kurt Vanmechelen and Jan Broeckhove

University of Antwerp, BE-2020 Antwerp, Belgium
kurt.vanmechelen@ua.ac.be

Abstract. The introduction of market principles is a promising approach for dealing with the complex issues that arise in Grid resource management. A key aim is to align the resource consumption and provisioning patterns of Grid participants through proper incentive mechanisms. An important research question in this regard is the choice of a market organization. A number of such organizations have been proposed to support an economically inspired form of Grid resource management. This paper presents a comparative, quantitative, analysis of the single-unit Vickrey auctions and commodity market organizations with regards to price stability, fairness, and communicative and computational requirements. Our analysis based on simulated market scenarios shows that both market organizations lead to similar outcomes but that a commodity market organization leads to more stable market behavior at the cost of higher communicative requirements.

Keywords: Commodity Markets, Vickrey Auctions, Grid Economics, Resource Management, Grids.

1 Introduction

Traditional resource management systems adopt a system-centric form of resource management where a scheduling component establishes a mapping from jobs to Grid resources. This mapping is based on system oriented metrics such as infrastructure utilization or throughput. To generate broad support for Grids, but also to develop usage models that are more attuned to the user's needs, it is important that this emphasis shifts to a more user-centric approach. As such, the focus should be on allocation algorithms that are driven by the user's valuation of their results. In this way, Grids will deliver the maximum utility to the individual user. Because of their strategic and selfish nature, one cannot expect users to accurately formulate their true valuations to the resource management system unless proper incentive mechanisms are installed.

A promising approach towards dealing with this issue, involves the use of an economics based resource manager [1] which takes resource utilization cost

D.J. Veit and J. Altmann (Eds.): GECON 2007, LNCS 4685, pp. 98–111, 2007.

into consideration and requires users to back their valuations with associated credits, of which they have limited supply. Such an economics based trading model, where consumers rent resources from providers, is an attractive method to manage resource allocation in Grid systems. Aside from applications within the Grid domain [2,3,4,5,6,7,8,9,10,11], (consult [1] for an overview), economic models for resource sharing have also been applied to agent systems [12,13], telecommunication networks [14] and to databases [15] and data mining [16].

One of the most important research questions in adopting economic principles for Grid resource management is the choice of a market organization. Multiple such organizations exist in economic literature and at present, it is unclear which organization is most suitable to support an economically inspired form of Grid resource management. From a usage model point of view it is fairly clear that adopting *combinatorial auctions* [17], in which a participant can submit a single bid for a combination of goods, is one of the most attractive organizations. It enables consumers to accurately define their valuations for specific collections of Grid resources that are required by their applications. As such, it allows for expressing valuations that are conditional on the *coallocation* of a set of Grid resources. This eliminates the *exposure problem* [18] users face when they need to participate in multiple auctions for acquiring the constituent parts of an allocation bundle. However, this approach suffers from high computational complexity which can mostly be attributed to the NP-completeness of determining the optimal set of winners in such an auction [19]. In addition, the lower bounds on the communicative complexity of the value elicitation process in combinatorial auctions also inhibit their applicability for large scale economies, certainly in the case of general bidder valuations and when aiming for exact efficiency [20].

In this contribution we analyse the performance of two market organizations for realizing Grid economies; single-unit Vickrey auctions and commodity markets. The price formation process is fundamentally different in both organizations. In the commodity market one takes the approach of performing global optimization for establishing an equilibrium price, by polling all market participants for their supply and demand levels at a particular price. Participants are required to communicate these supply and demand levels to a central process performing the optimization, also called the *Walrasian Auctioneer*. The auction market organization, on the other hand, is fully decentralized and lets prices emerge from the local interactions of the market participants in single-unit Vickrey auctions. The goal of this contribution is to investigate whether these two approaches lead to different outcomes in terms of established prices, fairness of allocations and communicative and computational requirements for establishing these allocations.

Limited work has been done on directly comparing both systems on these grounds. The study in [9] compares both organizations on price stability and infrastructural utilization. The authors postulate that “auctioneering is attractive from an implementation point of view but that it does not produce stable pricing or market equilibrium, and that a commodity market performs better from the standpoint of a Grid as a whole". Similar remarks on the stability of prices in a single-unit Vickrey auction market are made in [5].

2 Market Model

For the purpose of this study we resort to simulation for efficiently analysing both market organizations on a large scale. Therefore, modeling decisions have to be made concerning the type of Grid resources that are simulated and the behavior of the market's participants. The model adopted here is similar to the one described in [11].

2.1 Resources

A complete and accurate Grid resource model should include a large set of different resource types, each with their own specific attributes. Examples include CPU time, scratch and permanent storage, network bandwidth, main memory, and more specialized resource types such as specific hardware components. Aside from taking a decision on the scope of the simulated resources, a second important design choice concerns the extent to and manner in which these resource types are introduced as tradeable goods into the market. Fully exposing all resource types and attributes allows for a very accurate valuation of resources by Grid users. A downside to this approach is the resulting increase in the complexity of the market's pricing mechanism, the interactions between the market and its participants, and the participants' valuation logic. In this contribution, we take the approach of restricting our resource model to CPU resources and to consider these the single type of resources that are tradeable in the resource market.

Commodity Market. In order to introduce diversification related to CPU performance, we introduce different commodity categories for multiple classes of CPUs with respect to their performance (in terms of GFlops/s). Each category is characterized by a *performance ratio* which expresses the performance increase of using a CPU from a particular category compared to a CPU from the lowest category. These categories constitute substitutable commodities, as jobs can execute on both, although they will be valued differently by consumers. The term *resource category* refers to a partition within a resource type based on a specific resource attribute, e.g. performance. Resources belonging to the same type but to a different category are substitutable. In this contribution we consider only one type and three categories.

Auction Market. In the auction market, all CPU resources will be individually auctioned. This allows for a more accurate valuation by consumers and is an advantage over the more abstract resource model used in the commodity market. Nevertheless, we will adopt the same single-attribute characterization from the commodity market in the form of the performance ratio, in order to keep results comparable.

2.2 Consumers

Each consumer has a queue of CPU-bound computational jobs that need to be executed and for which resources must be acquired from providers through participation in the market. The dispatch of a job to the CPU is effected immediately after the necessary resource has been acquired. Every job has a nominal running time T, i.e. the time it takes to finish the job on a reference CPU. However, in our spending algorithms we do not assume that the consumer has knowledge of this running time.

Every consumer is provided with a budgetary endowment that is periodically replenished. The period for this replenishment is denoted by an *allowance period*. We do not assume a particular funding source for the consumers. In practice, funding rates could be determined by system administrators, could be set by consumers themselves through monetary payment, or could result from a feedback loop that redistributes the credits earned by the providers of a particular (virtual) organization to the users of that organization. In every simulation step, consumers are charged with the usage rate prices for all Grid resources that are currently allocated to their jobs. Consumers do not attempt to save up credits, but try to use their entire budget. However, expenditures are spread evenly across the allowance period because we assume that consumers do not have reliable estimates for the running times of their jobs. Therefore, we need to prevent them from agreeing to a price, a "cost" level, that would not be sustainable for them over the entire allowance period.

Commodity Market. In the commodity market, consumers have to decide on the demand level they are willing to express, given a price vector P suggested by the market. The components of that vector P_i represent the price per resource unit, per time unit of the i^{th} commodity category that is characterized by $PerformanceRatio_i$. Depending on the job mix a consumer has to schedule, certain resource categories will be preferred over others. This is expressed through the $ValuationFactor_i$ term. This leads to an adjusted price for each category, given by

$$AdjustedPrice_i = (P_i/PerformanceRatio_i)/ValuationFactor_i \tag{1}$$

The r.h.s reflects the price normalized to unit performance and factors in the valuation. The consumer expresses demand, limited by the current allowed rate of expenditure, in the category with the lowest adjusted price.

The use of the $ValuationFactor_i$ term in the adjusted price is a simple abstraction for the complex logic a consumer might follow to prefer one CPU category over another. An example of such a logic whereby a consumer is willing to pay more than double the price for a CPU of category 1, which is only twice as fast as one of category 2, is the following. Suppose the consumer has a job graph that includes a critical path and that the user adopts a spending strategy for optimizing total turnaround time. Such a consumer would be willing to pay more than the nominal worth of a CPU of category 2 for allocating jobs on the critical path, as they have a potentially large effect on turnaround time.

Auction Market. For the auction market, consumers have to decide on the amount of credits they are prepared to bid for a particular CPU, the amount of CPUs they will bid on, and the specific auctions they will participate in. They calculate a base level for their bids which depends on the remaining budget and adjust this bid with the characteristics of the CPU resource:

$$Bid_{cpu} = (Base\_Bid * PerformanceRatio_{cpu}) * ValuationFactor_{cpu} \qquad (2)$$

The calculation for the base bid level also takes into account a target *parallellisation degree* the consumer wishes to realize. At the start of the simulation, all consumers try to launch all of their available jobs in parallel and determine their bids accordingly. As trading progresses, consumers gradually learn the level of parallellisation that they are able to achieve, given their budgetary limits, and they adjust their expectations and base level bids accordingly. Consumers adopt the simple heuristic of participating in the auctions which currently host the lowest number of bidders.

2.3 Providers

Every provider hosts a number of CPUs that can be supplied to the computational market. Once a resource is allocated to a job, it remains allocated until the job completes. The market price at the time of resource acquisition will be charged as a fixed rate to the consumer for the duration of the job. This approach is consistent with the fact that we do not assume a prior knowledge of a job's running time. An alternative to a fixed rate is to allow a variability in the charged rate based on the market price evolution. Another option is to allow variability on the performance a consumer receives for a given rate over the job's execution period, an approach adopted in [3]. These alternatives allow for potentially faster reallocation of resources according to the dynamic market environment, but make budgetary planning and resource usage planning more difficult for consumers.

For the analysis presented in this contribution, providers will not set minimum prices for their resources and will supply all of their available resources to the market.

2.4 Market Pricing

In the commodity market, prices for the different CPU categories are dynamically set in every simulation step by an optimizer which adjusts the price in order to bring the market to equilibrium. The optimizer iteratively polls all market participants for their supply and demand levels for each CPU category. This information is used to define an excess demand surface i.e. the difference between current demand and supply as a function of the price vector. An example of such an excess demand surface for a commodity market with two substitutable CPU categories is shown in figure 1. Note that we use the Euclidian norm of the excess demand vector.

The market equilibrium point is the zero of this surface and fixes the price at which the market will trade at that point in time. The global zero search algorithm is a combination of the algorithm presented in [11], which is an adaptation of Smale's algorithm [21], and a pattern search algorithm [22] of which we use the implementation provided by Matlab.

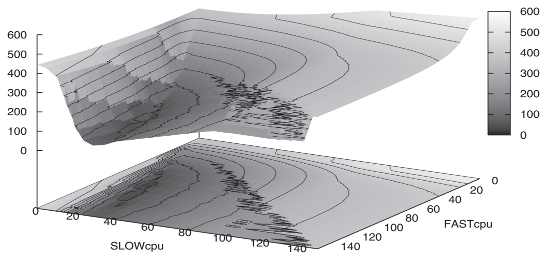

Fig. 1. Sample excess demand surface

In the auction market, each provider hosts a number of single-unit Vickrey auctions, one for each CPU that is available at that point in time. Consumers submit their sealed bids to the auctioneers of the CPUs they are interested in. The Vickrey auction allocates the CPU to the consumer with the highest bid, at the transaction price of the second highest bid (or zero if there is only one bidder). The fact that the consumer's transaction price does not depend on its own bid forms the basis for the *incentive compatibility* of the Vickrey auction. This means that a consumer has no incentive to place a bid which differs from its true value for the CPU, because no strategic advantage can be gained from this act.

3 Simulated Market Environment

We resort to a simulated market environment for analysing the commodity and auction market organizations. For this we use GES (Grid Economics Simulator) [23], a Java based discrete event simulator that we developed to support research into different market organizations for economic Grid resource management. The simulator supports both non-economic and economic forms of resource management and allows for efficient comparative analysis of different resource management systems. We currently have built-in support for commodity markets, different forms of auctions (English, Dutch, Vickrey, combinatorial and double auctions), fixed pricing as in [24], and implementations of other market mechanisms such as the proportional sharing approach found in Tycoon [3]. Non-economic resource management is supported through FIFO, round robin, and priority schedulers. The simulator is equipped with a user interface for supporting efficient analysis and configuration of market scenarios. A persistency framework allows for storing both scenario configurations and configurations of the UI layout. A screenshot of the UI is shown in figure 2.

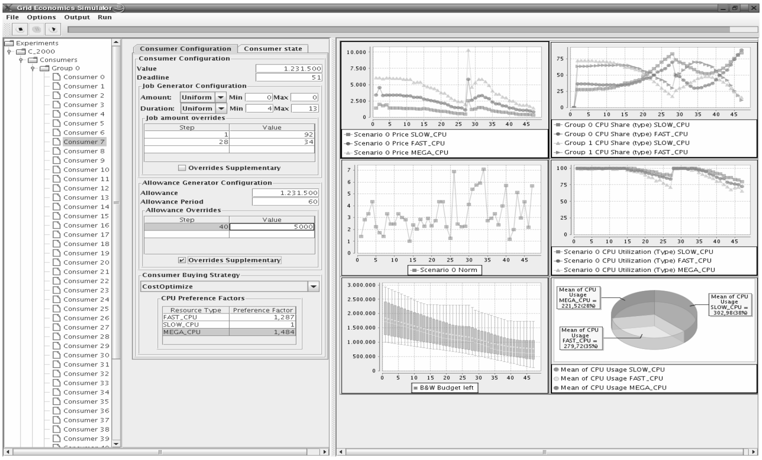

Fig. 2. Screenshot of the GES UI

The parameters of the scenario that we will use as the basis for our analysis are shown in table 1. For parameters that are specified with a range, we draw values from a uniform random distribution. Three groups of consumers with different budget levels are created by multiplying a consumer's base allowance with the respective *allowance factor* AF_i of its group. Note that in the context of this analysis, we keep the number of jobs in the consumer queues constant at the initial level by reinjecting a new job in the consumer's queue for every finished job. This results in a stable demand level which should lead to stable market prices.

Table 1. Simulation parameters

Parameter	Value
Number of consumers	100
Number of providers	50
Number of fastCPUs per provider	$\{1, 2, \cdots, 7\}$
Number of mediumCPUs per provider	$\{3, 4, \cdots, 11\}$
Number of slowCPUs per provider	$\{9, 10, \cdots, 17\}$
Performance ratio of fast vs slow	3.0
Performance ratio of medium vs slow	2.0
Valuation factors	[1.0,1.5]
Job running time in time steps	$\{4, 5, \cdots, 8\}$
Number of jobs per consumer (constant)	$\{150, 151, \cdots, 500\}$
Base Allowance	100,0000 * [1.0,1.5]
$\{AF_1, AF_2, AF_3\}$	{1.0, 2.0, 3.0}
Allowance period in time steps	800

4 Comparative Analysis

4.1 Dynamic Pricing

As shown in figure 3, the average prices paid by the consumers in the market for the different categories of resources are similar. The auction market shows a higher fluctuation in the price levels over the course of the simulation with a relative standard deviation [25] of 5.86% versus 1.62% for the commodity market. The deviation for the auction market prices does not include the instable price levels of the first 10 steps, if these are included the deviation increases to 9.62%. Whereas the commodity market immediately brings the market to equilibrium through global optimization, the participants in the auctions still have to optimize their *target parallellisation degree* and discover the amount of resources to bid for. This results in the extensive adjustments of the average CPU price paid at the beginning of the simulation.

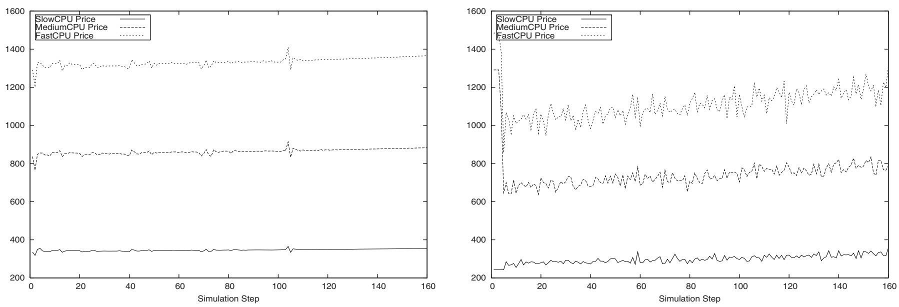

Fig. 3. CPU Prices for the commodity market (left) and auction market (right)

Although prices are slightly less stable in the auction market, they do follow the trend of supply and demand in the market as shown in figure 4. This scenario is similar to the one used in [9]. Periods of overdemand are followed by periods of underdemand through the injection of a set of jobs into the system at intervals of 45 simulation steps. In addition, jobs are not reinjected in the consumer queues on completion. Whereas the results in [9] indicate that such a scenario leads to very erratic pricing behavior for the auction market, showing price levels that do not reflect overall market supply and demand, the results are very different here. Although some parameters of the simulation differ, one of those being the fact that consumers have to coallocate disk and CPU resources in [9], our results do show that it is possible, using fairly simple bidding logic, to obtain meaningful and fairly stable average prices in the auction market. We note that the two price peaks for the slow CPU category in the commodity market scenario are caused by the fact that no slow CPU resources are available for trade at those time instances. The equilibrium optimizer generates a high price level for these

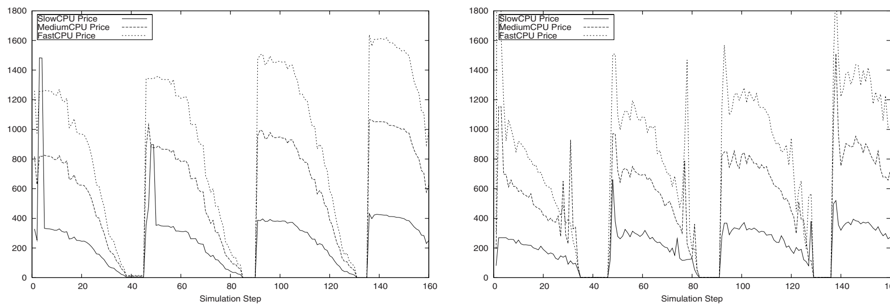

Fig. 4. CPU Prices in the varying load scenario for the commodity market (left) and auction market (right)

resources in order to remove all demand for them in the market and minimize excess demand.

We note that the average transaction price for resources is lower in the auction market. The difference in total revenue generated for the providers is 15.79%.

4.2 Fairness

The fairness of the allocations in an economic resource management system, denotes whether the level of budgetary endowment of a consumer is correctly translated into a corresponding share of the infrastructure allocated to that consumer. The graphs in figure 5 show that in both markets the average infrastructural shares of the three consumer groups converge to the budget shares of those groups. In the auction market, correspondence is not achieved in the first simulation steps. This can be explained by the fact that consumers are still learning the parallellisation degree that is sustainable by them in the current market situation. The commodity market does not require such judgment from its consumers and immediately brings about fair allocations. To investigate whether shares quickly adapt to sudden changes in the market, we swap the budget levels of the different consumer groups at step 80. As shown in figure 6, both market organizations are able to quickly adapt the allocations to reflect the new market situation. Instead of the cumulative average, figure 6 shows the instantanious shares at each simulation step, which are somewhat less stable for the auction market scenario.

From the providers' point of view, a fair operation of the market should lead to similar revenues among the different providers for the CPU resources sold. In the commodity market, we measured an average relative standard deviation of 2.09% on the nominal transaction price paid per CPU cycle in a single time step. This price is obtained by dividing the earned revenue on a set of CPUs by the performance factors of these CPUs. The origin of this deviation lies in the valuation factors consumers have towards different CPU categories and the differences in the number of CPUs each provider has of a particular category.

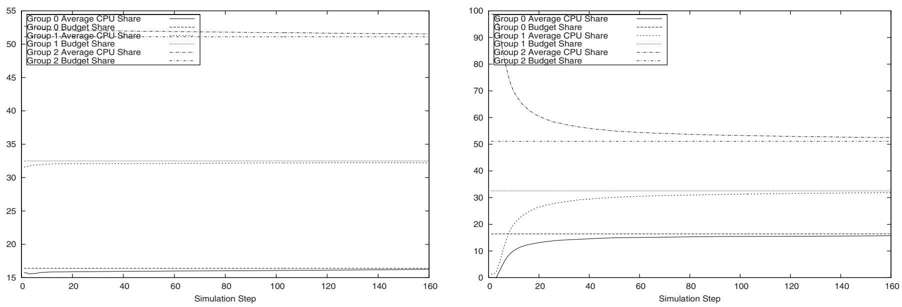

Fig. 5. Budget and infrastructural shares as a cumulative average for the commodity market (left) and auction market (right)

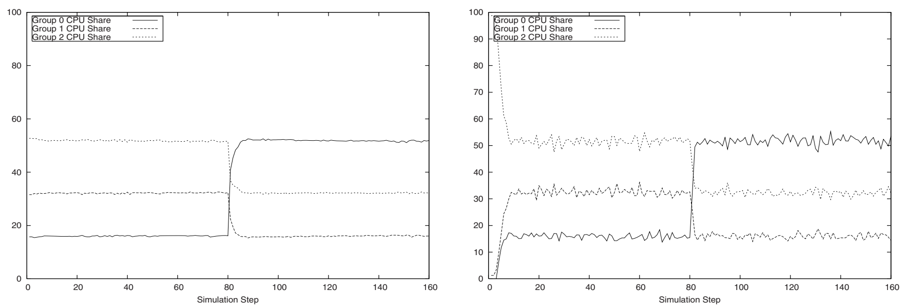

Fig. 6. Fairness under sudden budget change for the commodity market (left) and auction market (right)

In the auction market, prices emerge from the local interactions among the bidders and this results in less stable revenue streams for the providers. The average deviation was 6.67% in the auction market. Although a greater variance in revenue was observed on the transactions made at a single step of the simulation, the deviation on the total nominal revenue accrued by the providers at the end of the simulation was only 2.33%. For the commodity market this relative standard deviation was 2.06%.

In the less stable peaked demand scenario, the differences in revenue stability between the two market organizations increase. For the auction market, we measured a deviation of 43.74% on transaction prices for a single step and a deviation of 6.79% on the total accrued revenue. The respective deviations for the commodity market under the peak demand scenario were 4.03% and 2.31%.

4.3 Computational and Communicative Requirements

The number of resource categories introduced in the commodity market is a determining factor for its communication and computational requirements. This can be attributed to the fact that each additional CPU category increases the dimensionality of the equilibrium price optimization problem. Tables 2 and 3 show the effect of introducing more categories for the constant load and peaked load scenarios. The number of messages and running time are given per simulated time step, as well as the median of the excess demand norm over all simulation steps. The tables also include the number of network messages sent in the auction market for comparison.

Table 2. Market performance with respect to the number of CPU categories in the market (constant load scenario)

CPU categories	CM Messages	Running Time (ms)	Median norm	Auction Messages
1	4440	80	1.0	2996
2	6276	314	1.41	2541
4	47033	1187	5.48	1962
6	90812	2355	6.86	1675
8	157608	4539	8.94	1625

Table 3. Market performance with respect to the number of CPU categories in the market (peaked load scenario)

CPU categories	CM Messages	Running Time (ms)	Median norm	Auction Messages
1	3482	50	1.0	1646
2	33257	314	3.16	1576
4	93177	1084	7.55	1359
6	163750	2629	12.79	1312
8	261181	4808	72.02	1250

Introducing more categories allows the market participants to express their valuations for the different levels of CPU performance in a more fine grained manner. As a result, resource allocations will be better adapted to the real needs of the users. Although the computational costs do not inhibit a practical deployment of this market organization (they would allow for adapting the price within a timeframe of 5 seconds for the case of 8 CPU categories), the communicative requirements of the price optimization process might. This is especially true for large scale, wide-area deployments with higher communication delays, lower network bandwidth and higher network usage costs. Nevertheless, the communication burden can be reduced significantly through a dynamic deployment of the consumer bidding logic into the local environment of the price optimization process. In a Java based environment, this can be realized by allowing market participants to send an object representing their *bidding agent* to the JVM of the equilibrium optimizer when new prices are to be formed. The agent then reports the participant's supply or demand levels to a local component which aggregates these levels and passes them on to the optimizer through local method calls. Java's support for dynamic classloading allows the agent's code to be automatically downloaded when needed. This model has already been validated through a real-world deployment of the commodity market logic using the CoBRA framework [26]. Organizing multiple aggregators in a tree structure can further address the scalability issues of the commodity market. We also note that the median of the norm, which is a measure for the excess demand after commodity market prices are set [10], increases as we introduce more categories.

The amount of network messages sent in the auction market organization is significantly lower, especially for the scenarios with a higher number of categories. It diminishes as we introduce more categories because we keep the total processing capacity of the Grid constant, while introducing more types with higher performance factors. This leads to a lower amount of discrete resources that need to be auctioned. Note that an auction based framework which uses English auctions for example, can lead to significantly higher communication costs as a result of iterative overbidding in such auctions. Because of their strategy-proofness, single-unit Vickrey auctions require only one round of bidding, resulting in the minimum amount of communication necessary to establish a trade. Another important factor for the lower amount of communication is the fact that a consumer only participates in a limited number of auctions (according to its target parallellisation degree). On average, each auction attracted approximately five consumers in the simulated scenarios.

5 Conclusion

Both Vickrey auctions and commodity markets have been proposed as market organizations for establishing Grid resource management systems that are based on economic principles. In order to guide system designers in their choice for a particular organization, we have presented a comparative analysis of both options on the grounds of price stability, fairness and communicative and computational requirements. The commodity market organization results in a more stable environment with respect to prices and allocative shares. The main disadvantages of this organization are its limited support for fine-grained valuations because of the high communication costs when defining a large number of resource categories, and its centralized nature. The Vickrey auction organization leads to similar but less stable outcomes and supports fine-grained valuations at significantly lower communicative requirements.

References

1. Buyya, R., Abranson, D., Venugopal, S.: The Grid Economy. In: Proceedings of the IEEE, vol. 93(3), pp. 698–714 (2005)
2. Abramson, D., Buyya, R., Giddy, J.: A computational economy for Grid computing and its implementation in the nimrod-g resource broker. Future Generation Computer Systems 18(8), 1061–1074 (2002)
3. Feldman, M., Lai, K., Zhang, L.: A price-anticipating resource allocation mechanism for distributed shared clusters. In: EC '05: Proceedings of the 6th ACM conference on Electronic commerce, pp. 127–136 (2005)

4. AuYoung, A., Chun, B.N., Snoeren, A.C., Vahdat, A.: Resource allocation in federated distributed computing infrastructures. In: Proceedings of the 1st Workshop on Operating System and Architectural Support for the On-demand IT InfraStructure (2004)
5. Waldspurger, C.A., Hogg, T., Huberman, B.A., Kephart, J.O., Stornetta, W.S.: Spawn: A distributed computational economy. IEEE Trans. Softw. Eng. 18(2), 103–117 (1992)
6. Gomoluch, J., Schroeder, M.: Market-based resource allocation for Grid computing: A model and simulation. In: Int'l. Middleware Conference, Workshop Proceedings, pp. 211–218 (2003)
7. Regev, O., Nisan, N.: The POPCORN Market – an Online Market for Computational Resources. In: Proceedings of the 1st International Conference on Information and Computation Economies, pp. 148–157 (1998)
8. Chun, B.N., Culler, D.E.: User-centric performance analysis of market-based cluster batch schedulers. In: Proceedings of the 2nd IEEE/ACM Int'l. Symposium on Cluster Computing and the Grid, pp. 22–30 (2002)
9. Wolski, R., Brevik, J., Plank, J.S., Bryan, T.: Grid Resource Allocation and Control Using Computational Economics. In: Grid Computing: Making the Global Infrastructure a Reality, pp. 747–772. Wiley and Sons, Chichester (2003)
10. Vanmechelen, K., Stuer, G., Broeckhove, J.: Pricing Substitutable Grid Resources using Commodity Market Models. In: Proceedings of the 3th International Workshop on Grid Economics and Business Models, pp. 103–112. World Scientific, Singapore (2006)
11. Stuer, G., Vanmechelen, K., Broeckhove, J.: A Commodity Market Algorithm for Pricing Substitutable Grid Resources. Fut. Gen. Comp. Sys. 23(5), 688–701 (2007)
12. Bredin, J., Maheswaran, R.T., Çagri Imer, T., Başar, D.K, Rus, D.: Computational markets to regulate mobile-agent systems. Autonomous Agents and Multi-Agent Systems 6(3), 235–263 (2003)
13. Dash, R., Parkes, D., Jennings, N.: Computational Mechanism Design: A Call to Arms. IEEE Intelligent Systems 18(6), 40–47 (2003)
14. Gibney, M.A., Jennings, N.R., Vriend, N.J., Griffiths, J.-M.: Market-based call routing in telecommunications networks using adaptive pricing and real bidding. In: Proceedings of the 3th Int'l. Workshop on Intelligent Agents for Telecommunication Applications, pp. 46–61 (1999)
15. Stonebraker, M., Aoki, P.M., Litwin, W., Pfeffer, A., Sah, A., Sidell, J.: Mariposa: A wide-area distributed database system. VLDB J. 5(1), 48–63 (1996)
16. Joita, L., Rana, O., Freitag, F., Chao, I., Chacin, P., Navarro, L., Ardaiz, O.: A catallactic market for data mining services. Future Generation Computer Systems 23(1), 146–153 (2007)
17. Cramton, P., Shoham, Y., Steinberg, R. (eds.): Combinatorial Auctions. MIT Press, Cambridge (2006)
18. Bykowsky, M., Cull, R., Ledyard, J.O.: Mutually Destructive Bidding: The FCC Auction Design Problem. Journal of Regulatory Economics 17(3), 205–228 (1995)
19. Lehmann, D., Mller, R., Sandholm, T.: The Winner Determination Problem. In: Cramton, P., Shoham, Y., Steinberg, R. (eds.) Combinatorial Auctions, pp. 297–317. MIT Press, Cambridge (2006)
20. Lehmann, D., Mller, R., Sandholm, T.: The Communication Requirements of Combinatorial Allocation Problems. In: Cramton, P., Shoham, Y., Steinberg, R. (eds.) Combinatorial Auctions, pp. 265–294. MIT Press, Cambridge (2006)
21. Smale, S.: A Convergent process of price adjustment and global newton methods. Journal of Mathematical Economics 3(2), 107–120 (1976)

22. Lewis, R., Torczon, V.: A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds. SIAM Journal on Optimization 12(4), 1075–1089 (2002)
23. Vanmechelen, K., Stuer, G., Broeckhove, J.: Grid Economics Simulator (GES), CoMP Group, University of Antwerp, available from the authors on request (2007)
24. Buyya, R.: Economic-based Distributed Resource Management and Scheduling for Grid Computing, Ph.D. dissertation, Monash University, Australia (2002)
25. Snedecor, G., Cochran, W.G.: Statistical Methods, 6th edn. pp. 62–64. The Iowa State University Press, Ames, IA (1967)
26. Hellinckx, P., Stuer, G., Hendrickx, W., Arickx, F., Broeckhove, J.: User Driven Grid Research, the CoBRA Grid. In: Proceedings of the International Workshop on Grid Testbeds, colocated with CCGrid06, Singapore, pp. 1–8 (May 16-19, 2006)