A STRATEGYPROOF AUCTION MECHANISM FOR GRID SCHEDULING WITH SELFISH ENTITIES

V. Hastagiri Prakash
Indian Institute of Science
Bangalore - 560012, India

Y. Narahari
Indian Institute of Science
Bangalore - 560012, India

Keywords: Grid computing, Grid scheduling, Resource management, procurement auctions, parallel flow, selfishness, rationality, incentives, VCG Mechanisms.

Abstract: With major advances in computing technology and network performance, grid computing is strategically placed to become the future of enterprise and even personal computing. One of the most important issues concerned with grid computing is that of application scheduling. The type of scheduling algorithm used will depend on the type of the application. In a global grid setting, the individual users must be provided with an incentive to offer their resources. The situation becomes non-trivial because of the fact that these entities are intelligent, rational and selfish resource providers who, for strategic reasons, may not provide truthful information about their processing power and cost structure. In this scenario, apart from optimality of the algorithm used, strategy-proofness of the underlying mechanism becomes important. This paper presents a strategyproof mechanism based scheduling algorithm for parallel flow type applications in the form of a reverse auction.

1 INTRODUCTION

Started as a new computing infrastructure, grid computing is now becoming a mainstream technology for large scale resource sharing and system integration. With the power and presence of the grid computing systems poised to increase with time, fast, efficient and truthful mechanisms are needed to exploit the power of the grid to the fullest. For a detailed introduction to grid computing, see (Foster and Kesselman, 2002). Various design and operational issues exist in this paradigm of computing, chief among them being scheduling and resource management. For a detailed survey on grid scheduling systems and issues, the reader is referred to (Zhu, 2003). Due to the distibuted, heterogeneous and selfish nature of the participating entities, market based models have long been advocated as a panacea to the issue of grid scheduling, examples of which are (Buyya, 2002), (Wolski et al., 2002). The participating resource providers and consumers, being selfish, rational and intelligent, will attempt at tampering with the existing mechanisms as much as the rules allow them in order to optimize their profit or execution time. This scenario, though possibly profitable for the individuals, tends to bring down the overall efficiency of the grid system as a whole. In order to address this issue, we need to develop robust mechanisms which ensure efficiency in the presence of selfish and rational agents.

1.1 Motivation

Consider an enterprise with a number of branches spread geographically. Now the central administration needs to use a grid system to process data, collected from each of these branches, by a fixed deadline. Though there might be dependencies between the sub-jobs from a single branch, there lie no dependencies between jobs from different branches. So, the central authority can schedule these equal jobs independently. The authority would like to schedule the jobs on different sites on the grid for quick execution. But, when the job is actually given out, the sites may not disclose the true details of their specifications thereby leading to a sub optimal allocation both in terms of cost and time.

In the grid scenario, we often have jobs which can be split into a number of sub-jobs each of them being almost equivalent in terms of their resource requirement and the expected time of completion. This slight variation from the parameter sweep type scheduling gives the problem a new structure and helps us model

Hastagiri Prakash V. and Narahari Y. (2006).
A STRATEGYPROOF AUCTION MECHANISM FOR GRID SCHEDULING WITH SELFISH ENTITIES.
In *Proceedings of WEBIST 2006 - Second International Conference on Web Information Systems and Technologies - Internet Technology / Web Interface and Applications*, pages 178-183
DOI: 10.5220/0001255301780183

it uniquely. To the best of our knowledge, this type of application specific economic scheduling mechanism is not proposed anywhere in literature. The setting and solution for the problem inherently invokes game theoretic modeling with auctions as our choice of implementation.

1.2 The Problem

In this paper, we develop a scheduling algorithm and payment mechanism for the type of application scheduling problem discussed above. We set up the problem in the form of a reverse auction. If we were to have truthful information about the capabilities and costs of the resource providers, the scheduling problem can be cast as an optimization problem. We provide an efficient scheduling algorithm along with a payment structure that ensures truthful elicitation of valuations from the bidders. The grid user whose job is to be scheduled has a deadline which is known only to him. He then proceeds to conduct an auction inviting bids for single units of the job. The bids represent the valuations of the resource providers for a unit of the job when working at a given percentage of their capacity. In development of the scheduling algorithm, we assume that we have obtained truthful values for the valuations which is in turn ensured by the payment mechanism we propose.

1.3 Related Work

Market based mechanisms for distributed computing had been initially suggested in (Hogg et al., 1992) when they applied market principles to scheduling in a distributed system.Though the research precedes the popular advent of grid computing, Spawn was the first implementation of a distributed computational economy that acted as a precursor to many of the market based models that were developed later.

One very important work in this area was the dissertation work of Buyya (Buyya, 2002). In this work he identified the key requirements of an economic-based Grid system and developed a distributed computational economy framework called the Grid Architecture for Computational Economy (GRACE), which is generic enough to accommodate different economic models and maps well onto the architecture of wide area distributed systems.

Nisan et al (Nisan et al., 1998) suggested an economic model for grids called POPCORN for trading online CPU time among distributed computers. In their system a virtual currency called ”popcoin” was used as the unit of trade between buyers and sellers. The market was an auction based one and the social efficiency and price stability were studied using the Vickrey auction theory.

Some general classes of scheduling problems along with auction protocols for the same were studied in (Wellman et al., 1998). Decentralized economic protocols were suggested for scheduling. They had also compared direct revelation mechanisms to the market based methods.

Various game theoretic models, mechanisms and auction models were applied in grid scheduling issues in Grosu et al (Grosu et al., 2002), (Grosu and Chronopoulos, 2003), (Grosu and Chronopoulos, 2004), (Das and Grosu, 2005). These works dealt with mechanisms and models to cope with selfishness of the grid users and designed strategy-proof mechanisms and auction algorithms for generic scheduling on the grid.

Most of the work discussed dealt with distributed scheduling using a game theoretic approach or market based models. They did not look at specific problems arising in grid computing. Grid computing with its characteristic flow models and implementation issues is completely different from a conventional distributed system. Works propounding actual scheduling algorithms for the grid fail to take into account the selfish nature of the entities involved. It is this research gap that this paper attemps at addressing.

1.4 Our Contributions

The primary focus of this work is to ensure scheduling efficiency in the presence of rational and self-interested users and resource providers. We initially design the auction for the scheduling problem. Assuming that the bids received were truthful, we then formulate the auction assignment as an optimization problem. The specific structure of the problem allows us to develop an algorithm for effiecient deadline-based scheduling for parallel-flow type applications. In order to counter the degradation of efficiency due to actions of selfish agents, we propose a payment mechanism based on the famed Groves' Mechanism from the Vickrey-Clarkes-Groves(VCG) stable. As a result of this application of VCG mechanism, the payment mechanism induces the bidders to bid their true valuations thereby enabling the algorithm to calculate the optimal allocation.

1.5 Organization of the Paper

The paper is organized as follows: We give a very brief introduction to auctions and mechanism design in Section 2. We explain the model and formulate the problem in Section 3. Assumptions and limitations with respect to the model chosen are also presented in this section. In section 4, we develop the optimal allocation algorithm followed by the design of the payment mechanism. Finally, in section 5, we present our conclusions and scope for future research.

2 A PRIMER ON MECHANISM DESIGN

In this section, we give a brief introduction to the concepts of mechanism design with focus on VCG mechanisms. Most of the material in this section is adapted from (Narahari and Dayama, 2005).
Consider a set of agents (grid users or service providers) $N = 1, 2, ..., n$ with agent i having a type set Θ_i. The type set of an agent represents the set of perceived values of an agent. In the case of grid computing, the type set of an agent refers to the cost structure and deadline information of the users. Let Θ be the cartesian product of all the type sets of all the agents. Let X be the set of all outcomes. In our sense, an outcome refers to a possible allocation of jobs to providers and the associated payment made to them. A social choice function is a mapping from Θ to X which associates an outcome for every type profile. With respect to grid scheduling, the social choice function corresponds to an efficient way of allocating jobs to service providers. Let S_i denote the action set of agent i, that is S_i is the set of all actions available to the agent in a given situation. A given strategy s_i is a mapping from $\Theta_i \rightarrow S_i$. In a Grid setting, the strategy of the resource providers is the information they release about their cost structure. Suppose S is the cartesian product of all the strategy sets. A mechanism is then a tuple $(S_1, S_2, ..., S_n, g(.))$, where g is the mapping from S to X. That is, $g(.)$ maps every strategy profile into an outcome. A game with incomplete information can be associated with every mechanism. This game is called the game induced by the mechanism.

We say that a mechanism $\mu = (S_1, S_2, ..., S_n, g(.))$ implements a social choice function f if there is an equilibrium strategy profile $(s_1^*(.), s_2^*(.), ..., s_n^*(.))$ of the game induced by μ such that

$$g(s_1^*(\theta_1), s_2^*(\theta_2), ..., s_n^*(\theta_n)) = f(\theta_1, \theta_2, ..., \theta_n)$$

for all possible type profiles $(\theta_1, \theta_2, ..., \theta_n)$. That is, a mechanism implements a social choice function $f(.)$ if there is an equililibrium of the game induced by the mechanism that yields the same outcomes as $f(.)$ for each possible profile of types. Based on the type of equilibrium, two common types of implementations are *dominant strategy implementation* and *bayesian Nash implementation*. For a more detailed view of these concepts the reader is reffered to (Mas-Colell et al., 1995).
We now give some of the desirable properties of a mechanism:

Efficiency A general criterion for evaluating a mechanism is *Pareto efficiency*, meaning that no agent could improve its allocation without making at least one other agent worse off. Another metric of efficiency is *allocative efficiency* which is achieved when the total value of all the winners is maximized.

Individual rationality A mechanism is individual rational if its allocation does not make any agent worse-off than had the agent abstained from participating in the mechanism.

Incentive compatibility A mechanism is incentive compatible if the agents maximize their expected payoffs by bidding true valuations for the goods. It is desired that truthful bidding by agents should lead to a well defined equilibrium such as dominant strategy equilibrium or Bayesian Nash Equilibrium.

Strategy proofness A mechanism is said to be strategy proof if it is incentive compatibile and implements the desired social choice function in dominant strategies. This is a very strong concept and is a highly desirable property of a mechanism but because of its stringent requirements, is not present in most occasions.

2.1 Vickrey-Clarke-Groves Mechanisms

The simplest model of this group of mechanisms is the second price sealed bid auction (Vickrey auction) for a single item. The Vickrey auction is shown to be incentive compatible in dominant strategies (strategy proof). Clarke mechanisms and Groves mechanisms provide a broader class of incentive compatible mechanisms. All these mechanisms are collectively referred to as VCG (Vickrey-Clarke-Groves) mechanisms. These mechanisms induce truth revelation by providing a discount to each winning buying agent on his actual bid. This discount which is called the *Vickrey Discount* is actually the extent by which the total revenue to the seller is increased due to the presence of this bidder. If the agents are selling agents, then we have *Vickrey surplus* which is the additional amount given to a selling agent over and above what he has quoted. The VCG mechanisms are highly attractive because of the fact that generally they are allocatively efficient, individual rational, weakly budget balanced, and incentive compatible. But they are limited in their application by the fact that they assume quasi-linear utilities for the participating agents.

2.1.1 Groves Mechanism

A Groves mechanism is a direct revelation mechanism $M = (\Theta_1, \Theta_2, ..., \Theta_n, f(.))$ in which the function $f(\theta) = (k^*(\theta), t_1(\theta), ..., t_n(\theta))$ satisfies two properties:

1. Allocative efficiency
   $\forall\theta \in \Theta \forall k \in K$,

$$\sum_{i=1}^{n} v_i(K^*(\theta), \theta_{-i})) \geq \sum_{i=1}^{n} v_i(k, \theta_{-i}) \tag{1}$$

2. $\forall i \in N$, the payment received by agent i is of the form

$$t_i(\theta) = t_i(\theta_i, \theta_{-i}) = h_i(\theta_{-i}) + \sum_{j \neq i} v_j(k^*(\theta), \theta_j) \tag{2}$$

   where $h_i(\theta_{-i})$ is any arbitrary function $h_i : \Theta_{-i} \rightarrow \Re$ for each $i \in N$

2.1.2 Clarke's Mechanism

This is a special case of Groves mechanism when the $h_i(\theta_{-i})$ function is of a particular intuitive form

$$h_i(\theta_{-i}) = -\sum_{j \neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j) \tag{3}$$

3 THE MODEL

We consider a grid user in a grid computing system who is about to submit a job to be processed on the grid. The nature of the job is such that it can be parallelized into n approximately equal sub-jobs that can be executed in parallel without any dependencies. The user has a deadline d by which the entire job is to be completed. The user is ready to pay for the services of resource providers. Without loss of generality, the de facto mode of payment can be considered to be grid dollars G$ (Das and Grosu, 2005). It is to be noted here that a single resource provider can take up many sub-jobs. We consider a buyer-operated marketplace where the mechanism used is a procurement auction. The buyer sends out the specifications of the jobs on the grid broadcast system without actually revealing the deadline. The buyer then proceeds to invite bids from the resource provider for execution of a single sub-job in the following format.

$$(t_i, T_i, C_i, c_i)$$

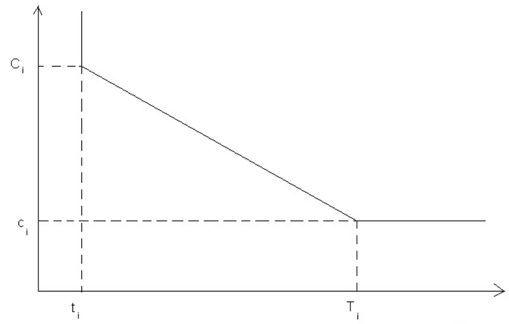

Figure 1: Cost Curve of a Resource Provider.

where t_i is the minimum time required by the i'th user to complete the sub-job at cost C_i and T_i is the time beyond which the cost is constant at c_i. Between t_i and T_i, the cost is assumed to be linear.

Now the problem for the buyer is to find upto n resource providers to whom the sub-jobs may be committed for execution. The grid user in addition to meeting his deadline would also like to minimize his total expenditure. Because the resource providers are selfish and rational and are thus looking to maximize their profit, they will not report their true cost curves. So, the buyer, in order to find the optimal allocation, has to employ an incentive compatible payment mechanism to elicit truthful bids from the resource providers. As soon as the user receives the bids, he can eliminate all bids with $t_i > d$ which removes resource providers whose processing power does not allow them to meet the deadline.

4 PROBLEM FORMULATION AND OPTIMAL ALLOCATION ALGORITHM

Assume that the user is left with m bids after elimination as explained in the previous section. Given the structure of the grid and the number of users present, we can safely assume that $m > n$.

Define,

$$\lambda_i = \frac{C_i - c_i}{T_i - t_i} \tag{4}$$

where λ_i becomes the slope of the cost curve for bidder i.

Now the user is aware of his deadline and sets up the allocation problem as a non-linear integer programming problem as follows:

$$l\min \sum_{i=1}^{m} \left[c_i + \max\left(\left(\mathrm{T_i} - \frac{d}{x_i} \right), 0 \right) * \lambda_i \right] * x_i \quad (5)$$

$$\begin{array}{l} \text{s.t} \\ \sum_{i=1}^{m} x_i = n \\ x_i \in \mathrm{N} \end{array}$$

where x_i is the variable which indicates number of jobs allocated to resource provider i. Though the solution of this integer programming problem is difficult, the structured nature of this problem leads to an elegant and efficient algorithm.

4.1 Optimal Allocation Algorithm

Knowing the deadline, d, the metric r_i is computed by the user as follows:

$$r_i = \begin{cases} \lambda_i, T_i > d \\ 0, T_i < d \end{cases} \quad (6)$$

Then the prices charged by each of the bidders for execution time d are computed using the formula $p_i = c_i + (T_i - d)r_i$. With these prices for a single sub-job execution, n minimum cost providers are selected initially. Then, using an iterative search, the optimal allocation is found among these n bidders. It is to be noted that we have assumed truthful and correct values of the bids in order to compute the optimal assignment. This assumption is justified by the payment mechanism which ensures this.

The winners determination algorithm can be summarized as follows:

compute $p_i = c_i + (T_i - d) * r_i \forall i = 1, 2, ..., m$

Choose $P_{(1)}, P_{(2)}, ..., P_{(n)}$ from sorted order $P_{(1)}, P_{(2)}, ..., P_{(m)}$

Allocate one sub-job to each of these n providers

```
set last=n

loop for i=1 to last
  loop for j=last to 1 and j ≠ i

    if deadline is met and cost does
    not increase

      increase allocation to i by 1

      decrease allocation to j by 1

      if j=0,remove j from list,last--
```

4.2 Coping with Selfishness

This algorithm produces an optimal solution to the scheduling problem assuming that the bids received were truthful. But this is not generally the case when the system consists of self-interested, rational resource providers. Now, this can be ensured by devising an appropriate payment scheme which incentivises the bidders to bid truthful values. On the lines of the famed VCG mechanisms, the payment to a resource provider is computed by giving a surplus to the winning bidder that is equal to the overall value the bidder adds to the revenue of the auctioneer.
Recall that the VCG mechanisms can be applied to scenarios in which the agents have quasi-linear utilities, or utilities of the form:

$$u_i(x, \theta_i) = v_i(k, \theta_i) + (m_i + t_i) \quad (7)$$

where u_i is the utility function, x is an outcome in the game, θ_i is the private valuation of player i, v_i is the cost incurred by player i in participating in this mechanism, k is the allocation vector, m_i is some form of endowment or participation fee and t_i is the payment received by player i.
It can be immediately seen that the model we have proposed clearly falls in this category because the utility to a resource provider is the difference between his cost of execution of the job using his resources minus the payment he recieves for completing the job. Hence, the VCG mechanisms can be applied directly in this setting.
Recall from section 2, the structure of the incentive to be given to an agent in the Clarke's mechanism

$$h_i(\theta_{-i}) = -\sum_{j \neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j) \quad (8)$$

This simply means that the loss to the grid user, who submits a job, due to the absence of each of the resource providers must be given back to the resource provider as an incentive to induce him to reveal his true valuation. This means that this calculated value must be given to the selected resource providers over and above their bid for the tasks. It is very simple to see that if a resource provider was not part of the final allocation as determined by the winners determination algorithm, then their incentive value is 0, while for those providers who were part of the final allocation this is simply the difference between the total payment made with and without that particular node in the bidding process. This means we have to run the winners determination algorithm k more times ($k \leq n$). Though this adds to the overall complexity of the process, considering the truthful values it induces, it is but a small price to pay.

4.3 Complexity Analysis

The complexity determining step in the above algorithm is the operations within the two loops which takes $\Theta(m^2)$ steps. Step 2 which involves sorting of m elements would take only $\Theta(mlogm)$ time. Since $m > n$ and the other steps are linear time computable, the entire algorithm is $\Theta(m^2)$ time solvable. But, the application of payment rule according to Clarke's mechanism demands that the algorithm be run maximum of n+1 times which adds a factor of m to the overall complexity of the process thereby yielding a $\Theta(m^3)$ complexity.
Thus the payment mechanism ensures truthful bids and the optimal allocation algorithm ensures an efficient and reasonably fast allocation of the sub-jobs.

5 CONCLUSION

The Algorithm developed in the paper computes an optimal allocation of jobs to resource providers and the payment mechanism which is inspired by the VCG mechanism ensures truthful dissemination of information. The Mechanism is highly robust because of its allocative efficiency and strategy proofness. The allocation algorithm and payment structure works optimally within the limitations and assumptions discussed in this paper. The optimal algorithm presents a clever way of circumventing the problems associated with the inherent non-linear integer programming formulation.

We have considered one model of application scheduling on the grid. Such strategy proof mechanisms must be developed for more models if we have to truly unleash the power of grid computing. A necessary improvement would be to estimate the efficiency of this mechanism by comparitive simulation with other scheduling mechanisms. Only through extensive simulations can we conclude that this mechanism would work in a grid setting. Another improvement would be to relax the assumption that all the sub-jobs are of equal size. One more relaxation could be made to the linear cost model of the resource providers, replacing it with a piece-wise linear model. Procurement auctions of the type involving piece-wise linear cost models can be studied in (Eso et al., 2001)

REFERENCES

Buyya, R. (2002). *Economic-based Distributed Resource Management and Scheduling for Grid Computing.* PhD thesis, School of Computer Science and Software Engineering, Monash University, Australia.

Das, A. and Grosu, D. (2005). Combinatorial auction-based protocols for resource allocation in grids. In *IPDPS05*.

Eso, M., Ghosh, S., Kalagnanam, J., and Ladayani, L. (2001). Bid evaluation in procurement auctions with piece-wise linear supply curves. Technical report, IBM Research Division, Yorktown Heights, NY 10598.

Foster, I. and Kesselman, C. (2002). *The Grid: Blueprint for a new computing infrastructure.* Morgan kaufmann publishers.

Grosu, D. and Chronopoulos, A. (2003). A load balancing mechanism with verification. In *IPDPS03*.

Grosu, D. and Chronopoulos, A. (2004). Algorithmic mechanism design for load balancing in distributed systems. In *IPDPS04*.

Grosu, D., Chronopoulos, A., and Leung, M. (2002). Load balancing in distributed systems: An approach using cooperative games. In *IPDPS02*.

Hogg, W. C., T., Huberman, B., J., K., and Stornetta, W. (1992). Spawn: a distributed computational economy. *IEEE transactions on software engineering*, 18(2):103–117.

Mas-Colell, A., Whinston, M., and Green, J. (1995). *Microeconomic Theory*. Oxford University Press, New York.

Narahari, Y. and Dayama, P. (2005). Combinatorial auctions for electronic business. *Sadhana*, 30(2-3):179–211.

Nisan, N., London, S., Regev, O., and Camiel, N. (1998). The popcorn market-online markets for computational resources. In *First International conference on Information and Computational Economies.*

Wellman, M., Walsh, W., Wurman, P., and MacKie-Mason, J. (1998). Some economics of market based distributed scheduling. In *ICDCS98*.

Wolski, R., Plank, J., Brevik, J., and Bryan, T. (2002). G-commerce: Market formulations controlling resource allocation on the computational grid. In *AAMAS 2002*.

Zhu, Y. (2003). A survey on grid scheduling systems. Technical report, Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong.