Resource Allocation in Market-based Grids Using a History-based Pricing Mechanism

Behnaz Pourebrahimi, S. Arash Ostadzadeh, and Koen Bertels
Computer Engineering Laboratory, Delft University of Technology
Delft, The Netherlands
{behnaz, arash, koen}@ce.et.tudelft.nl

***Abstract*-In an ad-hoc Grid environment where producers and consumers compete for providing and employing resources, trade handling in a fair and stable way is a challenging task. Dynamic changes in the availability of resources over time makes the treatment yet more complicated. Here we employ a continuous double auction protocol as an economic-based approach to allocate idle processing resources among the demanding nodes. Consumers and producers determine their bid and ask prices using a sophisticated history-based dynamic pricing strategy and the auctioneer follows a discriminatory pricing policy which sets the transaction price individually for each matched buyer-seller pair. The pricing strategy presented generally simulates human intelligence in order to define a logical price by local analysis of the previous trade cases. This strategy is adopted to meet the user requirements and constraints set by consumers/producers. Experimental results show waiting time optimization which is particularly critical when resources are scarce.**

I. Introduction

In High Performance Computing (HPC) terminology, Grid refers to an environment with the aim of hooking many independent or loosely coupled tasks to available idle processing resources provided by the workstations in the system. Condor [1] is a typical example of such systems that manages pools of hundreds of workstations around the world and allows the utilization of idle CPU cycles among them. Due to heterogeneities present in Grid environments, resource management is often based on approaches which are both system and user centric.

System centric approaches are traditional ones which attempt to optimize system-wide measure of performance such as overall throughput of the system. On the other hand, user centric approaches concentrate on providing maximum utilization to the users of the system based on their QoS requirements, i.e., a guarantee of certain levels of performance based on the attributes that the user finds important such as the deadline by which the jobs have to be completed [2].

Economic-based approaches provide an appropriate background in order to encourage resource owners to contribute their processing supplies to the Grid environment. This is the base of user centric performance, where the service received by each individual node tailored for one's own requirements and preferences, is considered in addition to the utilization of the system as a whole. Nimrod-G [3] is an instance of economic-based systems which introduces the concept of computational economy in managing and scheduling resources in Grids.

Meeting QoS constraints together with maintaining an acceptable level of system performance and utilization is the primary problem to tackle in ad-hoc Grid environments where the availability of resources and workloads are changing dynamically. This dynamic behavior in turn provokes challenges between the consumers and producers of resources in order to get hold of the required resources or tasks. In such a vibrating environment, delivering an appropriate degree of utilization both individually and globally is critical.

In the economic-based approaches, scheduling and managing resources are made dynamically at running time and are directed by the end-user preferences and requirements. Economic-based models have been used widely in resource allocation algorithms [4] [5]. A suitable platform for resource allocation in Grids is an auction model. As auctions allow consumers and producers of resources to compete in a dynamic environment where no global information is available and the price is fixed based on local knowledge. Several researches have been reported on auction mechanisms for resource allocation in Grids [6][7][8][9][10]. Our proposed strategy differs from the previous related approaches in two contexts. We introduce a sophisticated history-based dynamic pricing strategy adopted by consumers and producers to determine the preferred prices based on the requirements rather than using a static reservation price. We also investigate the market-based approach in a dynamic network, where the resources are not dedicated and the number and availability of the resources may change at any given time.

We utilize an economic-based model for scheduling and managing resources in market-based Grids. A Continuous Double Auction (CDA) protocol with discriminatory pricing policy is used as the basic platform for matchmaking where consumers and producers trade. Our proposed method is distinguished from previously introduced strategy [11] in a sense that a rational analysis of the previous trading cases weighing the matching time for each individual node, is conducted to settle a new price. Budget constraints for nodes are also applied in this implementation.

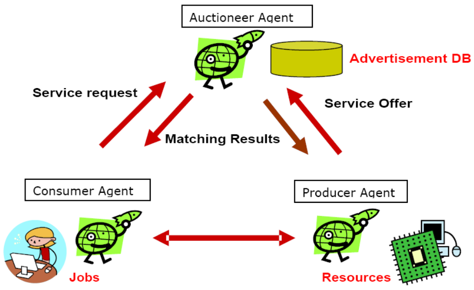

Fig. 1. Market-based Grid components

We examine accesses to the Grid resources for individual nodes in our model and compare it with a non-economic approach. Elapsed time for matching is calculated in favor of comparative studies. The results are promising and demonstrate a higher degree of waiting time optimization for individual nodes. Furthermore, system performance measurements demonstrate that the new strategy outperforms the previous methods. It provides more or less the same task utilization and is quicker in matchmaking.

The paper is organized as follows. Section II introduces the basic market-based Grid model and the proposed pricing mechanism. The experimental results concerning the waiting time for requests/offers are presented in section III. Finally, section IV summarizes the concluding remarks.

II. Market-Based Grid Modeling

A conventional market-based Grid model includes three different types of agents: Consumer (buyer), Producer (seller) and Auctioneer (matchmaking coordinator). There is one consumer/producer agent per node. A consumer/producer agent submits its request/offer to the auctioneer. The auctioneer agent manages the market, adopting a particular auction protocol. Fig. 1 depicts the components contained in such a system.

Continuous Double Auction (CDA) with discriminatory pricing policy is used as the economic-based protocol for matchmaking. CDA supports simultaneous participation of producer/consumer, observes resource/request deadlines and can accommodate the variations in resource availability which is the case in ad-hoc Grids. According to this market model, buy orders (bids) and sell orders (asks) may be submitted at any given time during the trading period. Whenever there are open bids and asks that can be matched or are compatible in terms of price and requirements (e.g. quantity of resources), a trade is executed immediately. The auctioneer seeks correspondence between buyers and sellers by matching offers (starting with the lowest price and moving up) with requests (starting with the highest price and moving down). When a task query arrives at the market place, the protocol searches all available resource offers and returns the best match which satisfies the task constraints, namely resource quantity, time frame and price. When a resource becomes available and several tasks are waiting, the one with the highest price bid is processed first. If no match is found, the task query is placed in a queue. The queries are kept in queue until the defined Time-To-Live (TTL) field is expired or a match is found. Transaction price is determined as the average of bid and ask prices.

A. Request / Offer Specifications

Each request/offer submitted by consumer/producer has a specification which contains different segments. These segments determine request or offer details, requirements and constraints.

- *Request.* A request message contains three segments:
    1. *Task Details* include information about the task such as execution time and Task ID. Execution time is an estimated processing time needed for execution. As different nodes have different hardware architectures, this time is calculated based on a reference processing unit. Task ID is a unique number denoting each task and is used in the case of multiple requests from the same consumer.
    2. *Task Deadline* specifies the deadline for the task execution.
    3. *Price Constraints* contain buyer price and buyer budget. Buyer price is the upper bound price the consumer is willing to pay for each unit of resource. Budget denotes the maximum budget currently available for the consumer.
- *Offer.* An offer message comprises three segments:
    1. *Resource Details* contain information about the resource characteristics such as CPU speed.
    2. *Resource Deadline* indicates the time interval during which the resource is available.
    3. *Price Constraint* denotes a seller price that is the value of each unit of resource, set as a lower bound.

B. History-based Dynamic Pricing Strategy

Consumer and producer agents join the market with an initial predefined price and dynamically update it over the time using an intelligent pricing strategy presented in this work. The price is defined as the value of each unit of resource in which the consumer and producer agents are willing to buy or sell. There is an upper limit for consumer price (bid price) which is rationally defined by the individual node budget, as *bidPrice*resourceQuantity<=Budget*. We also set a minimum value for producers below which they are not willing to offer their resources.

The agents perceive the demand and supply of the resources through their previous experiences and update their prices accordingly by careful inspections of the matching times in their former cases. Based on this strategy, ask and bid prices are defined respectively for producers and consumers as following:

$$p_a(t) = \max\{p_{\min}\ ,\ p_a(t-1) + \Delta p_a\} \tag{1}$$

and

$$p_b(t) = \min\{p_{\max}\ ,\ p_b(t-1) + \Delta p_b\} \tag{2}$$

Where *p(t)* is the new price and *p(t-1)* denotes the previous price. p_{min} is the minimum acceptable value for producers and p_{max} is the maximum affordable price for consumers, which is defined as:

$$p_{max} = Budget\,/\,resourceQuantity \tag{3}$$

resourceQuantity is the quantity of the needed resource. For example, it can refer to *job execution time* when CPU time is considered as the resource. Δp indicates whether or not the price is increasing. Δp for seller and buyer is defined based on the previous history of resource/task utilizations and timings at the corresponding seller or buyer.

$$\Delta p_a = (u_r(t) - u_{thR}) * \alpha * p_a(t-1) \tag{4}$$

and

$$\Delta p_b = (u_{thT} - u_t(t)) * \beta * p_b(t-1) \tag{5}$$

α and β indicate the factors corresponding to the rates at which the prices are increased or decreased. u_{thT} and u_{thR} are threshold values denoting the lower bound for task/resource utilizations. In other words, u_{thT} and u_{thR} can be considered as satisfaction thresholds for agents. Low values for these parameters imply that the agent is satisfied with a low usage of its resources or a low completion rate of its tasks. High values, on the contrary, denote more demanding expectations of the agents. The resource and task utilizations, $u_r(t)$ and $u_t(t)$, define respectively the level of previous utilizations of producer and consumer from the Grid market. They are computed locally by each agent considering the time spans of previous matchmakings in the history of each node. To make the utilization factor even more logical, we weigh the matchmakings made quickly (with small or no delay time after the request is posted) compared to the prolonged ones due to the current parameter specifications. The utilization formula is as follows.

$$u(t) = \sum_{i=t_1}^{t_2} x(i) * \frac{t_{TTL}(i)}{t_{match}(i)} \Bigg/ \sum_{i=t_1}^{t_2} \frac{t_{TTL}(i)}{t_{unit}(i)} \tag{6}$$

x(i) denotes one instance of request/offer within the time interval [$t_1 ... t_2$]. It has either the value of '1' for a matched request/offer or '0' for unmatched one. [$t_1 ... t_2$] is considered as the time interval during which *n* efforts have been made by the agent to buy or sell resources. In fact, *n* defines the number of former experiences considered for utilization estimate in the pricing strategy. This value can be dynamically set by the user to indicate the capacity of the matchmaking history memory. In our experiments, *n* is considered to be 10. t_{TTL} indicates the Time-To-Live field for each offer of the seller agent or the required deadline set by the buyer agent for a resource request. t_{unit} refers to one unit of time in a particular resource request. The agreement between matched ask-bid pair is made at a *transaction price* which is defined as:

$$p = k * p_a + (1-k) * p_b \tag{7}$$

where p_a and p_b are prices offered by the seller and buyer respectively. This is the common definition of k-double auction pricing rule [12]. For a fair trade, we have assumed k = 0.5, however the user is free to set this value according to individual requirements and preferences.

III. Experimental Results

To perform our experiments, we set up a Grid like environment based on a LAN in which our application test-bed is developed using J2EE and Enterprise Java Beans. JBoss application server is used to implement the auctioneer. The network consists of three types of agents. Consumers have certain tasks to perform for which they demand resources and producers have idle resources to offer, and auctioneer takes care of the matching process.

CPU time is considered as the resource in our system. For simplicity, we ignore allocation requirements and assume that tasks need CPU time only for execution. Whenever a consumer needs CPU time for performing a task, it sends a request to the auctioneer and similarly a producer announces its idle CPU time by sending an offer. The simulations are performed in an environment with 60 nodes. Each node is assigned a specific budget when joining the Grid. Each node creates a number of requests or offers during the simulation process. There are two user centric requirements which should be fulfilled by the system, namely deadline and budget.

We compare our method with a non-economic First-Come First-Served (FCFS) approach under similar conditions. It should be noted that in FCFS, no pricing constraint is defined and resources are allocated based on the resource quantity and deadline constraint. We conduct the experiments in a condition where resources are scarce and the number of generated tasks exceeds the available resources. This condition has been chosen in order to investigate the waiting time behavior when resources are scarce, which resembles a critical situation for Grid users.

Matching time analysis for each request/offer provides a valuable criterion for performance measurement. Using our proposed method, the time span between a request/offer issue and a corresponding match, is expected to be minimized, which results in quicker matchmaking. In this respect, the available resources in the Grid are not wasted and the throughput of the system is increased. Besides, the demanding nodes are not kept blocked with long delays in request satisfaction. Figure 2 depicts the elapsed waiting time for consumers after submitting a request till a match is found. The graph demonstrates the cases for FCFS and CDA. As inferred from the figure, the waiting time in CDA approach is quite

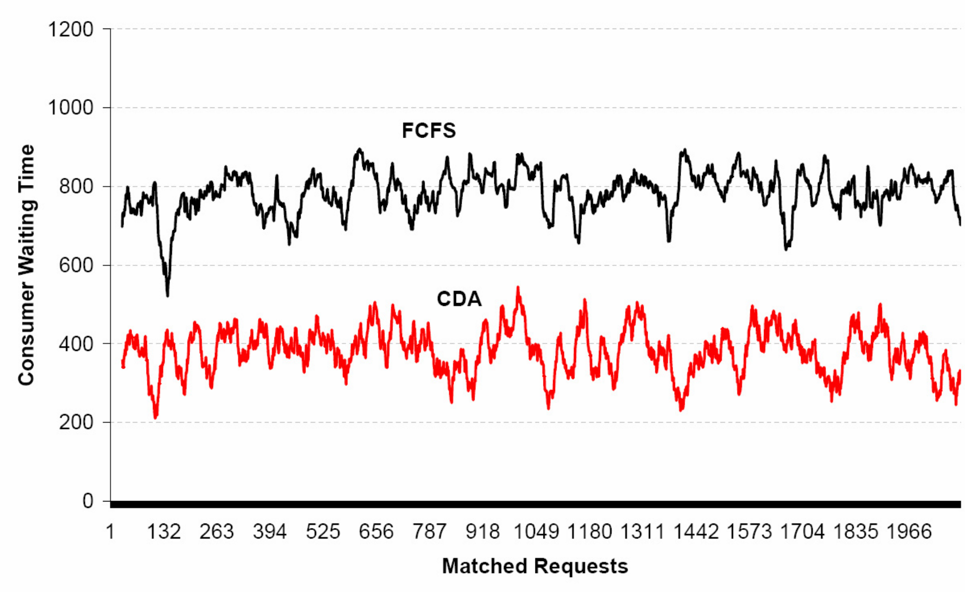

Fig. 2. Elapsed waiting time for consumers

lower compared to FCFS. This is not the case for producers, where both approaches show more or less the same values for the elapsed waiting times. The reason is due to scarce number of available resources and excessive number of potential tasks, which implies that a match can be quickly found for producers and they are not required to wait much.

IV. Concluding Remarks

In this paper, a market-based resource allocation model in ad-hoc Grids is introduced. In our dynamic proposed model, consumers and producers adopt a sophisticated history-based pricing strategy in order to reasonably update the prices based on their previous matchmaking experiences.

The distribution of the tasks among the resources in the Grid is studied with the new approach in the conditions where the resources are assumed to be scarce and the availability of tasks and resources varies over the time. Grid resource access is evaluated regarding the consumer/producer waiting time in the system and is compared with a non-economic conventional FCFS approach. The results show that our proposed strategy optimizes the waiting time for all submitted requests/offers, while FCFS demonstrate a low level of equitable utilization.

References

[1] M. Litzkow, M. Livny, and M. Mutka, “Condor - a hunter of idle workstations”, In Proceedings of the 8th International Conference of Distributed Computing Systems, June 1988.

[2] R. Buyya, D. Abramson, and S. Venugopal, The grid economy. In Special Issue on Grid Computing, vol. 93, pp. 698-714, 2005.

[3] R. Buyya, D. Abramson, and J. Giddy, “Nimrod-G: An architecture for a resource management and scheduling system in a global computational grid”, In Proceedings of The 4th Int. Conf. on High Performance Computing in Asia-Pacific Region, USA, 2000.

[4] R. Wolski, J. Brevik, J. S. Plank, and T. Bryan, Grid resource allocation and control using computational economies, In Grid Computing: Making The Global Infrastructure a Reality, John Wiley & Sons, 2003.

[5] R. Buyya, D. Abramson, J. Giddy, and H. Stockinger, Economic models for resource management and scheduling in grid computing, Concurrency and Computation: Practice and Experience, vol. 14(13-15), pp.1507-1542, 2002.

[6] C. Weng, X. Lu, G. Xue, Q. Deng, and M. Li, “A double auction mechanism for resource allocation on grid computing systems”, In GCC, page 269, 2004.

[7] J. Gomoluch and M. Schroeder, Market-based resource allocation for grid computing: A model and simulation, 2003.

[8] U. Kant and D. Grosu, “Double auction protocols for resource allocation in grids”, In Proceedings of the Int. Conf. on Information Technology: Coding and Computing, pp. 366-371, 2005.

[9] D. Grosu and A. Das, “Auction-based resource allocation protocols in grids”, In Proceedings of the 16th IASTED Int. Conf. on Parallel and Distributed Computing and Systems, pp. 20-27, November 2004.

[10] M. D. de Assuncao and R. Buyya, “An evaluation of communication demand of auction protocols in grid environments”, In Proceedings of the 3rd Int. Workshop on Grid Economics & Business, Singapore, 2006.

[11] B. Pourebrahimi, K. Bertels, G. Kandru, and S. Vassiliadis, “Market-based resource allocation in grids”, In proceedings of 2nd IEEE Int. Conf. on e-Science and Grid Computing, page 80, 2006.

[12] M. Satterthwaite and S. Williams, “The Bayesian theory of the k-double auction”, The Double Auction Market: Institutions, Theories and Evidence, Santa Fe Institute Studies in the Sciences of Complexity, pp. 99-123, 1991.