

Simulate Grid Resource Trading via Cognitive Agent: A Case Study

Yuhui Qiu

Faculty of Computer and Information Science
Southwest University, Chongqing, China
yhqiu@swu.edu.cn

Zhixing Huang

Faculty of Computer and Information Science
Southwest University, Chongqing, China
huangzx@swu.edu.cn

Abstract

In this paper, we explore the market-based grid resource trading system from the social perspective and the collaborative computing perspective. We firstly introduce a novel framework to simulate the trading system of the grid computing resources, then we give a case study based on this framework. In the case study, an evolutionary dynamics of the resource reservation game is played by agents on the network. We study the evolution of the system with cognitive agents: Study the evolution dynamics of system with agents can automatic adapt to their environment, they can exchange private information and learn experiences from each other. Finally, after analyzing the experiment results, some critical issues about designing the market-based trading system are discussed.

1 Introduction

Grid Computing has emerged as a new paradigm for next-generation computing. It supports the creation of virtual organizations and enterprises that enable the sharing, exchange, selection, and aggregation of geographically distributed heterogeneous resources for solving large-scale problems in science, engineering, and commerce. Grid Computing and also Peer-to-Peer Computing promises a flexible infrastructure for complex, dynamic and distributed resource sharing. In spite of a number of advantages in Grid computing, resource management and scheduling in such environments continues to be a challenging and complex undertaking. The future interconnect environment must absorb AI and distributed systems, inherit the advantages of the Web, Semantic Web, Grid and P2P technologies, and go beyond their scope with new principles [9].

In recent years, distributed computational economy [13] has been recognized as an effective metaphor for the management of Grid resources, as it: enables the regulation of supply and demand for resources, provides economic incentive for grid service providers, and motivates the grid ser-

vice consumers to trade off between deadline, budget, and the required level of quality-of-service. However, within these fields, numerous economical questions have to be solved, for example the evolutionary behavior of resources to grid users while assuring some specific quality of service for all users. In some recent practice of using market-based methods for resource allocation. Although there are some perfect theory results in Game Theory, Economics, etc, when we deployed microeconomic in the real resource allocator system, the user strategic behavior has been observed [5]. These behavior will change what the system should to be [3].

Many researchers adopt Agent-based Computational Economics (ACE) models to understand and analyze those complex economic systems. The social behavior consideration in multi-agent systems (e.g., cooperation, competition, proactiveness) has been recognized as a new dimension for enriching the autonomy of the coordination infrastructures.

In this paper, we explore the market-based grid resource trading system from different perspective: the social or behavioral perspective (Social Intelligence) and the collaborative computing perspective (Grid Intelligence) [16]. The remainder of this paper is structured as follows. In section 2, the ACE-based Grid Resource trading Frame used in this paper are proposed, and the model of small world is briefly introduced. In section 3, we analyze a resource trading case in which user agent reserve their computing service through participating in multiple sequential English auction, two novel risk-based heuristic bidding strategies also have been proposed in this section. In section 4, our model are evaluated through three experiments. Finally, in section 5 we present the conclusions and future work.

2 Grid Resource Trading Simulation Framework

In recent years, distributed computational economy has been recognized as an effective metaphor for the management of Grid resources. C. Kenyon [4] proposed a architecture which meets the needs for grid resource commer-

cialization. he pointed out that three necessary components: Value Expression, Value Translation and Value enforcement should be included in grid resource commercialization architecture. In this section, we propose a more concrete source trading simulation framework with five layer which also meets these needs.

Resource Layer: The basic information of the resources or the services been provided in the System is described in this layer. The basic information of the resource such as type, number, state, price and the service starting time, should be clearly depicted. Usually, the resources in the computing grid environment could be CPU, memory, disk, network bandwidth and the software licence. Furthermore, it also could be the Virtual Resources which combined the multiple concrete physic resources in the system.

Allocation Mechanism Layer: This layer includes the allocation or scheduling mechanisms of the resources. These mechanisms not only could be traditional queue-based allocation or scheduling methods, such as First Come First Serve, Shortest Job First, Highest Priority First, Proportional Based, etc., and these mechanisms also could be market-based allocation methods, For example, the allocation methods based on English auction, Vickrey auction and combinatorial auction, and so on. The allocation mechanism of the resources is specified by the owner of the resource, the resource user couldn't change these method. In addition, it has been strongly recommended that the function of advance reservation [7] should be supported in the allocation and scheduling mechanisms, and are currently being added to some test-bed of grid toolkit, such as Nimrod-G [13], etc.

Competitive Policy Layer: For the inherent dynamics and complex characteristics of the Grid resources, the competition strategies of the resources in Grid environment seems to be more complexity than in traditional centralized systems. It needs user carefully compares and selects the candidates of the service provider and then participates the completion of selected resource. When failure of the completion is taken place, user should repeat those operations. There are many possible strategies for competing these resource, especially for market-based allocation methods, such as Best-response strategy, Greedy strategy, Threshold strategy etc. We assume that most of the competitive policies in this layer could be provided by the service provider or the third-parties, user can freely selects the competitive strategy from this layer.

Adaptive Layer: The main function designed in this layer is to help user adapt the system's dynamics, it includes sense the change of environment, adjust the parameters of the competitive policies, save and analyze the historical data. For the dynamics of the environments, fixed competitive strategies are not always suitable, they should be adjusted according the change of the environment. For

example, in the market-based resource allocation system, combined with the predication of the price of the resource, most of these strategy can improve their performance. The methods which are adopted in Hybrid Intelligent System also could be used in this layer to improve the performance of the competitor. For comparing the influence of different learning methods, we assumed that the machine learning algorithms used in this layer are restricted in learning algorithms for single agent, in another words, these algorithms can only use the user local information or global information published by the system. And there is no cooperation between users in this layer.

Interaction Layer: The Interaction (user agent) layer include two parts of contents. On the one hand, it response to capture the needs of the user, such the deadline, budget, quality of service (QoS), and the preference between these attributes. On the other hand, this layer also includes the interaction between the different users, we could model the interaction between different users as the multi-agent cooperative learning. The interactive methods adapted in this framework which are differed from the traditional Agent-Oriented Computing which use global information to implement the cooperation between multi-agent. In the complex enjoyment, nobody can know all the information in the system, especially when some of the information are private.

To model the users' interaction in these complex environment, many researchers adopt Agent-based Computational Economics (ACE) [6] models to understand and analyze complex economic systems. L. Tesfatsion [10] defines Agent-based Computational Economics as the computational study of economies modeled as evolving systems of autonomous interacting agents. Starting from initial conditions, specified by the modeler, the computational economy evolves over time as its constituent agents repeatedly interact with each other and learn from these interactions.

In the studies in the ACE, the small world network has been widely used. Following an important body of literature in the field of socio-psychology and sociometrics initiated by Milgram [14]. The small world networks have been found in many areas so far [8], such as the knowledge, innovation diffusion processes, market organization etc. Small world can be viewed as a suitable model to simulate the interaction between humans, and it also can be used to study the emergence phenomena of the interaction between intelligent agents in the grid.

3 Trading Simulation: A Case Study

In this section, we specifically consider a special case study of resource advance reservation through user agents bid in multiple overlapping English auctions. As be noted

in GRAAP-WG ¹, the state of advance reservation has nine different states. For simplicity, in this paper we only consider the following three most important states: Requested, Declined, Booked.

- Requested: A user has requested a set of resources for a reservation. If the reservation accepted, it goes to being booked. Otherwise, it becomes declined.
- Declined: The reservation is not successfully allocated for some reason.
- Booked: A reservation has been made, and will be honored by the scheduler. From here, the reservation can become active.

That is, the booked reservation can not be altered or be cancelled.

3.1 Resources Provider

Suppose there are R resource/service providers willing to provide computation service, the services of the providers are homogenous. Suppose each service provider owns a computation pool with the capacity c_j , and computation service can be started at time s_j . Each service provider has its lowest price r_j for providing service (also call reserve price) and the minimal increase of the price λ_j in the auction. If one agent's bid to the service provider j is the highest bid in the last l_j rounds, the agent i will win this auction and the request of resource reservation is booked. As we have mentioned before, the agent's task size and the service's new start time are announced by provider and can be known by all the user agents. The services are reserved through auctions, and each service provider runs one auction independently.

3.2 Allocation Mechanism

Now, we briefly introduce the variant of ascending auction protocol which is different from the standard ascending auction [12] for the general continuous resource reservation protocol. In detail, we consider a multiple sequential English auction market, where each auction is given a start and an active bidding lasting time that may overlap with other auctions. We assume that English auctions work according to the following principles:

The auctions proceed in rounds. Each auction administrate the price of the service. We don't assume that the different auctions' rounds are synchronized, that is, all auctions move from one round to the next simultaneously. In

¹Advance Reservation State of the Art: Grid Resource Allocation Agreement Protocol Working Group, URL <http://www.fz-juelich.de/zam/RD/coop/ggf/graap/graap-wg.html>.

each round, after all the user agents propose their bids, the service provider will select the highest bidding agent as the active one, and then it will update its current price. If there are more than one highest bidding, the service provider randomly selects one as the *active* agent. If the request of the user agent is declined, the agent is *inactive*. If the agent i 's bidding to the service provider j and is active in the last l_j round, the agent i will win this auction and the request of resource reservation is booked. The service provider will reset the service price to its' *reserve* price, and update the start time of the next service, and then a new auction turn will begin.

Though simultaneous ascending auction protocol does not include announcements of bidder identities, the task size of the active agent will be announced by the auctioneer, and the number of user agents remaining in the system can be known by everyone.

3.3 Bidding Strategies

In each round, the user agent considers bidding if and only if it is not holding an active bid in an auction or it has not already obtained the service. If user agent i is inactive, the user agent will choose the service provider which it will request to and then decide whether or not to propose the next bid in this server. As we note below, each service provider has its reserve price, and the valid bids to the service provider j must be increased by λ_j .

In contrast to the stand-alone English auction, there is no *dominant strategy* that can be exploited in the multiple auction context [1] and there did not exist Nash equilibrium in even in the very simple scheduling problems [12]. So in this auction-based resource reservation model, the heuristic strategies are adapted. In addition, we make a distinction between the bidding strategies. If the agent's strategy can predict the prices of the next k turns of the auctions, we call the strategy level k strategy. And more specifically, if the agent takes into consideration all the future turns of the auctions, we call this strategy level $*$ strategy.

A key component of the successful strategy is able to make predictions about the likely closing prices of the various auctions, so that the agent can determine whether is should place a bid at current moment or whether is should be delay because better deals may subsequently become available. In [17], we propose two heuristic bidding strategies based on fixed price predication function, although these strategies can outperform above three strategies in most cases, however, it could be improved as following modifications.

We define $\phi \in [-1, +\infty)$ to limit the maximum absolute gradient and avoid the target price updating excessively with respect to change in the risk factor γ_i .

$$\varepsilon_i = \frac{(n(t) - n(t_k))(n(t_k) - 1)}{(n(t) - 1)^2} \times \phi \times \gamma_i \quad (1)$$

We use the following equation to predicate the closing price of the next k turn in the same server.

$$p_j(t_k) = r_j + \frac{(p_j(t_k) - r_j)(n(t_k) - 1)}{n(t)} \times (1 + \varepsilon_i) \quad (2)$$

In these equations, $p_j(t)$ is the current price and $n(t)$ is the current number of agent who are participating in the auctions at the current time t . Moreover, $p_j(t_k)$ is the predication of the price on server j and $n(t_k)$ is the number of agents remaining in the system at the time of t_k , $t_k < t$ and $n(t_k) \in [1, n(t) - 1]$. For example, suppose the number of agents currently in the system is n and the current auction will be closed soon. If we want to predict the likely closing price p'_j of the service j in the next turn, conservatively, we say the number of active agent remaining in the system will be $n - 1$. Through comparing the utilities of the current turn and the utilities of the future turns, the agents can make decisions on when and which auction they will bid to.

Intuitively, we can see that if there is only one user agent in the system, the price of the service on server j will fall down close to the server's reserve price. But as the number of agents increases, the competition of the service becomes severe, and the price of the service will increase.

RB-Greedy(1) strategy: unless the agent is active in some auction, based on Equation 2, the agent decides to bid in whichever auction currently has the highest evaluation, or decides not to bid if there exists a higher expected evaluation in the next turn than the current highest evaluation.

RB-Greedy(*) strategy: If the agent knows the average size of agents' tasks, it can use the following equation to predicate its task end time.

$$e_{ij} = s_j + \frac{t_j}{c_j} + \bar{t} \times \left[\frac{\bar{t} \times (n(t_k) - 1)}{\sum c_j} \right] \quad (3)$$

For $n(t_k) \in [1, n(t) - 1]$, based on Equation 2 and Equation 3, the agent can predicate the maximum utility by comparing different utilities. Unless the agent is active in some auction, the agent decides to bid in whichever auction currently with the highest evaluation, or decides not to bid if there exists a higher expected evaluation in future turns than the current highest evaluation. It indicates in the complex scenarios that the highest level of the strategy is not certainly with the best performance.

3.4 The Adaptive Layer

In the adaptive layer, the user agent uses a set of learning rule to update its risk factor to better fit prevailing mar-

ket conditions. Specifically, a learning algorithm is used to increase or decrease the risk factor. The Risk-Based strategy can risk-seeking that is the trader tries to achieve high profit but has a correspondingly lower probability of transacting), or risk-averse, which trades-off lower profit for a higher probability of transacting. The risk-neutral strategy is considers a bid or ask that maximizes its expected profit. Risk-Based strategy is flexible in that it can vary its risk attitude depending on the prevailing market conditions, to remain competitive.

We adapt the user agent's risk attitude by gradually changing its risk factor to a desired risk factor. η was chosen based on simulation result. Specifically,

$$\gamma(t+1) = \gamma(t) + \beta * \eta, \eta = \{-0.05, 0.05\} \quad (4)$$

Where $\beta \in (0, 1)$ is the learning rate of the algorithm and influences the moving rate of the bidding price. And the learning rules for user agent is that: when a reservation transaction price is p , if current trade price is lower that user's predicate price agent increase its risk factor else agent decrease its risk factor.

The role of the risk model is to generate the target prices given the user risk attitude, which is defined by its risk factor. If the target price equal to the transaction price implies that the user is risk-neutral. When a user agent adopts a risk-seeking attitude, it considers a target price that is below the transaction price, in order to obtain a higher profit margin. Conversely, a risk-averse attitude implies that the user place bids above actual trading price.

3.5 User Agent

Suppose there are N user agents, each agent with one task to fulfill before the deadline. Furthermore, we assume that agent i has a budget m_i , the size of the task of agent i is t_i and the deadline of the task d_i . The information of agents' task size, deadline and budget are private information, and the distributions of these information are all unknown to the agents. Suppose the current price of the service j is p_j , so the cost of user agent is: $b_{ij} = t_i p_j / c_j$. And, if the agent i 's task running on server j starts at s_{ij} , the end time of the task is $e_{ij} = s_{ij} + t_i / c_j$. The utility of agent is related to the cost of fulfilling the task and the task's end time. In this paper, we use a constant elasticity of substitution (CES) function to present user agent's utility u_{ij} . If $d_i \geq e_{ij}$ and $m_i \geq b_{ij}$,

$$u_{ij} = A_i (\delta_{i1} (d_i - e_{ij})^{\rho_i} + \delta_{i2} (m_i - b_{ij})^{\rho_i})^{1/\rho_i} \quad (5)$$

otherwise, $u_{ij} = 0$. So the agents must make trade-offs between the task finish time and the cost of the execution. The satisfaction of the user agent is defined as follows:

$$\theta_i = u_i / A_i (\delta_{i1} d_i^{\rho_i} + \delta_{i2} m_i^{\rho_i})^{1/\rho_i} \quad (6)$$

In addition, we adopt the small world network to model the interaction between the user agent. We suppose user agents are randomly or regularly located in the small world network according to their initial bidding strategies. The user agent can learn the successful experiences from his neighbors. Here, the experience restricts the type and the parameter of bidding strategy, other private information of the neighborhood will not be shared. The user agent can select the strategy of his neighbors with the highest accumulated satisfaction, and the user decide whether to change his bidding strategies with a fixed probability. The accumulated satisfaction ϑ is defined in following equation and τ is the learning rate.

$$\vartheta_i(t) = (1 - \tau) * \theta_i(t) + \tau * \vartheta_i(t - 1); \quad (7)$$

Because interactions between parts of a dynamic system are the source of both complexity and emergence. In next section, we will study the evolution dynamics of this trading system.

4 Experiment Analysis

In this section, we process the experiment in two perspectives. The first one is from traditional non-cooperative perspective that means user can not learn experience from their neighbors, and the second one from cooperative perspective, but as we above-mentioned the interaction are happened locally.

4.1 Experiment Setup

In the next section, we will investigate several bidding strategies to resource advance reservation. For simplicity, we assume that all the user agent with same utility function, and $\delta_{i_1} = 1$, $\delta_{i_2} = 1$, $A_i = 1$, $\rho_i = 2$, $l_i = 10$, $t_i = 10$, $d_i = 10$, $m_i = 10$; and we also assume that $c_j = 10$, $l_j = 10$, $\lambda_j = 0.5$.

4.2 Experiment 1

In this experiment, we assume that the number of server $M = 2$, in the Table 1 and Table 2 we list the average satisfaction and the average cost of each type when the total user agent number is $N = 10, 20, 30$, represents the situation when the supply is bigger, equal, and smaller than the demands individually.

From the Table 1 and Table 2 we can see Greedy(*) and RB-Greedy(*) strategies can obtain a higher satisfaction with the lowest price through price predication.

Table 1. The Average Satisfaction in Experiment 1

Strategy	$N = 10$	$N = 20$	$N = 30$
Fixed(0)	0.6083	0.5149	0.2116
Greedy(1)	0.7168	0.5210	0.2189
Greedy(*)	0.7409	0.6505	0.2112
RB-Greedy(1)	0.6471	0.5216	0.2086
RB-Greedy(*)	0.7520	0.6492	0.2023

Table 2. The Average Cost in Experiment 1

Strategy	$N = 10$	$N = 20$	$N = 30$
Fixed(0)	8.6000	9.2000	10
Greedy(1)	3.7500	8.3250	10
Greedy(*)	1.0000	1.0000	10
RB-Greedy(1)	8.2500	8.4000	10
RB-Greedy(*)	1.0000	1.0000	10

4.3 Experiment 2

Our experiments are performed as follows. All of the agents are located on a network, and we assume that user agent could change his bidding strategy when all the auction are closed, we call it is one generation. And 10% of agents change their strategy stochastically according to their gain earned in each generation. The new strategy is generated as a copy from the agent gained the highest satisfaction in the neighborhoods.

In this experiment, we consider two distinct scenarios: sufficient and insufficient resources. When $N \leq 20$, the resources are sufficient, user agents can improve their satisfaction and decrease the trading cost even through local interaction. In this scenario, the bidding strategies will converge to RB-Greedy(*) or Greedy(*). Experiment results showed that all the resource were traded near the lowest price even the resource were sold by auction. However, when $N > 20$, the resources is insufficient, the bidding strategy will not converge. Even Fixed(0) and Greedy(1) can perform well in this scenario, the agents are bidding with the highest cost, or else they will lost the opportunity of reserve the service. Low level strategies survive after the evolution of the game.

From the experiments we study some important factors which influence the outcome of the auction. First of all, the user bidding strategies play the most important role in the auctions. However, when users begin to cooperate, besides users' bidding strategies, the users' utility functions (budget and deadline) become the critical factor which determine the final price of the auctions.

Other factors such as the winner lasting time, the minimal increment price of the auctions, and the parameters of the small world, etc., will also affect the results of the auctions. Due to the space limitation, part of the detailed discussion we demonstrated in another paper [18].

5 Conclusion and Future Work

In this paper, we explore the market-based grid resource trade system from the social perspective and the collaborative computing perspective. We firstly propose a novel framework to simulate the trading system of the grid computing resources, and then we give a case study based on this framework. In the case study, an evolutionary dynamics of the resource reservation game played by agents on the network. We study the evolution of the system with cognitive agents: Study the evolution dynamics of system with agents can automatic adapt to their environment, they can exchange private information and learn experiences from each other.

After analyzing the experiment results, we can see that the human factors such as bidding strategy, utility function, etc. and societal issues interaction method will influence the performance of the trading system. So those aspects and other socio-economics aspects such as user true requirement, intersection between the physical world and the digital, collaboration and community should be taken into future investigation when we want to design and implement the economic-based resource management systems.

References

- [1] A. Byde, C. Preist, N. R Jennings: Decision Procedures for Multiple Auctions. *Proc. of the first International Joint Conference on Autonomous Agents and Multi-Agents System*, pp. 613-620, 2002.
- [2] A. Galstyan, K. Czajkowski, and K. Lerman: Resource Allocation in the Grid Using Reinforcement Learning. *Proc. of the International Conference on Autonomous Agents and Multi-Agent Systems AAMAS-03*, New York, NY, 2003.
- [3] B. Leloup. Dynamic Pricing on Agent-based Markets with Local Interactions. *Electronic Commerce Research and Applications*, 2003.
- [4] C. Kenyon, G. Cheliotis. Grid Resource Commercialization: Economic Engineering and Delivery Scenarios. Kluwer Press, 2004.
- [5] C. Ng, P. Buonadonna, B. N. Chun, A. C. Snoeren, A. Vahdat. Addressing strategic behavior in a deployed microeconomic resource allocator. *In 3rd Workshop on the Economics of Peer to Peer Systems*, August 2005.
- [6] D. Phan: From Agent-Based Computational Economics towards Cognitive Economics, *Cognitive Economics*, Bourguine P. , Nadal J. P. eds., Springer Verlag, pp. 371-398, 2004.
- [7] G. Cheliotis, C. Kneyon and R. Buyya: 10 Lessons from Finance for Commercial Sharing of IT Resources, *Peer-to-Peer Computing: The Evolution of a Disruptive Technology*, IRM Press, Hershey, PA, USA, 2004.
- [8] H. Zhuge. The Knowledge Grid. World Scientific Publishing Co., 2004.
- [9] H. Zhuge. The Future Interconnection Environment, *IEEE Computer*, 38 (4) (2005) 27-33.
- [10] L. Tesfatsion: Agent-Based Computational Economics: Growing Economies from the Bottom Up, *Artificial Life*, Volume 8, Number 1, pp.55-82, 2002.
- [11] M. He, N. R. Jennings and A. Prugel-Bennett: An Adaptive Bidding Agent for Multiple English auctions: A Neuro-Fuzzy Approach. *Proc. IEEE Conf. on Fuzzy Systems*, Budapest, Hungary. 2004.
- [12] M. P. Wellman, W. E. Walsh, P. R. Wurman and J. K. Mackie-Mason: Auction protocols for decentralized scheduling. *Game and Economic Behavior*. 2001.
- [13] R. Buyya, D. Abramson, J. Giddy and H. Stockinger: Economic Models for Resource Management and Scheduling in Grid Computing, *The Journal of Concurrency and Computation*, Volume 14, Issue 13-15, Wiley Press, USA, 2002.
- [14] S. Milgram: The Small-World Problem; *Psychology Today*, 1 may p. 62-67. 1967.
- [15] W. Smith, I. Froster and V. Taylor: Scheduling with Advanced Reservations, *Proc. of the IPDPS Conference*, May, 2000.
- [16] W. K. Cheung, J. M. Liu: On Knowledge Grid and Intelligence: A Survey. *Computational Intelligence, Volume 21*, Number 2, 2005.
- [17] Z. X. Huang, Y. H. Qiu: A Comparison of Advance Resource Reservation Bidding Strategies in Sequential Ascending Auctions, *Web Technologies Research and Development*, Springer, 2005.
- [18] Z. X. Huang, Y. H. Qiu: Resource Reservation Game in Small World. *Fifth International Joint Conference on Autonomous Agents And Multiagent Systems (AAMAS 2006)*, *Workshop on Emergent Intelligence on Networked Agents*, May 8-12, 2006, Hakodate, Japan.