# New algorithm for resource selection in economic grid with the aim of cost optimization using learning automata

Ali Sarhadi
Computer Engineering Department
Islamic Azad University – Malayer Branch
Malayer, Iran
Sarhadi_ali@iau-malayer.ac.ir

Mohammad Reza Meybodi
Computer Engineering Department,
Amirkabir University of technology
Tehran, Iran
mmeybodi@aut.ac.ir

*Abstract*— **In economic computational grids, resources have prices and the users must pay for executing their applications. The user determines his deadline and budget and then requests cost or time optimization. A resource selection service that adopts cost optimization strategy should select heterogeneous grid resources for heterogeneous user applications so that their execution finishes in the specified deadline with minimum cost. In this paper, new algorithms based on learning automata are proposed for this purpose. Using computer simulations, it is shown that the proposed algorithms have higher performance comparing to the existing algorithm.**

***Keywords-resource selection; learning automata; cost optimization; grid computing***

## I. INTRODUCTION

Grid [5] based computational infrastructure is a promising next generation computing platform for solving large-scale resource intensive problems. It couples a wide variety of geographically distributed computational resources (such as PCs, workstations, and clusters), storage systems, data sources, databases, Computational kernels and special purpose scientific instruments and presents them as a unified integrated resource. Economic model [1] for Grid environment presented newly. It encompasses a wide range of software technologies from local operating environments (operating or queuing systems) to global resource brokers and applications that are designed to exploit Grid capability. The interactions between these Components must be secure and adapt to the changing resource status. Internationally, there are many projects [3][6] actively exploring the design and development of different Grid system components and services for secure execution of applications on wide-area resources. One of the main problems in resource management is effective resource selection. In this paper we identify a new grid resource selection algorithm based on economic criteria using learning automata. Users (resource providers and consumers) in the Grid economy try to maximize his or her benefit. Various resources discover and selection algorithms have different policy such as system centric or user centric. We briefly discuss popular economic policy for resource trading and present new algorithm. It is shown that the proposed algorithms have higher performance comparing to the existing algorithms.

## II. LEARNING AUTOMATA

Learning Automata [4] are adaptive decision-making devices operating on unknown random environments. A Learning Automaton has a finite set of actions and each action has a certain probability (unknown to the automaton) of getting rewarded by the environment of the automaton. The aim is to learn to choose the optimal action through repeated interaction on the system. If the learning algorithm is chosen properly, then the iterative process of interacting on the environment can be made to result in selection of the optimal action. Figure 1 illustrates how a stochastic automaton works in feedback connection with a random environment. Learning Automata can be classified into two main families: fixed structure learning automata and variable structure learning automata (VSLA) [8]. In the following, the variable structure learning automata which will be used in this paper is described.

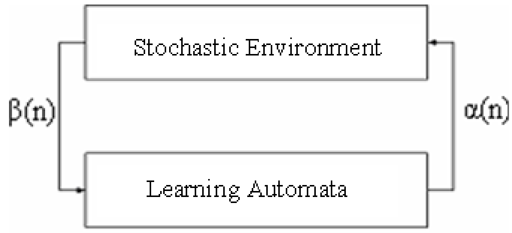


Figure 1. The interaction between learning automata and environment

A VSLA is a quintuple < α, β, p, T(α,β,p) >, where α, β, p are an action set with s actions, an environment response set and the probability set p containing s probabilities, each being the probability of performing every action in the current internal automaton state, respectively. If the response of the environment takes binary values learning automata model is P-model and if it takes finite output set with more than two elements that take values in the interval [0,1], such a model is referred to as Q-model, and when the output of the environment is a continuous variable in the interval [0,1], it is refer to as S-model. The function of T is the reinforcement algorithm, which modifies the action probability vector p with respect to the performed action and received response. Assume $\beta \in [0,1]$. A general linear schema for updating action probabilities can be represented as follows. Let action i be performed then:

A) Desire response

$$\begin{aligned} p_i(n+1) &= p_i(n) + a[1-p_i(n)] \\ p_j(n+1) &= (1-a)p_j(n) \qquad \forall j \quad j \neq i \end{aligned} \tag{1}$$

B) Undesired response

$$p_i(n+1) = (1-b)p_i(n)$$
$$p_j(n+1) = \frac{b}{r-1} + (1-b)p_j(n) \quad \forall j \quad j \neq i \tag{2}$$

Where a and b are reward and penalty parameters. When a=b, the automaton is called $L_{RP}$. If b=0 the automaton is called $L_{RI}$ and if 0<b<<a<1, the automaton is called $L_{R\varepsilon P}$. For more Information about learning automata the reader may refer to [4].

## III. ECONOMIC RESOURCE MANAGEMENT

In economic Grid environments, the producers (resource owners) and consumers (resource users) have different goals, objectives and strategies [1]. The most commonly used approaches for managing such complex environments are driven by *system-centric* and *user-centric* policies. System centric is a traditional approach to resource management that attempts to optimize system-wide measure of performance and User-centric approaches, on the other hand, concentrate on delivering maximum utility to the users of the system based on their QoS requirements, a guarantee of certain levels of performance based on the attributes that the user finds important such as the deadline by which his jobs have to be completed.

Enforcing QoS requires a system of rewards and penalties and, hence, it is common to find user-centric approaches driven by economic principles.

## IV. RESOURCE DISCOVERY

Resource discovery is one of the main functions of resource management for scheduling of application in grid environment.The grid resource discovery problem [7] can be defined as the problem of matching a query for resources, described in terms of required characteristics, to a set of resources that meet the expressed requirements. The problem is complicated by the fact that some resource information (e.g., CPU load or available storage) changes dynamically. Resource discovery techniques maintain the resource attribute and status information in a distributed database and differ in the way they update, organize, or maintain the distributed database. The challenge is to devise highly distributed discovery techniques that are fault tolerant and highly scalable.

Matching the needs of an application with available resources is one of the basic and key aspects of a Grid system. Resource Discovery is systematic process of determining which grid resources are the best candidate to complete a job with following trade-offs.

- In shortest amount of time.
- With most efficient use of resources.
- At minimum cost

The output of this phase of scheduling is set of resource that meets minimum requirement of users and in resource selection phase one of this resources with highest match with users application requirement will be selected.

## V. ECONOMIC RESOURCE SELECTION

The resource selection algorithm is responsible for selecting those resources that meet the deadline and budget constraints along with optimization requirements. Various resource selection mechanisms have been proposed. In this paper we used of new mechanism for selecting resource with economic criteria based on learning automata. In this section we briefly introduced our algorithm.

We need to describe how the learning automata's select resources. As the name learning suggests, automata's use their past experience to choose between the resources. In pervious section we introduce learning automata. in our model for each possible action (i.e., selecting a specific resource) the LA keeps environment response that indicates the efficiency of that resource in the past and based on this response give reward or penalty to this action. Qualifications for grant of reward or penalty will be introduced in algorithm. After each selection, the LA gets a response from environment (containing the price of resource and completion time), calculates the complete time, and translates it into reward r or penalty p for resource number i. For each new user, LA's chooses a resource with the highest reward. A resource with minimum cost that execute users job in specified deadline has a higher reward and will be selected.

```
Algorithm:
CO_LA_ RS()
1- Sorting the received requests in descending
 order in terms of the duration the pertinent
 resource is required
2- The beginning of the learning procedure
 in the learning automata connected to the
 customers
    2-1: reiterate them 5000 times
        2-1-1 Deplete the assignment queue
             of each Vendor
        2-1-2 Conduct the following stages
              for each buyer
              2-1-2-1 Opt for the germane vendor of
                      each purchaser and set the
                      purchase in the designation
                      queue of the pertinent vendor
        2-1-3 The beginning of the fining procedure:
              Inflict financial penalty on each
              purchaser who has opted for a resource
              that does not fulfill the temporal
              limitation
        2-1-4 The stage of the reward-
              proffering phase:
              If a purchaser is not fined and chooses a
              vendor who is identical or cheaper than
              the previous one, reward him or her with
              0.02 rating.
3. each buyer will be registered to the most appropriate
 seller subsequent to the convergence of results and the
completion of reiterations.
```

Figure 2. Proffered algorithm for resource selection based on economic criteria using learning automata.

## VI. EXISTING ALGORITHM

In this section existing algorithm will be introduced and in next section performance of presented algorithm and existing instance will be compared.

Buyya cost optimization algorithm (BCO_RS) [6] completes application processing by the deadline and minimizes the computational cost.

BCO_RS()
1. RESOURCE TRADING: Identify cost of each of the resources in terms of CPU cost per second and capability to be delivered per Cost-unit.
2. SORT resources by increasing order of Cost.
3. SELECTION: Repeat while there exist unprocessed jobs in application job list with a delay of scheduling event period or occurrence of an event AND the time and process expenses are
within deadline and budget limits:
[SELECTION ADVISOR with Policy]
a. For each resource predict and establish the job consumption rate or the available resource share through measure and Extrapolation.
b. For each resource based on its job consumption rate or available resource share, predict and establish the number of jobs a Resource can process by the deadline.
c. For each resource in order:
i. If the number of jobs currently assigned to a resource is less than the predicted number of jobs that a resource can consume, assign more jobs from unassigned job queue or from the most expensive machines based on job State and feasibility. Assign job to a resource Only when there are enough budgets Available.
ii. Alternatively, if a resource has more jobs than it can complete by the deadline, move Those extra jobs to unassigned job queue.
6. [DISPATCHER with Policy]
Repeat the following steps for each resource if it has jobs to be dispatched:
1. Identify the number of jobs that can be Submitted without overloading the resource. Our default policy is to dispatch jobs as long as the number of user jobs deployed (active or in queue) is less than the number of PEs in the resource.

Existing algorithmm for resource selection based on economic criteria

## VII. GRIDSIM TOLKIT FOR SIMULATION

In order to prove the effectiveness of presented resource selection algorithm and associated selecting algorithms, their performance needs to be evaluated under different scenarios such as varying the number of resources and users with different requirements. We have used a Java-based discrete-event Grid simulation toolkit called GridSim [2]. This toolkit supports modeling and simulation of heterogeneous Grid resources, users, brokers, and application models. With GridSim toolkit we evaluate the performance of presented algorithms with exist sample through a series of simulations.

## VIII. PERFORMANCE EVALUATION

To performance evaluation of presented algorithm and compare it with existing instance we classified grid resource in five classes. Each resource class is defined by a set of attributes which specify its characteristics. Each resource has a specific value for each attribute defined by the corresponding resource class.

An example of resource class is “computing resource” that defines the common characteristics of computing resources. These characteristics are described by attributes such as “OS name”, “CPU speed”, and “Free memory”. An instance of the “computing resource” class has a specific value for each attribute, for example, “OS name = Linux”, “CPU speed = 1000MHz”, and “Free memory = 1024MB”.

We employ presented algorithm and existing instance to select sightly resource for five number of different type applications that shown in table 2 and resolute presented.

## IX. RESOURCE CLASSIFICATION AND DESCRIPTION

| Resource class | Description | Mean price (G$/sec) |
|---|---|---|
| Computing resource | Computing capabilities provided by computers, clusters of computers, etc. | 2.7 |
| Storage resource | Storage space such as disks, external memory etc. | 1.4 |
| Device resource | Specific devices such as instruments, sensors, etc. | 1.8 |
| Software resource | Operating systems, software packages, Web service, etc. | 2.1 |
| Data resource | Various kinds of data stored in file systems or databases. | 2.2 |

## X. DIFFERENT TYPE OF APPLICATION

| Application | Description |
|---|---|
| MATLAB_app | This application need to most of computing resource, slightly software resource and partly Data resource. |
| QUERY_dat | This application need to most of Data resource, slightly Storage resource, partly Software resource and Computing resource. |
| NQW_net | This application need to most of Device resource, slightly Computing resource. |
| GCC_comp | This application need to most of Software resource and slightly Computing resource. |
| ASTRO_app | This application need to most of computing resource, slightly Device resource and partly Storage resource. |

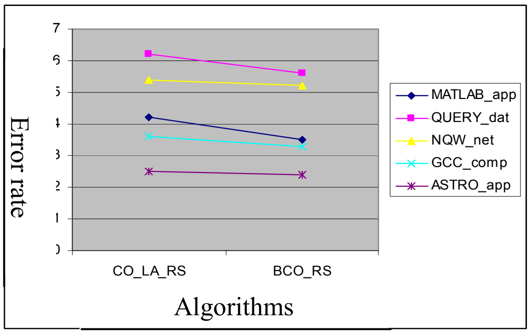


Figure 3. Error rate comparison for different type of application

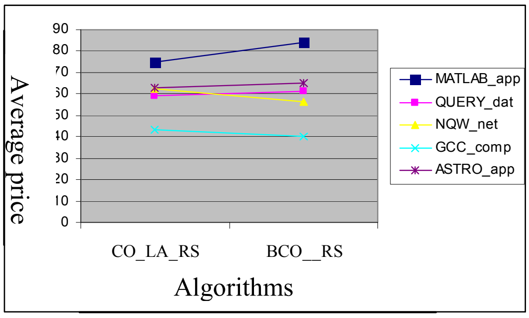


Figure 4. Average price required comparison for different type of application

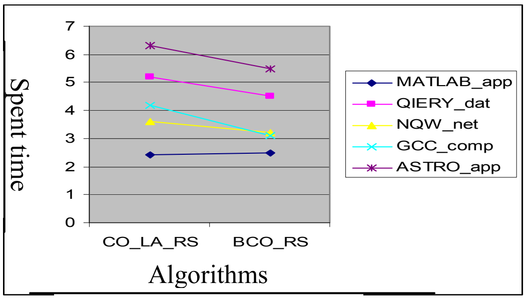


Figure 5. Comparison spent time comparison for different type of application

(Times is in terms of milliseconds)

## XI. EVALUATION PARAMETERS AND RESOULT

The most suitable options for algorithm evaluation have been propounded with regard to the effectuated securitizations. The error rate parameter are regulated with regard to each selection for instance in failure cases of choices, or in cases demanding re-selection or ignorance of customer's budget limitations. the spent time to each request that is to say the duration required for the completion of selection process with considered constraint. Another parameter is the average price determine average cost that consumer have to pay for purchase of selected resource for his or her application. Other parameters have taken to be constant.

## XII. EXPERIMENTAL RESULT

As the figures show average error rate for selecting resource in BCO_RS algorithm less than proffered algorithm because used mechanism based on learning automata in proffered algorithm to select desire resource using high repeat. And this operation cause increase of error rate. But average price required for purchase of desire resource is less than existing algorithm. Average spent time for selecting of desire resource in some of application is higher in proffered algorithm in comparison with existing algorithm. Because purpose of our algorithm is cost optimization we can partly waiver from other parameter.

## XIII. CONCLUSION

In this paper, we extend the traditional resource selection mechanism. Proffered algorithm for cost optimization is based on economic criteria. In our model we use learning automata for selecting resource with best qualification of price. Actually, the Problem's convexity characteristic is equipping each application of customers to learning automata. Moreover, the proposed concepts and algorithms can be readily put for different type of applications.

## ACKNOWLEDGMENT


This paper is supported by the Islamic Azad University-Malayer branch department of computer engineering 08514498765.


## REFERENCES


[1] R. Buyya, D. Abramson, and J. Giddy, *"An Economy Driven Resource Management Architecture for Global Computational Power Grids"*, Proceedings of the 2000 International Conference on Parallel and Distributed Processing Techniques and Applications, June 2000.

[2] R. Buyya and M. Murshed, *GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing*, Technical Report, Monash University, Nov. 2001. To appear in the Journal of Concurrency and Computation: Practice and Experience (CCPE), 1-32pp, Wiley Press, May 2002.

[3] Baker M., Buyya R., Laforenza D., *The Grid: International Efforts in Global Computing,* Intl. Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet (SSGRR 2000), Italy, 2000.

[4] K. Narendra and M. A. L. Thathachar, *Learning Automata: An Introduction*, Prentice Hall, Englewood Cliffs, New Jersey, 1989.

[5] Foster, I., and Kesselman, C. (editors), *The Grid:Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, USA, 1999.

[6] M. Baker, R. Buyya, and D. Laforenza, Economic-based Distributed Resource Management and Scheduling for Grid Computing, International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet (SSGRR 2000), l Aquila, Rome, Italy, July 31 - August 6. 2000.

[7] Iamnitchi and I. Foster, "On Fully Decentralized Resource Discovery in Grid Environments," IEEE International Workshop on Grid Computing,Denver, CO, 2001.

[8] K. Najim and A. S. Poznyak, *Learning Automata: Theory and Application*, Elsevier Science Ltd., Tarrytown, NY, 1994.