International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:2, 2009

Proffering a Brand New Methodology to Resource Discovery in Grid based on Economic Criteria Using Learning Automata

Ali Sarhadi, Mohammad Reza Meybodi, Ali Yousefi

Abstract**—**Resource discovery is one of the chief services of a grid. A new approach to discover the provenances in grid through learning automata has been propounded in this article. The objective of the aforementioned resource-discovery service is to select the resource based upon the user's applications and the mercantile yardsticks that is to say opting for an originator which can accomplish the user's tasks in the most economic manner. This novel service is submitted in two phases. We proffered an application-based categorization by means of an intelligent nerve-prone plexus. The user in question sets his or her application as the input vector of the nerve-prone nexus. The output vector of the aforesaid network limns the appropriateness of any one of the resource for the presented executive procedure. The most scrimping option out of those put forward in the previous stage which can be coped with to fulfill the task in question is picked out. Te resource choice is carried out by means of the presented algorithm based upon the learning automata.

Keywords**—**Resource discovery, learning automata, neural network, economic policy.

I. INTRODUCTION

COMPUTATIONAL grids[1-2] are among the recent approaches taken advantage of to solve the scientific, engineering and commercial predicaments on a ginormous scale. They arrange groundwork to share and integrate millions of resource which are geographically sporadic among dissimilar management scopes and organizations. Such grids comprise a set of inhomogeneous originators (personal computers, work stations, clusters and mainframes), infrastructural management systems (the operating system of the unit, the queue system, and so on and so forth), applied policies and programs (scientific, engineering and business) with disparate requirements (processors, input, output, memory and network). Grid contraptions and resource are proffered to users through the pertinent services. Discovery-resource is one of the services which users utilize directly. The aforementioned entity delves into a grid resource and selects the originator which the user has in mind. Diverse resource discovery services average availed in disparate grid projects conducted all over the world for instance iGrid [8] or P2p-based information service[9] , which average plied in European grid projects nonetheless the above discovery-resource services have 2 setbacks. They do not support application oriented queries that is to say users have to enter the specifications of the resource in query as search parameters. Therefore users themselves have to pinpoint which resource matches the applications they have in mind. The above human recognition procedure can't be ameliorated owing to the dynamic and inhomogeneous nature of the resource. The subsequent feeble point concerns the unsupported economic policies to ferret out the originator namely.

No policies have been propped up when we enter the economic grid [3] environment while users have budget limitations in opting for the resource in question. The only achieved task pertains to the economic resource [4-7] manipulation policies set into effect by dynamic a entity which comes true in the resource scheduling stage which eventuates after the resource discovery. Some efforts have been made to obliterate the weak points by presentation of a new methodology for resource discovery in grids.

A new architectural plan has been propounded discern the resource in the grid through intelligent neural network which support applied queries. Then tow available algorithms are utilized to resource discovery. We presented an algorithm based upon the learning automata eventually so that the resource is selected with economic criteria. These proposed algorithms have been simulated by means of GridSim [5-14] Toolkit. Their application has been delved into by the same toolkit. The rest of the article has been organized in this manner. Intelligent neural network and the learning automata are respectively presented in sections 2 and 3. The architectural features availed to pinpoint the extant resource are appraised in section 4. The new resource-discovering architecture based upon the intelligent neural network is proffered in section 5. The resource selection algorithm based upon the learning automata is presented in section 6. The simulation and the collation of results are propounded eventually.

II. INTELLIGENT NEURAL NETWORK

Intelligent neural networks [13] are among dynamic systems. They process empirical data and convey the scientific laws beyond them to the network structure. That is why such systems are named intelligent. They grasp generic rules based

This work was supported by Islamic Azad University - malayer branch. Ali sarhadi is with Islamic Azad University, malayer 2227878, Iran (phone: +98 918 311 3725; e-mail: sarhadi_ali@yahoo.com).
Mohammad Reza Meybodi is with the Computer Engineering Department, Amirkabir University of Technology Tehran 15875-4413, Iran (e-mail: mmeybodi@aut.ac.ir).
Ali yousefi , Islamic azad university –hamadan branch (phone: +98 918 811 4133; email: Ila.yousefi@gmail.com)

upon calculating the numerical data or exemplifications. Such systems make a blueprint of the neural network structure of the human brain through computational intelligence. According to Segal hypothesis human brain is made up of some elements known as neurons. Nervous plexuses are actually a collection of neurons which are active in a parallel manner.

A. The structure of the neural network

Type of Mono-layer neural network has been illustrated in figure 1. The input and output of the network have been respectively shown with p and a vector[16]. Each one of the input is connected to all the neurons in this structure. W Matrix has S rows and R columns.

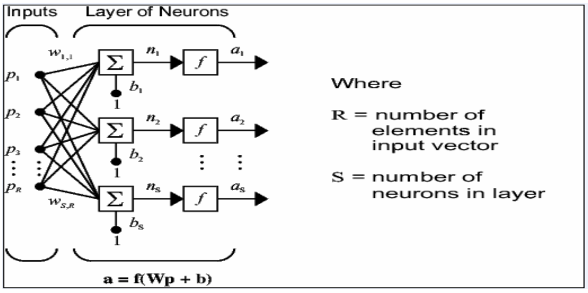

Fig. 1 Monolayer neural network

B. The learning rules of the neural network

One of the features of a neuron is the capability of learning. Neuron perimeter is taken to be constant in learning purposes nonetheless a network may alter and improve its comportment on a shared basis with a neuron. Each neuron may change the matching vectors in accordance with the learning rule.

1) Supervised algorithms

There is a collection of paired data in supervised algorithms called the learning data. When X input is fed into the output of the neural network which is a result of Y Network, it will be collated with ideal quantity. Then the error is calculated to regulate nexus parameters such as W in a manner that the next time that X input is fed into the same X input, the plexus output will be converged to Y.

2) Unsupervised learning algorithms

The vector of the optimum response does not input the network in unsupervised learning procedure. When the number of layers and neurons escalates, the learning procedure decelerates in supervised algorithms. This is the key defect of such algorithms. The resultant response of unsupervised algorithms is stored in long-term memory and the outputs are classified right from the beginning. A relationship is established among them through shared memory.

III. Learning Automata

A learning automata [11] is a machine with finite state which is capable of performing a limited number of tasks. Each selected task is appraised in a probable environment and a response is rendered to the learning automata. The learning automata utilize the aforementioned response to opt for the subsequent action. The learning automata entity learns how to pick out the best action out of the permitted ones. Figure 2 illustrates the relationship between the learning automata and the environment this environment can be limned by the ensuing tripartite entity: $E \equiv \{\alpha, \beta, c\}$ that $\alpha = \{\alpha_1, ..., \alpha_r\}$ which is a collection of inputs. $\beta = \{\beta_1, ..., \beta_m\}$ This is a set of outputs. $c = \{c_1, ..., c_r\}$ This is a collection of penalty probabilities. Each time β is a bipartite collection; the environment will be of P type. Thus $\beta_1 = 1$ known as the penalty and $\beta_2 = 0$ is a reward. In a Q environment $\beta(n)$ may vary within a restricted range of [0, 1] and in a S environment may have any quantity within [0, 1]. c_i Pertains to the unwanted result of a1 task. α_i Quantities remain unchangeable in static environment during a period while the aforementioned quantities alter in a volatile environment

The learning automata are categorized into two groups: those having a fixed structure and those bearing a variable structure. The learning automata with variable structures have been utilized in this article which will be briefly explicated hereinafter.

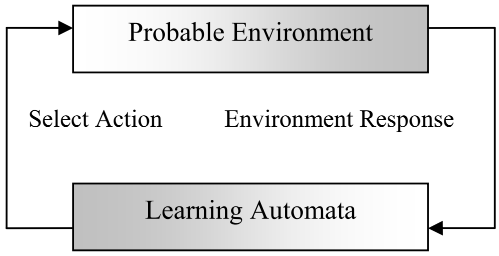

Fig. 2 Relation between learning automata and environment

A. The learning automata with volatile structures

Learning automata with variable structures are limned by the quadric-partite entity $\{\alpha, \beta, p, T\}$ where $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ is a collection of the automata tasks. $\beta = \{\beta_1, \beta_2, \ldots, \beta_m\}$ is a set of the automata inputs. $p = \{p_1, p_2, \ldots, p_m\}$ is the probability vector denoting the possible adoption of any of the tasks. $p(n+1) = T[\alpha(n), \beta(n), p(n)]$ is the learning algorithm. If at action is rendered during the nth stage in this type of the learning automata to receive optimum response from the environment, Pi(n) probability diminishes while other possibilities increase. Alterations are effectuated in a manner that Pi(n) sum will always be a constant equal to 1. the ensuing algorithm is a specimen of the linear learning algorithms:

Desire response (A

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)]$$
$$p_j(n+1) = (1-a)p_j(n) \qquad \forall j \quad j \neq i$$

Undesired response (B

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:2, 2009

$$p_i(n+1) = (1-b)\,p_i(n)$$

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

a is the reward parameter and b is the penalty parameter. Three modes can be conceived with regard to a and b quantities. When a and b are equal, the pertinent algorithm will be called L_{rp}. When b is so smaller than a, the germane algorithm will be named L_{rep}. If b is equal to zero, the pertinent algorithm will be called L_{ri}. For further studying of the learning automats and the pertinent applications you can refer to [6-10].

IV. Resource Discovery by MDS Architecture

Diverse resource may be shared by organizations and individuals in the grid. Grid users have little information about the pertinent resource. Therefore, they have limited efficacious usage of the resource in question. Grid information services have been devised to support searches, resource-discovery [17] and supervision of vital grid entia. There are various architectural plans for grid information services. MDS2 is the most sophisticated one.

The emergence of MDS2 discarded MDS1, the previous service, which used to be a pioneer in grid information services because it did not manage to fulfill the needs of the grid information services. MDS2 has a section known as grid resource information services and another part called grid indexed information service. Both of these parts can be configured. The aforementioned architecture is depicted in figure 3.

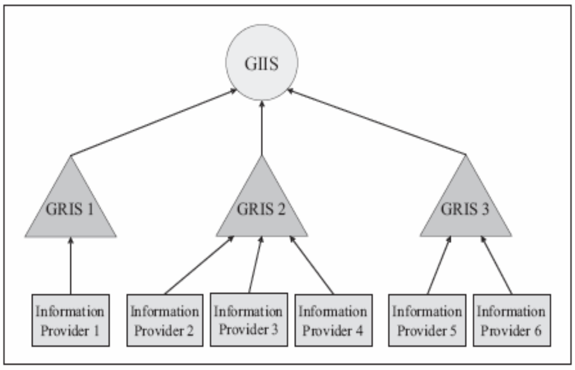

Fig. 3 MDS architecture

As illustrated in the above figure, users pan for collective directories marked with letter D by dispatching the seeking terms through the search protocols or they may look through the data providers marked with letter F and maintain the resource data nonetheless users have to enter the basic features of the resource as search parameter. The above procedure may be futile owing to the dynamic nature of resource and users' ignorance as regards them.

V. Architecture of the Resource Discovery Service Based on Intelligent Neural Network

Ere we proffer a new architecture based upon the intelligent neural network to pinpoint the resource discovery service in grid with regard to the cited quandaries of the resource discovery service. The neural network has been opted for on account of the fact that their efficiency is much higher than other similar techniques in classification predicaments based upon the obtained conclusions. The core of this service is a grid resource classifier based upon the intelligent neural network which access Meta directory service on a periodic basis. It also categorizes grid resource in terms of the diverse propounded applications within the computational grid environment. Users can summon the aforementioned services to pinpoint the applied point in query as the input search parameter instead of the specifications of the resource in query to select the best resource. Meta directory service reflects the situation of the computational grid environment and all the other information pertinent to the available entity in the grid. The aforementioned Meta directory service does not support application oriented query that is to say the selection of the resource in the user's mind with regard to the pragmatic aspect that he or she has set for it. The above defect is removed by application of the aforementioned architecture. Users can determine the particular practical aspect as the input vector of the neural network. The output vector of the neural network evinces the possibility of appropriateness of each one of the resource for the application in query.

For instance the output vector is formulated based upon the ensuing relationship based upon the information obtained from a particular resource.

T_k = (0.1, 0.1, 0.9… 0.1)

This output denotes the fact that the 3rd resource is more appropriate for this application than other ones.

A. Architecture of the application oriented resource discovery and selection based on neural network and learning automata

The architecture of the service utilized for application oriented resource discovery and selection based on neural network and learning automata has been illustrated in figure 4.

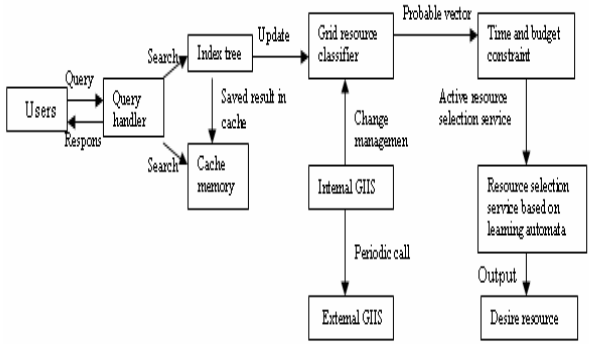

Fig. 4 Application oriented architecture for resource discovery

As depicted in the above slide, the aforementioned service is composed of components:

1) Internal GIIS

The operator of a grid resource discovery is an applied one. It summons extensive GIIS services on a periodical basis to obtain the situation of the grid environment and to become aware of any type of alterations in the resource situation.

2) Grid resource classifier

Grid resource classifier is a chief component in application oriented grid resource discovery services. It can implement applied classifications of the resource based upon the data received from grid resource through an intelligent neural network. GRC design will be expounded in the subsequent section. The above component effectuates the main task of the resource discovery service.

3) Index tree

This is a location where the conclusions produced by GRC are stored. The classification of the results from the grid resource is a pair amount. The identification key of the resource is in grid and specifies the predefined amount of the applied classification. The index tree is utilized to save such conclusions

4) Cache

There are plenty of resources in the grid environment. Thus we devise a cache to save the results of the last searches to enhance the efficiency of the search. The search manipulator guarantees the matching of cache with the index tree.

5) Query handler

The duty of this component is the processing of the users' query. The query handler looks through cache at first. If it comes across the identifiers of the important resource of the use in cache, the query handler will delete all the counters in this resource. Then it checks the compatibility of the cache contents and the index tree. Then the identifier of the resource in question is rendered to the user, otherwise the search runner looks through the index tree and updates the cache based upon the new upshot obtained from the index tree and returns it to the user afterwards. The aforementioned amount is rewritten in cache and the pertinent counter is input as zero.

VI. Resource Selection Aiming to Cost Optimization

Some resource was discovered for the task in query in the previous phase. The output vector of the neural network evinces the resource which can accomplish the pertinent task with varying qualification probabilities. One of the resources is picked out for the implementation of the specific application. Since we are located in the economic grid, resource owners demand money in return for presentation of their resource. Users also have to disburse specific amounts for the accomplishment of their tasks. Each one is trying to maximum his or her own profit. Thus users proffer their bids based upon the resource qualification probability vector which emerges in the output of the neural network. The bids of the users are set in the queue of each resource. Users proffer a mechanism to minimize the expenditures which is based upon the learning automata. Te set of learning automata pertinent to the purchasers is utilized. These learning automata have variable structures. They choose the vendors picked out by the customers. When all the purchasers opt for their preferred vendors, they will be rewarded positively or negatively in terms of the choice result appraisal. Purchasers do not do this task simultaneously however they pick out their vendors in accordance with the priority of being set in the queues. The observation of the aforementioned order is crucial because it affects the environmental response. Purchasers elect the vendors again. The above task is reiterated for sufficient times so that all the customers opt for their appropriate vendors. At last each buyer chooses his or her final vendor. The purchasers are sorted based upon the duration they require the resource to diminish the inactive time of vendors who have allotted high sums for their resource.

Contrary to the heuristic algorithms, this algorithm can't be finalized by one-time checking of purchasers and vendors but abundant reiterations are effectuated and a new choice is adopted in each iteration. The automata utilize the environment's response in their subsequent choices and approximate it ameliorates option. This trend keeps on until all the learning automata are converged and reiterations result in the smooth choices. This identical choice is the eventual one which purchasers utilize to opt for their vendors.

The choices have to be rendered to the vendors on a fleeting basis in each reiteration so that they can be appraised. Thus each vendor accesses an impermanent queue called the designation queue which is void at the beginning of each reiteration. As purchasers opt for vendors, they will be set in the queue of the pertinent seller. Hence 2 points are taken into account in the environment:

-Each purchaser must opt for the vendor who can obliterate temporal limitations.

-Each purchaser tried to pick out the cheapest possible seller.

Considering the above two points, this algorithm has been devised to obtain the best choice in each reiteration in positive and negative reward-bearing situations.

1. If a purchase opts for a vendor who is incapable of fulfilling the imminent requirements of the customer, the purchaser will have to undergo financial penalties.

2. If a purchaser favors a vendor who is cheaper than the previous seller, the customer will be rewarded.

Purchasers must set themselves in the designation queue of the pertinent vendors subsequent to the vendor's selection through the learning automata so that they can be positively or negatively rewarded by the environment. The aforementioned designation queue is a fleeting one. Each time a reiteration is made the pertinent queue is depleted. Such queues are examined in the environment. Then the germane buyers will be awarded positively or negatively.

The selected algorithm comprises both phases of the resource discovery and adoption.

A. Reviewing the reported algorithms and the propounded one

BCO Algorithm: a designation queue is allocated to each resource in this algorithm. The customers who have requested the resource in question are set in the designation queue in an ascending order. The purchasers are registered in the resource based upon the in an ascending order from the cheapest one and the number of customers that each resource can fulfill. These customers are deleted from the designation queue and the resources are apportioned to them in turns. This trend keeps on until the next round of bidding.

B. Blueprint propounded algorithm

The blueprint propounded in [4] is another algorithm which spots resource with the thrifty yardsticks and is based upon the bargaining paradigm sans collaboration and hierarchy cellular searches. The resource specifications and the vendor's negotiators are set in each data node. The purchase request of each resource has a negotiator for the resource customer. At first the panning is conducted in terms of non-economic benchmarks. If the panning proves to be successful, the negotiators of the purchaser and the vendor take action to set the price. Each one tries to maximize his or her own profit. If they reach an agreement, the resource will be rendered to the customer based upon the acquiesced price. If they fail, panning keeps on coming across another resource. Cellular hierarchy methodology has been utilized here to look through resource bearing non-economic touchstones.

C. The propounded algorithm based upon the neural network and the learning automata (LA_AND_NEURAL_CO)

This algorithm has been illustrated in below:

a) Resource discovery phase:

The training samples of the grid resource have been propounded here. Each one is a pair of <x, t>. X is the input vector of the plexus. T is the target output vector.

η is the learning rate and is equal to a numerical constant. It can be obtained in terms of the grid resource specifications for instance **η** = 0.05

$\mathbf{n_{in}}$ is the input vector of the network or the identical grid resource

$\mathbf{n_{out}}$ is the output vector of the network which is equal to the units' quantity in the outer layer or the same applied classification of the resource.

$\mathbf{n_{hidden}}$ is the number of the units in the hidden layer which we regarded as 3.

$\mathbf{X_{ji}}$ is the input from J unit to I unit

$\mathbf{W_{ji}}$ is the weight from I unit to J one.

1) The establishment of a fed forward network with $\mathbf{n_{in}}$ inputs, $\mathbf{n_{hidden}}$ number of units, and $\mathbf{n_{out}}$ number of external units

2) Initialize of the network weight data with random numbers within a range of – 0.05 to 0.05.

3) Repeat until the termination of the conditions:

4) ensue for each educational specimen <x, t>

4-1) Enter a sample X vector into the network. Calculate the Q_u output for each U unit

4-2) Figure out the δ_k error coefficient of each output unit K with the following formula:

$$\delta_k \leftarrow o_k(1-o_k)(t_k-o_k)$$

4-3) Figure out the δ_h error coefficient of each Hidden unit H with the following formula:

$$\delta_h \leftarrow o_k(1-o_k)\sum_{k\in outputs} w_{kh}\delta_k$$

4-4) Update each weight of the network with the Ensuing formula:

$$w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$$

4-5) W_{ij} quantity can be specified by means of the Ensuing formula:

$$\Delta w_{ji} = \eta\delta_j x_{ji} + \alpha\Delta w_{ji}(n-1)$$

b) Regulation of the cost parameter based upon the output vector of the neural network for the resource in question

c) Regulating the time parameter with regard to the application in question

d) The resource selection phase:

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 commencement 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

VII. EVALUATION PARAMETER

The most suitable options for algorithm appraisal have been propounded with regard to the effectuated scrutinizations. The error rate parameter expenditure improvement algorithms 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 and the response time to each request that is to say the duration required for the completion of the application. Another parameter is the registration duration as compared with other propounded algorithms namely the duration required by the purchaser to opt for a vendor. Other parameters have taken to be constant.

VIII. Performance Evaluation

The proffered algorithm selects and discovers resource through the learning automata**.** These algorithms have been implemented by means of GridSim simulation software. The conclusions of this algorithm have been collated with other extant algorithms based upon the cited parameters. These results are a mean of 20 times of simulations. The simulation environment has been regarded on a homogenous and inhomogeneous basis.(timing is in terms of milliseconds)

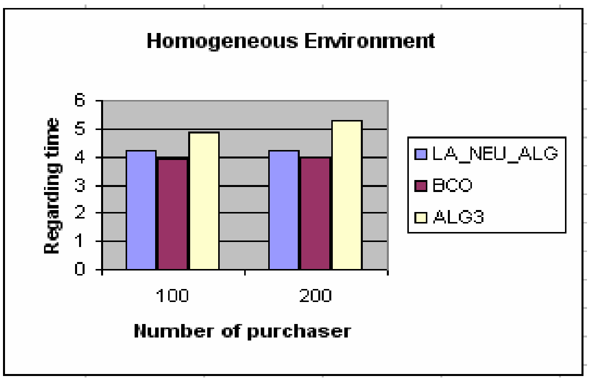

Fig. 6 Comparing the regarding time of the propounded algorithm and other algorithms of a homogenous environment

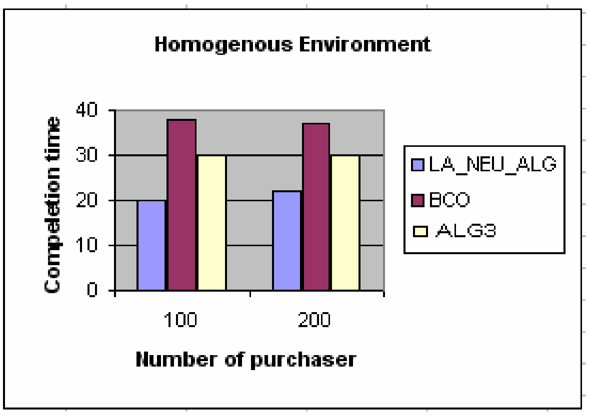

Fig. 7 Comparing the completion time of the propounded algorithm and other algorithms of a homogenous environment

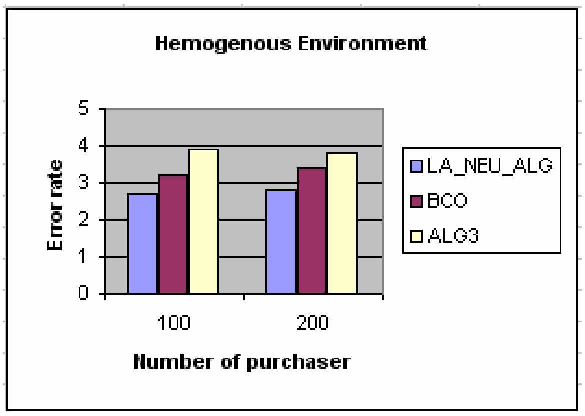

Fig. 8 comparing the error rating comparison of the propounded algorithm and other algorithms of a homogenous environment

IX. Conclusion

Resource discovery and selection is one of the phases for scheduling tasks in the computational grids. Hence in case efficient mechanisms and algorithms are utilized, the grid application will be enhanced. We presented a new approach for the resource discovery quandary in this article which is based upon the intelligent nervous plexuses. It was put forward for the resource selection services with thrifty touchstones. We proffered a brand new algorithm based upon the learning automata utilized by users. We collated it with 2 other extant specimens. Having considered the propounded parameters that is to say the registration duration, the task completion timing, and the error rating, we concluded that the registration process of the propounded algorithm based upon the learning automata takes further time because the learning process of the registration enhancement lasts longer. The average timing required for the completion of tasks in this algorithm is less than other ones because an appropriate resource is found during the learning process. Therefore, the error rate in this algorithm is drastically and acceptably less than other ones.

References

[1] I. Foster, C. Kesselman and S. Tuecke, "The anatomy of the Grid: Enabling scalable virtual organizations", *International Journal of Supercomputer Applications*, 2001.

[2] Viktors Berstis, "*Fundamentals of Grid Computing*", IBM Redbook series, November 2002, http://ibm.com/redbooks.

[3] R. Buyya, D. Abramson, and J. Giddy, "A Case for Economy Grid Architecture for Service-Oriented Grid Computing", *Proceedings of the 10th IEEE International Heterogeneous Computing Workshop*, April 2001.

[4] 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: Practice and Experience*, May 2002.

[5] M. Murshed, R. Buyya, and D. Abramson, "GridSim: A Portable and Scalable Toolkit for Modeling and Simulation of Application Scheduling for Parallel and Distributed Computing", Technical Report, Monash University, Oct. 2001. Available at: http://www.buyya.com/gridsim/.

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

[7] S. Lalis and A. Karipidis, "An Open Market-Based Framework for Distributed Computing over the Internet", *Proceedings of the First*

IEEE/ACM International Workshop on Grid Computing (GRID 2000), Bangalore, India, Springer Verlag Press, Germany, December 2000.

[8] Maxine Brown, Thomas DeFanti, Michael McRobbie, Alan Verlo, Dana Plepys, Donald McMullen, Karen Adams, Jason Leigh, Andrew Johnson, Ian Foster, Carl Kesselman, Andrew Schmidt, Steven Goldstein: The International Grid (iGrid): Empowering Global Research Community Networking Using High Performance International Internet Services. *Proceedings of INET*, 2003, pp. 3-9.

[9] Puppin, D., Moncelli S., Baraglia, R., Tonellotto, N.,Silvestri, F.: A peer-to-peer information service for the grid. In: Proceedings of the GridNets 2004 Workshop, San Jos´e, CA (2004)

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

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

[12] A. Galstyan, K. Czajkowski, and K. Lerman, "Resource allocation in the grid using reinforcement learning", *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'04)*, pp. 1314-1315, 2004.

[13] Carsten Peterson and Thorsteinn Rognvaldsson, *An Introduction to Artificial Neural Networks*, LU TP 91-23, September 1991 (Lectures given at the *1991 Cern School of Computing, Sweden*)

[14] R. Buyya and M. Murshed, "GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing", Journal of Concurrency and Computation: Practice and Experience, pp. 1-32, May 2002.

[15] 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: Practice and Experience*, May 2002.

[16] Rummelhart, D. E.; Hinton, G. E.; Williams, R. J., (2004).Learning representations by back-propagation errors. ature, 323, 533-536.

[17] S. Castano, A. Ferrara, S. Montanelli and G. Racca, “Matching Techniques for Resource Discovery in Distributed Systems Using Heterogeneous Ontology Descriptions”. Proc. Int. Conf. on Information Technology:Coding and Computing (ITCC’04), vol. 1, pp. 360-366, 2004.