© 2005 - 2009 JATIT. All rights reserved.

AGENT BASED RESOURCE BROKERING AND ALLOCATION IN WIRELESS GRIDS: REVISITED

MINA SEDAGHAT, MOHAMED OTHMAN

Department of Communication Technology and Network,

Faculty of Computer Science and Info. Tech.,

University Putra Malaysia, 43400 UPM Serdang, Selangor D.E., Malaysia

mina.sedaghatir@gmail.com , mothman@fsktm.upm.edu.my

ABSTRACT

Existing internet frameworks, like grids, wireless grids, P2P networks are now provide effective channels for gathering and processing widespread information, using the available resources within reasonable cost. Agent technology emerges as a suitable solution to combine with these frameworks for covering the existing challenges, like discovery, load balancing and resource management. Although different models introduced, practical solutions remain elusive and tend to exhibit underlying conflicts between different paradigms. This paper, aimed to revisit and discuss the characteristics on one of these system architectures based on agents. The base model is introduced, then our re-experiment is described and the results are compared, such that the challenges on this agent base model is highlighted and new ideas can be applied.

Keywords: *Wireless grids, Agents, Resource discovery, Cluster based architecture*

1. INTRODUCTION

Wireless grid is a computer network; consisting of wireless ad-hoc nodes which share their resources in a decentralized manner to achieve a higher performance in different criteria. Heterogeneity, non-stability of the users and resources, the dynamic nature and the decentralized control are some of key challenges of this type of network. As we go through other networks descriptions, it can be seen that some other environments like peer to peer networks, MANET (Mobile Ad-Hoc Network) or some specific grid models were tried to argue on similar challenges, as mentioned above. The problems like finding out an abstract virtual model on the constitution of these high churn nodes and issues like resource management, discovery and load balancing or handling the intermittent user departs and rejoins, are converged points in these environments. So an Analytical overview on wireless grid within interrelated topics can be applied to these environments too. As the main research purpose is to find out a suitable solution, for resource discovery in P2P networks, we examine and evaluate different available solutions with the same concern, like this one in wireless grid, and criticize them to come up with a precise idea on an improved model. Although results of the main objective are not presented in this article, it is been tried to give a brief discussion on a model selected as base, similar framework. We have studied the model mentioned in [1], with respect to both wireless grid, and P2P perspectives.

Rather than, available resource sharing infrastructures, mobile agent technology has become an alternative approach for the design of distributed systems to the traditional centralized architecture. Distributed system management can be distributed and scaled by the use of mobile agents. An agent model for P2P systems enables the description of complex systems with a higher level of abstraction [8]. Current grid systems are somewhat rigid and inflexible in terms of their interoperation and interactions, while agent systems are typically not engineered as serious distributed systems that are robust and secure and need scaling. Nevertheless, each is working its way towards the others' territory, as grids seek to become more flexible and agile, and agent systems seek to be more reliable and scaleable [7].

In environments like P2Ps, ad hoc's and wireless grids where, there is no guarantee for a node to be a fixed member, an independent way of interaction,

© 2005 - 2009 JATIT. All rights reserved.

seems necessary. The agent as an autonomous entity that can act on behalf of its client, the resource node or on the other hand, the user that request a node, seems to be a good solution. Available agent standards, like FIPA, specify a safe environment for agents and the language of communication between them that can be used as an upper application layer, which is not dependant on the underlying physical network any more. Other features of agents like intelligence and mobility used in distributed systems and still can be applied in these environments.

The purpose of this study is to revisit and discuss the characteristics on one of the architectures on wireless grid using agents, by M.N Birje et al.2006.Here we first briefly introduce their model as the base model and then describe our modifications and experiments on it. By comparing our results with theirs, we highlight the current problems of this agent base model.

The rest of this paper is structured as follows: Section 2 describes and introduces the base model [1]. Then our experiment on the base model will be described in detail in section 3. The results and the arguments are presented in chapter 4. The last section concludes the paper and points out the future directions.

2. AGENT BASED SYSTEM ARCHITECTUTRE, THE BASE MODEL

The model is based on a hierarchical agent domain that were similarly introduced and used in some other works like [2], [3], [4], and [5]. Although they mentioned that the work differs from the existing similar works , in the following ways: 1) it uses cluster based agent framework, 2) it supports faster discovery of resources irrespective of its location and then returns an optimum cost resource, 3) It balances the load across the grid.

The resource discovery in the model is applied in cluster based architecture. The clusters are categorized in two types: local clusters (Actual organizations) and higher level clusters (Virtual organizations). This constitution of the nodes, reminds the super peer perspective, used in P2P networks which utilise semi –centralized architectures.

A super-peer is a node that acts as a centralized server to a subset of clients. Clients submit queries to their super-peer and receive results from it, as in a hybrid system. However, super-peers are also connected to each other as peers in a pure system are, routing messages over this overlay network, and submitting and answering queries on behalf of their clients and themselves [6]. The model consists of an interactive agent domain and some non-agent components like resource nodes and storage components like a Cluster Resource Brokers (CRB) and a Base Station Servers (BS); try to resolve requests, by communicating with each other, through a Job Mobile Agent, JMA. The CRB stores information of the resource nodes in the actual organizations, whatever the nodes can provide as a service, in this case storage capacity and number of processing elements of each node, and also the name and the address of the nodes for further retrievals, the number of node in the cluster and the associated BS. The BS maintains information of its Virtual organization, names of the associated Actual Organizations (AOs) and the names and addresses of the CRBs and BSs. A BS is setup for each VO. A CRB is periodically or continuously updated with the status information of the node’s resources such as the number of available Processing Elements (PE), speed of PE’s, available Storage Elements (SE), current load of the node, minimum cost at which the resource is made available for other jobs [1]. The Actual Organization Resource Broker Agent (AORBA) and Virtual Organization Resource Broker Agent (VORMA) are the upper level agents that are responsible for maintaining information about their jurisdictions. In this case the AORBA is a broker, process the information of its own local cluster and VORMA is a higher level broker that deal with the information of its associated actual organizations. If the request can not be fulfilled within the node, the mobile agent, JMA, is created that carries the job requirements with it and then communicates with its AORBA in order to discover the required resource [1]. The AORBA tries to find the request within its cluster, and inform the JMA if it succeeded, if not the AORBA will interact with its associated VORMA. The process is continued, as the VORMA’s also interact with each other to resolve the request with the optimum cost. At the end the AORBA selects the node with a minimum cost and informs the address to the JMA. The JMA compares the cost with the maximum cost the user can afford. If it is less than or equal to the maximum cost then the job is allocated to the selected node. Otherwise the job is failed.

© 2005 - 2009 JATIT. All rights reserved.

3. EXPERIMENT

To evaluate the results on the base paper, we re-implement the model, using JADE (Java Agent Development framework) on a Pentium machine. The simulation was done with the same initial parameters.

Three different arrangements of the nodes in the environment are simulated. The environments are constitute by c=6, c=7, c=8, where c is the number of clusters in the environment. The number of the nodes are distributed randomly between, <n1, n2> where, <n1=1- n2=5> in c=6, <n1=1 - n2=10> in c=7, <n1=1-n2=15> in c=8. The allocated bandwidth in each cluster is v=40 Mbps. The JMA requirement for bandwidth is a random value between <p1=1-p2=5>. The requests are generated for storage elements <r1=1, r2=3> and Processing elements with <q1=1, q2=1000> MIPS. The time out for the JMA is considered T=2 seconds. The basic price of a resource is x = 1000 unit, and Maximum price the user can afford is y=1200 unit. The JMA size is j=4kb, and the JMA migration time for each hop, is distributed between <t1=200, t2=300>. The jobs are generated, every a=50 *msec*.

The main work, evaluate some performance parameters like bandwidth utilization, Average resource discovery time, Rejection ratio, Resource utilization and agent overhead. Although graphs and more detailed analyses are given on the first three mentioned parameter. We focus and re-analyze the results on Average resource discovery time, Average bandwidth utilization, and Rejection Ratio, and introduce a new parameter, number of hops, and discuss about the affects of this parameter on the other three. The performance parameters above are calculated based on these formulas:

Average bandwidth utilization =

$$\frac{\sum \text{Bandwidth required for each job for its completion}}{\sum \text{Bandwidth available in the environment.}}$$

Average Resource discovery time =

$$\frac{\sum \text{Time taken by all agents to discover the resources}}{\text{The number of agents that are generated}}$$

Rejection Ratio=

$$\frac{\text{Number of JMAs that don't acquire the resources}}{\text{Number of JMAs that are requesting the resources}} * 100$$

4. RESULTS AND ARGUMENTS:

4.1. Average Bandwidth utilization:

For calculating the average bandwidth consumption, first we articulate the given formula above:

The requirement bandwidth for each job (RB_Job) is the sum of the product of the JMA size and the number of hops for its completion and the JMA bandwidth requirement for that request (JMA_BR).

RB_Job = JMA_BR + JMA size * number of hops (1)

The average bandwidth utilization is the ratio of the RB_Job to the available bandwidth in the network. When the available bandwidth in the network is the product of the number of clusters in each arrangement of the environment, (e.g. c=6, c=7, c=8) and the allocated bandwidth for each cluster. As it can be seen, in (1), two main effective factors in average bandwidth utilization are number of hops and JMA bandwidth requirement. JMA bandwidth requirement depends on type of the request generated, in our case; it is a random number between <1-5> that is distributed uniformly to show the different possible conditions. Figures 1, 2 and 3 shows the acquired result, based on the new experiment. The base paper mentioned that an increase in the number of jobs will increase the bandwidth utilization, as it can be seen in the graphs. Also the result shows that when, as number of clusters and resources in clusters increase, the bandwidth utilization is reduced. This is because, the allocated bandwidth is increasing suddenly and not in a uniform manner, when just one cluster is added. It can be argued that, the allocation of the bandwidth in the environment is not relevant with the requirements of potential requests for the bandwidth.
As mentioned previously, two main factors that affect bandwidth utilization parameter are number of hops and JMA requirement for bandwidth. As bandwidth utilization parameter is strictly related to JMA requirement for bandwidth, and this factor is considered as a random value, we observed that the bandwidth utilization range is clearly depend on the random values that are generated by the random function. In this case Fig 1, And Fig 2 shows the different result range with different distribution of values. Also we can't, say that, the environments with more number of clusters, necessarily, have a lower values in bandwidth utilization, as it closely depend on the requests size. So we present both graphs to show this non-relevancy. The second factor that is noticeable is the number of hops and

© 2005 - 2009 JATIT. All rights reserved.

the relation between this factor and the bandwidth utilization. As it shows in Fig 3, the number of hops doesn't affect the bandwidth utilization parameter so much, when the size of the JMA is just 4kb, in compare with the JMA requirement for bandwidth that is between <1-5> Mbps. So, in this case, as much as JMA travel the environment to discover the resource, the number of hops doesn't play the main role on increasing the bandwidth utilization results.

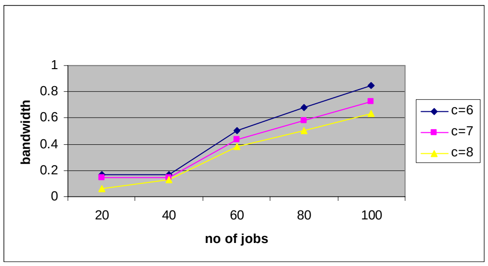

Fig 1 : The bandwidth utilization results in re-experiment, Lower Generated numbers of JMA_BR

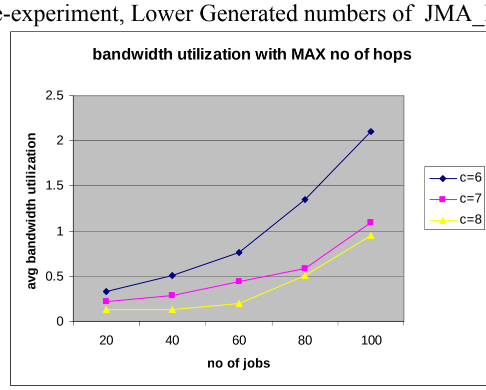

Fig 2 : The bandwidth utilization results , In re-experiment with a uniform distribution.

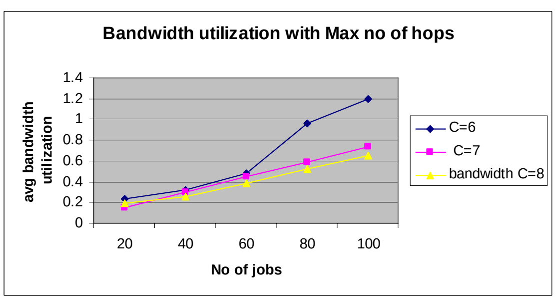

Fig 3 : The bandwidth utilization results , in re-experiment with max number of hops

4.2. Resource Discovery Time

In calculating the resource discovery time, the given formula is interpreted as follows:

The time taken for each agent to discover the resource is the product of the JMA migration time in each hop and the number of hops till the JMA completion of the task. So the formula can be re-write as:

$$\left\{\sum_{n=1}^{n}\frac{\text{JMA migration time in each hop * no of hops}}{n}\right\} \qquad (2)$$

'n' is the number of generated agents to discover the resource. The base model, report that resource discovery time increased with an increase in the number of jobs. It also shows the average discovery time increases with an increase in the number of hops, as the JMA moves from cluster to cluster [1]. The relation of the resource discovery time, in different environments can be also studied. Based on the previous results, as the number of clusters and available resources increase, the resource discovery time increases too. But during our simulation, we observed that, when the number of clusters in the environment increase, the number of nodes or available resources in each cluster increases too (e.g. for c=8 n=<1-15> when in c=6, n= <1- 5>). When the number of available resources in each cluster increases, the chance for the request to be fulfilled in its first visiting clusters will be increased. So the job of the JMA will be completed in less number of hops. In this case, it is expected that the resource discovery process, takes less time in the environments with more number of clusters and nodes. Fig 5 shows this relationship between the size of the environments and the resource discovery time. A little conflict is raised, in this case between the results on the base model, and new experiment. We tried to find out the reason, by corresponding with the author. Two reasons are given to resolve the conflict. Firstly, the information of the bigger environments includes searching the bigger databases. The BS's and CRB's are these storage centers that collect the information about their nodes and lower peers. So the database searching time can be mentioned as the other factor in calculating the resource discovery time. But we believe that searching a database, with the current size, a range between 30-120 entities, doesn't have that much affect on results. The second point, mentioned as the possible conflicting issue was the random generation of the number of nodes in the clusters. As the arrangement of the environment is based on random values, it means that, clusters with bigger ranges for nodes (e.g. c=8 n=<1-15>), doesn't necessarily includes more number of resources. So the first assumption, that there is a higher chance for bigger clusters for being completed in less number of hops, seems not that much strong. But the point is, when the value range of the random function defined larger, the chance for generating the bigger values will be

© 2005 - 2009 JATIT. All rights reserved.

higher too. So although , the number of nodes in the bigger environments, shouldn't be necessarily more than the smaller environments, but still there is more chance for them to generate the more number of resources. So we still think that the first assumption, can be valid, and resource discovery time in bigger environments, is much less than those with less number of available resources.

The convergence at the end of the graph shows that, for more no of jobs, the average resource discovery time become constant, as the jobs are fulfilled in the first visited clusters .It can be analyzed that the number of jobs doesn't affect the time in the bigger environments, it is also because as the VO's become larger (with more available resources), the discovery can be done in less number of hops. So the Resource discovery time, will become constant, when we compare a constant number of jobs in clusters with different number of resources. Although as the number of jobs increases in a way that the proportion of the number of resources to the number of jobs, become equivalent, then a valid comparison can be made.

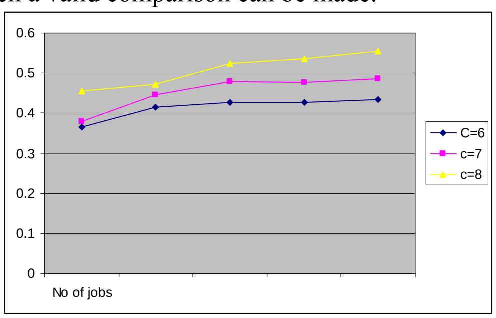

Fig 5. Resource discovery time results , in the re-experiment , considering searching time factor

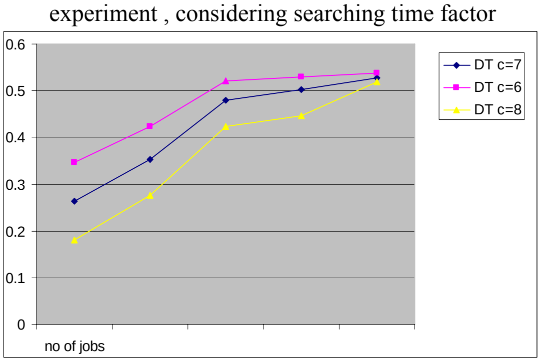

Fig 6. Resource discovery time results , in the re-experiment without considering searching time factor

4.3. Rejection Ratio

For a grid with a fewer number of resources, usually an increase in the number of jobs increases the number of job rejections. If the number of jobs exceeds the grid accommodating capacity, the rejection ration increases exponentially [1]. The rejection ratio is considered the ratio of the number of JMAs that don't acquire the resources to the number of JMAs that requesting the resources. The model, defines a TTL, for the mobile agent for about 2 seconds. So one of the factors, that causes a rejection, rather than non-availability of the requested resource, is the JMA timeout, before finding the resource that is probably available, within the next clusters. The scarcity of bandwidth or weak power signal level of nodes or the mobility of nodes that can meet the job requirements are some of the other factors [1]. Fig 8 shows the results on rejection ratio in the re-experiment.

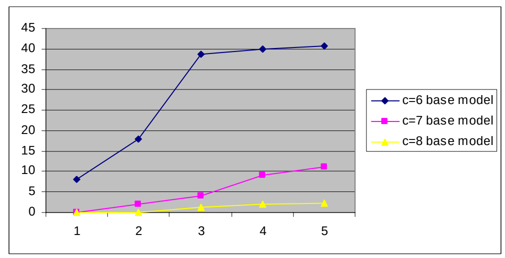

Fig 8. Rejection Ratio , in the re-experiment

4.4. Number of Hops

During the illustration of the bandwidth utilization, and resource discovery section, we encountered a new parameter, as number of hops that was one of the main factors that affect the above parameters. On the other hand, the rejection ratio is also had a direct relation, with resource discovery time, and so on the number of hops too. So studding the number of hops as one the main parameters can be helpful. During calculation bandwidth utilization, it was observed that the number of hops was not impress the parameter, as the size of the JMA, was so smaller than the size of the request carried by the Agent. But in case the size of JMA , increases during the travel around the environment, like when it gathers , some information about the status of visited nodes, or learn about the an optimized route during the roam , the number of hops will be one of the important factors , in bandwidth utilization .

The trade of between the number of hops and resource discovery time parameter discussed in previous section. When each migration of the JMA, takes a finite time, as much the JMA, find the requested resource in less number of hops, the resource discovery time, became shorter. So it is obvious that in environments with more number of resources, the required time for finding a resource is smaller than those with a less number of the resources.

© 2005 - 2009 JATIT. All rights reserved.

5. CONCLUSION

The base model used a cluster based agent framework, in which the nodes are clustered un-purposely, and without any common values, they all provide the same type of resource with no subjective or semantic categorization. By this type of clusterization, it has been tried to localize the searches, and reduce the JMA travel zone. However, these non-oriented clusters , doesn't help that much , in reducing the number of hop counts in the discovery process , as JMA needed to travel all the clusters , with no clue or priority to visit , for finding its requested resource. This type of arrangement causes more discovery time, more bandwidth consumption, as the requests would be fulfilled in more number of hops and higher risk of rejection, as the time has an effect on the rejection.

In this paper, it has been tried to revisit and discuss the characteristics of one the agent based resource discovery models in a wireless grid, M.N Birje et al.2006. The model and the re-experiment of the model was described and compared. In future implementation, we plan to work on another type of environment constitution that covers the current challenges, and examining a new JMA with additional responsibilities, rather than working as a message transfer agent.

REFERENCES

[1] M. N. Birje, S. S. Manvi, Bhanu Prasad, "Agent based Resource Brokering and Allocation in Wireless Grids" , IEEE International Conference on Services Computing 2006.

[2] J.Cao, DJ. Kerbyson, GR. Nudd, "Performance Evaluation of an Agent-Based Resource Management Infrastructure for Grid", Computing, Proceedings of 1st IEEE/ACM International Symposium on Cluster Computing and the Grid. Brisbane, Australia, 311–318, May 2001.

[3] Yanxiang He, Weidong Wen, Hui Jin, Haowen Liu, "Agent-based Mobile Service Discovery in Grid Computing", Proceedings of the 2005 The Fifth International Conference on Computer and Information Technology (CIT'05) , 2005 IEEE

[4] Pei-You ZHUI, JI Gao , Bo-OU Jiang, Fan Zhang , Hang Gou , "An agent federations based infrastructure for grid monitoring and discovery" , Proceedings of the Fifth International Conference on Machine Learning and Cybernetics, Dalian, 13-16 August 2006]

[5] M. Fukuda and D. Smith, "UWAgents: A mobile agent system optimized for grid computing", in Proc. of the 2006 International Conference on Grid Computing and Applications – CGA'06. Las Vegas, NV: CSREA, June 2006, pp. 107–113.

[6] Beverly Yang Hector Garcia-Molina, Designing a super peer network, Proceedings of the 19th International Conference on Data Engineering (ICDE'03), 2003 IEEE

[7] Foster, I., Jennings, N. R., and Kesselman, C. "Brain Meets Brawn: Why Grid and Agents Need Each Other". Proceedings of 3rd Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS), New York, USA, 2004.

[8] H. M. Kelash, M. Amoon, G. M. Ali, and H. M. Faheem, "A social agent interface for resource management in distributed systems," *Proceedings of CIMCA*, Vienna, November