Decentralized Dynamic Load Balancing and Intersection Trust in Mobile Ad Hoc Grids

Don Abraham[1] and VetriSelvi Vetrian[2]

[1] **Department of Information Science and Technology, Anna University**
Chennai, Tamil Nadu – 600025, India

[2] **Department of Computer Science and Engineering, Anna University**
Chennai, Tamil Nadu – 600025, India

Abstract

A mobile ad hoc network is an autonomous system connected through wireless links. It does not have any fixed infrastructure, and the mobile nodes in the network coordinate among themselves for communication. This network can be formed by heterogeneous mobile devices like laptops, PDAs, cell phones etc. Highly capable device provides some services to low capability device and this will bring in a grid environment into the mobile ad hoc networks. A balanced allocation of load is a critical issue in such grids, considering its limitations in terms of power and availability. A decentralized approach is favored for a dispersed functioning of the grid, without draining a single chosen node. It is also essential to augment the resource allocation with an effective authorization mechanism, as the ad hoc grids lack permanent trusted central authority.

In this paper, dynamic virtual organizations are constructed within this grid based on the resource commonalities of devices. Upon this setup we implement a decentralized load balancing mechanism that is able to perform a dynamic load balancing among nodes within a VO i.e. intra-VO load balancing using the Local Load Balancing Algorithm and also between the dynamic VOs i.e. inter-VO load balancing using the Global Load Balancing Algorithm for efficient utilization of resources and enhancing the performance of computational grid. When jobs are sent across VOs due to resources unavailability within its own VO, a trust check is performed based on the intersection trust factor between the VOs. The effectiveness of the algorithm is substantiated by the simulation results.

Keywords: *Mobile Ad hoc Grid, Dynamic virtual Organization, Decentralized Dynamic Load Balancing, Local Load Balancing Algorithm, Global Load Balancing Algorithm, VO-Intersection Trust.*

1. Introduction

A heterogeneous ad hoc network (Li and Hue 2004) is the sharing of resources and services across various devices. For instance, mobile devices like laptops, PDAs, mobile phones, etc., have different computation capabilities, power, hardware and software functions. Nodes with higher computation capabilities and power can share the resources with devices of lesser capabilities. Thus a mobile ad hoc grid that allows such sharing can be formed [1]. Grid technology [2][3] is applying the resources of many computers in a network to problems that requires a great number of computer processing cycles or access to large amounts of data. The availability of wirelessly connected mobile devices has grown considerably in recent years, creating an enormous collective unexploited potential for resource utilization. It is a natural idea to integrate the resource aggregation model of Grid with mobile ad-hoc networks, so as to build a mobile ad-hoc Grid platform that can be instantly constructed anytime, anywhere. Having been constructed from a group of mobile devices, an ad-hoc grid would allow the networked devices to accomplish a specific mission that maybe beyond an individual's computing or communication capacity [4]. At the heart of the Grid is the ability to discover, allocate, and negotiate the use of network-accessible capabilities which will also be applicable to the mobile ad hoc grids. The essential mode to implement grid is through Virtual Organizations. A temporary or permanent coalition of autonomous individuals that pool resources, capabilities and information to achieve common business objectives are called Virtual Organization [5]. The VOs are formed based on certain commonalities its constituent nodes share. These VOs are dynamic in case of Mobile Ad hoc Grids, hence are known as dynamic Virtual Organizations.

Load balancing is a technique to distribute workload evenly across two or more computers, network links, CPUs, hard drives, or other resources, in order to get optimal resource utilization, maximize throughput, minimize response time and avoid overload. Due to heterogeneous and dynamic nature of the grid system, scheduling and load balancing in such environment is significantly complex and challenging [7]. The mobile and ad hoc nature of the Mobile Ad hoc Grid makes it e ven more difficult. An efficient load balancing is a k ey factor in achieving high performance for distributed applications on dynamic environments like that of MAG. In general, load-balancing algorithms can be roughly categorized as centralized [8] or decentralized [9][10] in terms of the location where the load balancing decisions are made. For the centralized approach [11], many authors argue that this approach is not scalable, because when the system size increases, the chosen central controller may become a system bottleneck. In case of mobile ad hoc grids, it is further formidable as the nodes are mobile and power-limited. Decentralized way of load balancing proves to be efficient approach in a mobile ad hoc environment where the burden of the load balancing would be assigned to several regionalized leaders, thereby preventing quick battery drain of a particular central node and also reducing the traffic to it, which decreases the waiting time of requests as multiple nodes are available to provide service. Dynamic load balancing algorithms [12] [13] attempt to use the runtime state information to make more informative decisions in sharing the system load and can potentially provide better performance than static algorithms. This makes the entire system faster and efficient [7].

When the Grid is intended to be used for business purposes, it is necessary to share resources with unknown parties. Such interactions may involve some degree of risk since the resource user cannot distinguish between high and low quality resource providers on the Grid. The inefficiency resulting from this asymmetry of information can be mitigated through trust mechanisms [14].To establish trust, traditional grids use various methods, mostly centrally oriented ones, such as certification authorities, VO management servers or credential pools [15]. An ad hoc grid environment is characterized by the absence of a permanent central trusted authority; therefore collaborating entities must establish and maintain a trust relationship among themselves. Authorization mechanism based on intersection of VOs can be used to build trust relationships between such grid entities in cases when standard solutions have failed [16].

The nodes in all ad hoc networks are not completely in random motion; there is some steadiness and regularity in its movement patterns. In case of ad hoc networks found in universities formed by laptops, PDAs, smart phones, etc. the nodes have a low mobility and high regularity patterns in movement and time. Such ad hoc scenarios where grid functionality is most likely to be implemented have a higher degree of stability making the core concept viable [17].

This paper is organized as follows. Section 2 discusses the background and related work. Section 3 introduces the concept of mobile ad hoc Grid with virtual organizations. Section 4 deals with the construction of virtual organizations within the ad hoc grid. In Section 5 the proposed Decentralized Dynamic Load Balancing Algorithm is explained in detail. In Section 6 the intersection based trust mechanism is presented. Section 7 evaluates the algorithm based on simulation results. Section 8 concludes the paper.

2. Related Work

Several researches have been carried out with regard to mobile ad hoc grids, as well as various unconventional methods of load balancing in grids.

Ihsan et al [6] have proposed a mobile ad hoc service grid that maps the concepts of grid on to ad hoc networks. The availability of the service in a node is broadcast to all one-hop neighbors. Since the grid is formed within one-hop neighbors, there is a ch ance for resource discovery to fail when there is no service provider within one hop. In this grid, each node is responsible for maintaining the resource look up table, which can be a b urden to devices with less storage capabilities. Vetri Selvi et al [1] have proposed a mobile ad hoc grid architecture using the trace based mobility model. They propose an architecture for a mobile ad hoc grid and address the challenges due to mobility by considering a trace model for the movement of the nodes, called Trace Based Mobility Model (TBMM) [17].TBMM captures the history of the movement patterns of the nodes and identifies regularity in these movements. This regularity in movement is used to predict the stability of the nodes.

A grid load balancing scheduling algorithm based on statistics thinking has been set out by Lu et al [18]. They use a hierarchical structure model of resource management, i.e. resource management and scheduling are of multi-levels, each resource has its own scheduling system, and users only need to submit the assignment to a central manager. The algorithm is centralized and hence it is not suitable for ad hoc grids where there is no permanent central authority. Suri et al [7] have proposed an efficient decentralized load balancing algorithm for grid. They use a decentralized grid system model for providing load balancing in grid comprising of clusters, where cluster's server is treated as Coordinator Node. Each cluster has multiple Worker Nodes with different processing powers.

Table 1: Grid Resource Table maintained by GHN

Node ID	SPN /CN	VOM / Non-VOM	Resources				Price / Fee	Battery	Position	Stability
			Processor Speed	Available Memory	Disk Space	Bandwidth				

The jobs generated by users are directly submitted to their coordinator nodes, where a dynamic load balancing algorithm is invoked to perform decentralized load balancing within clusters and between them. This algorithm has been deployed in wired environment.

Arenas et al [14] have dealt with the classical Virtual Organization lifecycle, augmented with trust management actions. VO lifecycle phases for infrastructure grids have also been discussed. Azzeden et al [19] have dealt with integrating trust into grid resource management. These trust technologies for infrastructure grid has been adopted for wireless grids with needed variations. Huraj et al [16] have proposed VO intersection trust in ad hoc grid environment. The aggregated trust of the systems that exist in intersection of potential user's VO and resource owner's VOs must be greater or equal to a particular threshold that is based on resource owner policy. The higher the threshold, the higher the resulting trust.

Though the concept of wireless grids are widespread these days [20], the conventional methods of load balancing persists, which does not consider the additional challenges of mobility and power limitations of underlying ad hoc network, unlike wired grids. Here we propose a decentralised dynamic load balancing mechanism suitable for a mobile ad hoc grid with virtual organizations. The idea is augmented with a simple trust mechanism appropriate for ad hoc environment.

3. Mobile Ad hoc Grid with Virtual Organizations

Primarily, grid infrastructure is formed over an ad hoc network based on a specific framework that includes certain light weight mechanisms apt for ad hoc environment [1]. In a mobile ad hoc grid, difficulties that arise due to mobility are overcome by using the stability estimation of the trace based mobility model [17].

Mobile Ad Hoc Grid: The grid is comprised of service provider nodes (SPNs) that are highly capable and consumer nodes (CNs) are the nodes that require services. The grid formation and resource management between the CNs and SPNs is achieved by an SPN that is willing to act as grid head node (GHN). The GHN is responsible for maintaining all the resource details about the member nodes in a grid resource table (GRT) as shown in table 1.

The initial grid is formed based on the physical location of the nodes, over this we provide a logical cluster called virtual organizations. These VOs are formed based on resource similarity. Each VO has a manager node that facilitates services to the other nodes of the VO. Virtual Organization Manager (VOM) is the most stable SPN whose stability is estimated using trace based mobility model. Jobs specific to that VO are submitted by the other nodes of the VOM in a decentralized scenario. Load balanced job distribution is done within the VO. Virtual Organization member Nodes are the nodes of the VO that includes both SPNs and CNs. They request service or provide service according to their functionality through the respective VOM. Fig. 1 depicts a typical mobile ad hoc grid with two dynamic virtual organizations VO1 and VO2, visibly showing their role.

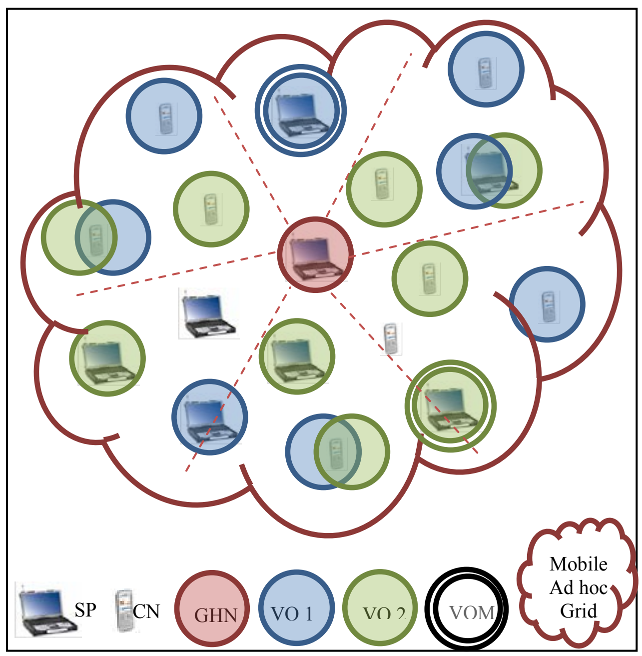

Fig. 1 Ad hoc Grid with Virtual Organizations

4. Dynamic virtual Organization Construction

A temporary or permanent coalition of autonomous individuals that pool resources, capabilities and information to achieve common business objectives are called Virtual Organization. Dynamic VOs in a MAG are the dynamic logical clusters within a physical grid cluster. The construction and management of these VOs in a MAG includes the following steps

- VO Identification
- VO Intimation
- VO Registration
- VO Confirmation
- VOM Utilization Table Maintenance
- VO Node Table maintenance

4.1 VO Identification and Intimation

The identification is the foremost step in which the resource requirements for the VO to be formed are specified by any CN to the GHN. The GHN scans its grid resource table to find the set of nodes that satisfy these resource criteria. Out of these selected nodes, the node with maximum stability is chosen as the manager of that VO. The GHN provides a new VO ID and also intimates about the chosen manager to all the nodes of similar resources.
.

4.2 VO Registration

After receiving the intimation message about the VO and its VOM, the willing nodes register themselves with the VOM to become a member of that VO. This registration includes their resource reputation / credentials exchanged in form of trust certificates. Thus the trust value is collected by the VOM from each member of the VO.

4.3 VO Confirmation

The weighted mean trust value is calculated by the VOM, from the resource reputation given by the nodes. A confirmation message is sent by the VOM to the members of the VO, to confirm their membership in the VO on satisfying the trust requirements for that VO. It is possible for same node to have different trust values on different VOs, because the weights for each resource used for calculation of weighted average trust may vary from VO to VO, depending upon the need and nature of the particular VO. Thus all nodes within a VO trust each other.

4.4 VOM's Utilization Table

The VOM constructs and maintains a utilization table as shown in Table 2 that is used for local load balancing. The table is updated every time it receives a registration message from its members. The utilization field is updated dynamically for every job processed by the VO, because the load balancing is completely reliant on this value. The table includes details such as node ID, VO ID, node type, trust value, utilization and battery power.

Table 2: Utilization Table maintained by VOM

VO Node ID	SPN / CN	Trust	VO ID	Utilization	Battery

4.5 VO's Node Table

Each and every VO member Node maintains a node table, which contains the VO membership details of that node. The membership details of a node include VO ID, corresponding VOM ID and the associated trust value. This is a very important data structure, as a node can be a member of more than one VO.

5. Decentralized Dynamic Load Balancing

A decentralized grid system model is used for providing load balancing in grid comprising of VOs. Each VO consists of a VO Manager and several VO Nodes (both SPNs and CNs). The SPNs have greater processing powers and they provide resources for completion of the jobs submitted by CNs. Jobs generated by CNs are directly submitted to their VOMs. Local load balancing is done within VO using Local Load balancing Algorithm (LLBA) at the VOM. In case of overload of the VO, the job is scheduled to another VO through GHN where the Global Load Balancing Algorithm (GLBA) is employed.

Local Load Balancing Algorithm: If a new task arrives at VOM of any VO_m then, VOM does the following.

1. Add it to its task set.
2. Determine the least loaded SPN in VO_m from the dynamically updated utilization table, SPN_{min}.

$Load_{min}$ = min ($Load_{SPN\ k}$);for k={all SPNs of VO_m}
SPN_{min}=SPN corresponding to $Load_{min}$

Load is computed using the weighted sum of squares method as follows,

$$Load_{SPN} = \sqrt{\sum_{i=1}^{n} w_i L_i^2} \quad (1)$$

Where, n is the number of resources considered
L_i is the Utilization Percentage (or) Load Index of resource i of the SPN
w_i is the VO-specific weight assigned for resource i, such that
$\sum_{i=1}^{n} w_i = 1$

3. If (Load(SPN_{min})< *overload_threshold*), then
   Assign the job to SPN_{min}
   Update Load of SPN_{min} and VO_m
   Else
   Forward job to GHN

Gobal Load Balancing Algorithm: If a task arrives at GHN from any VOM due to VO overload, GHN does the following

1. Find the VO with minimum load from the table dynamically maintained with the GHN, VO_{min}

   In case of VOs, the load of a VO is given by the arithmetic mean formula as follows,

$$Load_{VO} = \frac{\sum_{j=1}^{k} Load_{SPN\_j}}{k} \quad (2)$$

   Where, k is the total number of SPNs in the VO under consideration. $Load_{SPN\_j}$ is the load of SPN_j.

2. Check Trust
   If (Intersection_Trust >= *trust_threshold*)
   Dispatch job to VOM of selected VO_{min}.
   Else
   Select the next least loaded VO (go to Step 1)

The following components are employed in implementing the proposed decentralized dynamic load balancing in MAG. The interaction of these components is described by figure 2.

- Job Submission
- Load Calculation and Monitoring
- Job Distribution
- Job Completion

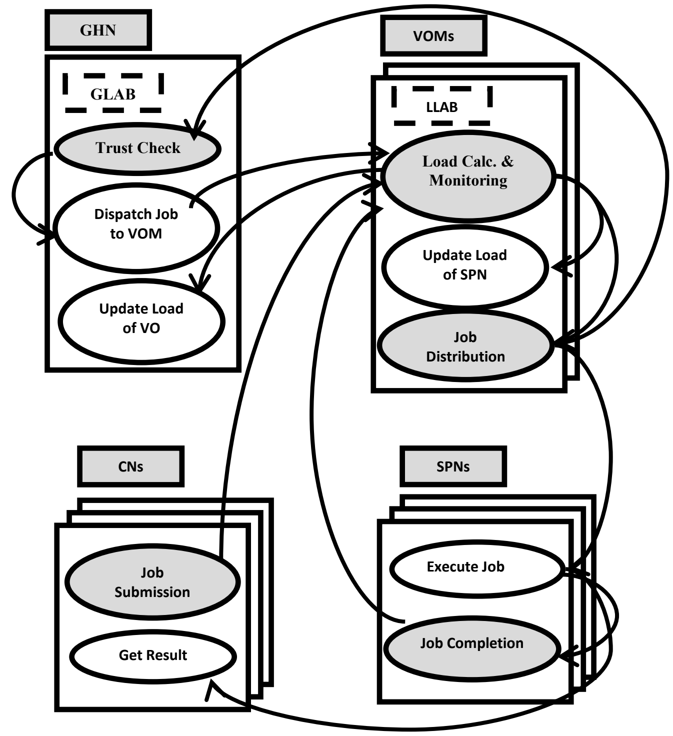

Fig. 2 Components involved in Dynamic Decentralized Load Balancing

5.1 Job Submission

The Jobs are submitted by the member CNs, to the respective VOM of the VO to which the job belongs. These jobs include the details such as job ID, VO ID, amount of each resources required by them, and also the period for which these resources are needed for execution of the job. This job description is used by the VOM to make a load aware choice of SPN for allocating the job for execution. In any general scenario arrivals of this kind follow Poisson distribution and the Service time follows Exponential distribution. Jobs are submitted on Poisson intervals with Exponential job durations as followed in many simulation experiments of Grids [21] and Mobile Ad hoc Grids [1].

5.2 Load Calculation and Monitoring

As jobs arrive at the VOM, the local load balancing algorithm is invoked. The least loaded SPN within the same VO is found by scanning the utilization table dynamically maintained by the VOM. As jobs are directed to that SPN, the increased in load of the respective SPN is calculated using Eq. (1) and is dynamically updated in the utilization table. With the increasing load of SPNs, the overall load of the VO also increases. This is calculated using Eq. (2), and intimated to the GHN. If the load of the least loaded SPN of a VO is greater than a specific threshold load value, then we understand that all the SPNs are overloaded and hence the entire VO is also overloaded. In this case the job is forwarded to the GHN, where the global load balancing algorithm is invoked. GLBA identifies the least loaded VO from the table containing the load of each VO as calculated and dynamically updated by the respective VOMs. Before the job is dispatched to the VOM of the minimum utilized VO, a trust check is done as explained in section 6. We continue forwarding the job to that VO on grounds of sufficient trust; else we go for the next least loaded trust worthy VO.

5.3 Job Distribution

This module distributes jobs to the SPNs in a load balanced fashion with the aid of load monitoring facility after loads are calculated and updated. When all SPNs available within the VO are overloaded, the job is forwarded to the GHN for execution at some other under-utilized VO.

5.4 Job Completion

On completion of a task the VOM is notified by the SPN who finished executing the job, for relaxing the load of the particular SPN and the corresponding VO. The load calculation and updating facility is again invoked for this purpose. Then the results are presented to the CN which submitted the corresponding job.

6. Intersection Trust

In a grid, the decision regarding resource access is up to the resource owner. Generally, the resource owner checks if the potential user is member of the same VOs as the resource owner or, if there are any trustworthy attribute authorities which have signed the certificates. The resource is granted on fulfillment of the above conditions. But there is no central trusted authority for certificate signing in a mobile ad hoc grid which makes the latter condition unlikely. When resources outside the VO are to be accessed, a robust trust mechanism becomes crucial for efficient and prudent utilization of limited resources available in ad hoc grids. Such a novel trust mechanism is implemented at the GHN as follows.

6.1 Intersection Trust Check

In the present work, such a trust model well adapted for ad hoc setup has been integrated with the load balanced resource allocation process. In the VO intersection trust, when a CN requests resources of a SPN, the SPN provides services without any hesitation if they belong to the same VO, where as a trust check is required if the job is from a different VO. As devices can belong to more than one VO, the boundaries of the VO overlap. We first consider the number of nodes that lie in this region of overlap or intersection. These nodes are the ones that are members of both the requesting VO as well as the resource owner VO. The average trust of these nodes is checked against an owner-specific threshold trust, on clearing which, the resources are allocated for the job from the alien VO. Due to the dynamicity of the setting the intersection members tend to change over time, hence an intersection trust check is done for every job that is sent across VOs. There is no permanent trust between VOs. The scenario is depicted in Fig. 3.

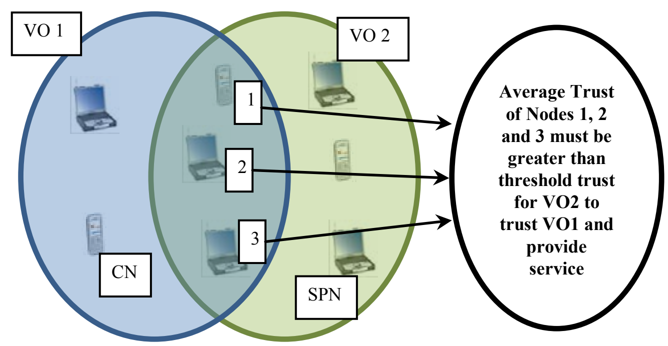

Fig. 3 VO Intersection Trust Mechanism

7. Performance Evaluation with Simulation Results

Simulation studies have been carried out to evaluate the mobile ad hoc grid architecture. The simulation tool used is Glomosim [22].

Table 3: Parameters for Simulation

Parameters	*Values*
Number of Nodes	50
Simulation Time	1000 Seconds
Terrain Dimension	(1000 X 1000) meters
Mobility	Trace Based Mobility Model
Radio-Tx-Range	15 dBm
MAC Protocol	802.11
Routing Protocol	DSR Protocol

The parameters used for the simulation are given in Table 3. The mobility model used for the nodes is a trace-based model derived from Ansim [23] depicting a University scenario.

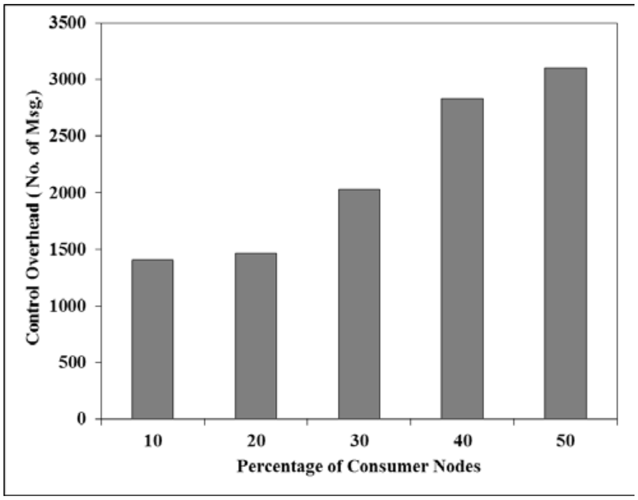

Fig. 4 Grid Formation Overhead with Increasing CNs

A Mobile ad hoc grid has been simulated in this set up using 2 highly stable GHNs and the rest of the nodes acting as SPNs and CNs. Each grid consists of 2 VOs. The overhead in forming a grid is comprised of additional grid-forming messages and VO management messages that are communicated among the nodes to form the grid with dynamic VOs. For the overhead analysis of grid formation and VO construction, the number of CNs is increased in steps of 10%, keeping 25% of nodes as SPNs.

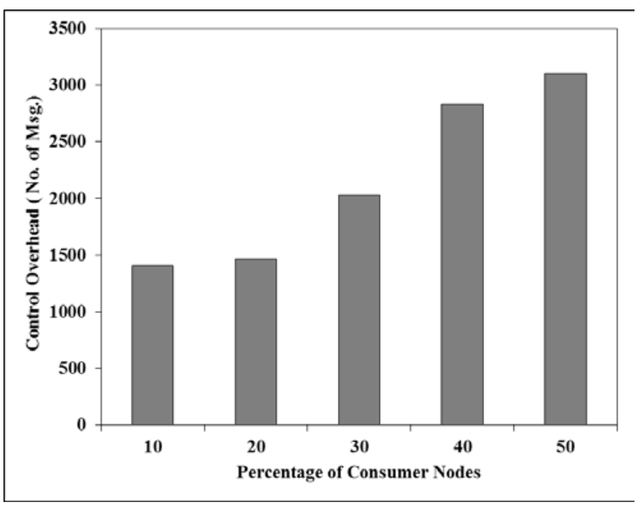

Fig. 5 VO Construction Overhead with Increasing CNs

Fig.4 gives the overall control overhead involved in grid formation over a mobile ad hoc network and Fig. 5 depicts the overhead incurred on VO management. The increase in overhead with increasing number of nodes demanding service is evident from the results.

For the analysis of the Decentralized Dynamic Load Balancing (DDLB) algorithm, the similar set up is used with 50% of nodes acting as CN and the remaining 50% as SPNs. The arrivals followed Poisson distribution, with service rates following an exponential distribution. The time taken for execution of every subsequent 50 jobs is measured with DDLB and also without DDLB.

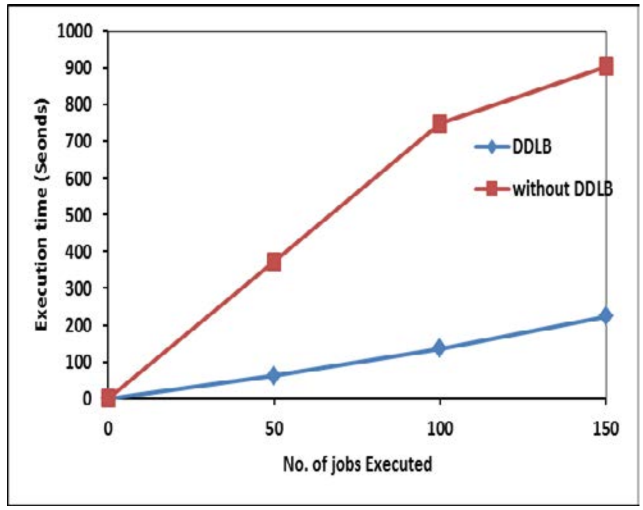

Fig. 6 Impact of DDLB on Job Execution Time

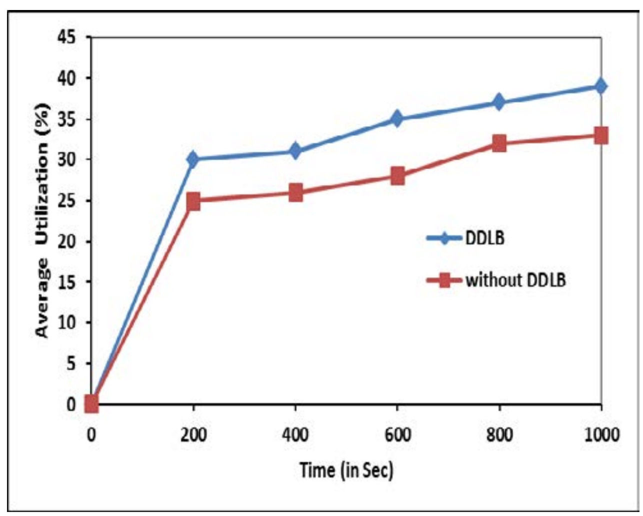

Fig. 7 Impact of DDLB on Average Utilization

It is obvious from Fig. 6, that the more jobs are completed earlier with DDLB, without having to wait longer as the resource allocations are load balanced. On the other hand, without DDLB, job distribution is done arbitrarily leading to high job rejection rates and longer waiting time as certain nodes that are overloaded are repeatedly allocated, when there are many under-utilized ones. Fig. 7 clearly shows that average utilization of nodes is higher with DDLB, than without DDLB. In a volatile environment such as mobile ad hoc grid, which has limited resource capabilities, most efficient use of all the available resources is of paramount importance. In this regard the proposed algorithm has performed superior than the usual method according to the simulation results.

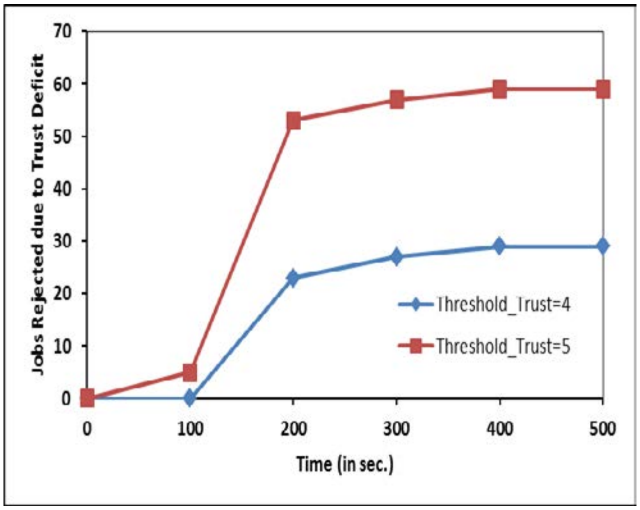

Fig. 8 Jobs Rejected on Trust Deficit due to VO Intersection trust mechanism

The number of jobs rejection due to trust deficit when sent across VOs, employing the novel VO intersection trust mechanism has also been studied. Considering jobs arriving at Poisson intervals and service rate of exponential distribution, a comparison of number of jobs rejected at two different threshold trust values is made. It is seen in Fig 8 that with higher threshold, more jobs are rejected over time, as trust requirements could not be met.

8. Conclusions

In this paper we have dealt in detail about formation of a grid overlay with dynamic VOs over a mobile ad hoc network using lightweight algorithms. The frail nodes of a mobile ad hoc network can accomplish jobs beyond its capability in such a grid environment. In such a setup, for judicious usage of available resources an efficient method for load balancing is set out. The effectiveness of the proposed decentralized dynamic load balancing algorithm is substantiated by the simulation results. The overhead involved in grid maintenance and VO management have also been analysed based on the simulation experiments. An ideal trust mechanism for mobile ad hoc grid, based on VO intersections has also been incorporated into the system for security concerns

References

[1] V. VetriSelvi, Shakir Sharfraz, Ranjani Parthasarathi "Mobile Ad Hoc Grid Architecture Using a Trace Based Mobility Model," *International Conference on Grid and Pervasive Computing*, 2007, pp. 274-285.

[2] I. Foster, C. Kesselman, and S. Tuecke, "The anatomy of the grid, enabling scalable virtual organizations", *International Journal on Supercomputer Applications*, , 2001 vol. 15, no. 3.

[3] I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke, "The physiology of the grid, an open grid services architecture for distributed systems integration," *Open Grid Service Infrastructure WG, Global Grid Forum*, 2002.

[4] Zhuoqun Li, Lingfen Sun and Emmanuel C. Ifeachor, "Challenges of mobile ad-hoc grids and their applications in e-healthcare," in *2nd International Conference on Computational Intelligence in Medicine and Healthcare*, 2005.

[5] Theo Dimitrakos, David Golby, Paul Kearney "Towards a Trust and Contract Management Framework for Dynamic Virtual Organisations", *In eAdoption and the Knowledge Economy: Proceedings of e-challenges*, 2004.

[6] Imran Ihsan, Muhammed Abdul Qadir, Nadeem Iftikhar, "Mobile Ad- Hoc Service Grid- MASGRID," *World Enformatika Conference*, , 2005 pp 124-127.

[7] Suri P.K., Singh, M. "An Efficient Decentralized Load Balancing Algorithm for Grid," *IEEE 2nd International, Advance Computing Conference*, 201, pp. 10-130.

[8] K. Li, "Optimal load distribution in non-dedicated heterogeneous cluster and grid computing environments," *Journal of Systems Architecture:the EUROMICRO Journal*, vol.54(2), , 2008 pp. 11-123.

[9] K. Lu, R. Subrata, and A. Zomaya, "On the performance-driven load distribution for heterogeneous computational grids," Journal of Computer and System Sciences, vol. 73(8), , 2006 pp. 1191-1206.

[10] Ruchir Shah, Bhardwaj Veeravalli, and Manoj Misra, "On the Design of Adaptive and Decentralized Load-Balancing Algorithms with Load Estimation for Computational Grid Environments," *IEEE Transactions on parallel and distributed systems,* vol. 18(12), 2007, pp.1675-1686.

[11] D. Grosu, and A.Chronopoulos, "Algorithmic Mechanism Design for Load Balancing in Distributed Systems," *IEEE Transactions on Systems*, Man and Cybernetics, vol. 34(1), 2004, pp.77 – 84.

[12] Albert Y. Zomaya, and Yee-Hwei Teh, "Observations on Using Genetic Algorithms for Dynamic Load-Balancing," *IEEE Transactions on parallel and distributed systems*, vol. 12, 2001, pp. 899- 911.

[13] D. Gu, L. Yang, and L. Welch., "A predictive, decentralized load balancing approach," *Proceeding of 19th IEEE International Parallel and Distributed Processing Symposium*, 2005, pp. 131-133.

[14] Alvaro Arenas, Michael Wilson, Brian Matthews, "On Trust Management in Grids," *Proceedings of the 1st*

international conference on A utonomic computing and communication systems, , 2007 Article No.: 4.

[15] I. Foster and C. Kesselman, "The Grid: Blueprint for a New Computing Infrastructure", Second Edition, 2004.

[16] Huraj L., Reiser H., "VO Intersection Trust in Ad Hoc Grid Environment," *Fifth International Conference on Networking and Services*, 2009, pp. 456 – 461.

[17] V. VetriSelvi, Ranjani Parthasarathi , "Trace Based Mobility Model for Ad Hoc Networks," *Third IEEE International Conference on W ireless and Mobile Computing, Networking and Communications*, , 2007 pp. 81 – 88.

[18] Bin Lu and Hongbin Zhang "Grid Load Balancing Scheduling Algorithm Based on Statistics Thinking," *The 9th International Conference for Young Computer Scientist*, , 2008 pp. 288 – 292.

[19] Farag Azzedin and Muthucumaru Maheswaran, "Integrating Trust into Grid Resource Management Systems", *International Conference on P arallel Processing*, , 2002 ,pp. 47 – 54.

[20] Gang Li, Hongmei Sun, Huahao Gao, Haiyan Yu, Yue Cai, "A survey on W ireless Grids and Clouds", *International Conference on Grid and Cooperative Computing*, 2009 , pp. 261 – 267.

[21] Dimitri Bertsekas, Robert Gallager, "Data Networks", 2nd Edition, *Prentice-Hall India*, 1999 , pp. 174-176.

[22] http://pcl.cs.ucla.edu/projects/glomosim, 2000.

[23] http://www.i-u.de/schools/hellbrueck/ansim.