Trust Based Load Distribution in a Data Grid Environment

Shaik Naseera,
Sri Venkateswara University, Tirupati, India

Dr. K.V.Madhu Murthy
Professor, Department of CSE
Sri Venkateswara University, Tirupati, India

Abstract— **The basic idea of a load balancing system is to transfer partial of system workload from busy nodes to some idle nodes. The target resource selection for task migration is a key factor in the load balance strategy. As grid is dynamic in nature, resources may join or leave the grid at any instance of time. It's inefficient to select a target resource that often leaves the grid and hence the tasks may be rescheduled for several times. When the cost of load transmission is not negligible, the overhead of the load balancing strategy will be unacceptable. In that case, the cost of propagating system load will be extremely high. To improve the efficiency of the load balancing process, one can select trustworthy resources from the list of available resources. In this paper we present a load distribution strategy that considers the trustworthiness of the resource as a benchmark for participation in the load balancing process. The simulation results show that the load distribution among the trustworthy resources results better system throughput compared to load distribution to all available resources in the grid.**

Keywords-Data Grid, Load Distribution, Communication Overhead, Trust, Resource Reliabilty

I. INTRODUCTION

To provide a high performance computing environment is the one of the primary goals of the grid computing. The computation load on different work units may differ significantly. This difference becomes the major limitation of the system performance and hence cause the under utilization of system resources.

To utilize the spare capacity of these resources, the workload is distributed from overloaded resources to under loaded resources. As grid is dynamic in nature, it is hard to realize the load balance in such environment. The load balancing for these resources in the grid network is influenced by the factor of resource reliability. Resource reliability is a measure of fraction of time for which a resource is available for use. Grid network is a combination of reliable and unreliable set of resources. A reliable resource provides better service compared to unreliable resource.

Load distribution strategy in grid environment deals, when one resource is overloaded it make use of unused computing power in the network by distributing the load from overloaded resource to an under loaded resource. When the load is migrated from the overloaded resource (sender) to the under loaded resource (receiver), the receiver must be reliable enough such that it completes the assigned task with the assured performance. Therefore, resource discovery process plays an important role in the load balancing process.

To minimize the potential failure in allocating the resources in load balancing process, we emphasize on resource discovery by taking into account the trustworthiness of the resource providing the service. The trust value of a resource is computed by integrating the direct trust and reputation of the resource.

The reminder of the paper is structured as follows. Section 2 discusses about the related work. Section 3 discusses about the load balancing problem in grid, section 4 discusses about the grid framework, section 5 discusses about the simulation results and section 6 presents the conclusions and future enhancements.

II. RELATED WORK

The load balancing strategies are divided into two types: static load balancing such as RRS, WRRS, LCS, DHS, RS [2-5] and dynamic load balancing strategies such as SWP arithmetic [4-5]. In static load balancing approach, the load states in the cluster are known and in dynamic load balancing approach, the load states are unknown before load balancing process Grid is a dynamic environment, resources may join or leave the grid at any instance of time. Therefore it is hard to realize the load balance among the nodes offering processing power.

Load distribution is essential for efficient utilization of resources in the grid environment. Therefore our work is focused on the utilization of the resources that offer reliable service. The load distribution policy considers the reliability of the resources for participation in the load distribution process. The existing load distribution algorithms like RAND [21], ACWN [22], PRAND [23], LADE [24], PACWN [25], LADF [26] consider the workload of the processors and adapt a load distribution rule to distribute the load among the available processors.

Much research on load balancing in P2P systems are also carried out. Buers et al. proposed in [14] a simple "the power of two" load balancing strategy. In their approach, multiple hash functions are used to generate the file_id's. In case a new file is inserted, it will be given multiple identifiers, the file will be stored on the least loaded peer. But it increases computational overhead for routing request and it is static allocation algorithm. Zhiyong and Bhuyan proposed a novel load balancing algorithm [19], in which file access history and peer heterogeneity properties are taken into account to determine the load distribution. The algorithm dynamically performs the load redistribution during system running time if overloaded peers are appeared. Some algorithms [27-29] use the concept of virtual servers and peer heterogeneity to address the load imbalance state.

Research on load balancing in grid, Dobber [15] proposed a dynamic load balance strategy which predict the future load situation according to history information and reschedule newly incoming request to balance the overall system load. Algorithms in [16][17] mainly focuses on selection of suitable jobs and bind them with suitable resources. A fitted random sampling based load balancing scheme proposed in [12] eliminate the need for monitoring the resource availability.

The success of load balancing process is mainly depending on the communication overhead. In a dynamic environment, to limit the cost of propagating system load one can select target from its neighbors [6][11]. Research efforts using agent-based load balancing strategies [18] use agents to exchange load information among different resources. An ant-like self-organizing agent's mechanism is introduced and achieved overall grid load balancing through a collection of very simple load interactions [3].

Our work is mainly inspired from the work done by authors in [1] for trusted grid computing with security assurance at all resource sites. The authors have proposed a new fuzzy-logic based trust model for trusted grid computing with security assurance. They designed a secure grid outsourcing system for secure scheduling a large number of autonomous and indivisible jobs to grid sites and observed considerable performance gain for large workloads under risky conditions. We adapted the fuzzy-based task mapping proposed in [1] in our trust-aware load distribution problem for task mapping to trustworthy resources and improve the performance of the load distribution process.

III. LOAD BALANCING PROBLEM IN GRID

Workload and resource management are two essential functions provided at the service level of the grid software infrastructure. To improve the global throughput of the grid, workloads have to be evenly scheduled among the available resources. When the load/task is distributed from the overloaded (sender) resource to the under loaded (receiver) resource, the receiver must be reliable enough and must complete the migrated task with the assured Quality of Service (QoS). A QoS is the ability to serve a job by providing quality and reliable resource for completing the assigned task. Selection of a quality and reliable resource yield excellent and quality results. Grid provides facility for sharing of a large scale, distributed and heterogeneous computing resources [7]. Reliable resources must be selected for participation in the load balancing process so as to reduce the job turn-around time and hence improve the system throughput.

Several attempts have been made to select resources based on user requirements and resources availability, but to the best of our knowledge there has been no attempt directly address the load balancing process to provide selection of participating resources on the basis of trust. The success of a load balancing algorithm mainly depends on the reliability of the participating resources in the load balancing process. Hence overloaded resource must trust the performance of an under loaded resource.

A. Problem Specific to Grid Computing

Grid is a heterogeneous environment in which resources span across the multiple administrative domains in a wide spread geographical area. The resources in the grid may join or leave the grid at any instance of the time. It means the probability of some resource failures is naturally high. Efficient resource management techniques are required to monitor the dynamic behavior of the resources. Due to this nature, the load balancing process becomes more complex in grid compared to traditional homogeneous parallel and distributed systems. These observations notifies that it is very difficult to achieve load balancing for all the under loaded resources in the grid, rather we propose a method to distribute the load among the trustworthy resources that guarantee the QoS commitment.

B. Notion of Trust

In this paper, we use a trust model that deal with behavior trust. We follow the definition of trust proposed by Muthucumaru in [9] as below.

"Trust is the firm belief in the competence of an entity to act as expected such that this firm belief is not a fixed value associated with the entity but rather it is subject to the entity's behavior and applies only within a specific context at a given time".

Table 1: Description of the trust levels

Trust level range	Description
1-2	very untrustworthy
3-4	untrustworthy
5-6	medium trustworthy
7-8	trustworthy
9-10	very trustworthy

The firm belief is a dynamic value and spans over a set of values ranging from very trustworthy to very untrustworthy [30], and in our work we mapped the trust value with numeric range on a 10 point scale as shown in Table 1. We compute the trustworthiness of the resource based on the direct trust and reputation as shown in (1).

Let Ri be the ith resource in grid G. Ti be the trust-value of the resource Ri. We define Ti as

$$Ti(Ri) = \text{Direct-Trust} \times \alpha + \text{Reputation} \times \beta \quad (1)$$

Where $\alpha + \beta = 1$.

Direct-Trust is a measure of the trustworthiness of a resource based on consumer's experience. Reputation is an expectation about a resource behavior based on information or observation about its past behavior [10].

We maintain a table of Direct-Trust and Reputation for each resource in the grid. Every resource is associated with an appropriate type counter for Direct-Trust and Reputation, which gets automatically updated when the resource completes/fails the assigned task.

C. Trust-Aware Load-Distribution Model

The load distribution process is composed of two steps. The first step called trust-driven step, in which the information about the trustworthiness of the resources are retrieved and trust-value is computed as in (1). In the second step, the task mapping step deals with the process of distributing the load to the qualified resources using fuzzy-logic by aggregating the capability of resource. The trust-aware load distribution system (TLDS) proposed in this paper evaluate the trust-value of all grid resource providers and facilitate the selection of suitable resource for load distribution. We adapt the fuzzy-logic based load distribution model for task mapping from overloaded resource to under-loaded resources. The flow of TLDS is depicted as shown in Fig 1.

- Trust Value Computation : This is the first step of a TLDS in which the trust value of a resource is computed by aggregating the direct-trust and reputation of the resource using parameter weighting factors α and β in (1) based on the user recommendations.
- Trust-Value Classification : Based on the trust value, the resources are classified into four distinct strata such as "Very Trustworthy", "Trustworthy", "Un-Trustworthy" and "Very Un-Trustworthy" according to AbdhulRahman and Halies [10]. The advantage of trust strata is that the selection between the resources is simplified.

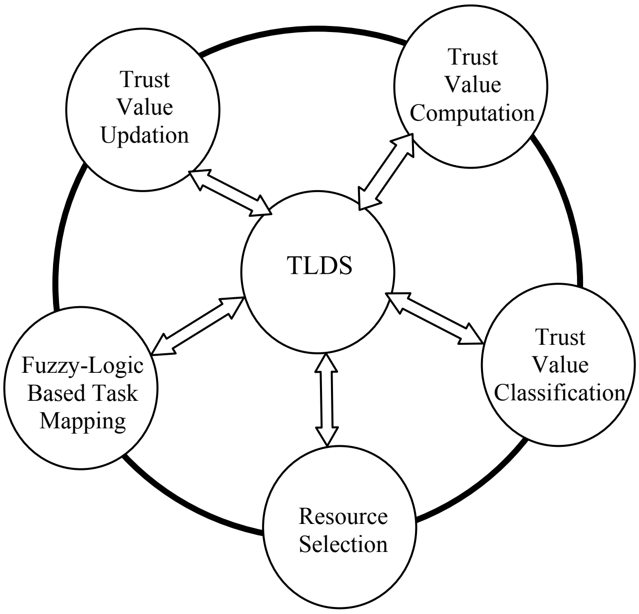

Fig 1. Trust-Aware Load Distribution Model

- Resource Selection: The trust strata facilitate the selection of suitable resources for the participation in the load balancing process. Our load distribution model selects the trustworthy and very-trustworthy resources for the participation in the load balancing process.
- Fuzzy-Logic Based Task Mapping: As shown in Fig 2, Task mapping from overloaded resource to under-loaded resources are done by using fuzzy-logic.
- Trust value updation: To reflect the dynamism in grid, the trust value of the resource is updated based on the state of the job executed. The trust-value of a resource is updated by changing the appropriate type counters associated with the reputation and direct-trust indices of a resource.

D. Motivations for Trust-Aware Load Distribution:

A resource must have certain basic capabilities before it can participate in the load balancing process. These basic capabilities include assured CPU rating, lower workload level and sufficient storage capacity. The workload level and storage capacity is dynamic and vary with respect to time. Though the resource is capable of executing the migrated task from the overloaded resource due to some intermittent resource failures makes the resource incapable of executing the assigned task which was previously acceptable at that resource.

To make the load balancing process efficient, it is necessary to consider the trustworthiness of the participating resource. Therefore the matching of overloaded resource jobs to under loaded trustworthy resources is an important issue to be considered. There is no significant work that addresses this issue directly. In this paper we propose a trust based approach for load balancing problem. In our work, we propose a two level fuzzy-logic based trust model to enable the aggregation of trust parameters with resource capabilities as shown in the Figure 3. The fuzzy logic based task mapping makes the mapping of tasks to resources easy.

IV. GRID FRAMEWORK

We carried out the simulation for a grid topology proposed by Yagoubi and Slimani in [11].

A. Grid Topology.

Grid computing as shown in Fig. 4 is finite set of G clusters C_k, interconnected by gates gt_k, k Є {0,...,G-1}, where each cluster contains one or more sites S_{jk} interconnected by switches SW_{jk} and every site contains some Computing Elements CE_{ijk} and some storage elements SE_{ijk}, interconnected by a local area network.

The grid topology is viewed as a hierarchical grid structure in which the leaves of the tree correspond to the computing elements of a site, and root is a virtual node associated to the site and is denoted by G/S/M, where G is the number of clusters that compose a grid, S the number of Sites in the grid and M the number of Computing Elements. The generic model is a non cyclic connected graph where each level has specific function.

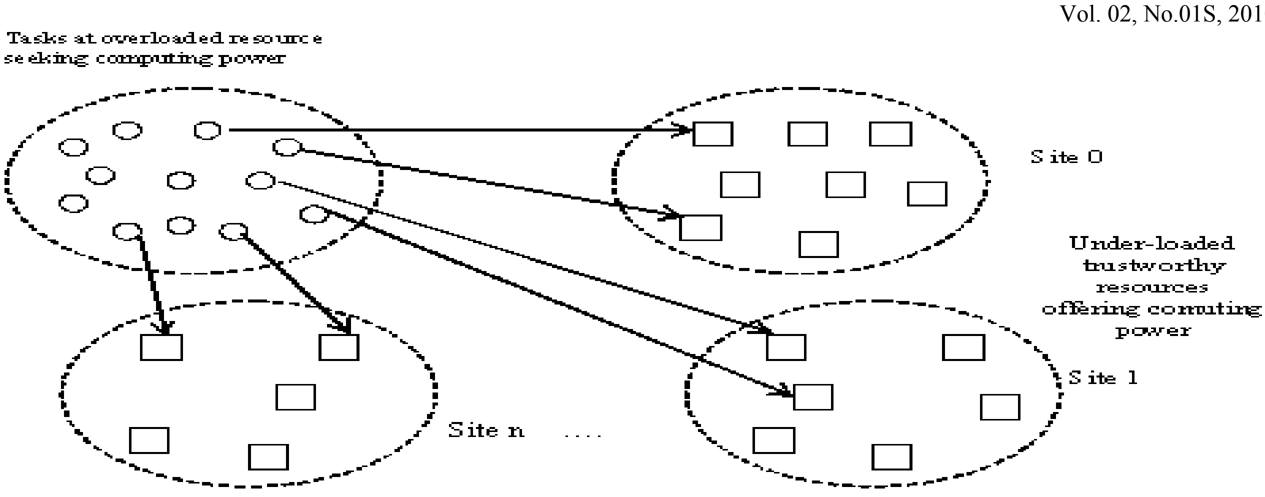

Fig 2. Distribution of Load from overloaded resource to under loaded trustworthy resource

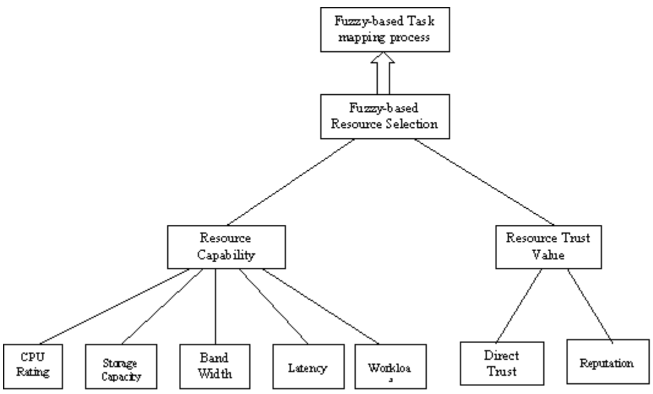

Fig 3. Fuzzy-logic based parameter aggregation

Table 2: Reliability level

Reliability	Failure Rate
Highly Reliable (HR)	0% ≤ failure rate ≤ 3%
Reliable (R)	3% ≤ failure rate ≤ 10%
Marginable Unreliable (MU)	10% ≤ failure rate ≤ 30%
Unreliable (U)	30% ≤ failure rate ≤ 60%
Highly Unreliable (HU)	60 % ≤ failure rate ≤ 90%
Very Highly Unreliable (VHU)	90% ≤ failure rate ≤ 100%

V. Experimental Results

To select the list of resources for participation in the load distribution process, we have adapted the reliability levels for various categories of the resources proposed by Nathan [13] as shown in Table 2.

To evaluate the performance of our load distribution strategy, we have developed a load balancing grid simulator implemented using java. We implemented grid topology proposed in [11] and experiments are conducted for varying number of CE's, SM's and CM's. We conducted the experiment for CE's ranging from 20 to 150 connected to SM's and CM's ranging from 2 to10.

Grid is a heterogeneous environment, each CE has different trustworthiness and varying number of CE's are connected to sites. Hence grid possesses resources in different reliability range sets as depicted in table 3.The resource participation from the reliable set is more compared to participation from an unreliable resource set. The fuzzy based task mapping identifies the list of qualifying resources from the resource set based on the threshold value. The threshold value represents minimum trust value required for a resource for participation in the load distribution process.

Table 3: Resource Set

Resource Set	HR	R	MU	U	HU	VHU
Reliable	60%	20%	10%	4%	3%	3%
Un Reliable	5%	10%	30%	20%	20%	15%
Mixed Set	20%	20%	20%	20%	10%	10%

The experiment is conducted under three different test cases as listed below with a threshold value of 60%. .

A. Best-Case: The load distribution process is considered to be a best-case if the load distribution is completed among the intra site CE's.
B. Average-Case: The load distribution process is considered to be in Average case, if the load distribution is to be carryout with the participation of inter site CE's.

C. Worst-Case: The load distribution is in Worst case, if the load distribution is to be carryout with the participation of inter cluster CE's.

(i) Load Distribution Time Vs CE's

The simulation time units(iterations) taken for the load distribution process for all the cases are depicted as shown in Fig 4, Fig 5 and Fig 6. we observed that the load distribution process to reliable CE's is less compared to the load distribution process to all CE's.

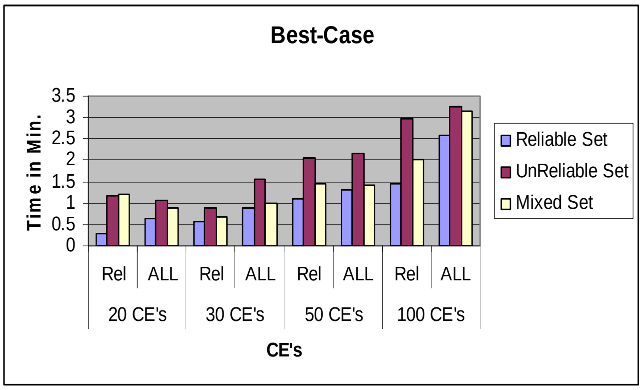

Fig 4. Best-Case Load Distribution

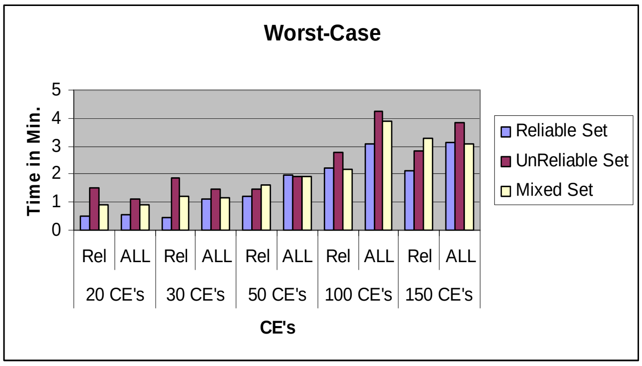

Fig 5. Average-Case Load Distribution

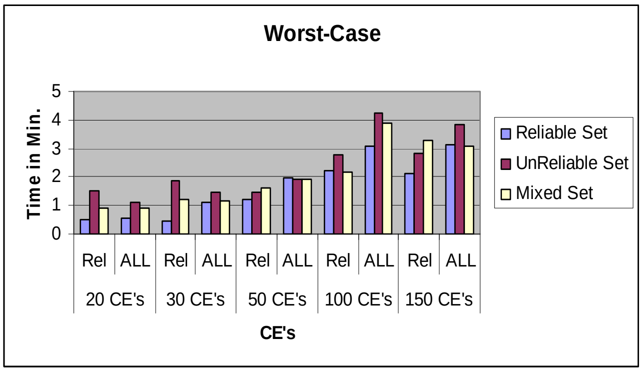

Fig. 6. Worst-Case Load Distribution

(ii). Communication overhead Vs CE's

Communication overhead is a measure of the number of messages exchanged between the sender and receiver during the load distribution process. The results are depicted as shown in the Table 4 for various resource sets. We observe that the communication overhead involved in the load distribution process is less for the participation of reliable CE's compared to the participation of all CE's.

Table 4: Communication Overhead

CE's	Reliable Set		Unreliable Set		Mixed Set	
	All	Reliable	All	Reliable	All	reliable
20	37	32	96	34	100	31
50	102	49	190	46	132	35
75	99	59	284	64	182	72
100	165	162	516	195	437	158

(iii) Failure rate Vs CE's

Failure rate is a measure of number of unsuccessful attempts made by the resources during the execution of the assigned tasks. The results for reliable set, un-reliable set and mixed set are depicted in the Fig 7, Fig 8 and Fig 9 respectively. The results shows that in any resource set, the total number of unsuccessful attempts are more with all CE's participation compared to the participation of the reliable CE's. Hence the given jobs are completed at faster rate with the reliable CE's rather than all CE's.

VI. CONCLUSIONS

It is demonstrated from the results that the load distribution process with trust worthy resources completes the execution of the jobs faster than the load balancing among all the resources in the grid. The experiment is conducted for CE's ranging from 20 to 100 and it is observed that the TLDS performs better for any resource set with more than 20 CE's in the Grid.

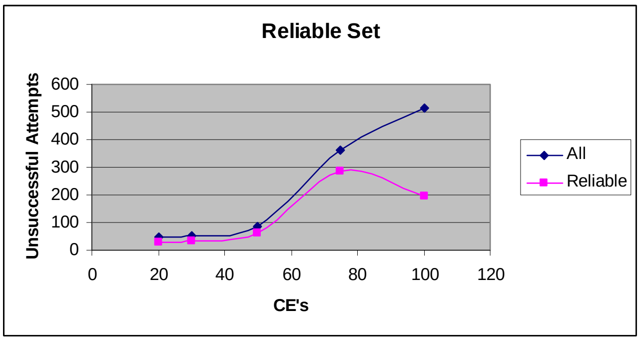

Fig. 7. Unsuccessful Attempts in Reliable Set

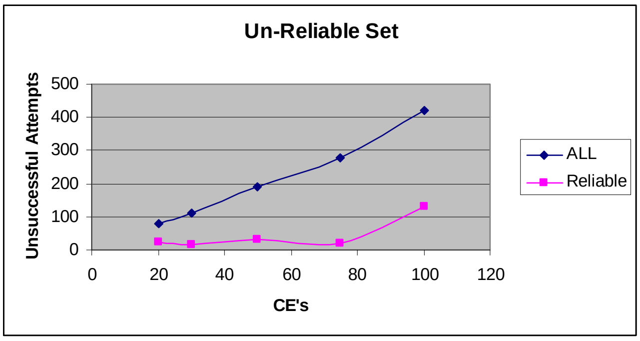

Fig 8. Unsuccessful Attempts in Un-Reliable Set

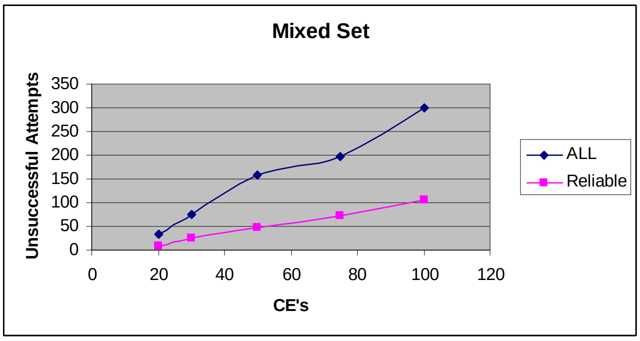

Fig 9. Unsuccessful Attempts in Mixed Set

REFERENCES

[1] Shanshan S, Hwang K and Yu-Kwong K, "Trusted Grid Computing with security Binding and Trust Integration", Journal of Grid Computing, 2005.

[2] Cardellini V, Colajanni M, P.S. Yu, "Redirection Algorithms for load sharing in distributed web server systems", Proceedings of 19th IEEE International Conference on Distributed Computing Systems, 1999, pp: 528-535.

[3] Junwei Cao. "Self-Organizing Agents for Grid Load Balancing", In Proceedings of 5th IEEE/ACM International Workshop on Grid Computing (GRID 2004), in conjunction with IEEE/ACM Supercomputing Conference (SC 2004), Pittsburgh, PA, USA, November 2004, pp. 388-395.

[4] T Schroeder, S Goddard, B. Ramamurthy, "Scalable Web Server Clustering Technologies", IEEE Network, May/June 2000, pp: 38-45.

[5] H. Bryhni, E.Klovning, O. Kure, " A Comparision of Load Balancing Techniques for Scalable Web Servers", IEEE Network, July/August 2000. PP 58-64.

[6] Shu W, Kale L.V, "A Dynamic Scheduling Strategy for the Chare2Kernei Systems" In Proceedings of Supercomputing '89, pp 389-398, Reno, Nevada, 1989.

[7] Foster I and Kesselman C, Globus: The Grid Blueprint for a future computing Infrastructure, Morgan Kaufmann, 1999.

[8] H.D. Karatze, "Job Scheduling in Heterogeneous Distributed Systems", Journal of Systems and Software 1994, pp: 203-212.

[9] Azzedin F, Maheswaran M, "Evolving and Managing Trust in Grid Computing Systems", In Proceedings of IEEE Canadian conference on Electrical Computer Engineering, 2002.

[10] Abdhul Rahman A, Hailes S, "Supporting Trust in Virtual Communities", Hawaii International Conference on System Sciences, 2000.

[11] Yagoubi B and Slimani Y, "Dynamic Load Balancing Strategy for Grid Computing", PWASET volume 13, May 2006, ISSN 1307-6884.

[12] O.A. Rahmeh, P. Johnson and S.Lehmann, "A fitted Random Sampling Scheme for Load Distribution in Grid Networks" PWASET vol 24, October 2007, ISSN 1307-6884.

[13] Nathan Griffiths and Kuo-Ming Cho. Experience-Based Trust: Enabling Effective Resource Selection in a Grid Environment. Research Report LS-RR-409. Department of Computer Science, University of Warnick, Dec 2004.

[14] J.W. Byers, Considine J and Mitzenmacher M. "Simple load balancing for distributed hash tables", In proceedings of 2nd international workshop on peer-to-peer systems (IPTPS), Berkeley, C.A, pp 80-87, 2003.

[15] Menno Dobber, Ger Koole, Rob van der Mei. Dynamic load balancing experiments in a grid. In proceedings of the 5th IEEE International Symposium on cluster and the grid, vol 2, page 1063-1070, Cardiff, UK, 2005.

[16] Livny M and Melman M. Load balancing homogeneous broadcast distributed systems. In proceedings of the ACM Computer Network Performance eSymposium, Page 47-55, College Park, Maryland, United States, 1982.

[17] Kunz T. The influence of different workload descriptions on a heuristic load balancing scheme. IEEE transaction on software engineering, 17(7): 725-730, 1991.

[18] Junwei Cao. Agent-based grid load balancing using performance driven task scheduling. In proceedings of 17th IEEE International parallel and distributed processing Symposium (IPDPS 2003), Nice, France, April 2003.

[19] Zhiyong Xu and Laxmi Bhuyan. Effective load balancing in P2P systems. In proceedings of 6th IEEE international symposium on cluster computing and grid (CCGRID'06), Singapore, May 2006.

[20] Menno D, Koole G, Rob van der mei. Dyanamic load balancing experiments in a grid. In proceedings of 5th IEEEinternational symposium on cluster computing and grid (CCGRID'05), Cardiff, UK, May 2005.

[21] R.M.Karp and Y. Zhang. A randomized parallel branch-and-bound procedure. In proceedings of the 20th annual ACM symposium on theory of computing, pp 290-300, 1998.

[22] L.V. Kale. Computing the performance of two dynamic load distribution methods. In proceedings of international l conference on parallel processing, pp 8-12, 1988.

[23] S. Tschoeke, R. Luling and B. Monien. Solving the traveling salesman problem with a distributed branch and bound algorithm on a 1024 processor network. In proceedings of international parallel processing symposium, pp 182-189, 1995.

[24] A.B.Sinha and L.V.Kale. A load balancing strategy for prioritized execution of tasks. In proceedings of 7th international parallel processing symposium, pp 230-237, 1993.

[25] C.Z. Xu and F.C.M. Lau. The generalized dimension exchange method for load balancing in k-ary n-cubes and variants. Journal of parallel and distributed computing. 24(I):72-85, January 1995.

[26] C.Z.Xu, B Monien, R. Ldling and F.C.M.Lau. An analytical comparison of nearest neighbor algorithms for load balancing in parallel computers. In proceedings of 9th international parallel processing symposium, pp 472-479, 1995.

[27] B. Godfrey, K.Lakshminarayana, S. surana, R.M. Karp ans I Stoica, Load balancing in dynamic structured P2P systems, In proceedings of IEEE INFOCOM 2004.

[28] J. Ledlie and M. Seitzer, " distributed secure load balancing with skew, heterogeneity and chum, In proceedings of IEEE INFOCOM, 2004.

[29] A. Rao, K.Lakshminarayana, S. surana, R.M. Karp ans I Stoica, Load balancing in structured P2P systems, In proceedings of the 2nd international workshop on peer-to-peer systems(IPTPS), Berkeley, CA, pp 68-79,2003.

[30] Farag Azzedin and Ali Rizvi. Understanding and Measuring trust evolution process in peer-to-peer computing systems, PWASET Volume 21 May 2007 ISSN 1307-6884