www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 7 July, 2014 Page No. 7177-7180

Efficient Load Balancing in Clusters in Hierarchical Structure

Viney Rana, Sunil Kumar Nandal
(Dept. Of Computer Science& Engineering)GJUS&T, Hisar, Haryana, India
Vinu.rana67@gmail.com
nandal_sunil@yahoo.com

Abstract--***Load balancing is a process of transferring load from one node to another to increase the performance of the whole system as each node is equally loaded in structure. The aim of this paper is to implement load balancing in hierarchical structure by taking different size of cluster at different level and compare its results with simple load balancing approach. The results show that our proposed hierarchical algorithm is more efficient than simple load balancing.***

Keywords--Load balancing, clustering, hierarchical structure, load index, efficiency, node management.

I. INTRODUCTION

Load balancing is the allocation of the workload among a set of co-operating computational elements (CEs). In large-scale distributed systems, the CEs are physically or virtually distant from each other, there are communication related delays that can significantly alter the expected performance of load-balancing policies that do not account for such delays. This is a problem in systems for which the individual units are connected by means of a shared communication medium such as the Internet,wireless LANs, ad-hoc networks. Moreover, the system performance may greatly vary since it incorporates heterogeneous nodes that are not necessarily dedicated to the application at hand. In such cases, an actual implementation becomes necessary to understand the load-balancing strategies and their reactions when employed in several environments since mathematical models may not always capture the unpredictable behaviour of such systems.

The scheduling algorithm which use the load- balancing approach are based on the fact that an even load distribution helps in better resource utilization. This is done by transferring the workload from heavily loaded nodes to lightly loaded nodes to ensure good overall performance, measured by the response time of processes. From the resource point of view, the metric to measure performance is the total system throughput.

Designing Issues in Load-Balancing Algorithms

Various issues are involved in designing a good load-balancing algorithm, such as deciding policies for load estimation, process transfer, static information exchange, location, priority assignment, and migration limitation.

The load estimation policy determines how to estimate the workload of a node in a distributed system. The classification of load estimation policies are as -

- Measuring the number of processes running on a machine
- Capturing CPU busy time

The process transfer policy decides whether the process can be executed locally or there is a need for remote execution. There is a need to decide a policy which indicates whether a node is heavily or lightly loaded, called the *threshold policy*. The classification of threshold policy is as –

- Single- level threshold
- Two- level threshold

*Static information exchange policy*determines how the system load information can be exchanged among the nodes. The proposed load- balancing algorithm uses the following state information exchange policies –

- Periodic broadcast
- Broadcast when state changes
- On-demand exchange of state information
- Exchange by polling

Location policy determines the selection of a destination node during process migration. Location policies are classified as –

- Threshold policy
- Shortest location policy
- Bidding location policy
- Pairing policy

*The priority assignment policy*determines the priority of execution of a set of local and remote processes on a particular node. The priority rules can be classified as –

- Selfish priority assignment policy
- Altruistic priority assignment policy
- Intermediate priority assignment policy

*The migration limiting policy*determines the number of times a process is allowed to migrate. Two policies are used –

- Uncontrolled
- Controlled

Clustering

A definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”. A cluster is basically a collection of objects which are “similar” between the
m.

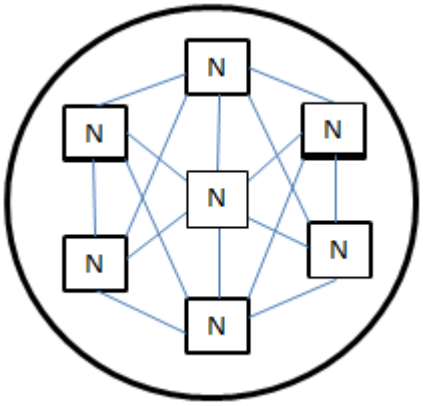


Fig1 Cluster of seven node using mesh topology

A computer cluster consists of a set of loosely connected or tightly connected computers that work together .The nodes in a cluster are usually connected to each other through fast local area networks ("LAN"), with each node (computer used as a server) running its own instance of an operating system. Clusters are usually made to improve performance and availability over that of a single computer, also much more cost-effective than single computers of comparable speed or availability.Computer clusters are used everywhere ranging from small business clusters with a number of nodes to some of the fastest supercomputers in the world.

"Load-balancing" clusters are configurations in which cluster-nodes share computational workload to provide better overall performance. A server cluster may assign different queries to different nodes or workstation, so the overall response time will be optimized.Different approaches to load-balancing may significantly differ among applications, for example a high-performance cluster used for scientific computations would balance load with different algorithms from a web-server cluster which may just use a simple round-robin method by assigning each new request to a different node.

Cluster management

The major challenges in the use of a computer cluster is the cost of administrating it which can at times be as high as the cost of administrating N independent machines, if the cluster has N nodes.

II. RELATED WORK

Research work is done in the field of load balancing by many researchers. Load balancing algorithm is applied on different existing technology for sharing load in network or client server model. Paul Werstein, Hailing Situ et. al. have propose a load balancing algorithm for distributed use of a cluster computer which uses load information including CPU utilisation, CPU queue length,memory utilisation and network traffic to decide the load of each node. The experimental results when compared to an algorithm using only CPU queue length shows that the proposed algorithm performs well. RutujaJadhav, I Priyadarshini,et. al. [1] have focused to achieve better performance results even in case of node failure using regenerative theory in dynamic load balancing. The proposed method assumes n nodes and each node is associated with back up node. Back up nodes will not serve the tasks but will transfer the un-served tasks to other under loaded nodes. At the balancing instant overloaded nodes are transferring extra load to the under loaded nodes, performing load balancing of the system. Even in case of node failure back up nodes take care of the un-served tasks. Experimental results conclude that even in case of node failure or regeneration event their proposed method will help in improving the performance.

Siu-Cheung Chau, Ada Wai-CheeFu[3]: proposes a simple and efficient load balancing methodto balance loads between computing clusters that are far apart from each other. An improved dimension exchange method (DEM) for synchronous load balancing for hypercube architecture is presented. The improved DEM requires the same number of communication steps and roughly the same task migrations comparing to the original dimension exchange method. Results shows that the improved DEM provides much better load balancing for computing clusters that are quite far apart.Jaswinder Pal Singh[15]: proposes algorithm to parallelize adaptive grid generation on a cluster by using a variety of partitioning and dynamic load balancing procedures such as global graph based partitioning, local graph based method, refinement tree partitioning etc. and proposed a new algorithm. Jean Ghanem[2] propose a software implementation architecture where several distributed load-balancing strategies could be tested and verified under different environments. They experimentally investigate network delays that are the main factor in degrading the performance of the load distribution strategies. They develop an improved policy that adapts to the system parameters such as transfer delays, connectivity, and CE computational power.

Abdel Rahman H. Hussein, SufianYousef, et. al.[5] propose an enhancement on weighted clustering algorithm (EWCA), that leads to a high degree of stability in the network and improves the load balancing. A comparison was conducted to measure the performance of our algorithm with original WCA in terms of numbers of clusters formed with satisfy load balancing , topology stability, and number of clusterhead change. The simulation results show that their enhancement clustering algorithms have a better performance on average. Yong Meng TEO, RassulAYANI[8]: Their research focuses on an experimental analysis of the performance and scalability of cluster-based web servers. Dispatcher and web server service times used in the simulator are determined by carrying out a set of experiments on the testbed. They observe that the round robin algorithm performs much worse in comparison with the othertwoalgorithms (least connected and least loaded) for low to medium workload. S.Ayyasamy, S.N. Sivanandam[13] propose a cluster based replication architecture for load-balancing in peer-to-peer distribution systems. It is an intelligent replica placement technique and it also consists of an effective load balancing technique.Experimental results shows that proposed architecture attains less latency and better throughput with reduced bandwidth usage.

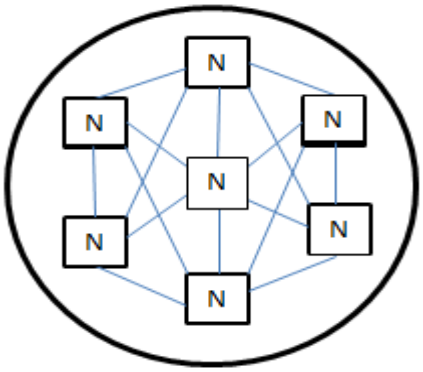
M. Banikazemi, S. Prabhu, et.al.[11] proposes a new profile-based approach to perform load balancing in heterogeneous clusters.They have proposed a new PBLB algorithm which uses application specific information (profile) to come up with a (near) optimal load distribution. They have incorporated the effect of external load into PBLB and hence come up with a complete strategy for static load balancing. Results shows that Performance evaluation of this algorithm on a 12-node testbed indicates that with increased heterogeneity degree, the execution times can be reduced by up to 46%.

III. PROPOSED WORK

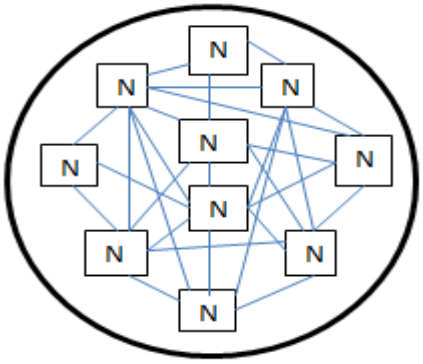
In this paper, a cluster based hierarchical architecture for load balancing in distributed systems is proposed. Load is calculated on the basis of number of processes on a node. Nodes in clusters are similar with equal processing power. A group of heterogeneous cluster is made for load balancing. Size

of cluster is determined by the number of nodes it contains. Homogeneous clusters are placed at one level. An assumption is made that the network is ideal means there is no delay in network or transferring load from one node to another or from one cluster to another cluster. The various simulation steps involved in the procedure are:-

1. Input no. of cluster in the structure, node of nodes in cluster and total capacity of cluster.



(a)



(b)

Fig 2 Structure of cluster for seven node(a) and ten node(b)

2. Processes are generated randomly on the nodes in cluster and calculate the load on each node.
3. Load is calculated on the basis of CPU queue length means number of processes.
4. Randomly elect a cluster head and arrange the load of nodes in ascending order and also calculate overload amount and space of each node.

Similarly more level can be added in the network and also number of cluster. This is the procedureused for load balancing algorithm in hierarchical structure. Mesh topology is used for connected node in cluster.

IV. EXPERIMENTAL RESULT

In this section, we evaluate the performance of the proposed load balancing in hierarchical structure using simulation. We simulated a three level hierarchical structure for various numbers of heterogeneous cluster at different levels. Range of cluster is varying from fifteen to ninety. Thesimulation result is shown below in graph-

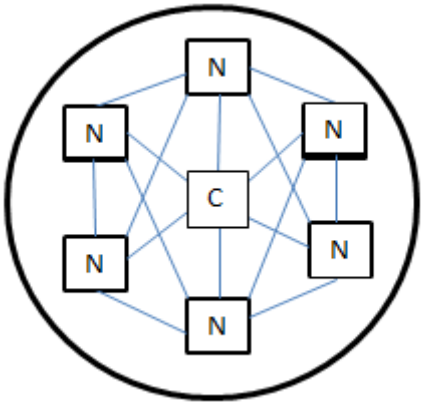


Fig 3 One node is elected as cluster head, here c indicate cluster head

5. Transfer load from overloaded node to idle node to manage the load of cluster.

6. Calculate the total overload amount and space at each cluster and transfer load between cluster at same level.

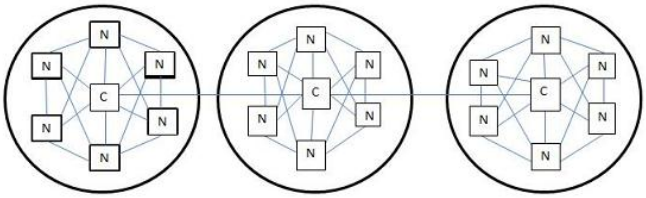


Fig 4 Structure of node and cluster at one level

Cluster heads are connected with each other and load is transfer between clusters through cluster head.

7. Transfer load from one level to another for balancing load in whole structure.

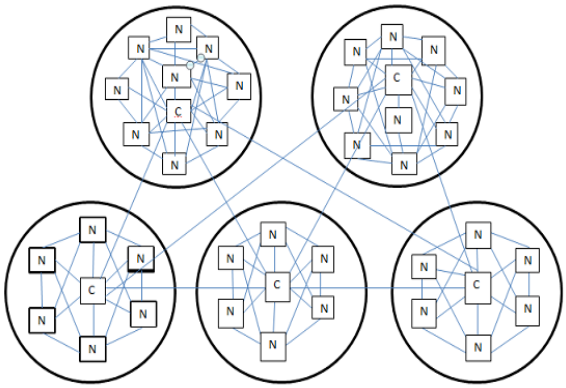


Fig 5 Network structure for twolevels in hierarchical load balancing

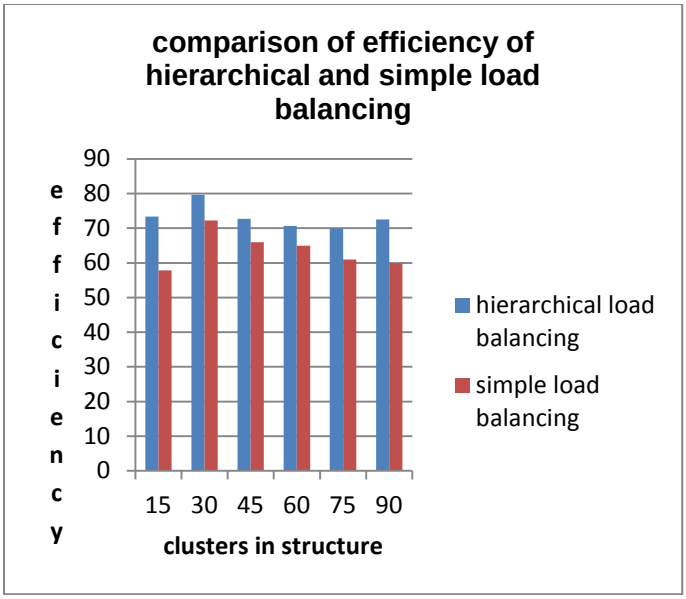


Fig 6 Comparison of simple and hierarchical load balancing algorithm

As shown in the above graph, the hierarchical structure gives better performance than simple load balancing algorithm. Also the management of nodes and cluster become easy in this hierarchical structure.

V. CONCLUSION

In this paper, we proposed a procedure for load balancing in hierarchical structure. The load is calculated on the basis of CPU queue length. We conducted simulation that shows the performance of load balancing in hierarchical structure. The simulation results shows that hierarchical load balancing is more efficient than simple load balancing in grouping of nodes. Also management of clustersbecome easy in hierarchical structure.

REFERENCES

[1] RutujaJadhav,I Priyadarshini,SnehalKamlapur,"Performance Evaluation in Distributed System using Dynamic Load Balancing",International Journal of Applied Information Systems (IJAIS), vol. 2, issue 7, pp. 36-41, 2012

[2] Jean Ghanem, "Implementation of Load Balancing Policies in Distributed Systems",june 2004

[3] Siu-Cheung Chau, Ada Wai-Chee Fu, "Load Balancing between Computing Clusters", Proceedings of the 8th International Scientific and Practical Conference of Students, Post-graduates and Young Scientists. Modern Technique and Technologies, pp. 548-551, 2002

[4] Laercio L. Pilla, Christiane PousaRibeiro, Daniel Cordeiro, Chao Mei, AbhinavBhatele,Philippe O. A. Navaux, Francois Broquedis, Jean-Francois Mehaut, Laxmikant V. Kale, "A Hierarchical Approach for Load Balancing on Parallel Multi-core Systems",41st International Conference on Parallel Processing, pp. 118-127, 2012

[5] Abdel Rahman H. Hussein, SufianYousef, and Omar Arabiyat, "A Load-Balancing and Weighted Clustering Algorithm in Mobile Ad-Hoc Network", may2005

[6] Xiao Qin, Hong Jiang, Adam Manzanares, XiaojunRuan, Shu Yin, "Communication-Aware Load Balancing for Parallel Applications on Clusters",jan 2010

[7] Gae-won You, Seung-won Hwang, and Navendu Jain, "Scalable Load Balancing in Cluster Storage Systems", 2012

[8] Yong Meng TEO, Rassul AYANI, **"**Comparison of Load Balancing Strategies on Cluster-based Web Servers",Simulation, vol. 77, issue 5-6, pp. 185-195, 2001

[9] BelabbasYagoubi, MeriemMeddeber, "Distributed Load Balancing Model for Grid Computing",vol. 12,pp. 43-60, 2012

[10] Hongbo Jiang, ArunIyengar, Erich Nahum, Wolfgang Segmuller, Asser Tantawi, and Charles P. Wright, "Design, Implementation, and Performance of A Load Balancer for SIP Server Clusters",IEEE/ACM Transactions on Networking, vol. 20, issue 4, pp. 1190-1202, 2012

[11] M. Banikazemi, S. Prabhu, J. Sampathkumar,D. K. Panda, T. W. Page, P. Sadayappan, "Profile-Based Load Balancing for Heterogeneous Clusters" **,** 2000

[12] Hongbo Jiang, ArunIyengar, Erich Nahum, Wolfgang Segmuller, Asser Tantawi, Charles P. Wright, "Load Balancing for SIP Server Clusters",IEEE INFOCOM 2009 - The 28th Conference on Computer Communications, pp. 2286-2294, 2009

[13] S.Ayyasamy, S.N. Sivanandam, "A Cluster Based Replication Architecture for Load Balancing in Peer-to-Peer Content Distribution",International journal of Computer Networks & Communications, vol. 2, issue 5, pp. 158-172, 2010

[14] Luis Aversa, AzerBestavros, "Load Balancing a Cluster of Web Servers",Conference Proceedings of the 2000 IEEE International Performance, Computing, and Communications Conference, pp. 24-29, 2000

[15] Jaswinder Pal Singh, "Dynamic Load Balancing for Cluster Computing", 2005

[16] Yong FuHongan Wang, Chenyang Lu, RamuSharat Chandra, "Distributed Utilization Control for Real-Time Clusters with Load Balancing",27th IEEE International Real-Time Systems Symposium, pp. 137-146, 2006

[17] Ossama Othman, Douglas C. Schmidt, "Optimizing Distributed System Performance via Adaptive Middleware Load Balancing", 2002