International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 5, Issue16, Dec – Jan 2016, 59-64
© IASET

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

DECENTRALIZED LOAD BALANCING IN HETEROGENEOUS DISTRIBUTED SYSTEMS USING TRAINING BASED APPROACH

MOHD HAROON[1] & MOHD HUSAIN[2]

[1]Research Scholar (TMU Moradabad), India

[2]Research Supervisor, India

ABSTRACT

Load balancing and job scheduling both are the most important attributes in parallel system and distributed system, once the new jobs are generated, then the scheduler decide whether the generated job is executed, load balancing is one of the important activities, by load balancing technique, load of the entire computing system is balanced, resulting that improving of system throughput, and response time, resource utilization is also improved by the load balancing approach.

A new model is proposed, in this model a new load distribution approach has been defined, and during compile time further reallocation of the load is also defined in this model, by the help of this approach, a processor network has been created, in this network, data movement, data reallocation, process or thread creation is also understandable.

The proposed algorithm is adaptable and scalable, this algorithm is applied in adhoc network, the generated work is randomly distributed among all processors, and if any computing node is out of order then the load reallocation method can justify the further allocation of the load.

KEYWORDS: Distributed System, Load Balancing, Data Arrival Rates, Data Execution Rates

INTRODUCTION

A dynamic load balancing policy is used in distributed systems, the reason being in static load balancing, new generated program or data cannot assign during run time of the processing unit, so all data generated device may be affected, and it signifies the reduction of performance of the entire network,. Every computation unit have a sufficient load for execution , and every computing unit is autonomous ,the distribution of the jobs is dynamically decide by the scheduler and load balancer, during run time dispatcher watch the status of the processing unit and assigned the jobs, some time processing unit is not finished their input jobs at that time new jobs has arrived in this situation , the coming jobs is transferred into a processor waiting queue, and in this case load balancer can further allocate the waiting queue job to available computing unit in a entire network.

The dynamic load distribution is totally different with static load distribution , different approach are used in dynamic load distribution like state-polling of the computing devices, mutual communication between the computing unit, feedback process ,random data assignment process, probabilistic data assignment process, all the job assignment process used to reduce the overhead of data assortment, in general mutual information feedback process is used by the load balancer, supposed machine 1 send the data to machine 2, then before sending the data , machine 1 send the it send data transfer message to machine 2 , if machine 2 is read t accept the data , then machine 2 reply by the message, after that machine 1 send the data, this technique is known as mutual agreement technique, majority of load balancing approach is based on this technique.. A random topology is used for the organization of the distributed. In a network every computing unit have autonomous load for execution, and the distribution of load is randomly decided by the load balancer, the main attribute in this paper is the execution time of the job, execution time of the jobs is also dynamically changed, and it will depend on several parameter like, length of the program, number of loop used in a program, and the processor frequency, CPI of the program etc. similar program can be decided by the similar requirement of the resource by the program, like some program is the CPU bound, and some other are the I/O bound, that similarity can be decided by the resource requirement by the program.

In this model distributed system is supposed have M computing units, and the number of jobs is grouped into N. And the task of group i is arrived at computing machine j with the rate α i, j. Thus, $\alpha i = \sum_{j=1}^{m} \alpha ij$ j=1 is the total incoming rate of class i tasks to the system j. suppose μij is the execution rate of task I at computing node j, we allow μij = 0, is signify the machine j cannot execute the task i, each task class can be executed by at least one machine. In execution rate matrix having several task, and the task can be assigned from the execution rate matrix.

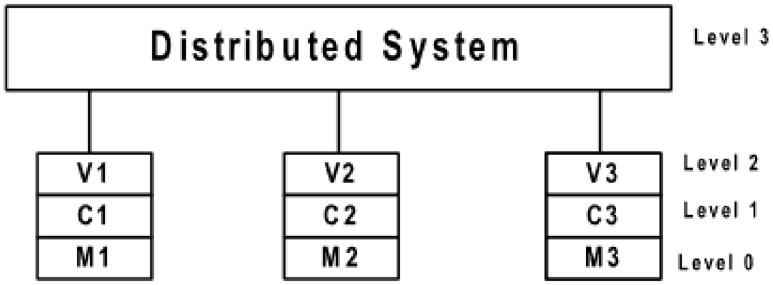
In this model computing unit communicate to opposite computing unit, mean in this model all processing unit is communicate with adjacent computing unit, they communicate with non adjacent computing unit[3,4].

The number of job allotted to each processor by the$Ci = \sum_{k=1}^{p} ti * M$. Where ci is the no of task allocated to each processor.

Several model has obtain on the communication of the processing unit with their adjacent, but in model high bandwidth network are used, and all processing unit communicate with that bandwidth, the quantity of information is transferred by this link in one step, many units of tasks are migrated within the same message across a link in one step, provided moving every task incurs movement of solely an affordable quantity of information [12].

SYSTEM MODEL

A distributed system having m computing unit is supposed, all computing unit is connected with communication network, and the devices shred by n users, total job incoming rate in this model is Ψj. Total incoming rate of user j jobs is $\sum_{k=1}^{n} \Psi k$ .All the jobs in the system are assumed to be of same size [7, 11]. The service rate of node i for the job j is μij.



For the stability of the network, the total job incoming rate must be less than the total processing rate of the system:

$\sum_{i=1}^{m} \Psi j < \sum_{i=1}^{n} \mu i$

For stability, the part of the load at computing node i is not maximizing the service rate of node I, means every time load allotment should be depend on the service rate of the computing devices j.

“Out of user k jobs arriving at node i, the ratio x ij^k of jobs is forwarded upon incoming through the communication means to another node ($j \neq i$) to be processed there.

The remaining ratio $Xijk^{=}1- Xij^k$ processed at node I”.

LINEAR TRAINING BASED SIMILARITY SCHEDULING STRATEGY (LBS)

In linear based scheduling strategy, we suppose the job i is computed on computing machine j with the execution rate μij. Where the decision variables are λ and β ij for i = 1. . . N, and j = 1. . . M.

“Then $\sum_{j=1}^{m}$ βijμij>= λαi   for all i=1, 2, 3….N (1)

$\sum_{i=1}^{n}$ βij $<= 1$ For all j = 1. . . M, (2)

βij ≥ 0 For all i=1, 2….n and j=1, 2, 3…m” (3)

Letβij, i = 1. . . N, j = 1. . . M, be an optimal solution to the allocation load.

The LBs policy data arrival and execution are clearly defined, the computing node are communicating with each other for data exchanged, sending machine send the request message for date transfer with the execution rate [9, 10]. Receiving machine relies the sending machine with the expected completion time after, those if both machines are agree with the execution rate and expected completion time, and then they exchanged their load.

Consider a system with two computing machines and two group of tasks (M = 2, N = 2). Assume initially that α and μ are known by both machines:

$$\alpha = \begin{pmatrix} 1 & 1.45 \\ 1 & 1.45 \end{pmatrix} \quad \mu = \begin{pmatrix} 9 & 5 \\ 2 & 1 \end{pmatrix}$$

The allocation Load gives

$$\begin{pmatrix} 0 & .5 \\ 1 & .5 \end{pmatrix}$$

In this manner, every arriving of the jobs that fit in with above situations, and machine are frequent communicate to each other’s by the link. At the seasons of their approaching, errands that fit in with class 2 are allocated to the machine, either machine 1 or 2 that have the most punctual expected finishing time. Despite the fact that machine 1 has the quickest rate for class 1.

A different performance factor can relate by the given attributes, one is the ready time of machine and expected task execution time of machine. Let RTk, n(m) be the ready time of machine m at the n th mapping event (assignment of a task to a machine) of the kth iteration, and ETC(t, m) is the estimated time to compute task t on machine m[6,8]. A generalized completion time (CT) function of task t on machine m (where λ and η are arbitrary values) is

CT (t, m, n, k) = λ ・ ETC(t, m) + η ・ RTk, n(m).

We can then define the completion time, CT, of a new task t on machine m

SYSTEM PERFORMANCE

performance of the system is depend on several parameter like, speed up factor, execution time of the system, MIPs rate, CPI, clock rates. if the computing system have higher clock rates it signify the fastest execution of the program, if the execution time of the program is minimized, it mean increasing of system performance, increasing of system throughput [12], decreasing the overhead inside the systems, if system performance accordingly, then lod balancer have minimum complexity to migrates the load one computing nodes to another computing nodes.

Speed-up factor: speed up is one of the factors that affect the performance of the system. Speed up is calculated by the following formula.

“S =F (Algorithm, System, Schedule)

S =SPT / CP = sequential processing time/ concurrent processing time

S = actual speedup on an n processor system

$Si=RC/RP \times n$

RC == Relative Concurrency (processor utilization)≤1

RP = Relative Processing requirement≥1

N = number of computational unit

M = number of tasks

Pi= Computation time of task I”

CONCLUSIONS

In this paper a highly decentralized, heterogeneous, scalable system has discussed, and in this model new approach of dynamic load balancing has also suggested, in this model system can communicate with their opposite computation nodes by the communication channel, after that they share their load by the ready time , a linear scheduling algorithms has also discussed, in this algorithm arrival rates of the jobs[11], execution rates of the jobs, and expected communication time of the job has discussed, by the help of above equation incoming jobs are scheduled on the computational units, and machine compute their load without storing the jobs or load in a waiting queue, in this approach length of waiting queue is also minimized , mutual feedback policy is improve by the linear scheduling approach. And expected execution time,

REFERENCES

1. Said Fathy El-Zoghdy. “A Load balancing Policy for Heterogeneous Computational Grids” Vol. 2, No. 5, 2011
2. S. Xian-He, W. Ming, and GHS: “A performance system of Grid computing”, in: Proceedings of the 19th IEEE International Symposium on Parallel and Distributed Processing, 4–8 April 2003.
3. X. Tang and S. T. Chanson. “Optimizing static job scheduling in a network of heterogeneous computers”. In Proc. of the Intl. Conf. on Parallel Processing, pages 373–382, August 2000.

4. Mohd Kalamuddin Ahmad, Mohd Husain, "Required Delay of Packet Transfer Model for Embedded Interconnection Network, International Journal of Engineering" Research, vol 2, issue 1, Jan 2013.

5. Kalamuddin Ahmad, A.A. Zilli Mohd. Mohd. Husain, "A Statistical Analysis and Comparative Study of Embedded Hypercube", International Journal of Computer Applications, Volume 103, Oct 2014.

6. Mohammad Haroon, Mohammad Husain, "Analysis of a Dynamic Load Balancing in Multiprocessor System", International Journal of Computer Science engineering and Information Technology Research, Volume 3, March 2013.

7. Mohammad Haroon, Mohammad Husain, "Different Scheduling Policy for Dynamic Load Balancing in Distributed System", 3rd international conference TMU Moradabad,

8. Mohammad Haroon, Mohammad Husain, "Different Types of Systems Model For Dynamic Load Balancing", IJERT, Volume 2, Issue 3, 2013.

9. Mohammad Haroon, Mohammad Husain, "Different Policies For Dynamic Load Balancing", International Journal of Engineering Research And Technology, Volume 1, issue 10, 2012.

10. Mohd Haroon Ashwani Singh, Mohd Arif, "Routing Misbehavior in Mobile Ad Hoc Network", IJEMR, Volume 4, Issue 5, October 2014.

11. shahaid husain mohd haroon, riyazuddin, "Different technique of load balancing in distributed system: A review paper", IEEE, 2015/4/23.

12. Mohd Haroon, Mohd Husain, Manish Madhav Tripathi, Tameem Ahmad, Vandana "Server Controlled Mobile Agent". International Journal of Computer Applications (0975-8887), 2010/12.

13. M Haroon, M Husain," Interest Attentive Dynamic Load Balancing in Distributed Systems", IEEE xplore.

14. S Srivastava, M Haroon, A Bajaj, "Web document information extraction using class attributes approach", IEEE xplore.