International Journal of Research

eISSN: 2348-6848 & pISSN: 2348-795X Vol-5 Special Issue-13

International Conference on Innovation and Research in Engineering, Science & Technology

Held on 23rd & 24th February 2018, Organized by Tulsiramji Gaikwad Patil College of Engineering & Technology, Nagpur, 441108, Maharastra, India.

# Algorithmic Implementation Of Dynamic Load Balancing In Wireless Lan

Mr.Ashwini Khawaskar[1], Prof.Rajesh Babu[2]

[1]M.Tech Student TGPCET, Nagpur.

[2]Assistent Profesor, CSE Dept, TGPCET, Nagpur.

[1]khawaskarashwini@gmail.com,rajeshbabu.cse@tgpcet.com[2]

***Abstract:***

*Load balancing involves distribution of all computational and communicational activities over two or more processors, links or any other computational devices present in the network.The proposed scheme checks each AP's load state and balances overloaded APs by immigrating some wireless stations towards under loaded one, periodically. Immigration can be accomplished with considering throughput value between wireless station and under loaded APs. The system overload is uniformly distributed among under loaded cells that wireless stations from overloaded cell moves to most under loaded cell first. Each wireless stations has different throughput with different APs, respectively. The load balancing algorithm mostly considers throughput improvement in imbalance state of the system.To reduce this deficiency, dynamic load balancing schemes is used to significant performance gains in terms of system throughput and data loss due to overloading.*

.

***Keywords***

*Dynamic load balancing algorithm, immigration, wireless local area networks.*

## 1. Introduction

The basic idea of a load balancing is to equalize loads at all computers by transferring loads to idle or heavily loaded computers. The Load adjusting is disseminating handling and interchanges exercises equitably over a PC organize so no single gadget is overpowered. It is a procedure to circulate the Workload equally crosswise over at least two PCs, arrange joins, CPUs, hard drives, or different assets. The reasons for stack adjusting are to accomplish ideal asset usage, expand throughput, limit reaction time, and stay away from over-burden. The conventional load adjusting with stack unit relocation starting with one preparing component then onto the next when stack is light on some handling components and overwhelming on some other preparing components. In the event that one of the web server gets over-burden or goes off, all things considered interchange server initiates and access the asked for stack. Load adjusting is finished by relegating specific administration time for each procedure so as to guarantee that few solicitations are dealt with without causing activity. At the end of the day, particular time is appointed to each procedure in the server for its execution and the procedure no more remain in the server once benefit time expands. When stack balancer works effectively, benefit time diminishes for each procedure decreases. Load adjusting is especially helpful for parallel and dispersed frameworks where we need to share the workload to get the most extreme throughput from the System. Most circulated frameworks are portrayed by the dissemination of both physical and coherent components. The design of a disseminated framework is for the most part measured in nature. In WLANs, cell breathing can be actualized by controlling the transmission energy of an AP's signal parcels. Note that we don't change the transmission energy of information parcels to abstain from corrupting customers' execution. All the more particularly, when information parcels decreases, the AP may see higher information bundle misfortunes, or even adjust to a lower sending rate, both of which debase the customer's execution. In correlation, changing the transmission energy of reference point parcels just influence how customers connect with APs, and does not influence the misfortune rate or sending rate of information bundles, which coordinates our objective well. Finding the fitting force task at APs to consequently accomplish stack adjusting is a testing issue. All the more particularly, when information parcels decreases, the AP may see higher information bundle misfortunes, or even adjust to a lower sending rate, both of which corrupt the customer's execution. In correlation, changing the transmission energy of reference point parcels just influence how customers connect with APs, and does not influence the misfortune rate or sending rate of information bundles, which coordinates our objective well.To balancing the load is particularly useful for parallel and distributed systems where we have to share the workload to get the maximum throughput.

## 2. Proposed load balancing algorithm

Load balancing is a technique of distributing the total load to the individual nodes of the collective system to the facilitate networks and resources to improve the response time of the job with maximum throughput in the system. The important things which said about load balancing are estimation of load, load comparison, different system stability, system performance, interaction between the nodes, nature of work to be transferred, selecting of nodes and many other ones to consider while developing such algorithm. Dynamic load balancing algorithms are those algorithms which search for the lightest server in the network and then designated appropriate load on it. In this, work load is distributed among the processors at runtime. The algorithms in this category are considered complex, but have better fault tolerance and overall performance. In this project basically five policies are used such as:

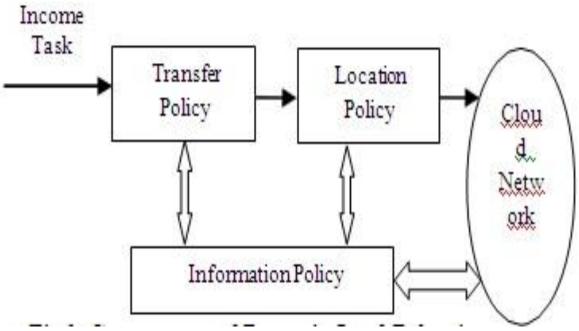
2.1 **Transfer Policy :**
In this policy it is responsible to determine when a task should be transferred from one node to the other node.

**2.2 Selection Policy :**
In this policy it focuses on choosing the processor for load transfer so that the overall response time and throughput may be improved.

**2.3 Location Policy** :
In this policy it determines the availability of essential resources for providing services and makes a selection based on location of resources.

**2.4 Information Policy** : In this policy it acquires workload related information about the system such as nature of workload and the average load on each node.

**2.5 Load Estimation Policy:** In this policy it determines the total workload of a node in a system.



There are various issues while dealing with load balancing in a cloud computing environment. Each load balancing algorithm must be such as to achieve the desired goal. Some algorithms aim to achieving higher throughput, minimum response time, and maximum resource utilization.

## 3. Dynamic Algorithms

Dynamic algorithms use system state information (the loads at nodes) to make load balancing decisions. Dynamic algorithms have t he potential to outperform the static algorithms, since they are able to exploit the short term fluctuations in the system to improve performance. But they incur overhead in the collection, storage and analysis of system state.

## 4. Packages used in Project

The project is being prepared using .NET and we are using remote server for load balancing.

Following are the packages used in our project: Windows form application,Remote server 4.5.

## 5. Load balancing technique for web servers.

**5.1 Round-Robin load balancing**:
Round-robin is one of the simplest scheduling algorithms for processes in an operating system, which assigns time slices to each process in equal portions and in order, handling all processes without priority. Round-robin scheduling is both simple and easy to implement, and starvation-free. Round-robin scheduling can also be applied to other scheduling problems, such as data packet scheduling in computer networks. The name of the algorithm comes from the round-robin principle known from other fields, where each person takes an equal share of something in turn.

## 6. References

[i] E. Villegas, R. V. Ferr, and J. P. Aspas, "Load balancing in wireless LANS using 802.11k mechanisms," in Proc. IEEE Symposium on Computers and Communications, 2006, pp. 844–850.

[ii] G. P. Collotta, V. M. Salerno, and G. Scata, "Distributed load balancing approach for industrial IEEE 802.11 wireless networks," in Proc. 18th IEEE International Conf. on Emerging Technologies and Factory Automation, Krakow, 2012, pp. 1-7.

[iii] M. Collotta, L. L. Bello, E. Toscano, and O. Mirabella, "Dynamic load balancing techniques for flexible wireless industrial networks," in Proc. IECON 36th Annual Conference on IEEE Industrial Electronics Society, 2010, pp. 1329-1334