**LOAD BALANCING IN NETWORK OF SERVERS**

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

**Bachelor of Technology** In

**Computer Science and Engineering**

**School of Engineering and Sciences**

Submitted by

**Samah Maaheen Sayyad-AP20110010397**

**Devi Chinmayi Vulchi-AP20110010405**

**Susmitha Amudalapalli-AP20110010361**

**Deepthi Kolli-AP20110010374**

**Juhita Naga Priya Velagapudi-AP20110010378**

**Himaja Koneru-AP20110010406**

**A picture containing text

Description automatically generated**

Under the Guidance of

**Dr. Muzakkir Hussain Md**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**[December, 2022]**

# Certificate

Date: 09-Dec-22

This is to certify that the work present in this Project entitled “**LOAD BALANCING IN NETWORK OF SERVERS**” has been carried out by **Samah Maaheen, Devi Chinmayi Vulchi, Susmitha Amudalapalli, Deepthi Kolli, Juhita Naga priya Velagapudi,** **Himaja koneru** under my supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology in **School of Engineering and Sciences**.

**Supervisor**

Dr. Md. Muzakkir Hussain

Department of Computer Science and Engineering

E-mail id: [muzakkirhussain.m@srmap.edu.in](mailto:muzakkirhussain.m@srmap.edu.in)

# Table of Contents

Certificate iii

[Table of Contents](#_1fob9te) iv

Abstract vi

[Statement of Contributions](#_tyjcwt) vii

[List of Fi](#_4d34og8)gures viii

[Introduction](#_17dp8vu) **1**

[1.1](#_3rdcrjn) Motivation and Contribution 1

1.2 State Space Representation 2

[Methodology](#_lnxbz9) **3**

2.[1](#_35nkun2) Problem Description 3

Results and [Discussion](#_44sinio) **5**

[Concluding Remarks](#_2jxsxqh) **7**

[Future Work](#_z337ya) **8**

[References](#_3j2qqm3) **9**

# Abstract

In this practical world almost all the systems are in the form of a network. Managing these networks is quite challenging. One of such real-world networks is the university computer system where the computers are the nodes and they form the network and are assigned with some jobs. So, allocating the upcoming jobs to the existing system by managing their availability is the challenge. This problem is termed as the Load Balancing in the network server. To solve this problem, we have implemented the BFS algorithm in our work.

# Statement of Contributions

Task I: Responsible for analysis of problem,

Task II: Responsible for research of problem statement ,

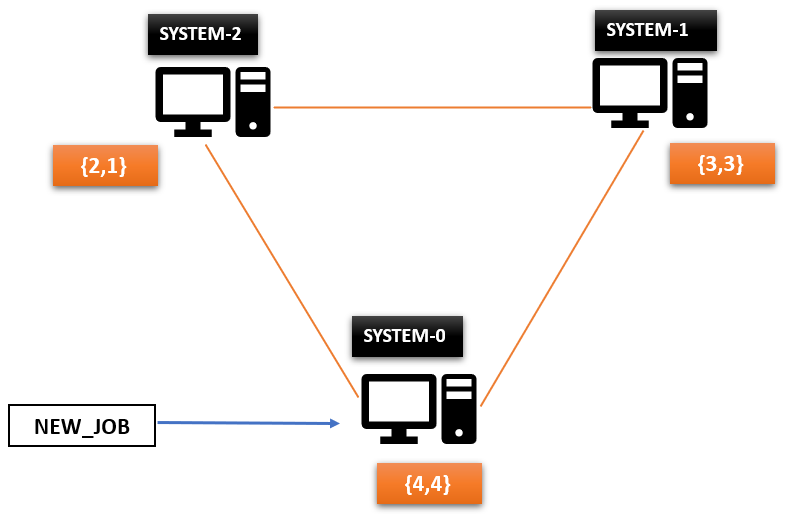
Task III: Responsible for implementation of problem,

Task IV: Responsible for simulation of source code,

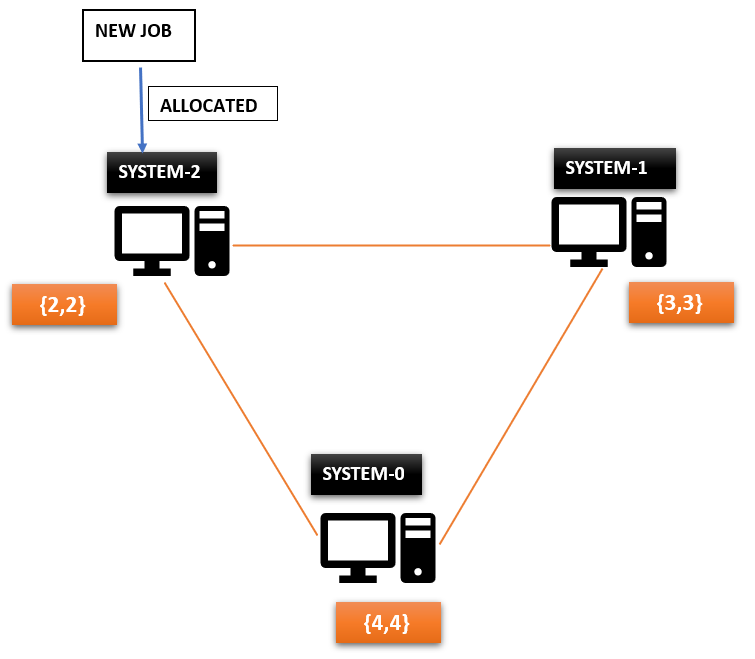
Task V: Responsible for manuscript writing,

Above mentioned are the frameworks contributed by the team.

# List of Figures



**Figure 1. Arrival of a new job**



**Figure 2. Allocation of new job with minimum number of migrations**

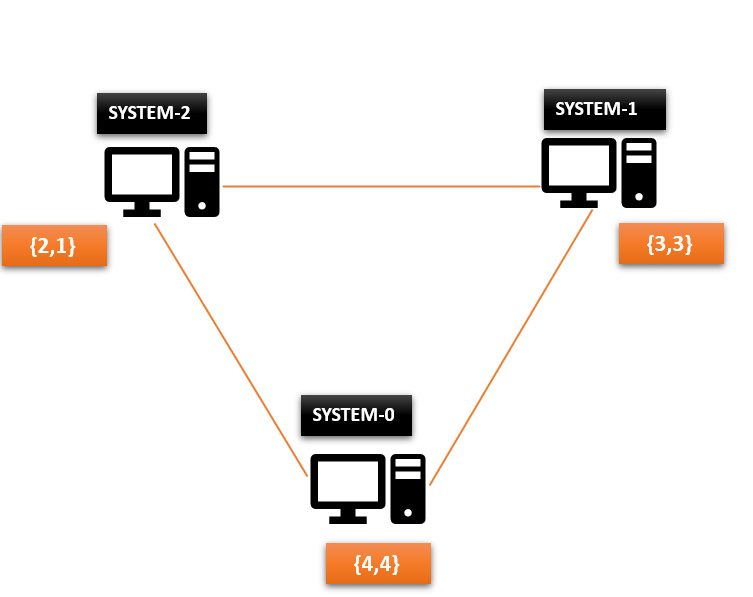
# Introduction

Load balancing refers to how efficiently we can distribute the incoming network traffic across a group of backend servers. The main aim of the project "Load Balancing in Network of Servers" is to design an algorithm to minimize the total number of migrations and report the number of migrations in a given system. The input to the system is the number of nodes and their weight or capacity and the interconnections among the nodes. And jobs are submitted randomly at any node. In this project the nodes represent the computer system, and a weight is assigned to that node to represent the maximum number of applications that node can execute. The edges connecting the computer systems represent the direct connections between the nodes/ computer systems. In this project we have found the minimum number of migrations using the search technique "Breadth First Search" .

**1.1 Motivation**

In the present day every real life application is part of the network system in order to have an easy way of communication.But we aren’t familiar with the path chosen to transfer the data from host to host. It might be the longest path or the shortest path. In order to optimize the path cost and time taken to transfer the data we have chosen this project.

**1.2 State Space Representation**



## 

Let us consider the above network. In this network we have the following configurations.

1. Number of systems = 3

2. Names of the systems are = {0,1,2}

3. Neighbours of each system in the network are as follows:

Neighbours={0: [1,2], 1:[2,0], 2:[1,0]}

4. Weight and the current status [weight, current status] of the systems are as follows:

SYSTEM (0) = [ 4,4]

SYSTEM (1) = [3,3]

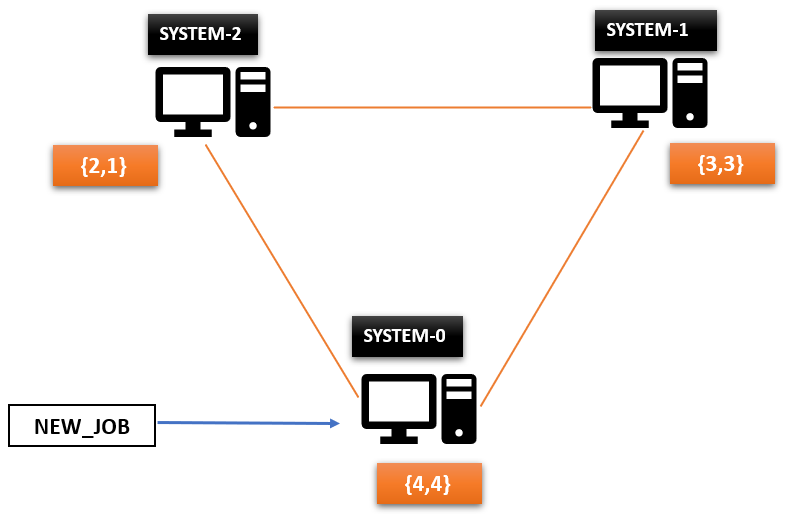
SYSTEM (2) = [2,1]

Here weight represents the capacity of the system and the current status represents Number of tasks the system is executing at present.

**Methodology**

**2.1 Problem Description:**

• Now let us consider a new job that has arrived at system number 0.



**STEP-1:**

• In this as the new job is arrived at the SYSTEM-0 so according to the proposed algorithm, we need to check whether the SYSTEM-0 is free or not as we can see from the above network SYSTEM- 0 capacity is 4 and the current status is 4 which means it is busy so it can’t be allocated with any job.

**STEP-2:**

• As the SYSTEM-0 is busy now the task is to check whether its neighbours are free or not to do this we are going to apply the BFS algorithm with the root node as 0.

• The BFS algorithm returns the level order traversal from the root node.

• So, in this case the traversal will be T = {0,1,2}

**STEP-3:**

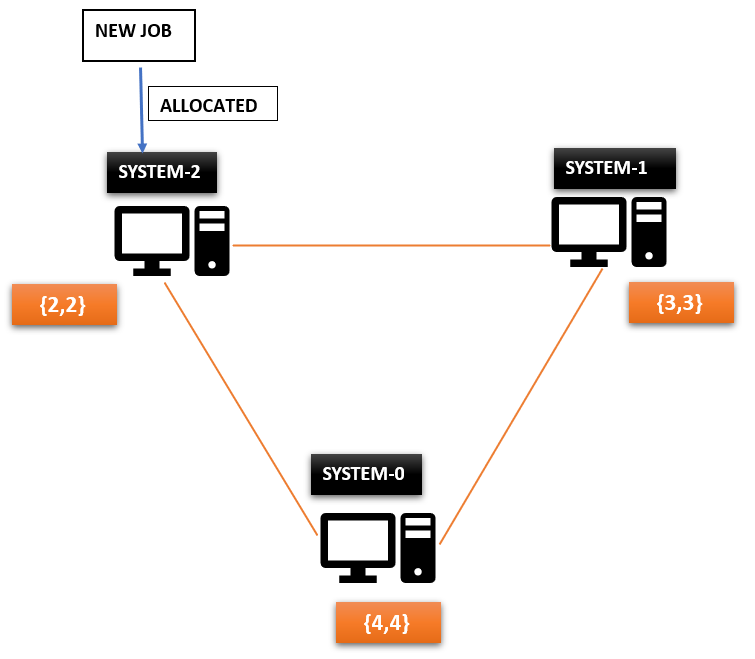
• As the neighbours of the node 0 are obtained, now the task here is to check each neighbour whether they are free or not. If we come across any of the neighbours which have available slots for the new job then a new job is going to be allocated to that system.

• From the T= {0,1,2} as 0 is busy now next node 1 is going to be checked

• In this case as the capacity of node 1 is 3 and the current status is 3 which means it is completely busy so it can’t be allocated with any task.

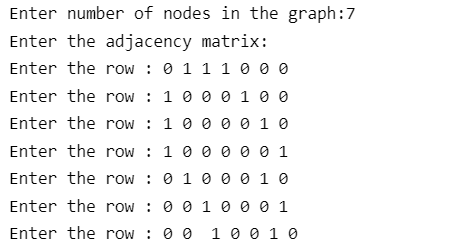
• As node 1 is busy now the next node 2 is checked from the T= {0,1,2}.

• The capacity of node is 2 and its current status is 1 that means 1 slot is available. So now the new job is allocated to the node 2. And the number of migrations will be 1 in this case.

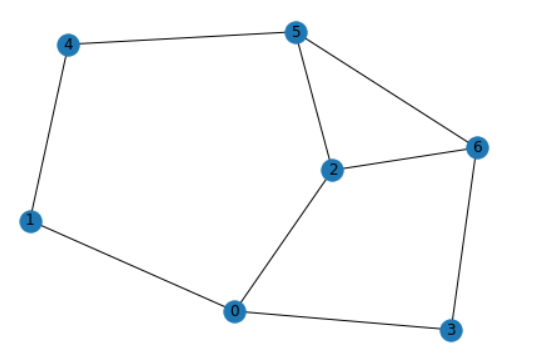


**Results and Discussion**

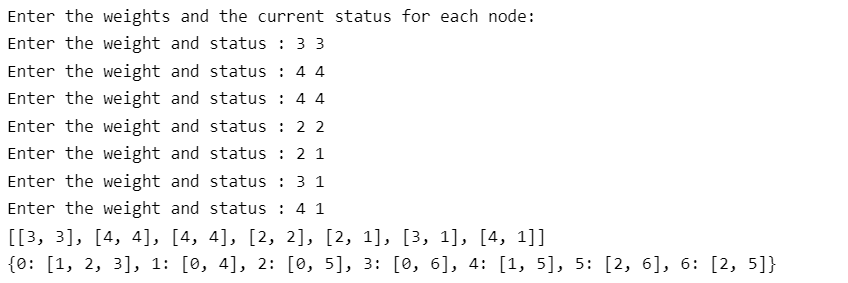
The following is the **working process** of our algorithm



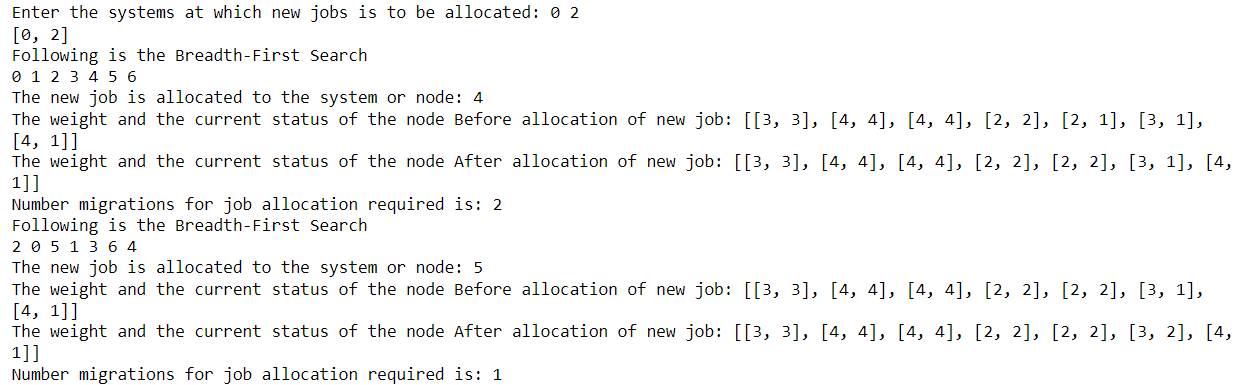
**Figure 1. Implementation of network of systems using a graph**



**Figure 2. Network consisting of 7 systems**



**Figure 3. Assigning weights and current processing status to each job and printing the neighbors of each job**



**Figure 4. Allocating jobs to the desired systems by making minimum number of migrations**

**Properties of Algorithm Applied:**

|  |
| --- |
| Optimal : YES  Complete : YES  Time Complexity : O( bd )  Space Complexity : O( bd ) |

where,

b is the branching factor of the search network

d is the depth of the least cost solution in the network

# Concluding Remarks

Finally, 'Load Balancing in Network of Servers' assists in allocating newly arrived jobs to existing systems with the least number of migrations. Such networks are also common in the real world and solving these tasks is an important job. This task can be solved using different algorithms but we have solved using the BFS algorithm as BFS will find the shortest path between starting point and any other reachable node. Also, BFS will never get trapped in a blind alley, which means unwanted nodes. If all of the network's systems are busy, new jobs must wait to be processed.

# Future Work

Our future scope is to work using the Load Balancing Algorithms to reduce the time and space complexity. Here in this problem the jobs are scheduled only in one network. We further want to extend our work using multiple networks. Apart from this we have used a simple graph for designing a network but we would like to further work using the different network topologies like Ring topology, Mesh topology , Bus Topology e.t.c.

# References

1. <https://docs.aws.amazon.com/elasticloadbalancing/latest/network/introduction.html> (Accessed on 20th Nov, 2022)
2. <https://medium.comvatsalunadkat/advantages-and-disadvantages-of-ai-algorithms-d8fb137f4df2> (Accessed on 24th Nov, 2022)
3. <https://www.hackerearth.com/practice/algorithms/graphs/breadth-first-search/tutorial/> (Accessed on 30th Nov, 2022)