The Research Monograph Series in Computing, Electrical & Communication Networks
Vol. 1 (01), January 2023, pp. 43–55

MOBILITY AWARE LOAD BALANCED SCHEDULING

Here, we propose Mobility Aware Load Balanced Scheduling algorithm. In this approach we categories jobs into Computing-focused and Communication-focused. The compute-intensive jobs are allocated to the resources with shorter round-trip-time, high CPU speed and capacity. The communication-intensive jobs are allocated to resources with low mobility and high reliability. The outcome of this approach results in more effective load balanced job allocation across the mobile nodes. By simulation results, we show that the proposed approach gives optimized results compared to the existing techniques.

The Mobile devices have advantages over fixed computing resources such as mobility, portability, and pervasiveness. The strength of the mobile grid allows it to be applied on location-restricted fields such as wildfire prevention, disaster management, and e-health system, etc. Some of the Challenges of job scheduling in Mobile Grid Environment are the device mobility, the type of environment, type of the grid architecture, task processing policy, tasks interrelations, Battery power and frequent disconnections that affect the resource availability thereby causing devices to become unreliable for job scheduling [3, 6].

In the Grid Environment, the scheduler plays a crucial role in job allocation and overall network performance. An intellectual grid scheduler optimizes standard scheduling objectives such as allocating jobs to proper mobile devices, minimize uncertainty in job execution and strive to optimize scheduling objectives such as maximum throughput, response time, balance available resources with security requirements and minimize energy consumption [1, 2, 3, 4, 5]. The existing job scheduling algorithms lacks to verify the node's capability and optimize the job execution [6].

PROPOSED SOLUTION

In this paper, we propose a Mobility Aware Load Balanced Scheduling Algorithm for the mobile grid environment. This MALBS provides effective load balanced mobile Grid environment. In this technique, the server upon receiving the request from grid controller analyzes the job. It categorizes the job into computing-focused and communication-focused jobs. The compute-intensive jobs are allocated to the resources with shorter round-trip-time (RTT), high CPU speed and capacity. The communication-intensive jobs are allocated to resources with low mobility and high reliability. The main advantage of this approach relies on job scheduling based on nodes execution capability.

The following paragraphs demonstrate ahead, the parameters that are used in the module.

ESTIMATION METRICS

Round Trip time

Let t_{tx} represent the time at which a data packet originates from the node. Let trx represent the time at which a data packet is received at the originator node. The round trip time (RTT) is defined as the difference between the transmission and reception time of the data packet at the sender node [12]. This is given using Equation.4.1.

$$RTT = (\mathrm{t}_{rx} - \mathrm{t}_{tx}) \tag{4.1}$$

Reliability (R)

Reliability characterizes the ability of a Grid system to execute the job correctly irrespective of failures. Reliability plays a significant role in the specific job process.

CPU speed and capacity

The capacity C(t) of the resource at time 't' is estimated using CPU speed and processing power. It is given using the following Equation 4.2.

$$\mathrm{C(t)} = \frac{P_{comp}}{AP_{comp}} \tag{4.2}$$

Where P_{comp} = Total computational power of the processors in a mobile node.

AP_{comp}= Available computational power of the processors of the mobile node.

Received signal strength (RSS)

The Signal Strength of a data transmitted by a node/resource can be estimated using Friis Equation 4.3 [17].

$$RSS = \frac{P_{tx} * \alpha * \beta * h_{tx}^2 * h_{rx}^2}{d^4 * L} \tag{4.3}$$

Where P_{tx} is the transmission power, α is the transmitter gain, β is the receiver gain, d is the distance between the transmitter and sink, Lis the system loss, h_{tx} is the height of transmitting antenna and h_{rx} is the Height of receiving antenna.

Node mobility (Mo-P)

The Node mobility (Mo-P) represents the mean distance traversed by the mobile node for a Time period (T) within in the client set. The Grid controller (G_Con) tracks and stores the movement history and location of each mobile node. The movement history estimates the duration which the mobile nodes are available within the access point of the particular client.

Let $CS_i = \{CS_1, CS_2, CS_3 \ldots CS_n\}$ be the client sets. Z_i^j be the jth zone in the ith client set. The mobility model of the nodes is determined using the joint distribution of sequence G_i which is given using Equation (4.4)

$$Pr(G = [z_1, z_2, \ldots z_n]) = Pr(Location\ of\ N_i\ in\ z_{21}^{g1} \wedge z_2^{g2} \ldots. \wedge z_n^{gn} \ldots.. \tag{4.4}$$

The above equation can be demonstrated using the following case. For a Time(T), If G(2) = 4, then the mobile node is located in the 4th zone of the client set CS_2.

Estimation of Response Time

The combination of the transmission time of job (t_{xxy}), waiting time of job in the resource queue (t_{wxy}) and job service time on the resource (t_{sxy}) represents the response time of a job. The response time (t_{resxy}) of a resource y for a job x is defined by Equation 4.5

$$t_{resxy} = t_{xxy} + t_{wxy} + t_{sxy} \tag{4.5}$$

t_{wxy} is computed by summing all service times of jobs that are in the waiting queue of resource ‘y’ arrived prior to job ‘x’. It is represented using Equation 4.6.

$$t_{wxy} = \sum\nolimits_{j=1}^{q} t_{sjy} \tag{4.6}$$

t_{xxy} is defined using Equation 4.7

$$t_{xxy} = \frac{Z_x}{B_{xy}} \tag{4.7}$$

Where Z_x = the size of a given job x.

B_{xy} = the bandwidth between the grid scheduler and the resource ’y’ on which the job x can be executed. t_{sxy} can be defined by:

$$t_{sxy} = \frac{t_{sx}}{\lambda_y} \tag{4.8}$$

Where t_{sx} = service time required for job x.

λ_y= speed of resource y [16].

MOBILE GRID ARCHITECTURE

Figure 4.1 demonstrates the mobile grid architecture. The grid controller is a server of grid community that interlinks the clients and offers distributed services. The proxy server fetches the job from Grid Controller (G_Con) through the router and replies the resultant information to the controller. The proxy server (PS) acts as dominant in the job allocation system. The major role of proxy corresponds to the information service (IS) and the job scheduler service (JSS). IS gathers the details of mobile resources utilizing the information provided by the resource providers. JSS performs job scheduling adapting an efficient scheduling technique to select an appropriate resource from any one of the client set for completing the job request. Each client set is said to include mobile nodes (N_i) with access points (AP_i). This access point (AP_i) is connected via wired or wireless mode to the proxy server.

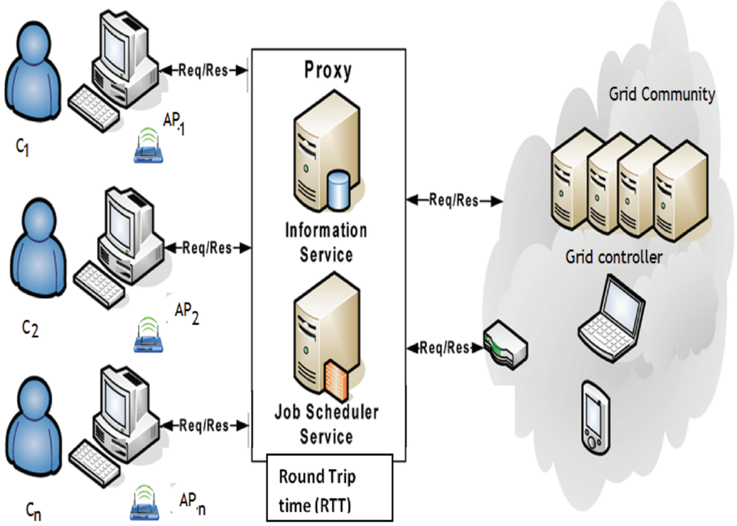

Figure 4.1. Proposed Mobile Grid Architecture.

MOBILITY-AWARE LOAD BALANCED JOB SCHEDULING TECHNIQUE

In our approach, we categorize the job into Computation-focused and Communication-focused jobs.

Computation-focused jobs: (J_{comp})

The Compute-intensive or computation-focused jobs imply, sharing and coordinated utilization of resources independent of their physical nature and location that demands a lot of computation [15]. Its application includes meteorological simulations, data intensive applications, research on DNA sequences and nano-materials.

Communication-focused jobs (J_{comm})

The Communication-focused jobs are characterized by the large quantity of data, communicating among processors. These data appear to be frequent large messages. The application includes telemedicine, disaster management and scientific collaboration [14].

Job Scheduling Process

The client sends a resource requisition to the Grid Controller. The Grid Controller forwards the requested message to the proxy server. The proxy server fetches the (R_req) and analyzes the job category. With the help of IS the scheduler identifies suitable resources that are capable of executing the job request. It then estimates the response time of all the nodes in the client-set as per Equation (5). The proxy then analyzes the parameters such as round trip time (RTT) and CPU speed (S_{CPU}) and capacity (C(t)), mobility (Mo_P), received signal strength (RSS) and reliability (R). It sorts the list of the suitable resources for the job, according to the response time.

In the case of Computation focused job, to achieve load balanced scheduling, it investigates the response time. For a response time greater than the threshold, the proxy selects a node that has Round Trip Time (RTT) lesser than a threshold ($RTT<RTT_{th}$), high CPU speed ($S_{CPU}>S_{th}$) and high CPU ($C(t)>C_{th}$) capability. For a response time lesser than the threshold, the job can be assigned to a node which has either Round Trip Time ($RTT <RTT_{th}$) or high CPU speed($S_{CPU}>S_{th}$) or high CPU ($C(t) >C_{th}$) capability. Where RTT_{th}, $S_{th,}$ $C_{th}$are threshold value for Round Trip Time, CPU speed and CPU capacity respectively.

If the job is communication focused, the proxy monitors the node within each client-set for its Received Signal Strength (RSS) and mobility(Mo_P). It assigns the job to a node with lesser mobility (Mo_P<T_{th}) and with a greater signal strength ($RSS>R_{th}$). If the response time is lesser than the threshold, the job can be assigned to a node which has either lesser mobility (Mo_P<T_{th}) or greater signal strength ($RSS>R_{th}$).

Upon selecting the node, the proxy server sends the reply message (R_rep) about the availability of suitable node to the controller. Upon receiving R_rep, the appropriate job is scheduled to the node via proxy server.

This efficient job allocation technique presents a balanced load across entire client-set and minimizes the time required to transmit and execute jobs. We define the above technique as Mobility Aware Load Balanced Scheduling Algorithm as shown below in Algorithm-1.

Mobility Aware Load Balanced Algorithm-1

```
1. set threshold values for variables $T_{min}$, $RTT_{th}$, $S_{th}$, $C_{th}$, $T_{th}$, $R_{th}$
2. Calculate $t_{resxy}$ for all the nodes// $t_{resxy}$ is the Response Time
3. if job ($J_i$) is received from the Grid Controller
   3.1 if Compute Intensive Job
       3.1.1 if Response time is greater than $T_{min}$
             Select node with $RTT <RTT_{th}$and $S_{CPU}>S_{th}$and $(C(t) >C_{th})$
             Else if Response time is lesser than $T_{min}$
```

```
                Select node with $RTT < RTT_{th}$ or $S_{CPU} > S_{th}$ or $(C(t) > C_{th})$
3.2  if Communication Intensive Job
        3.2.1  if Response time is greater than $T_{min}$
                Select node with $(Mo\_P < T_{th}$ and $(RSS > R_{th})$
                Elseif Response time is lesser than $T_{min}$
                 Select node with $(Mo\_P < T_{th})$ or $(RSS > R_{th})$
```

SIMULATION RESULTS

In this section, we examine the performance of our Mobility Aware Load Balanced Scheduling Algorithm (MALBS) with an extensive simulation study based upon the Network Simulator Version-2 (Ns-2)[18]. The simulation topology is given in Figure 4.2. Various simulation parameters are given in Table 4.1.

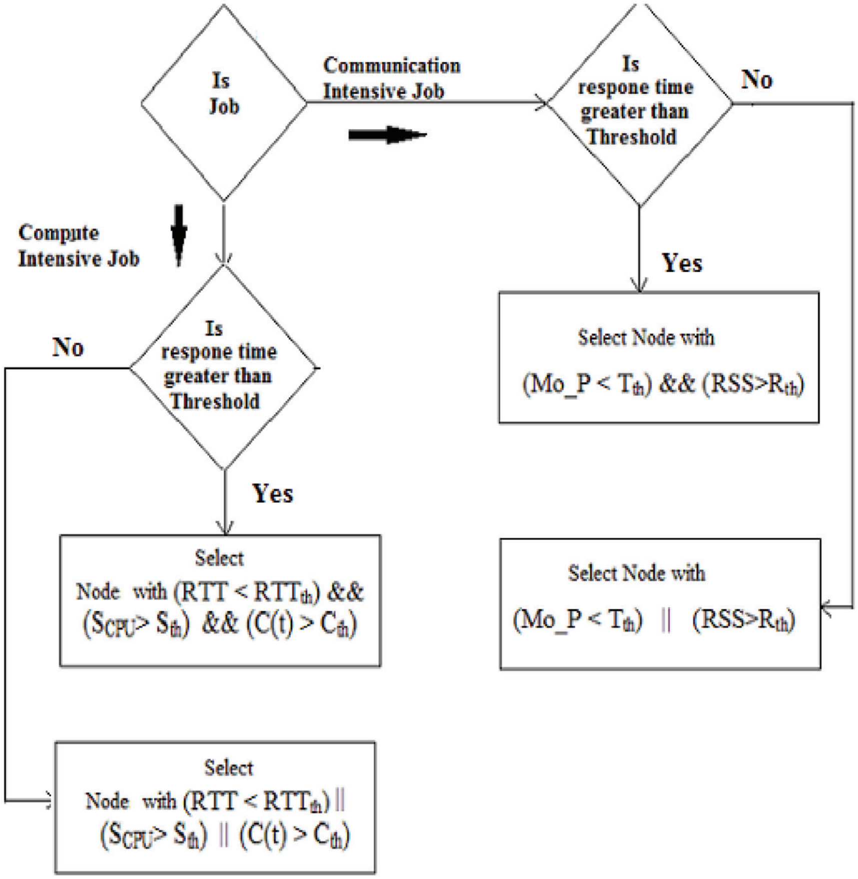

Figure 4.2. Modular diagram for MALBS Algorithm.

Table 4.1. Simulation Settings.

Mobile Nodes	9
Users	6
Clusters	3
Area Size	1000 × 1000
Mac	802.11
Radio Range	250 m
Routing Protocol	DSDV
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	512
Maximum Rate	250, 500, 750 and 1000 Kb
Speed of mobile nodes	5 m/s to 25 m/s
Transmit Power	0.660 w
Receiving Power	0.395 w
Idle Power	0.335 w
Initial Energy	10.1 J

For our simulation study, we formed three clusters containing three nodes in each cluster. The User1, User2 and User3 are allowed to send packets to the mobile nodes. The User1's packets are passed to cluster1 through cluster head AP1, User2's packets to cluster2 through cluster head AP2 and User3's to cluster3 through cluster head AP3. The packet size of the traffic is 512 bits. The speed of the mobile nodes is set to 5m/s. The simulation is done for 50 sec.

PERFORMANCE METRICS

In our experiments, we measure the following metrics.

i. Packet Delay: It measures the average end-to-end delay occurred while executing a given task.
ii. Received Bandwidth: It is the ratio of the number of bits successfully transmitted to the receiver per sec.
iii. Packet Drop: The total number of packets dropped during the data transmission.

In the simulation study, we simulate MN1, MN4 and MN9 as movable nodes. Nodes MN3, MN5 and MN8 are partial available nodes and MN2, MN6 and MN7 as stationary nodes.

EXPERIMENT FOR COMPUTE INTENSIVE JOB

In our experiment, based on our MALBS algorithm defined in Section 4.2, Nodes MN2, MN4 and MN9 are assumed selected for compute intensive jobs and data Packets are sent to the nodes through their respective cluster heads. The performance of the MALBS algorithm is validated by measuring the overall throughput, packet delay and packet drop for various load values such as 250 Kb, 500 Kb, 750 Kb and 1000 Kb. The results are compared with the Balanced Scheduling Algorithm (BSA technique)[6].

In a BSA technique communication-intensive jobs are assigned to fully available nodes and computation intensive jobs are allocated to partially available nodes. According to BSA technique a node is said to be fully available if, it has a communication link and is capable of executing the job. It is said to be partially available if it is capable of executing the job, but may or may not have strong communication link [6]. Therefore, the nodes MN3, MN6 and MN7 are selected for compute intensive jobs.

Figure 4.3 shows the throughput measured for various load values in case of both proposed MALBS and BSA techniques. From the figure, it can be concluded that the throughput of MALBS is 20% higher than the BSA technique.

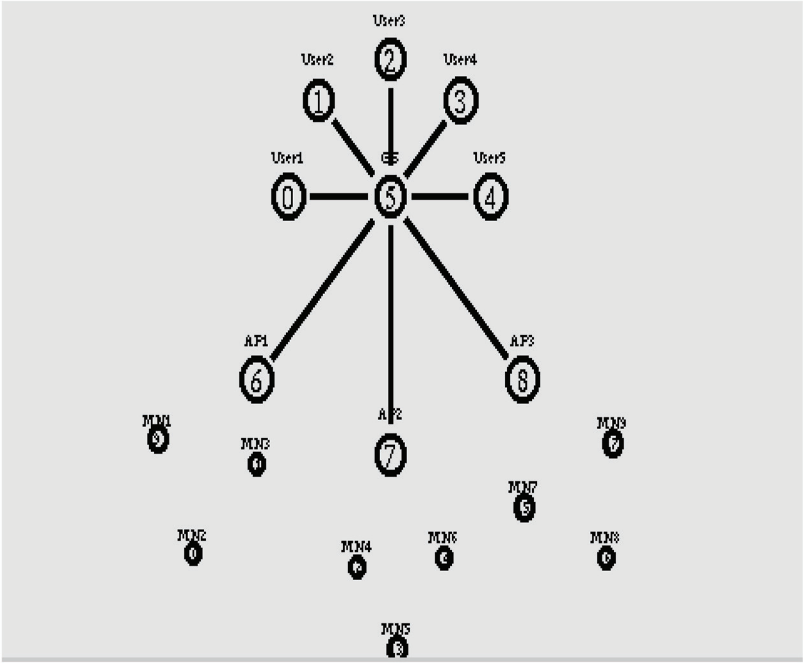

Figure 4.3. Simulation Setup.

Figure 4.4 shows the packet delay measured in case of both MALBS and BSA techniques for various load values. From the figure, we can see that the overall packet delay of MALBS is lesser than the BSA technique.

Figure 4.5 shows the packet drop measured against various load values for both techniques. From the figure, it can be concluded that the packet drop of MALBS is 60% lesser than BSA technique.

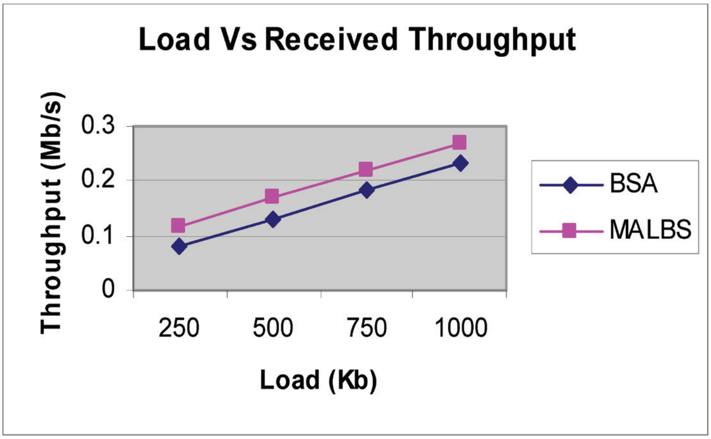

Figure 4.4. Load Vs Received Throughput.

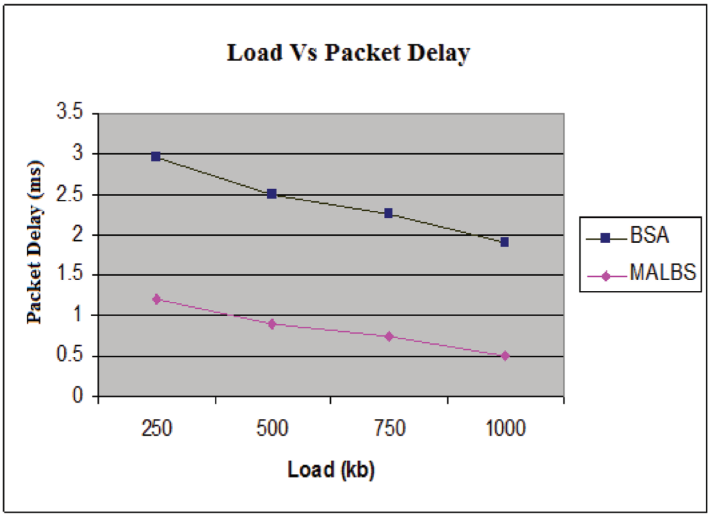

Figure 4.5. Load Rate Vs Packet Delay.

EXPERIMENT FOR COMMUNICATION INTENSIVE JOB

In the case of communication intensive Jobs, we assume MN1, MN5 and Mn9 based on the criteria defined in Section 4.2.3.3 as the suitable node. The packets are sent to the mobile nodes through their respective cluster heads. The overall throughput, packet delay and packet drop are measured for various speed values such as 5, 10, 15, 20, 25 m/s. The results are compared to BSA technique, where the communication intensive jobs are allocated only to the fully available nodes. In the simulation study for BSA technique the nodes MN2, MN5 and MN8 are selected for the communication intensive job.

Figure 4.6 shows the throughput received in case of both the techniques for various speeds of the nodes. From the figure, we can see that the received throughput of MALBS is 30% higher than the BSA technique.

Figure 4.7 shows the packet delay measured in case of both the techniques for various speeds of the nodes. From the figure, we can see that the packet delay of MALBS is 82% lesser than the BSA technique.

Figure 4.8 shows the packet drop measured in case of both the techniques for various speeds of the nodes. From the figure, it can be seen that the packet drop of our proposed MALBS is 41% lesser than the BSA technique.

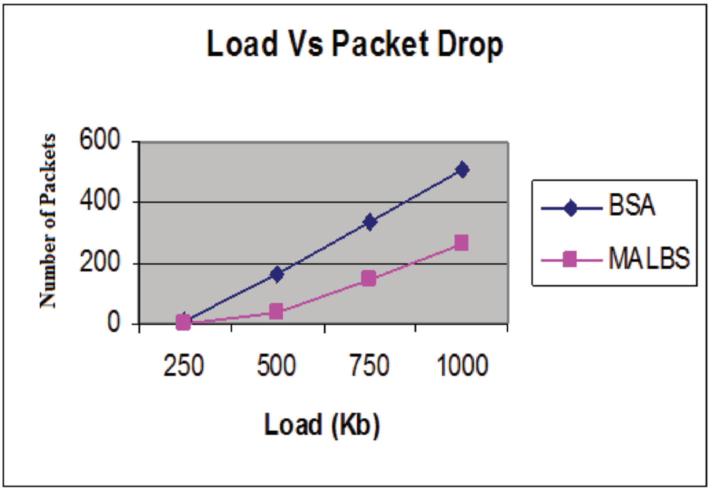

Figure 4.6. Load Vs Packet Drop.

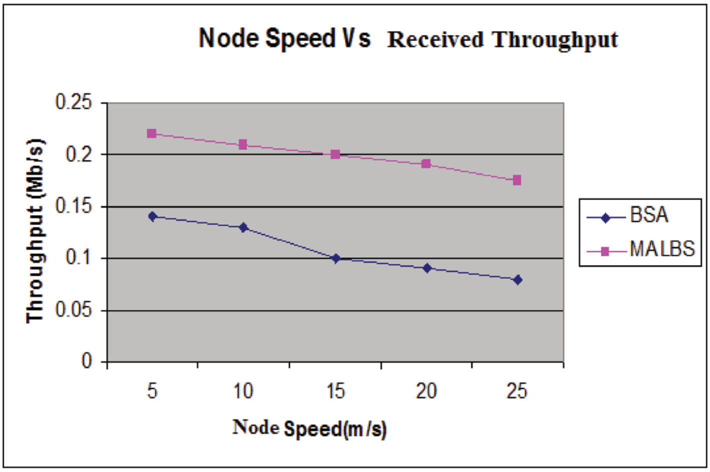

Figure 4.7. Node Speed Vs Received Throughput.

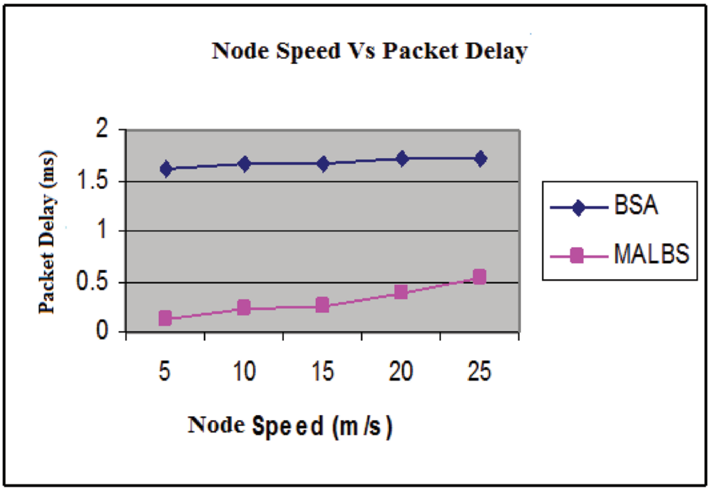

Figure 4.8. Node Speed Vs Packet Delay.

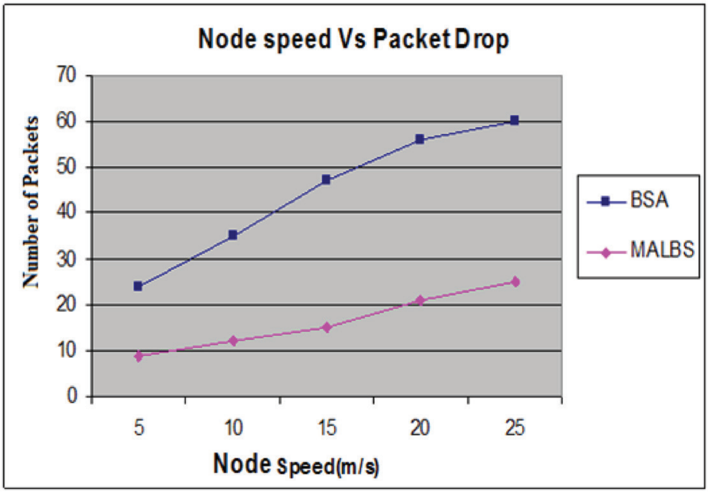

Figure 4.9. Node Speed Vs Packet Drop.

SUMMARY

Here, we designed a Mobility Aware Load Balanced Scheduling Algorithm for the mobile grid environment. In this technique, two job categories such as computing-focused and communication-focused jobs are taken into consideration. The server allocates the resources with shorter round-trip-time (RTT) and high CPU speed and capacity of the computing-intensive jobs. The communication-intensive jobs are allocated to resources with low mobility and high reliability. This approach provides an efficient, balanced job scheduling across the entire client set in the mobile grid network. By simulation, we compare our MALBS algorithm with BSA algorithm based technique. We have shown that our proposed MALBS approach minimizes the network overhead and increases the performance compared to BSA technique.