Priority EDF Scheduling Scheme for MANETs

Abel Mukakanya Muwumba[1](✉), Godfrey Njulumi Justo[1],
Libe Valentine Massawe[1], and John Ngubiri[2]

[1] College of Information and Communication Technologies,
University of Dar es Saalam, Dar es Salaam, Tanzania
abelmuk@gmail.com, njulumi@gmail.com, massawe.libe@udsm.ac.tz,
liebetz@yahoo.co.uk
[2] Makerere University, Kampala, Uganda
ngubiri@cis.mak.ac.ug

Abstract. Analytical EDF Priority schedulers are not common in Mobile Ad-hoc Networks (MANETs). Some researchers like Abhaya et al. have proposed a classical preemptive Earliest Deadline First (EDF) scheduler. The goal of this EDF scheduler was to favor higher priority packets thereby reducing their waiting times. Accordingly, favoring higher priority queues end up increasing the waiting times of lower priority queues. We improve Abhaya's approach and adopt it to the MANETs environment. We numerically study the performance of the Adopted and Improved Adopted Abhaya Earliest Deadline First (IEDF) models for different packet queues. Our analytical results show that the IEDF model shortens the waiting times of packets of the different queues at various system loads in comparison to the Adopted Abhaya EDF model.

Keywords: Deadline · Model · Packets · Preemptive · Waiting time

1 Introduction

Mobile Ad-hoc Networks consists of a set of mobile nodes that communicate with one another across multiple hops in a distributed manner [1]. The traffic transmitted by the nodes in MANETs is real-time and some is non-real time. Both delay-sensitive and non-delay-sensitive applications usually coexist in the same network, making QoS provisioning to be a critical issue [2]. Real-time traffic is delay sensitive, therefore, the design of an efficient priority scheduling scheme that will ensure that the mobile nodes in MANETs transmit traffic to the desired expectations of the users under strict deadline constraints becomes crucial. In designing priority-based scheduling policies, the ultimate goal is to avoid job starvation [3]. Unfortunately, MANETs technology does not specify any specific scheduling scheme leaving it open to researchers and scholars to innovate in this area. Whereas priority scheduling algorithms like Earliest Deadline First (EDF)

H. Gao et al. (Eds.): ChinaCom 2019, LNICST 312, pp. 66–76, 2020.
https://doi.org/10.1007/978-3-030-41114-5_6

have been sought to be among the viable solutions in transmitting traffic in wireless networking environments. Unfortunately, at the time of this study, we did not come across any specific EDF analytical models in MANETs. However, there exist a classical preemptive EDF model proposed by Abhaya et al. [4,5] that we have adopted to the MANETs environment. Therefore, in this paper, we study the Adopted Abhaya EDF model. The numerical results from the Adopted Abhaya EDF model show a performance degradation in waiting time for different packets in different priority queues at various system loads. We propose an Improved Adopted Abhaya Earliest Deadline First (IEDF) model that shortens the average waiting times of packets of the different queues. Therefore, in this paper we address the following specific gaps in the Adopted Abhaya EDF scheduler research. (i) the starvation of low priority queue packets. (ii) We address the poor performance of the Adopted Abhaya EDF model at high network traffic loads.

We use the Adopted and Improved EDF schemes to determine which packet in a particular queue of traffic should be served first among the set of waiting or remaining packets in different queues at a specific node in MANETs. The remainder of this paper is organized as follows: In Sect. 2 we describe the some Related Works. This is followed in Sect. 3 by the EDF schemes. Section 4 presents the Improved Adopted Abhaya EDF model. The Results and Discussions are presented in Sect. 5. Conclusion and Future Research are finally presented in Sect. 6.

2 Related Work

Liu and Layland [6] suggested the most popular real-time scheduling algorithms, EDF. The EDF policy assigns a deadline to each packet, which is used by the scheduler to define the order of service. The highest priority job is the one with the earliest deadline. It essentially schedules the jobs in a greedy manner which always picks the jobs with the closest deadline. An efficient Quality of Service architecture using inter layer communication with a highly efficient real time scheduler design at the network layer with improved Rate Monotonic Algorithm and Earliest Deadline First scheduling that efficiently schedules multiple real time applications without missing any of their deadline was proposed [7].

A number of EDF priority scheduling policies have been proposed in the previous works by altering the existing ones and adding new constraints to enhance performance [4,8–12]. An analytical method for approximating the performance of a two-class priority M/M/1 system was presented [8]. In this model the prioritized class-1 jobs were considered to be real-time and served according to the EDF scheduling policy, and the non real-time class-2 jobs were served according to the FIFO policy. One limitation with this model is that it is not an exact analytical solution for the analysis of EDF, even for a system with purely real-time jobs.

A multi-queue EDF and its variant Flexible Earliest Deadline First (F-EDF) was proposed [9]. The solution [10] investigated mean sojourn times in multi-class queues with feedback and their application to packet scheduling in communication networks. A Packet Scheduling algorithm consisting of EDF algorithm and

Least Slack Time algorithm is proposed for scheduling the various multimedia applications [11]. This Scheduling algorithm is used to reduce the transmission delay and to achieve better QoS requirements.

A preemptive M/M/1/EDF and non-preemptive M/M/m/EDF model is presented [10]. In this model exponentially distributed service times are assumed and these may not suitably represent web services workloads because the services could be used in exposing any type of system. A non-preemptive and work-conserving M/G/1/./EDF model which is supported by general workloads is proposed [12]. A preemptive EDF scheduling scheme that approximates the mean waiting time for a given class based on the higher and lower priority tasks receiving service prior to the target and the mean residual service time experienced was proposed [4,5]. The goal of this EDF scheduler was to favor higher priority packets thereby reducing their waiting times. The limitation with this approach is that favoring higher priority queues yielded increased waiting times of lower priority queues.

3 The EDF Schemes

3.1 The Preemptive EDF Model

Abhaya et al. [4,5] proposed an algorithm that applies on web services middleware. In the algorithm the requests, jobs and tasks are received by the middleware, and selected requests are serviced at each server using the EDF scheduling algorithm. The model considers mean waiting time of multiple streams of packets serviced by node that acts as server. When packets arrive at the node, the scheduler classifies the packets into high and low priority queues. Each packet in the priority queue is assigned a deadline and the packets with high priority and with short deadlines are serviced first. The generic expression of mean waiting time for any queue i is given in Eq. (1).

$$\overline{W}_i = [\frac{\overline{W^i}_0}{1-\sigma_i} + \sum_{k=i+1}^{N} \rho_k max(0, \overline{W}_k - D_{k,i})] + \sum_{k=1}^{i-1} \rho_k min(\overline{W}_i, D_{i,k}) \quad (1)$$

According to Abhaya et al. [4,5] the symbols in Eq. 1 are explained as follows: $\overline{W}_i$, is mean waiting time for a packet of stream/priority i; $\overline{W}_k$, is mean waiting time for a packet of stream/priority k; $\overline{W}_0^i$ is mean time delay experienced by an arrival from stream i, from the packets already in progress; $D_{k,i}$ is the difference in the deadline offsets of streams i and k; $D_{i,k}$ is the difference in the deadline offsets of streams k and i; N is the number of independent streams through which requests arrive at the system following a Poisson process; and ρ_k is the system load due to queue k packets.

3.2 Adopting the Abhaya EDF Model to MANETs

MANETs have a unique characteristic behaviour, where within their network each node has the potential to act as a data source, a data sink, and/or a router

```
input  : Consider a preemptive M/G/1 queue
output: Mean waiting time for any class $i$

For all incoming jobs classify into Priority Classes;
Assign a deadline to each job;
for $i = 1 \leftarrow N$ do
    for $j = i + 1 \leftarrow N$ do
        $D_{i,j} = d_j - d_i$;
    end
    Compute the;
    Service times;
    Second moments;
    System loads;
    Mean residual service;
    Probability of a request from stream;
    Mean delay experienced by a new arrival;
    Mean waiting time;
end
```

Algorithm 1. The EDF Abhaya model

[13]. The basic structure of MANETs constitutes mobile nodes that are connected to transfer packets from source to destination mobile nodes, and there is an intermediary node between transmitting and receiving node acting as a router [14]. It is possible to have multiple servers in a single MANETs environment. We exploit this property of MANETs in the adopted EDF Abhaya model.

We make the following changes in the Abhaya model. (i) We adopt an M/G/m queue which is a multi-server system. Specifically, we use the M/M/m queueing system with arrival rates λ and service time $\overline{X}$. (ii) Compute the waiting probability following a non-preemptive M/M/m queue for server utilization. (iii) We compute the mean residual service time for a request of stream/priority i is for a M/M/m system. (iv) We change from preemptive scheduling to non-preemptive scheduling because all network data is useful and once it has been assigned a deadline, preempting an on-going job leads to several re-transmissions and results into wastage of resources.

We present the classic Adopted Abhaya EDF model in Algorithm 2. The Adopted Abhaya EDF scheduling algorithm determines the way packets are processed by the M/G/m scheduling system depending on deadline priority factor. Four priority queues i.e., $P1$-high, $P2$-medium, $P3$-normal and $P4$-low are considered at the intermediary node (router). Routers transmit packets by selecting the packet with shortest deadlines in the high priority queue. If any packet exists in the high priority queue, then it is selected and transmitted. Else, a packet with the shortest deadline is selected from the medium priority queue and transmitted. If there does not exist any packet in the medium priority queue also, then normal priority queue is considered. Finally, the low priority queue is taken into account. This procedure is continued for every packet in the MANET traffic.

The adopted generic expression of mean waiting time for any Queue i is given by Eq. (2).

```
input  : Consider a non-preemptive M/G/m
output: Mean waiting time for any Queue $i$
For all incoming Jobs classify into Priority Queues;
Assign a deadline;
for $i = 1 \leftarrow N$ do
    for $j = i + 1 \leftarrow N$ do
        $D_{i,j} = d_j - d_i$;
    end
    Compute the;
    Service times;
    Second moments;
    System loads;
    Waiting Probability of a request;
    Mean mean residual service time;
    Mean delay experienced by a new arrival;
    Mean waiting time;
end
```

Algorithm 2. The Adopted EDF Abhaya model

$$\overline{W}_i(Adp) = [\frac{\overline{W}^i{}_0}{m(1-\sigma_i)} + \sum_{k=i+1}^{N} \rho_k max(0, \overline{W}_k - D_{k,i})] + \sum_{k=1}^{i-1} \rho_k min(\overline{W}_i(Adp), D_{i,k}) \tag{2}$$

Like in Eq. 1, and according to Abhaya et al. [4,5] the symbols in Eq. 2 are explained as follows: $\overline{W}_i(Adp)$, is the mean waiting time for a packet of stream/priority i for Adopted EDF Abhaya model; $\overline{W}_k$, is mean waiting time for a packet of stream/priority k; $\overline{W}_0^i$ is Mean time delay experienced by an arrival from stream i, from the packets already in progress; $D_{k,i}$ is the difference in the deadline offsets of streams i and k; $D_{i,k}$ is the difference in the deadline offsets of streams k and i; N is the number of independent streams through which requests arrive at the system following a Poisson process; and ρ_k is the system load due to queue k packets. The other additional parameter m stands for the number of indentical servers in an M/G/m queue.

We make the following assumptions: In our system, 4 sources, 2 routers and 10 destination mobile nodes are the three main components. A router can send the packet to the destination mobile nodes via wireless Ad Hoc networks. The system has two (02) identical servers.

3.3 Performance of the Adopted Abhaya EDF Model

We first look at the performance of the Adopted Abhaya EDF model under the M/M/m queueing system. Our goal here is to show the weakness of the Adopted

Abhaya EDF in terms of penalizing low priority class packets in favor of higher priority packets. Due to limited space, we only show delay performances of the Adopted Abhaya EDF for four priority classes. We present more results of the Adopted Abhaya EDF performance when we compare it with the Improved Adopted Abhaya EDF in Sect. 5.

Figure 1 shows the waiting time as function of total load. We observe that: (i) $P1$ packets have a better performance overall among the all compared priority packets making the Adopted Abhaya EDF model to penalize low priority packet. (ii) waiting time for $P3$ and $P4$ packets increases uniformly with increasing load. (iii) waiting time for $P1$ and $P2$ packets increases uniformly with increasing load up to the 0.6 network load, beyond this load it drops up to 0.75 network load; and again it gradually increases till the end.

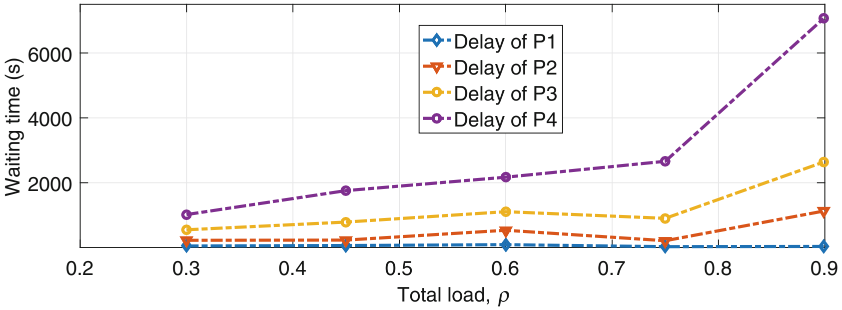

Fig. 1. Waiting time as a function of total load

4 Improving the Adopted Abhaya EDF Model

We indicate the following improvements in the Adopted Abhaya EDF model. (i) The component $\sum_{k=i+1}^{N} \rho_k max(0, \overline{W}_k - D_{k,i})$ is removed to avoid excessive waiting time for low priority packets. (ii) The component $\sum_{k=1}^{i-1} \rho_k min(\overline{W}_i(Adp), D_{i,k})$ is changed to $\sum_{i=2}^{N-1} \rho_i min(\overline{W}_i(imp), D_{i+1,i})$. It is these modifications that are responsible for shortening the waiting times for the low priority packets. We present the Improved Adopted Abhaya EDF model in Algorithm 3. In the Improved Adopted EDF Abhaya model, traffic consists of N number of independent queues that are classified by the scheduler into priority queues. Each queue is identified by $i, i = 1, 2, ..N$ and is associated with a different deadline. Packets from the same queue get assigned a constant deadline offset. Scheduling of packets among different queues uses IEDF and requests from the same queue are serviced in a First-Come First-Served basis. We consider the scenario of four priority queues where $N = 4$ i.e., $P1$-high, $P2$-medium, $P3$-normal and $P4$-low. Packets from $P1$ are assumed to have shorter deadlines than packets from $P2$, $P3$ and $P4$. In a typical priority based system, higher

```
input  : Consider a non-preemptive M/G/m queue
output: Mean waiting time for any queue $i$
For all incoming jobs classify into Priority Queues;
Assign a deadline;
for $i = 1 \leftarrow N$ do
    for $j = i + 1 \leftarrow N$ do
        $D_{i,j} = d_j - d_i$;
    end
    Compute the;
    Service times;
    Second moments;
    System loads;
    Waiting Probability of a request;
    Mean mean residual service time;
    Mean delay experienced by a new arrival after removal of excessive delay
    componets;
    Mean waiting time;
end
```

Algorithm 3. The Improved Adopted EDF Abhaya model

priority packets are always serviced ahead of lower priority packets, since the priority is determined by the absolute deadline in the system under consideration. Since packets from $P1$ have higher priority over $P2, P3$ and $P4$ packets, under a high arrival rate of $P1$ packets; $P2, P3$ and $P4$ packets are served after $P1$. For the case of four priority queues $N = 4$, we have four view points of the average delay of a tagged packet in a specific queue after delay. (i) The average delay of $P1$ packets. The tagged $P1$ packet will experience the following delays: $P1$ packets found in the queue will be serviced before the tagged packet, in this case $\overline{N}_{1,1} = \lambda_1 \overline{W}_1$. Where λ_1 is the arrival rate of $P1$ queue packets, and $\overline{W}_1$ is the mean waiting time for $P1$ queue packet. $P1$ packets from stream 1 arriving at the system after the tagged request will be served later i.e., $M_{1,1} = 0$. Therefore, the *average waiting time*, $\overline{W}_1$ for a tagged packet in $P1$ queue is given as;

$$\overline{W}_1 = \overline{W}_0 + \rho_1 \overline{W}_1 \tag{3}$$

(ii) The average delay of $P2$ packets. The following are the delays experienced by the tagged $P2$ packets; delay due to $P1$ packets found in the queue when the tagged packet arrives, these packets will experience delay given by; $N_{1,2} = \lambda_1 \overline{W}_1$. Delay due to $P2$ packets found in queue $N_{2,2} = \lambda_2 \overline{W}_2$. Where λ_2 is the arrival rate of $P2$ queue packets, and $\overline{W}_2$ is the mean waiting time for $P2$ queue packet. Once the tagged packet arrives at the system, it will not wait for a portion of $P1$ packets to be served before it. These packets will have deadlines earlier than the tagged packet. Because packets from $P1$ that arrive after the tagged packet, they will be served after the tagged packet. Therefore $M_{1,2} = 0$, and the delay experienced by the tagged packet can be expressed by;

$\overline{W}_2 = \overline{W}_0 + \overline{X}_1\lambda_1\overline{W}_1 + \overline{X}_2\lambda_2\overline{W}_2$. The *average waiting time*, $\overline{W}_2$ for a tagged packet in $P2$ queue is given as;

$$\overline{W}_2 = \overline{W}_0 + \rho_1\overline{W}_1 + \rho_2\overline{W}_2 \tag{4}$$

(iii) The average delay of $P3$ packet. The following are the delays experienced by the tagged $P3$ packet: delay due to $P1$ packets in queue $N_{1,3} = \lambda_1\overline{W}_1$; $P2$ packets in queue $N_{2,3} = \lambda_2\overline{W}_2$; and $P2$ packets in queue $N_{3,3} = \lambda_3\overline{W}_3$. Where λ_3 is the arrival rate of $P3$ queue packets, and $\overline{W}_3$ is the mean waiting time for $P3$ queue packet. Given the waiting time of $P3$, the tagged packet may be in the queue for a time period less than $D_{3,2}$ given that $W_3 < D_{3,2}$. The delay can be estimated as $M_{2,3} = \lambda_2 min(\overline{W}_3, D_{3,2})$.
$\overline{W}_3 = \overline{W}_0 + \rho_1\overline{W}_1 + \rho_2\overline{W}_2 + \rho_3\overline{W}_3 + \overline{X}_2\lambda_2 min(\overline{W}_3, D_{3,2})$. $D_{3,2}$ is the difference in the deadline offsets of streams 3 and 2. The *average waiting time*, $\overline{W}_3$ for a tagged packet in $P3$ queue is given as;

$$\overline{W}_3 = \overline{W}_0 + \rho_1\overline{W}_1 + \rho_2\overline{W}_2 + \rho_3\overline{W}_3 + \rho_2 min(\overline{W}_3, D_{3,2}) \tag{5}$$

(iv) The average delay of $P4$ packet. The following are the delays experienced by the tagged $P4$ packet: delay due to $P1$ packets in queue $N_{1,4} = \lambda_1\overline{W}_1$; delay due to $P2$ packets in queue $N_{2,4} = \lambda_2\overline{W}_2$; delay due to $P3$ packets in queue $N_{3,4} = \lambda_3\overline{W}_3$; and delay due to $P4$ packets in queue $N_{4,4} = \lambda_4\overline{W}_4$. Where λ_4 is the arrival rate of $P4$ queue packets, and $\overline{W}_4$ is the mean waiting time for $P4$ queue packet. Given the waiting time of stream 4, the tagged packet may be in the queue for a time period less than $D_{4,3}$ given that $W_4 < D_{3,2}$ and less than $D_{4,3}$ given that $W_4 < D_{4,3}$. The delay can be estimated as $M_{2,3} = \lambda_2 min(\overline{W}_3, D_{3,2})$ and $M_{3,4} = \lambda_3 min(\overline{W}_4, D_{4,3})$. $D_{4,3}$ is the difference in the deadline offsets of streams 4 and 3. $\overline{W}_4 = \overline{W}_3 + \rho_4\overline{W}_4 + \overline{X}_3\lambda_3 min(\overline{W}_4, D_{4,3})$.

The *average waiting time*, $\overline{W}_4$ for a tagged packet in $P4$ queue is given as;

$$\overline{W}_4 = \overline{W}_0 + \rho_1\overline{W}_1 + \rho_2\overline{W}_2 + \rho_3\overline{W}_3 + \rho_4\overline{W}_4 + \rho_2 min(\overline{W}_3, D_{3,2}) + \rho_3 min(\overline{W}_4, D_{4,3}) \tag{6}$$

Given the scheduling discipline considered, the mean waiting times must satisfy the conservation law for M/G/m queues [15,16]. Because of the m servers;

$$\sum_{k=1}^{i} \rho_k W_k = \frac{\sigma_i \overline{W^i}_0}{m(1-\sigma_i)} \tag{7}$$

Note: $\sigma_i = \sum_{k=1}^{i} \rho_k$. Substituting Eq. (7) into Eqs. (3), (4), (5) and (6), become Eqs. (8), (9), (10), (11) respectively.

The *average waiting time*, $\overline{W}_1$ for a tagged packet in $P1$ queue is:

$$\overline{W}_1 = \frac{\overline{W^1}_0}{m(1-\sigma_1)} \tag{8}$$

The *average waiting time*, $\overline{W}_2$ for a tagged packet in $P2$ queue is:

$$\overline{W}_2 = \frac{\overline{W^2}_0}{m(1-\sigma_2)} \tag{9}$$

The *average waiting time*, $\overline{W}_3$ for a tagged packet in $P3$ queue is:

$$\overline{W}_3 = \frac{\overline{W}^3{}_0}{m(1-\sigma_3)} + \rho_2 min(\overline{W}_3, D_{3,2}) \tag{10}$$

And the *average waiting time*, $\overline{W}_4$ for a tagged packet in $P4$ queue is:

$$\overline{W}_4 = \frac{\overline{W}^4{}_0}{1-\sigma_4} + \rho_2 min(\overline{W}_3, D_{3,2}) + \rho_3 min(\overline{W}_4, D_{4,3}) \tag{11}$$

The generic equation for the *average waiting time*, $\overline{W}_i(imp)$ for the IEDF for P_i queue is given by;

$$\overline{W}_i(imp) = \frac{\overline{W}^i{}_0}{m(1-\sigma_i)} + \sum_{i=2}^{N-1} \rho_i min(\overline{W}_i(imp), D_{i+1,i}) \tag{12}$$

5 Results and Discussions

The intention of this Section is to experiment the IEDF and benchmark it against the adopted Abhaya EDF model in Sect. 3.2. The evaluation of the models was carried out using analytical methods. The main metrics measured are average waiting time. We implemented EDF models in Matlab to evaluate the performance.

5.1 The Analytical Results

The waiting times of the four priority queues were computed for both models using an iterative process at system loads, $\rho = 0.3$, 0.45, 0.6, 0.75 and 0.9. Table 1 shows the estimated waiting times the four priority queues for the Adopted and Improved EDF models. We assumed the same parameters for deadlines, deadline differences, service times and second moments for various system loads.

Table 1. Waiting times for Adopted and Improved Abhaya EDF models-four priority queues

Load	AEDF	AEDF	AEDF	AEDF	IEDF	IEDF	IEDF	IEDF
	P1	P2	P3	P4	P1	P2	P3	P4
0.30	51.335	225.843	549.792	1019.489	51.335	207.775	500.911	876.030
0.45	64.232	232.496	788.239	1758.080	64.232	205.624	672.913	1354.04
0.60	91.612	535.430	1107.482	2171.020	91.612	431.022	816.768	1299.689
0.75	29.225	212.829	902.871	2661.403	29.225	170.263	663.700	1392.291
0.90	36.283	1127.756	2639.266	7067.499	36.283	258.811	1175.739	4324.818

5.2 Discussions of the Results

Figure 2 is a graphic representation of the waiting time against total load for four priority queues of the EDF and IEDF models. Results obtained show that: (i) the highest priority $P1$ packets had the lowest waiting times at all loads for the Adopted Abhaya EDF and Improved Adopted Abhaya EDF models. Packets that arrive are serviced on first come first serve basis with minimal delay. This validates the claim that in multi-server Queuing Systems when the average service rate is more than average arrival rate then there are no or minimal delays. (ii) the waiting time for $P2$, $P3$ and $P4$ packets increases with increasing total load in both the Adopted and Improved Adopted Abhaya EDF models. This confirms fact that in multi-server Queuing Systems increasing total load, and when the number of packets in the system is more than or equal to the number of servers then all servers will be busy resulting into increasing longer waiting times. (iii) At any instant the waiting time for $P4$ is significantly higher than $P3$, $P2$ and $P1$ packets in the Adopted Abhaya EDF model; We further note that the Improved Adopted Abhaya EDF model provides bigger relative improvements in waiting times for $P4$, $P3$ and $P2$ packets. This validates the authors two claims that (a) by EDF favoring higher priority packets ends up increasing the waiting times of lower priority packets. (b) low-priority queue packet starvation is avoided by the IEDF model. (iv) at higher system loads, IEDF model provides higher improvements in waiting times for packets compared to EDF. High system loads are associated with high arrival rates resulting into long delays for lower priority class packets.

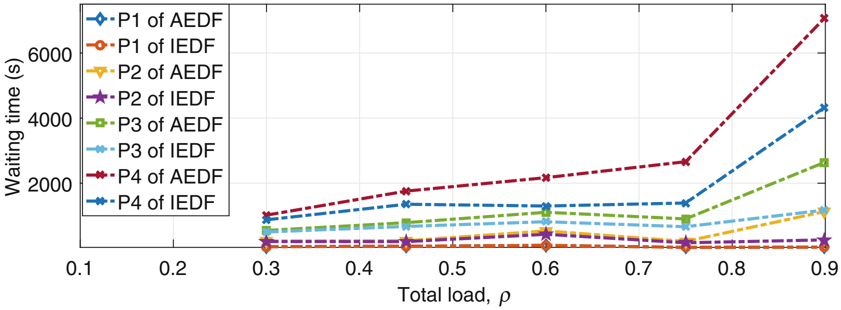

Fig. 2. Waiting time as a function of total load

6 Conclusion and Future Research

We developed a novel IEDF scheduling scheme that reduces the average waiting time of the priority queue packets. We compared average waiting times of four priority queues i.e, $P1$-high, $P2$-medium, $P3$-normal and $P4$-low at various system loads for the IEDF and AEDF models. From the results obtained, the IEDF shortens the waiting times of packets in all queues as compared to AEDF. In future research a new algorithm should be developed that will minimize relative performance gaps, study starvation trends and effects of selective preemption based on remaining processing time of lower queue packets.

References

1. Kuo, W., Chu, S.: Energy efficiency optimization for mobile ad-hoc networks. IEEE Open Access J. **4**, 928–940 (2016)
2. Chen, W., Guan, Q., Jiang, S., Guan, Q., Huang, T.: Joint QoS provisioning and congestion control for multihop wireless networks. EURASIP J. Wirel. Commun. Network. **2016** (2016)
3. Malik, S., Ahmad, S., Ullah, I., Park, D.H., Kim, D.: An adaptive emergency first intelligent scheduling algorithm for efficient task management and scheduling in hybrid of hard real-time and soft real-time embedded IoT systems. Sustainability **11**(8), 2192 (2019)
4. Abhaya, V., Tari, Z., Zeephongsekul, P., Zomaya, A.Y.: Performance analysis of EDF scheduling in a multi-priority preemptive M/G/1 queue. IEEE Trans. Parallel Distrib. Syst. **25**(8), 2149–2158 (2014)
5. Abhaya, G.: Towards achieving execution time predictability in web services middleware. Ph.D. dissertation, School of Computer Science and Information Technology, College of Science, Engineering, and Health, RMIT University, Melbourne, Victoria (2012)
6. Liu, C., Layland, J.W.: Scheduling algorithms for multiprogramming in a hard-real-time environment. J. ACM **20**(1), 46–61 (1973)
7. Rath, M., Pati, B., Pattanayak, B.K.: Cross layer based QoS platform for multimedia transmission in MANET. In: Proceedings of the 11th International Conference on Intelligent Systems and Control (ISCO), January 2017
8. Kargahi, M., Movaghar, A.: A two-class M/M/1 system with preemptive non real-time jobs and prioritized real-time jobs under earliest-deadline-first policy. Sci. Iran. **15**(2), 252–265 (2008)
9. Dehbi, Y., Mikou, N.: Priority assignment for multimedia packet scheduling in MANET. In: Proceedings of the International Conference on Signal Image Technology and Internet Based Systems, November 2008
10. Barhoun, R., Namir, A.: Packet scheduling of two classes flow. Int. J. Comput. Sci. Inf. Technol. (IJCSIT) **3**(4) (2011)
11. Arunkumar, B., Avudaiammal, R., Swarnalatha, A.: QoS - based packet scheduler for hybrid wireless networks. Int. J. Netw. (IJN) **1** (2015)
12. Chen, K., Decreusefond, L.: An approximate analysis of waiting time in multi-classes M/G/1/./EDF queues. In: Proceedings of the 1996 ACM SIGMETRICS International Conference on Measurement and Modeling of Com, Las Vegas, NY,USA (1996)
13. Charles, A., Bensraj, R.: Enhanced weighted fair-queuing technique for improving the QoS in MANET. Int. J. Adv. Eng. Technol. **7** (2016)
14. Kumar, V.: Improving quality of service in mobile ad-hoc networks (MANETs) using adaptive broadcast scheduling algorithm with dynamic source routing protocol. J. Comput. Theor. Nanosci. **14**(9), 4370–4376 (2017)
15. Bolch, G., Greiner, S., Meer, H.D., Trivedi, K.S.: Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications. Wiley-Blackwell, Hoboken (2006)
16. Kleinrock, L.: Queueing Systems Volume 2: Computer Applications (1976)