

TLSA: A Two Level Scheduling Algorithm for Multiple packets Arrival in TSCH Networks

Manjunath G. Asuti^{1*}, and Prabhugoud I.Basarkod²

¹ School of Electronics and Communication Engineering, REVA UNIVERSITY
Bengaluru,560064-India

[e-mail : manjunathgasuti@reva.edu.in]

² School of Electronics and Communication Engineering, REVA UNIVERSITY
Bengaluru,560064-India

[e-mail: basarkodpi@reva.edu.in]

*Corresponding author : Manjunath G. Asuti

*Received January 26, 2019; revised December 28, 2019 ; accepted January 27, 2020;
published August 31, 2020*

Abstract

Wireless communication has become the promising technology in the recent times because of its applications in Internet of Things(IoT) devices. The IEEE 802.15.4e has become the key technology for IoT devices which utilizes the Time-Slotted Channel Hopping (TSCH) networks for the communication between the devices. In this paper, we develop a Two Level Scheduling Algorithm (TLSA) for scheduling multiple packets with different arrival rate at the source nodes in a TSCH networks based on the link activated by a centralized scheduler. TLSA is developed by considering three types of links in a network such as link i with packets arrival type 1, link j with packets arrival type 2, link k with packets arrival type 3. For the data packets arrival, two stages in a network is considered. At the first stage, the packets are considered to be of higher priority. At the second stage, the packets are considered to be of lower priority. We introduce level 1 schedule for the packets at stage 1 and level 2 schedule for the packets at stage 2 respectively. Finally, the TLSA is validated with the two different energy functions i.e., $y = e^{ax} - 1$ and $y = 0.5x^2$ using MATLAB 2017a software for the computation of average and worst ratios of the two levels.

Keywords: Time-Slotted Channel Hopping, IoT, IEEE 802.15.4e, Packet Scheduling, Energy cost Function.

1. Introduction

Internet of Things (IoT) comprises of interconnection of various heterogeneous devices in which the data packets arrive at the source node with different rates. The processing of the packets arrival rate can affect the reliability and energy efficiency of the network. The data collected by the gateway needs an optimal scheduling policy for the existing traffic generated in the Wireless Sensor Networks [1]. Due to huge demand of the traffic generated in the Wireless Sensor Networks, it is essential to analyze the energy efficiency of the wireless links. At the same time, it is difficult to analyze the energy efficiency of a wireless networks because of the co-channel interference. Therefore, it is essential to optimize the energy efficiency of a network [2]. Scheduling of high priority real-time and low priority non real-time packets at the source node plays an important role in reducing the energy consumption and end-to-end delay of nodes [3].

Subsequently, the scheduling of packets in real-time traffic with deadline constraints has become an perineal problem in many real-time applications. A packet is dropped if it is not transmitted within the deadline constraints or due to their arrival rate, size and bandwidth [4]. In general, scheduling of packets should maximize the system performance by providing better QoS requirements and the packets arrival rate forms an important aspect in a TSCH networks which has a direct impact on the energy consumption of the node. Additionally, less emphasis is given to the packets arrival rate at the source node whenever a particular link is scheduled in TSCH networks [5] – [12]. As TSCH networks are used in many IoT applications there is a need to focus on the packets arrival rate at the source node and schedule them based on the application requirements.

In this paper, we develop a Two Level Scheduling Algorithm (TLSA) for scheduling the multiple packets at source node depending on their arrival rate in a TSCH networks. The number of packets which arrive may be equal to either 1 or more than 2 in a given timeslot. At each arrival of the packets the source node decides either to send the packets or queue them in a buffer or defer some of the packets and transmit them in future timeslots.

A centralized scheduler acts as a coordinator in a TSCH network and it is responsible for the multiple packets scheduling by allocating them to different timeslots based on the active link chosen and the packets arrival rate. Centralized scheduler has got the complete information about the active link, number of nodes in a network and the type of packets arrival at the source node.

The remainder of this paper is organized as follows. In section 2, related work in the field of packet scheduling in Wireless Sensor Networks is presented. In section 3, the Network Model is introduced. In section 4, the problem statement is defined. In section 5, the proposed Two Level Scheduling Algorithm (TLSA) architecture is explained. In section 6, the cost of scheduling each link is calculated. In section 7, some performance metrics are defined. In section 8, the performance evaluation of TLSA is presented. In section 9, TLSA is compared with other packet scheduling techniques. Finally, the conclusion is made in section 10.

2. Related Work

Bo Ji et al [13] proposed an optimal link Scheduling scheme for packets in mutihop networks without the requirement of per-hop, per-flow information of the network and reduced the delay by introducing simple queue structure. Further, they analyzed the proposed

scheme with the help of fluid limit techniques in order to achieve the required optimality in the network. Juan et al [14] addressed the problem of packet scheduling in real-time traffic generated in adhoc networks by assigning a deadline for the number of packets to be transmitted. Reema et al [15] proposed a unique mechanism called dynamic scheduling to provide better QoS requirements in Internet of Things (IoT) with dynamic allocation of bandwidth for the different applications. Saima et al [16] focused on scheduling of the message in IoT environment by identifying the node failure in the network. Later, they proposed an efficient method for recovery of the failure node with achievement of better response time and consumption of energy.

Gomathi et al [17] developed a new packet scheduling method which is Nearest Job Next with combination (NJNC). NJNC provides the assistance for data collection with servicing for multiple requests. Zhu et al [18] proposed an Adaptive packet scheduling scheme for real time traffic generated in WLAN mesh networks by adaptively adjusting the packet length and introducing intra-class service differentiation. Some of the improvements achieved by them are end-to-end delay, throughput, packet loss rate and QoS requirements. Lutful et al [19] introduced three different classes of priorities for packet scheduling in wireless networks by assigning higher priority for real time packets and next two lower priorities for type-1 non real-time packets and type-2 non real-time packets. Jing Zhang et al [20] focused on energy efficiency fairness of the nodes in the wireless powered communication networks. They also proposed an efficient algorithm for the optimization of energy efficiency with the subsequent reduction of computational complexity in the wireless powered communication networks. Finally, they achieved a significant improvement in energy efficiency fairness of nodes when compared to the baseline scheme.

Djabir et al [21] considered the IIoT data scheduling by dividing them into high and low priority in hierarchical fog servers deployed across different tiers in a network. Jing Yang et al [22] models the data packets arrivals and energy harvested in a network by considering two different types of scenarios for the data arrival at the source node. In the first scenario, they considered the packets are already available at the source node and in second scenario the packets are available only during the transmission. They also proposed an optimal offline scheduling algorithm which minimizes the time taken to deliver all the packets to reach the destination. El Gamal et al [23] considered two different kinds of channels i.e downlink channel and uplink channel. In downlink channel they developed an algorithm called move right which solves the problem of packet scheduling by imposing an additional constraints such as packet deadline and buffers. In case of uplink channel they develop a move right express algorithm which utilizes an look ahead buffer and outperforms better compared to an optimal offline schedule.

E. Uysal-Biyikoglu et al [24] proposed an algorithm for efficient transmission of packets in a wireless link which minimizes the energy consumed by the network. They also developed an optimal schedule algorithm in which the transmission rate varies for a given amount of time whenever packets are transmitted. W. Chen et al [25] focused on transmission of packets efficiently with the consideration of packet delay constraint with fading channel conditions and finally they compared the performance of delay and energy in types of different scenarios. Yanting wu et al [26] introduced a greedy offline scheduling algorithm which schedules the packets depending on their arrivals, additionally they also developed an online algorithm and finally they compared the two algorithms with respect to the competitive ratio.

N. Ehsan et al [27] considered the job arrivals at the source node as random arrival process and addressed the problem of delay optimal allocation in a multiple access channel by

proposing different rate allocation policies which minimized the delay in the system. Murtaza Zafer et al [28] models the channel variation with respect to time by stochastic process and data packets with certain delay constraints, they develop an optimal rate control policy for data packets and channel in the network. Murtaza Zafer et al [29] adapts a rate-control mechanism which minimizes the transmission energy expenditure by meeting the QoS metric. They also developed an online policy for arrival of packets with deadline constraints.

Paulo Salvador et al [30] models the packets arrived and their size distribution with the help of Markovian arrival process by proposing a traffic model by adapting a parameter fitting procedure to provide a match between autocovariance and marginal distribution of the packets that arrived. O.Ozel, E.Uysal-Biyikoglu [31] – [32] models the transmission of data packets over a fading channel. K.Tutuncuoglu et al [33] proposed an optimal transmission policy mechanism by adapting an energy harvesting method because of the limited energy in the node.

Kirik Pruhs [34] introduced an online scheduling scheme for the server systems. N.Bansal et al [35] considers the speed scaling problem with an objective to reduce the total flow and energy consumed in the network. Further, they also developed an online algorithm to schedule an unfinished job at the speed of energy cost function. M. Lin et al [36] Introduced a geographical diverse data centers in internet-scale systems. They reduced the switching and operation costs in the network with optimization of the convex function. Yujae et al [37] investigated the impact of packet arrival rates on mean packet delay with coexistence of Wi-Fi AP and cellular systems. Later, they analyzed the maximum packet arrival rate for both the systems.

3. Network Model

Consider a TSCH network with a centralized scheduler as illustrated in Fig. 1. There are three types of links in a network i.e., link i, link j, link k and each link with their own data packets arrival rate. Whenever a particular link is chosen by the centralized scheduler, the packets are scheduled based on their arrival rates. We assume that packets arrive at link i with constant rate, packets at link j with burst arrival rate, and packets at link k with random arrival rate. The network is divided into two stages i.e., stage 1 and stage 2. At stage 1, the data packets arrive with higher priority, and at stage 2, the data packets arrive with lower priority. At each stage, any one of the link is active at any particular time instance and the centralized scheduler performs the packet scheduling by allocating the timeslots to all incoming packets. A timeslot may be allocated to a packet which has arrived immediately or it may be deferred for some time. Given a list of links in a network with their addresses, the scheduler should select a particular active link and assign a packet scheduling algorithm so that all the incoming packets are scheduled in a TSCH network.

3.1 Illustration of TSCH Slotframe, packets arrival and timeslots

In TSCH networks, the link scheduling is carried out with the number of slotframes. Each slotframe consists of n number of timeslots. Let us consider 1 slotframe for link scheduling in which each slotframe consists of 10 timeslots (each of 10 ms duration). Let the number of packets scheduled per timeslot be 10.

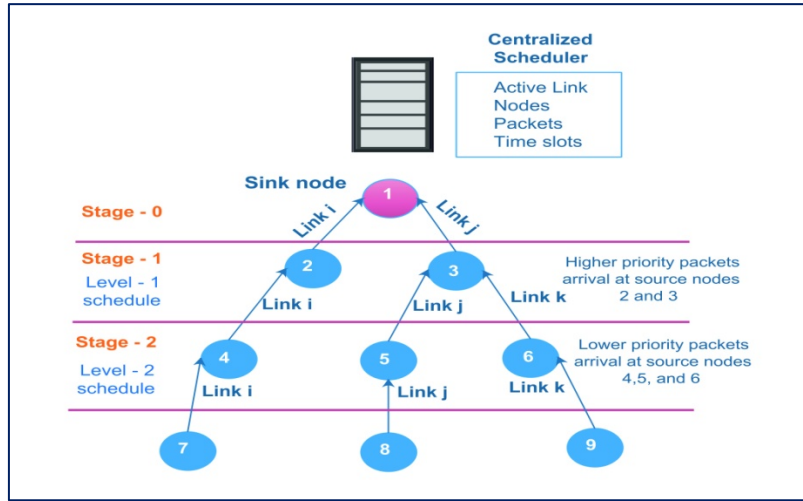
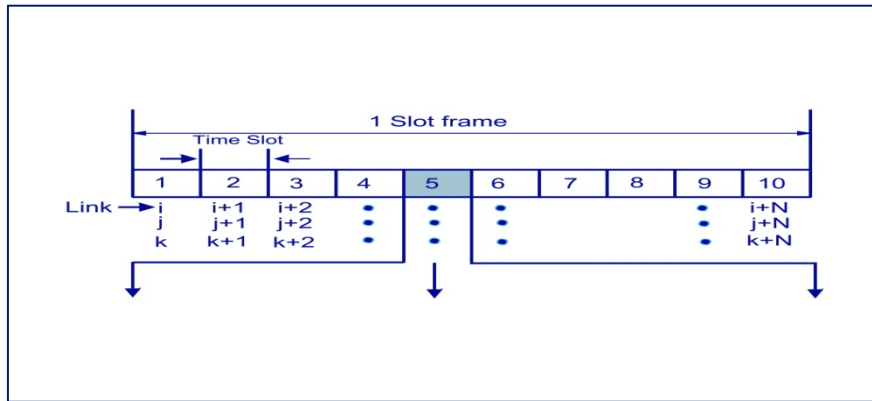
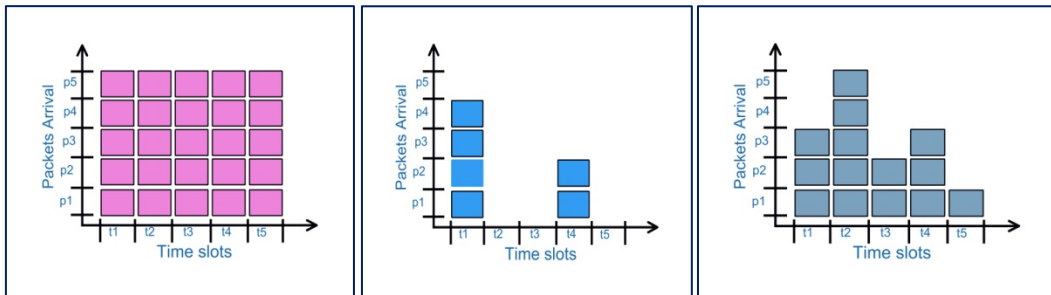


Fig. 1. TSCH Network with Centralized scheduler and different links



(a)



(b)

(c)

(d)

Fig. 2. (a) TSCH slotframe (b) Packets arrival for link i (c) Packets arrival for link j (d) Packets arrival for link k

Fig. 2 illustrates the slotframe, timeslots, and the number of packets arrival in the three types of active links in a TSCH network. We employ a recurrent scheduling approach [38] for each of the link activated in the network. Let $i, i + 1, i + 2, \dots, i + N$ represent the recurrent link scheduling sequence for the constant arrival of packets (equal number of packets arrival at each timeslot) whenever link i is activated. Let $j, j + 1, j + 2, \dots, j + N$ represent the recurrent link scheduling sequence for the burst arrival of packets (unequal number of packets arrival at each timeslot) whenever link j is activated. Let $k, k + 1, k + 2, \dots, k + N$ represent the recurrent link scheduling sequence for the random arrival of packets (random number of packets arrival at each timeslot) whenever link k is activated.

3.2 Assumptions

The following are the assumptions for the proposed TLSA.

- For simplicity, we assume frequency is fixed with multiple timeslots in a TSCH networks.
- All the information such as number of active links, number of packets available at the source node, number of timeslots is available at the centralized scheduler.
- The problem addressed is to first find the active link in a network and then allocate the timeslots for the incoming packets depending on their arrivals.
- Centralized scheduler does not have any information about the packets in queue, packets dropped, packets that collide.
- The Packet size, number of timeslots and transmission rate is fixed.
- All the data packets are available at the initial timeslot i.e. $t=1\text{msec}$.
- Channel is assumed to be noise free.

3.3 Preliminaries

3.3.1 Cost of a link

In TSCH network, the cost of a link is computed based on the number of data packets scheduled in different timeslots. In the network model illustrated in **Fig. 1**, for each link, the packets arrival rate is different and it is essential to estimate the cost of each link scheduled. For illustration, let us estimate the cost of two links i.e., link i and link j . Let link i be scheduled at certain time instant with timeslots t_1^i and t_2^i , the corresponding packets be p_1^i and p_2^i respectively. Let $p_1^i = p_2^i = 2$, be the number of packets scheduled at the two timeslots $t_1^i = t_2^i = 1$. Therefore, the total number of packets scheduled are $2+2=4$, and the timeslots required are $1+1=2$. Lastly, the cost of scheduling link i will be equal to $4 \times 2=8$. Let link j be scheduled at certain time instant with timeslots t_1^j and t_2^j and the corresponding packets be p_1^j and p_2^j respectively. Let $p_1^j = p_2^j = 1$, be the number of packets scheduled at the two timeslots $t_1^j = t_2^j = 1$. Therefore, the total number of packets scheduled are $1+2=3$, and the total timeslots required are $1+1=2$. Lastly, the cost of scheduling link j will be equal to $2 \times 3 = 6$. Therefore, the cost of scheduling link j is less when compared to the cost of scheduling link i for the two timeslots t_1 and t_2 .

3.3.2 Energy Function

In our network model, we assume three types of packets arrival in TSCH networks. Therefore, we define an energy function to determine the power spent on scheduling the packets on different timeslots i.e., if more number of packets are scheduled on a particular timeslot then the energy spent at that time instant is more compared to other. For illustration, we consider three types of packets arrival: pattern generator type 1 (more packets at one

timeslot) with energy function E1, pattern generator type 2 (different packets at each timeslot) with energy function E2, and pattern generator type 3 (equal packets at each timeslot) with energy function E3. Let the total number of packets to be scheduled be 6, and the total timeslots available are 3. **Table 1** shows the Energy function computed for different types of pattern generators.

Table 1. Energy Function

Packets Type	Total number of Packets available for schedule	Packets Scheduled at timeslot t1	Packets Scheduled at timeslot t2	Packets Scheduled at timeslot t3	Energy function
Pattern generator type 1	6	6	0	0	Since all the packets are scheduled at timeslot t1, therefore more energy is spent at timeslot t1 ($E1 > E2, E3$). Energy cost is more at t1
Pattern generator type 2	6	1	3	2	Since all the packets are distributed on different timeslots, therefore, the energy spent for scheduling all the packets is $E1 < E3 < E2$. Energy cost is moderate
Pattern generator type 3	6	2	2	2	Since all the packets are scheduled at constant rate on all timeslots, therefore, the energy spent for all the timeslots is $E1=E2=E3$. Energy cost is less

From **Table 1**, it is observed that more energy is spent for scheduling the pattern generator type 1 in comparison with other types. Hence, the energy function is used to evaluate the energy spent on scheduling the packets in a particular timeslot in the network. TLSA is evaluated with two energy functions such as $y = e^{ax} - 1$ and $y = 0.5x^2$. The energy function $y = e^{ax} - 1$ is an exponential function in which a represents the constant integer value and x represents the number of packets at a particular timeslot. Similarly, the energy function $y = 0.5x^2$ is a polynomial function in which x represents the number of packets at a particular timeslot. The two energy functions represent the increasing power in a wireless network with the number of packets scheduled at a particular timeslot.

4. Problem Formulation

Consider a TSCH network with packets at the source node with different arrival rates. The packets are considered to be of higher or lower priorities depending upon their service

requirements and the traffic generated in a network. Problem formulation is to schedule all the incoming packets arriving at the source node by allocating different timeslots based on the active link chosen by the centralized scheduler.

We consider two stages in a TSCH network. At stage 1, the packets are considered to arrive with higher priorities. At stage 2, the packets are considered to arrive with lower priorities. At each stage, there are three types of links (link i, link j, link k) to be chosen by the centralized scheduler. Once the link is active there is a need to allocate the timeslots for all the incoming packets.

4.1 Case i

At stage 1: Let $p_1^{i,j,k}, p_2^{i,j,k}, \dots, p_{N-1}^{i,j,k}, p_N^{i,j,k}$ be the packets at the source node with the different arrival rates and the corresponding timeslots be $t_1^{i,j,k}, t_2^{i,j,k}, \dots, t_{M-1}^{i,j,k}, t_M^{i,j,k}$.

Where N is the total number of packets at stage 1, M is the total number of timeslots available at stage 1 in a network.

If i denotes the link for constant arrival of packets at stage 1, then

$$\sum_{N=1}^{NcMax} p_N^i = l, l \in const, \text{ where } N \text{ varies } 0 \leq N \leq Ncmax \quad (1)$$

If j denotes the link for burst arrival of packets at stage 1, then

$$\sum_{N=1}^{NbMax} p_N^j = l, \text{ where } l \text{ varies } 0 < l < lmax, N \text{ varies } 0 \leq N \leq Nbmax \quad (2)$$

If k denotes the link for random arrival of packets at stage 1, then

$$\sum_{N=1}^{NrMax} p_N^k = Random(l), \text{ where } l \text{ varies } 0 < l < lmax, N \text{ varies } 0 \leq N \leq Nrmax \quad (3)$$

4.2 Case ii

At stage 2: Let $p_{11}^{i,j,k}, p_{21}^{i,j,k}, \dots, p_{N1-1}^{i,j,k}, p_{N1}^{i,j,k}$ be the packets at the source node with the different arrival rates and the corresponding timeslots be $t_{11}^{i,j,k}, t_{21}^{i,j,k}, \dots, t_{M1-1}^{i,j,k}, t_{M1}^{i,j,k}$.

Where N1 is the total number of packets at stage 2, M1 is the total number of timeslots available at stage 2 in a network.

If i denotes the link for constant arrival of packets at stage 2, then

$$\sum_{N=1}^{NcMax} p_{N1}^i = l, l \in const, \text{ where } N1 \text{ varies } 0 \leq N1 \leq Ncmax \quad (4)$$

If j denotes the link for burst arrival of packets at stage 2, then

$$\sum_{N=1}^{NbMax} p_{N1}^j = l, \text{ where } l \text{ varies } 0 < l < lmax, N1 \text{ varies } 0 \leq N1 \leq Nbmax \quad (5)$$

If k denotes the link for random arrival of packets at stage 1, then

$$\sum_{N=1}^{NrMax} p_{N1}^k = Random(l), \text{ where } l \text{ varies } 0 < l < lmax, N1 \text{ varies } 0 \leq N1 \leq Nrmax \quad (6)$$

The objective of the TLSA is to schedule all the incoming packets at stage 1 and stage 2 with level 1 and level 2 respectively. Firstly, we estimate the cost of scheduling each link of a TSCH network. Later, we determine and analyze the energy spent on scheduling all the packets with the help of two energy functions such as $y = e^{ax} - 1$ and $y = 0.5x^2$. Finally, we compute the average ratio, worst ratio, and competitive ratio of two levels in a TSCH network.

Table 2. List of Symbols used

Symbol	Description
L_{act}	Active link in a network
M	Maximum number of timeslots
M_{cmax}	Maximum number of timeslots for constant arrival of packets
M_{bmax}	Maximum number of timeslots for burst arrival of packets
M_{rmax}	Maximum number of timeslots for random arrival of packets
N	Maximum number of packets
N_{cmax}	Maximum value for constant arrival of packets
N_{bmax}	Maximum value for burst arrival of packets
N_{rmax}	Maximum value for random arrival of packets
l	Number of packets in a particular timeslot t_M^i or t_M^j or t_M^k
Δt	Time instance
t_M^i, t_{M1}^i	Number of timeslots available in i^{th} link at stage 1 and stage 2
t_M^j, t_{M1}^j	Number of timeslots available in j^{th} link at stage 1 and stage 2
t_M^k, t_{M1}^k	Number of timeslots available in k^{th} link at stage 1 and stage 2
p_N^i, p_{N1}^i	Number of packets in the i^{th} link at stage 1 and stage 2
p_N^j, p_{N1}^j	Number of packets in the j^{th} link at stage 1 and stage 2
p_N^k, p_{N1}^k	Number of packets in the k^{th} link at stage 1 and stage 2
$p_{TN}^i, p_{TN}^j, p_{TN}^k$	Total number of packets scheduled by i^{th} link, j^{th} link, and k^{th} link
$t_{TM}^i, t_{TM}^j, t_{TM}^k$	Total number of timeslots for the i^{th} link, j^{th} link, and k^{th} link
C_T^i, C_T^j, C_T^k	Cost of scheduling i^{th} link, j^{th} link, and k^{th} link
p_{TN}^{i-k}	Total number of packets scheduled by the network
C_T^{i-k}	Total cost for scheduling all the links

Lemma 1: At each stage of a network, two active links are scheduled independently at a particular time instance.

Proof: Let us consider two links (link i and link j) which are located at stage 1 in a network. Let i^{th} link be active at the time instant Δ_{t1} . If the total number of timeslots available in the i^{th} link are $\sum_{M=1}^{McMax} t_M^i$ and the corresponding packets (packets arrival type 1) in the i^{th} link are $\sum_{N=1}^{NcMax} p_N^i$. The timeslots $\sum_{M=1}^{McMax} t_M^i$ are allocated to $\sum_{N=1}^{NcMax} p_N^i$ whenever the i^{th} link is activated. Let j^{th} link be active at the same time instant Δ_{t1} . If the total number of timeslots available in the j^{th} link are $\sum_{M=1}^{MbMax} t_M^j$ and the corresponding packets (packets arrival type 2) in the j^{th} link are $\sum_{N=1}^{NbMax} p_N^j$. The timeslots $\sum_{M=1}^{MbMax} t_M^j$ are allocated to $\sum_{N=1}^{NbMax} p_N^j$ whenever the j^{th} link is activated. The i^{th} link is scheduled for the packets arrival type 1 at time instant Δ_{t1} and j^{th} link is scheduled for the packets arrival type 2 at the same time instant Δ_{t1} . Therefore, at Δ_{t1} , two active links are scheduled independently because of the different packets arrival type in a network.

Lemma 2: At each stage of a network, a particular link may be active for any number of times in a network.

Proof: Let us consider one link (link i) located at stage 1 in a network. Let i^{th} link be active at the time instant Δ_{t1} . If the total number of timeslots available in the i^{th} link are $\sum_{M=1}^{McMax} t_M^i$ and the corresponding packets (packets arrival type 1) in the i^{th} link are $\sum_{N=1}^{NcMax} p_N^i$. Then, the timeslots $\sum_{M=1}^{McMax} t_M^i$ are allocated to $\sum_{N=1}^{NcMax} p_N^i$ whenever i^{th} link is activated. Let $(i+1)^{th}$ link be active at time instant Δ_{t2} . If the total number of timeslots available in the $(i$

$(i+1)^{\text{th}}$ link are $\sum_{M=1}^{McMax} t_M^{i+1}$ and the corresponding packets (packets arrival type 1) in the $(i+1)^{\text{th}}$ link are $\sum_{N=1}^{NcMax} p_N^{i+1}$. Similarly, the timeslots $\sum_{M=1}^{McMax} t_M^{i+1}$ are allocated to $\sum_{N=1}^{NcMax} p_N^{i+1}$ whenever the $(i+1)^{\text{th}}$ link is activated. Here $\Delta_{t_1} \neq \Delta_{t_2}$, Δ_{t_1} and Δ_{t_2} are the different time instants in a network. Therefore, a particular link is active for any number of times in a network because the i^{th} link is activated at time instant Δ_{t_1} and the $(i+1)^{\text{th}}$ link is activated at time instant Δ_{t_2} . Similarly, it can be proved for j^{th} link and k^{th} link.

Lemma 3: In a two stage network, there exists some packets which are not scheduled in level 1 are scheduled in level 2.

Proof: Let us consider two stages in a network i.e., stage 1 and stage 2. At stage 1, if the total number of timeslots available in the i^{th} link are $\sum_{M=1}^{McMax} t_M^i$ and the corresponding packets (high priority) in the i^{th} link are $\sum_{N=1}^{NcMax} p_N^i$. Then, the timeslots $\sum_{M=1}^{McMax} t_M^i$ are allocated to packets $\sum_{N=1}^{NcMax} p_N^i$ whenever link i is scheduled (level 1). Similarly, at stage 2, if the total number of timeslots available in the i^{th} link are $\sum_{M=1}^{McMax} t_M^i$ and the corresponding packets (low priority) in the i^{th} link are $\sum_{N=1}^{NcMax} p_{N1}^i$. Then, the timeslots $\sum_{M=1}^{McMax} t_M^i$ are allocated to packets $\sum_{N=1}^{NcMax} p_{N1}^i$ whenever the i^{th} link is scheduled (level 2). Since packets at stage 2 arrive with lower priorities they are not scheduled by level 1 but scheduled by level 2. Therefore, level 1 schedules high priority packets and level 2 schedules low priority packets. Similarly, it can be proved for j^{th} link and k^{th} link.

Lemma 4: At each stage of a network, the number of packets arrived is equal to the number of packets scheduled at any particular timeslot.

Proof: Let us consider the i^{th} link located at stage 1 in a network. Let the i^{th} link be active at the time instant Δ_{t_1} . Let $p_1^i = l_1$ be the number of packets arrived at timeslot t_1^i , and $p_2^i = l_2$ be the number of packets arrived at timeslot t_2^i at time instant Δ_{t_1} . Let t_1^i be the timeslot allocated to p_1^i packets and let t_2^i be the timeslot allocated to p_2^i packets at the time instant Δ_{t_2} . If $l_1 = l_2$, then the number of packets arrived at time instant Δ_{t_1} , and the number of packets scheduled at time instant Δ_{t_2} are same for the two timeslots t_1^i and t_2^i . Therefore,

$$\frac{\sum_{M=1}^{McMax} t_M^i}{\sum_{N=1}^{NcMax} p_N^i} = 1 \text{ (constant)}$$

5. TLSA Architecture and Description

5.1 TLSA Architecture

Two Level Scheduling Algorithm (TLSA) Architecture is illustrated in [Fig. 3](#). It shows three types of packets arrival i.e., packets arrival type 1, packets arrival type 2, and packets arrival type 3. All the packets arrive at the source node with different rates so that TLSA will schedule all the incoming packets in different timeslots by selecting either level 1 or level 2 schedule. In our model, we assume packets arrival with type 1 when the centralized scheduler has selected link i for scheduling, the packets arrival with type 2 when the centralized scheduler has selected link j for scheduling, and packets arrival with type 3 when the centralized scheduler has activated link k for scheduling.

TLSA schedules all the incoming packets arriving at the source node in a TSCH network depending on their priorities i.e., we assume two kinds of priorities such as high priority

(Real-time packets) and low priority (Non-real-time packets). We process the real-time packets with higher priority in order to reduce the end-to-end delay and non-real-time packets with lower priority. Initially, TLSA checks for the available timeslots in a TSCH network. After, a particular link has been chosen by the centralized scheduler the packets may arrive (implies for all 3 packets arrival type) at the source node with the higher or lower priorities depending on the traffic generated in a network. Therefore, level 1 will schedule all the incoming packets with higher priorities (packets located at stage 1) in the different timeslots. Later, level 2 will schedule all the incoming packets with lower priorities (packets located at stage 2) in the different timeslots. In the proposed model, the number of packets arrival at the source node may be equal to 1 or more than 2 on a given timeslot.

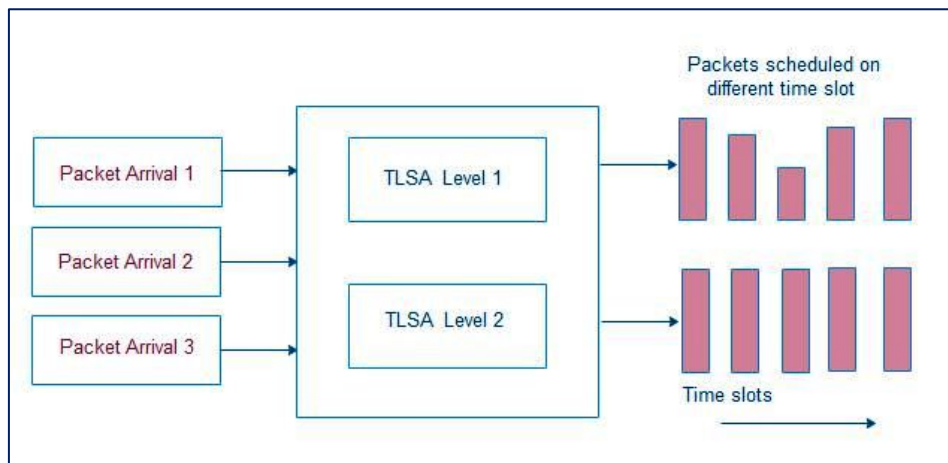


Fig. 3. TLSA: Two Level Scheduling Algorithm Architecture

5.2 TLSA Description

Algorithm 1 starts by initiating three types of active links, data packets arrival with different rates and the timeslots available in a TSCH network. All the packets arriving at the source node are scheduled by the level 1 and level 2 schedules. Upon the activation of a particular link, the centralized scheduler will select the suitable algorithm for scheduling of all the packets with the timeslot allocation. We generate three types of packets arrival at the source node depending upon the maximum value (0.8) set for the packet generation. Packet arrival type 1 is generated when the pattern selector is less than 0.3. Packet arrival type 2 is generated if the pattern selector lies in between 0.3 to 0.5. Packet arrival type 3 is generated if the pattern selector is greater than 0.5. Later, depending upon the active link chosen and the packets arrival type algorithm 2 or 3 or 4 are executed. Algorithm 1 schedules all the incoming packets with Level 1 and Level 2. Algorithm 1 ends up by computing the competitive ratio, worst ratio and average ratio for Level 1 and Level 2 schedules. Algorithm 2, 3, and 4 describes the scheduling of all three types of packets arrival whenever links i , j , and k are activated. Each algorithm starts by initializing the number of timeslots and the packets arrival for level 1 schedule (line 1). Similarly, initializing the number of timeslots and the packets arrival for level 2 schedule (line 2). Next, each Algorithm checks whether the initial timeslot is less than the last timeslot for level 1 and level 2 schedules (line 5), and then each algorithm will check the packets arrival type (line 6). Later, each algorithm checks for the availability of timeslots (line 7), based on the availability of timeslots each algorithm predicts the priorities (higher or lower priorities) of every incoming packets and

their stages (stage 1 or stage 2) where they are located (lines 10 to 11). Algorithm 2, 3, and 4 initiates level 1 to schedule all the incoming packets (high priorities) located at stage 1 (lines 12 to 19) and level 2 to schedule all the incoming packets (low priorities) located at stage 2 (lines 24 to 31). Finally, each algorithm checks whether all the packets have been scheduled by the centralized scheduler (line 32), algorithm 2, 3, and 4 goes to step 4 if packets are pending (line 33). Algorithm 2, 3, and 4 ends at line 38.

Algorithm 1: TLSA: Two Level Scheduling Algorithm

```

1: Input: Link i, Link j, Link k, timeslots, no of packets arrival with different rates
2: Begin
3: Computation of the active link (Link i, Link j, Link k)
4: Select the active link  $\sum_{act=i}^q L_{act}$  // Centralized scheduler selects the active link
5: Set p_max=0.8 // set maximum value for generation of packets
6: if (pat_sel < 0.3) then
7:   Select link i for scheduling with packet arrival type 1
8:   goto algorithm 2 // Initiate type 1 packets scheduling
9: else if (pat_sel >=0.3 && pat_sel < 0.5) then
10:  Select link j for scheduling with packet arrival type 2
11:  goto algorithm 3 // Initiate type 2 packets scheduling
12:else
13:  Select link k for scheduling with packet arrival type 3
14:  goto algorithm 4 // Initiate type 3 packets scheduling
15:end if
16: Output: Level 1 Schedule, Level 2 Schedule
17: Compute the cost ratio of Level 1 and Level 2 schedules // Competitive ratio
18: Compute the Worst ratio of Level 1 and Level 2 schedules
19: Compute the Average ratio of Level 1 and Level 2 schedules
19: End

```

Algorithm 2: Link i Schedule for packets type 1

```

1: Input: timeslots =  $t_1^i, t_2^i, \dots, t_{M-1}^i, t_M^i$ , packets arrival =  $p_1^i, p_2^i, \dots, p_{N-1}^i, p_N^i$ 
2: Input: timeslots =  $t_{11}^i, t_{21}^i, \dots, t_{M1-1}^i, t_{M1}^i$ , packets arrival =  $p_{11}^i, p_{21}^i, \dots, p_{N1-1}^i, p_{N1}^i$ 
3: Output: Level 1 schedule, Level 2 schedule
4: Begin
5: While ( $t_1^i < t_M^i$  &&  $t_{11}^i < t_{M1}^i$ ) do
6:   if packets arrival rate is  $\sum_{N=1}^{NcMax} p_N^i = l$ ,  $l \in const$  then
7:     if timeslots  $\neq \emptyset$  then // check for the available timeslots in the network
8:       for ( $i = t_1^i, t_2^i, \dots, t_{M-1}^i, t_M^i$ ) do //timeslots for level 1 schedule
9:         for ( $j = p_1^i, p_2^i, \dots, p_{N-1}^i, p_N^i$ ) do // packets for level 1 schedule
10:        for every incoming packets do // iteration 1 for Level 1 schedule
11:          if (data packets arrived = stage 1 && high priority) then
12:            Initiate Level 1 Schedule :
13:            Schedule  $t_M^i = p_1^i$  // Allocate last timeslot to the first packet arrived
14:             $t_M^i = t_M^i - 1$  // decrement timeslot for the next packets
15:             $p_1^i = p_1^i + 1$  // next packets to be scheduled
16:          end if
17:        end for
18:      end for
19:    end for
20:  for ( $i = t_{11}^i, t_{21}^i, \dots, t_{M1-1}^i, t_{M1}^i$ ) do //timeslots for level 2 schedule

```

```

21: for (j = p11i, p21i, - - - - pM1-1i, pM1i) do //packets for level 2 schedule
22:   for every incoming packets do // iteration 2 for Level 2 schedule
23:     if (data packets arrived=stage 2 && low priority) then
24:       Initiate Level 2 Schedule :
25:       Schedule tM1i = p11i // Allocate last timeslot to the first packet arrived
26:       tM1i = tM1i - 1 // decrement timeslot for the next packets
27:       p11i = p11i + 1 // next packets to be scheduled
28:     end if
29:   end for
30: end for
31: end for
32: if (∑N=1NcMax pNi ≠ ∅ && ∑N=1NcMax pN1i ≠ ∅) then
33:   goto step 4 // schedule remaining packets by Level 1 and Level 2
34: end if
35: end if
36: end if
37: end while
38:End

```

Algorithm 3: Link j Schedule for packets type 2

```

1: Input: timeslots = t1j, t2j, - - - - tM-1j, tMj, packets arrival= p1j, p2j, - - - - pN-1j, pNj
2: Input: timeslots = t11j, t21j, - - - - tM1-1j, tM1j, packets arrival= p11j, p21j, - - - - pN1-1j, pN1j
3: Output: Level 1 schedule, Level 2 schedule
4:Begin
5: While (t1j < tMj && t11j < tM1j) do
6:   if packets arrival rate is ∑N=1NbMax pNj = l, where l varies 0 < l < lmax then
7:     if timeSlots ≠ ∅ then // check for the available timeslots in the network
8:       for (i = t1j, t2j, - - - - tM-1j, tMj) do // timeslots for level 1 schedule
9:         for (j = p1j, p2j, - - - - pN-1j, pNj) do // packets for level 1 schedule
10:        for every incoming packets do // iteration 1 for Level 1 schedule
11:          if (data packets arrived =stage 1 && high priority) then
12:            Initiate Level 1 schedule:
13:            Schedule tMj = p1j // Allocate last timeslot to the first packet arrived
14:            tMj = tMj - 1 // decrement timeslot for the next packets
15:            p1j = p1j + 1 // next packets to be scheduled
16:          end if
17:        end for
18:      end for
19:    end for
20:    for (i = t11j, t21j, - - - - tM1-1j, tM1j) do // timeslots for level 2 schedule
21:      for (j = p11j, p21j, - - - - pN1-1j, pN1j) do // packets for level 2 schedule
22:        for every incoming packets do // iteration 2 for Level 2 schedule
23:          if (data packets arrived =stage 2 && low priority) then
24:            Initiate Level 2 Schedule:
25:            Schedule tM1j = p11j //Allocate last timeslot to the first packet arrived
26:            tM1j = tM1j - 1 // decrement timeslot for the next packets
27:            p11j = p11j + 1 // next packets to be scheduled
28:          end if

```

```

29: end for
30: end for
31: end for
32: If (  $\sum_{N=1}^{NbMax} p_N^j \neq \emptyset$  &&  $\sum_{N=1}^{NbMax} p_{N1}^j \neq \emptyset$  ) then
33:   goto step 4 // schedule remaining packets by Level 1 and Level 2
34: end if
35: end if
36: end if
37: end while
38: End

```

Algorithm 4: Link k Schedule for packets type 3

```

1: Input: timeslots =  $t_1^k, t_2^k, \dots, t_{M-1}^k, t_M^k$ , packets arrival =  $p_1^k, p_2^k, \dots, p_{N-1}^k, p_N^k$ 
2: Input: timeslots =  $t_{11}^k, t_{21}^k, \dots, t_{M1-1}^k, t_{M1}^k$ , packets arrival =  $p_{11}^k, p_{21}^k, \dots, p_{N1-1}^k, p_{N1}^k$ 
3: Output: Level 1 schedule, Level 2 schedule
4: Begin
5: While (  $t_1^k < t_M^k$  &&  $t_{11}^k < t_{M1}^k$  ) do
6:   if packets arrival rate is  $\sum_{N=1}^{NrMax} p_N^k = \text{Random}(l)$ , where  $l$  varies  $0 < l < lmax$  then
7:     if timeSlots  $\neq \emptyset$  then // check for the available timeslots in the network
8:       for (  $i = t_1^k, t_2^k, \dots, t_{M-1}^k, t_M^k$  ) do // timeslots for level 1 schedule
9:         for (  $j = p_1^k, p_2^k, \dots, p_{N-1}^k, p_N^k$  ) do // packets for level 1 schedule
10:          for every incoming packets do // iteration 1 for Level 1 schedule
11:            if (data packets arrived = stage 1 && high priority) then
12:              Initiate Level 1 schedule:
13:              Schedule  $t_M^k = \text{Random}(p_1^k)$  // Allocate timeslot randomly to first packet arrived
14:               $t_M^k = t_M^k - 1$  // decrement timeslot for the next packets
15:               $p_1^k = p_1^k + 1$  // next packets to be scheduled
16:            end if
17:          end for
18:        end for
19:      end for
20:      for (  $i = t_{11}^k, t_{21}^k, \dots, t_{M1-1}^k, t_{M1}^k$  ) do // timeslots for level 2 schedule
21:        for (  $j = p_{11}^k, p_{21}^k, \dots, p_{N1-1}^k, p_{N1}^k$  ) do // packets for level 2 schedule
22:          for every incoming packets do // iteration 2 for Level 2 schedule
23:            if (data packets arrived = stage 2 && low priority) then
24:              Initiate Level 2 Schedule :
25:              Schedule  $t_{M1}^k = \text{Random}(p_{11}^k)$  // Allocate last timeslot randomly to first packet arrived
26:               $t_{M1}^k = t_{M1}^k - 1$  // decrement timeslot for the next packets
27:               $p_{11}^k = p_{11}^k + 1$  // next packets to be scheduled
28:            end if
29:          end for
30:        end for
31:      end for
32:    if (  $\sum_{N=1}^{NrMax} p_N^k \neq \emptyset$  &&  $\sum_{N=1}^{NrMax} p_{N1}^k \neq \emptyset$  ) then
33:      goto step 4 // schedule remaining packets by Level 1 and Level 2
34:    end if
35:  end if
36: end if
37: end while
38: End

```

6. Computation of Cost

In this section, we estimate the cost of scheduling the individual link based on the number of packets scheduled in a TSCH network. Finally, the total cost of the network is computed with addition of all the links.

6.1 Cost of link i

If the total number of timeslots available for scheduling i^{th} link is $t_{TM}^i = \sum_{M=1}^{McMax} t_M^i$ and the total number of packets to be scheduled by i^{th} link is $p_{TN}^i = \sum_{N=1}^{NcMax} p_N^i$, then the cost for scheduling i^{th} link is given by

$$C_T^i = t_{TM}^i \times p_{TN}^i \quad (7)$$

6.2 Cost of link j

If the total number of timeslots available for scheduling j^{th} link is $t_{TM}^j = \sum_{M=1}^{MbMax} t_M^j$ and the total number of packets to be scheduled by j^{th} link is $p_{TN}^j = \sum_{N=1}^{NbMax} p_N^j$, then the cost for scheduling j^{th} link is given by

$$C_T^j = t_{TM}^j \times p_{TN}^j \quad (8)$$

6.3 Cost of link k

If the total number of timeslots available for scheduling k^{th} link is $t_{TM}^k = \sum_{M=1}^{MrMax} t_M^k$ and the total number of packets to be scheduled by k^{th} link is $p_{TN}^k = \sum_{N=1}^{NrMax} p_N^k$, then the cost for scheduling k^{th} link is given by

$$C_T^k = t_{TM}^k \times p_{TN}^k \quad (9)$$

Therefore, the total number of packets scheduled by i^{th} link, j^{th} link, k^{th} link is

$$p_{TN}^{i-k} = p_{TN}^i + p_{TN}^j + p_{TN}^k \quad (10)$$

Total number of timeslots available for scheduling i^{th} link, j^{th} link, k^{th} link is

$$t_{TM}^{i-k} = t_{TM}^i + t_{TM}^j + t_{TM}^k \quad (11)$$

Total cost of the network for scheduling i^{th} link, j^{th} link, k^{th} link is

$$C_T^{i-k} = C_T^i + C_T^j + C_T^k \quad (12)$$

7. Performance Metrics

In this section, we define the parameters used to measure the performance of level 1 and level 2 schedules in the proposed methodology.

7.1 Competitive Ratio

Competitive ratio compares the optimal solution provided by the level 1 and level 2 schedules. It is defined as the ratio of total cost estimated for scheduling the packets at level 1 to the total cost estimated for scheduling the packets at level 2.

7.2 Worst Ratio

Worst ratio compares the cost ratio obtained by the individual schedules i.e., either level 1 or level 2 with the total cost ratio obtained by level 1 and level 2 schedules. If the total cost ratio is less in comparison with the individual cost ratios, then it adds to the worst case ratio of the proposed algorithm.

7.3 Average Ratio

The average ratio of the proposed algorithm is based on the total cost obtained by level 1 and level 2 schedules to the number of simulation rounds conducted. It is defined as the ratio of total cost obtained by scheduling level 1 and level 2 schedules to the total number of simulation rounds conducted.

8. Performance Evaluation

In this section, we evaluate the performance of TLSA using MATLAB 2017a software. We generate three types of packets arriving at the source node as constant packets arrival (type 1), burst packets arrival (type 2), and random packets arrival (type 3). Level 1 schedules all the incoming packets (higher priorities) located at stage 1 and level 2 schedules all the incoming packets (lower priorities) located at stage 2. We simulate the performance of TLSA to observe the energy cost functions ($y = e^{ax} - 1$ and $y = 0.5x^2$) and the delay occurred for scheduling all the packets on a time-slotted channel. Finally, we compute the average and worst ratios of the two levels. **Table 3** provides the simulation parameters used in the proposed model. Also, **Table 4** provides the average ratio and the worst ratio computed for the two energy functions.

Table 3. Simulation Parameters

Parameter	Value
Number of Iterations	800
Maximum number of TimeSlots	200
N value	50
Maximum burst duration	5
Maximum slots between two arrivals	50
Maximum packets arrivals in a Single slot	20
Pattern Selector for packets arrivals	0.5

Table 4. Energy function

Energy Function	Average ratio			Worst ratio		
	Constant arrival	Burst arrival	Random arrival	Constant arrival	Burst arrival	Random arrival
$y = e^{ax} - 1$	1.6948	1.0804	1.6528	1.6948	1.5254	1.8879
$y = 0.5x^2$	1	1.0496	1.0819	1	1.1825	1.1569

8.1 Generation of packets

Fig. 4 illustrates the generation of constant packets arrival (type 1), burst packets arrival (type 2), and random packets arrival (type 3) at the source node. We assume the total number of timeslots equal to 200, the maximum number of packets as 2 for constant packets arrival, 20 for burst packets arrival, and 3 for random packets arrival.

8.2 Energy function $Y = e^{ax} - 1$

Fig. 5 demonstrates the scheduling of constant arrival of packets whenever the link i is scheduled. It is observed that the level 1 schedules all the incoming packets (higher priorities) at the rate of $l=2$ (packets) upto $t_M^i = 80$ timeslots with high energy cost and the remaining packets are scheduled at the rate of $l=1$ (packet) upto $t_M^i = 120$ timeslots with low energy cost. On the other hand, level 2 schedules all the incoming packets (lower priorities) at the rate of $l=1$ (packet) upto $t_{M1}^i = 50$ timeslots and the intermittent packets at the rate of $l=2$ (packets) upto $t_{M1}^i = 100$ timeslots. Later, the remaining packets are scheduled at the rate of $l=1$ (packet) upto $t_M^i = 150$ timeslots. We notice that with level 2 schedule, more energy cost is spent on intermittent packets in comparison with the rest of the packets. Finally, the average and worst ratios are computed to be 1.6948 and 1.6948 respectively.

Fig. 6 demonstrates the scheduling of burst arrival of packets whenever the link j is scheduled. It is observed that level 1 schedules all the incoming packets (higher priorities) at the rate of $l=1$ (packet) upto $t_M^j = 103$ timeslots with low energy cost and few packets at the rate of $l=2$ (packets) with high energy cost. On the other hand, level 2 schedules all the incoming packets (lower priorities) at the rate of $l=1$ (packet) on all the t_{M1}^j timeslots with equal energy cost distribution. Also, we notice that some packets are deferred in between 21 to 31 timeslots. Finally, the average and worst ratios are computed to be 1.0804 and 1.5254 respectively.

Fig. 7 demonstrates the scheduling of random arrival of packets whenever the link k is scheduled. It is observed that level 1 schedules all the incoming packets (higher priorities) at the rate of $l=2$ and 1(packet) in all t_M^k timeslots randomly. Similarly, level 2 schedules all the incoming packets (lower priorities) at the rate of $l=1$ (packet) on all t_{M1}^k timeslots except few slots. It is noticed that level 1 schedules all the incoming packets with more energy cost without deferring the packets. Similarly, the level 2 schedules all the packets with less energy cost with packets deferred on timeslots. Finally, the average and worst ratios are computed to be 1.6528 and 1.8879 respectively.

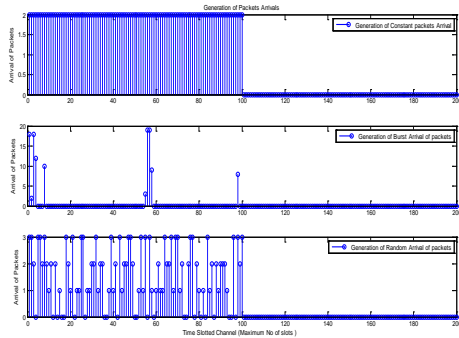


Fig. 4. Generation of Packets arrival

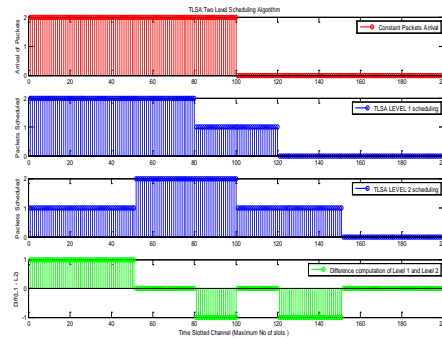


Fig. 5. TLSA Schedule for constant arrival of packets with energy function ($y = e^{ax} - 1$)

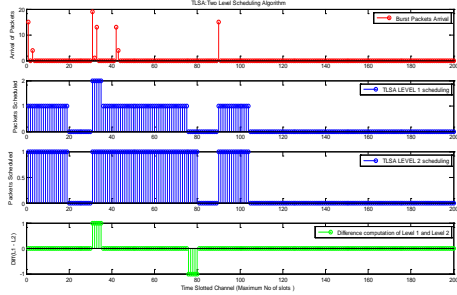


Fig. 6. TLSA Schedule for Burst arrival of packets with energy function ($y = e^{ax} - 1$)

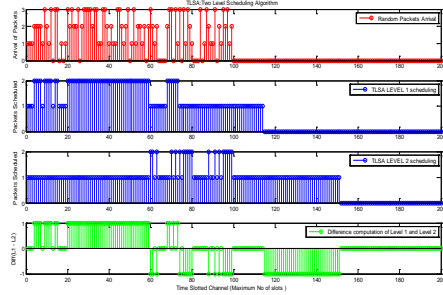


Fig. 7. TLSA Schedule for Random arrival of packets with energy function ($y = e^{ax} - 1$)

8.3 Energy function $Y= 0.5 x^2$

Fig. 8 demonstrates the scheduling of constant arrival of packets whenever the link i is scheduled. It is observed that level 1 and level 2 schedules all the incoming packets (higher and lower priorities) at the rate of $l = 2(\text{packets})$ in all t_M^i and t_{M1}^i timeslots with equal energy cost distribution. Finally, the average and worst ratios are computed to be 1 and 1 respectively.

Fig. 9 demonstrates the scheduling of burst arrival of packets whenever the link j is scheduled. It is observed that level 1 and level 2 schedules all the incoming packets (higher and lower priorities) at a decaying rate in all t_M^j and t_{M1}^j timeslots. We notice that level 1 schedules all the incoming packets with more energy cost in comparison with level 2 schedule. Finally, the average and worst ratios are computed to be 1.0496 and 1.1825 respectively.

Fig. 10 demonstrates the scheduling of random arrival of packets whenever the link k is scheduled. It is observed that level 1 and level 2 schedules all the incoming packets (higher and lower priorities) at the rate of $l = 3(\text{packets}), l = 2(\text{packets})$ and $l = 1(\text{packet})$ on all t_M^k and t_{M1}^k timeslots randomly. We notice that an almost equal energy cost distribution for level 1 and level 2 schedules. Finally, the average and worst ratios are computed to be 1.0819 and 1.1569 respectively.

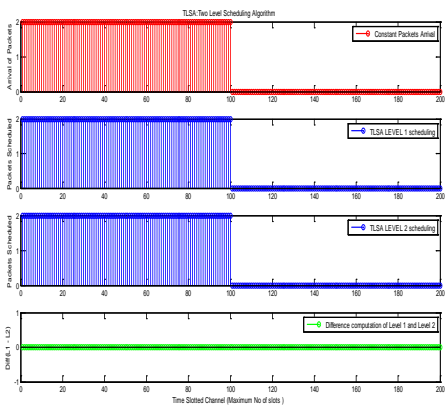


Fig. 8. TLSA Schedule for constant arrival of packets with energy function ($y = 0.5x^2$)

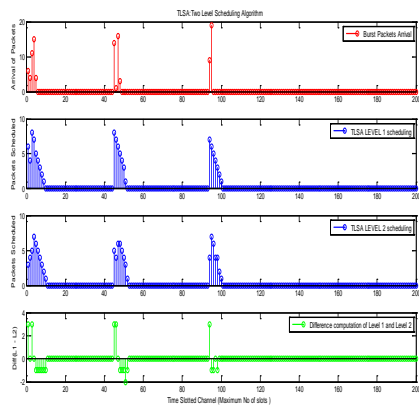


Fig. 9. TLSA Schedule for Burst arrival of packets with energy function ($y = 0.5x^2$)

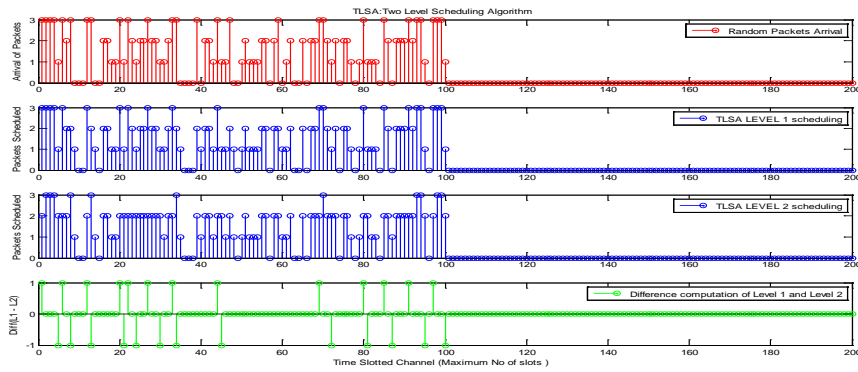


Fig. 10. TLSA Schedule for Random arrival of packets with energy function ($y = 0.5x^2$)

9. Performance Comparison

In this section, we compare the performance of TLSA with existing packet scheduling techniques such as Dynamic Multilevel Priority (DMP) scheduling and First cum First served (FCFS) scheduling. Fig. 11 shows the average waiting time (microseconds) of packets versus the number of nodes in a network. We observe that, the average waiting time required for TLSA is less when compared with DMP and FCFS scheduling schemes. Fig. 12 shows the end-to-end delay (microseconds) of packets versus the number of nodes in a network. It is observed that, TLSA can reduce the delay occurred in scheduling the packets because it utilizes two stages for scheduling. Finally, from Fig. 13 and Fig. 14, we are able to achieve a better fairness ratio and schedulability ratio using TLSA.

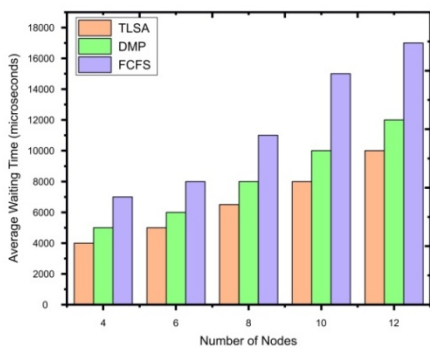


Fig. 11. Average waiting time

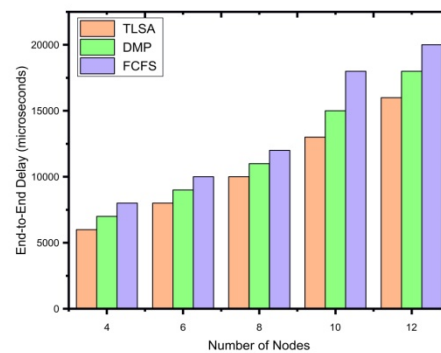


Fig. 12. end-to-end delay

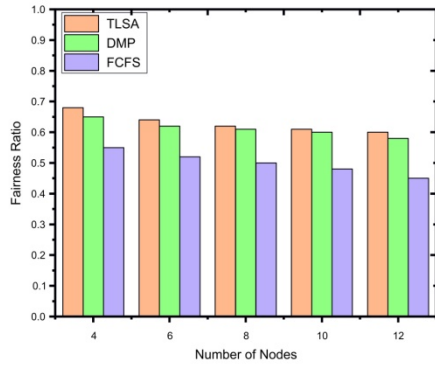


Fig. 13. Fairness Ratio

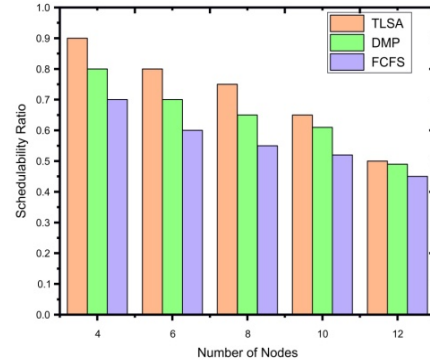


Fig. 14. Schedulability Ratio

10. Conclusion

In this paper, we have focused on scheduling of multiple packets with different arrival rate at the source node in a TSCH networks. We have developed a Two Level Scheduling Algorithm (TLSA) for all the incoming packets at the source node based on the active link chosen by the centralized scheduler. Later, we have estimated the cost of scheduling each link and analyzed the energy spent on scheduling each links using two energy functions $y = e^{ax} - 1$ and $y = 0.5x^2$. Furthermore, we have computed the average ratio, worst ratio, and competitive ratio of the two levels in a TSCH network. Finally, the TLSA performance is compared with the existing packet scheduling techniques in terms of average waiting time, end-to-end delay, fairness ratio, and schedulability ratio. The proposed model shows better performance than other packet scheduling techniques.

References

- [1] D.Kim, T.Lee, S.Kim, B.Lee, Y.Hee, "Adaptive packet scheduling in IoT Environment based on Q-learning," *Procedia Computer Science*, vol. 141, pp.247-254, 2018. [Article\(CrossRefLink\)](#)
- [2] Jing Zhang, Lin Xiang, Derrick, Minho Jo, and Min Chen, "Energy Efficiency Evaluation of Multi-Tier Cellular uplink transmission under maximum power constraint," *IEEE Transactions on Wireless Communications*, vol.16, no 11, pp.7092-7107, November 2017. [Article\(CrossRefLink\)](#)
- [3] Nidal Nasser, Lutful Karim, Tarik Taleb, "Dynamic Multilevel Priority Packet Scheduling scheme for Wireless Sensor Network," *IEEE Transactions on Wireless Communications*, vol.12, no 4, pp.1448-1459, April 2013. [Article\(CrossRefLink\)](#)
- [4] Kang, wang, Juan and ying, "On the Performance of Largest-Deficit-First for Scheduling Real-Time Traffic in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol.24, no.1, pp .72-84, February 2016. [Article\(CrossRefLink\)](#)
- [5] Darbandi, and Myung Kyun Kim, "Path Collision-aware Real-time Link Scheduling for TSCH Wireless Networks," *KSII Transactions on Internet and Information systems*, vol 13, no 9, pp.4429-4445, September 2019. [Article\(CrossRefLink\)](#)
- [6] Seohyang, Hyung, and Chongkwon, "ALICE: Autonomous Link-Based Cell Scheduling for TSCH," in *Proc. of IPSN-19, ACM*, April 2019.
- [7] Apostolos, Dimitrios, and Christos, "Collision-Free Advertisement Scheduling for IEEE 802.15.4-TSCH Networks," *Sensors*, 19(8), pp.1789, 2019. [Article\(CrossRefLink\)](#)

- [8] Sukho, Dong, Kim, and Kangseok, "Escalator: An Autonomous Scheduling Scheme for Convergecast in TSCH," *Sensors*, 18(4), 1209, April 2018. [Article\(CrossRefLink\)](#)
- [9] Yuvin, and Sang, "A Virtual slotframe technique for reliable multi-hop IEEE 802.15.4e time-slotted channel hopping network," *International Journal of Distributed Sensor Networks*, vol.14(7), pp. 1-12, June 2018. [Article\(CrossRefLink\)](#)
- [10] Alper, Sedat, "DIVA: a distributed divergecast scheduling algorithm for IEEE 802.15.4e TSCH Networks," *Wireless Networks, Springer*, September 2017. [Article\(CrossRefLink\)](#)
- [11] Mei, Zhu, Zhao, and Chen, "Scheduling for Data Transmission in Multi-Hop IEEE 802.15.4e TSCH Networks," *Mobile Network Application, Springer*, 23, 119-125, May 2017. [Article\(CrossRefLink\)](#)
- [12] Ren-Hung, Chih-Chiang, and Wu-Bin, "A Distributed Scheduling Algorithm for IEEE 802.15.4e Wireless Sensor Networks," *Computer Standards & Interfaces, Elsevier*, vol.52, pp.63-70, January 2017. [Article\(CrossRefLink\)](#)
- [13] Bo Ji, Changhee Joo and Ness B.Shroff, "Throughput-Optimal Scheduling in Multihop Wireless Networks without Per-Flow Information," *IEEE/ACM Transactions on Networking*, vol.21, no.2, pp .634 -647, April 2013. [Article\(CrossRefLink\)](#)
- [14] Juan, Srikant and Ying, "Scheduling for Optimal Rate Allocation in Ad Hoc Networks with Heterogeneous Delay Constraints," *IEEE Journal on Selected Areas in Communications*, vol.29, no.5, pp .979-987, May 2011. [Article\(CrossRefLink\)](#)
- [15] Reema, Navin kumar, Namratha and T.Srinivas, "Packet Scheduling scheme to Guarantee QOS in Internet of Things," *Wireless Personal Communication, Springer*, 100, 557-569, 2018. [Article\(CrossRefLink\)](#)
- [16] Saima Abdullah, Kun Yang, "An Energy Efficient Message Scheduling Algorithm considering node Failure in IoT Environment," *Wireless Personal Communication, Springer*, 79, 1815-1835, 2018. [Article\(CrossRefLink\)](#)
- [17] Gomathi, Mahendran, "An Efficient Data Packet Scheduling schemes in wireless sensor networks," in *Proc. of 2nd International conference on Electronics and Communication system, ICECS*, pp.542-547, 2015. [Article\(CrossRefLink\)](#)
- [18] Zhu, Qin and Lai, "Adaptive packet scheduling scheme to support Real-time Traffic in WLAN Mesh Networks," *KSII Transactions on Internet and Information systems*, vol 5, no 9, pp.1492-1512, September 2011. [Article\(CrossRefLink\)](#)
- [19] Lutful, Nidal, Tarik and Abdullah, "An Efficient Priority Packet Scheduling Algorithm for wireless sensor network," in *Proc. of IEEE ICC*, pp.334-338, 2012. [Article\(CrossRefLink\)](#)
- [20] Jing Zhang, Qingjie, Derrick, and Minh Jo, "Optimal Energy Efficiency Fairness of Nodes in Wireless Powered Communication Networks," *Sensors*, 17(9), 2125, September 2017. [Article\(CrossRefLink\)](#)
- [21] Djabir, Lyes and Hussein T. Mouftah, "Industrial IoT Data Scheduling based on Hierarchical Fog Computing: A Key for Enabling Smart Factory," *IEEE Transactions on Industrial Informatics*, vol.14, no 10, pp.4590-4602, October 2018. [Article\(CrossRefLink\)](#)
- [22] Jing Yang, Sennur Ulukus, "Optimal Packet scheduling in an Energy harvesting communication system," *IEEE Transactions on Communication system*, vol.60, no.1, pp.220-230, January 2012. [Article\(CrossRefLink\)](#)
- [23] E. Uysal-Biyikoglu, B. Prabhakar, and A. El Gamal, "Energy-Efficient packet transmission over a wireless link," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 487-499, Aug. 2002. [Article\(CrossRefLink\)](#)
- [24] W. Chen, U. Mitra, and M. Neely, "Energy-Efficient scheduling with individual delay constraints over a fading channel," in *Proc. of 2007 WiOpt*, pp. 1-10, 2007. [Article\(CrossRefLink\)](#)
- [25] El Gamal, C. Nair, B. Prabhakar, E. Uysal-Biyikoglu, and S. Zahedi, "Energy-efficient scheduling of packet transmissions over wireless networks," in *Proc. of 2002 IEEE Infocom*, vol. 3, pp. 1773-1782, 2002. [Article\(CrossRefLink\)](#)

- [26] Yanting wu, Rajgopal Kannan, Bhaskar Krishnamachari, “Efficient Scheduling for Energy-Delay Tradeoff on a Time-Slotted Channel,” *Technical Report, CENG-2015*, 2015. [Article\(CrossRefLink\)](#)
- [27] N. Ehsan and T. Javidi, “Delay optimal transmission policy in a wireless multi-access channel,” *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3745–3751, Aug. 2008. [Article\(CrossRefLink\)](#)
- [28] Murtaza Zafer, Eytan Modiano, “Delay-constrained energy efficient data transmission over a wireless fading channel,” in *Proc. of 2007 Information Theory Applications Workshop*, pp. 289–298, 2007. [Article\(CrossRefLink\)](#)
- [29] Murtaza Zafer ,Eytan Modiano, “A Calculus approach to energy-efficient data transmission with quality of service constraints,” *IEEE/ACM Transactions on Networking*, vol. 17, pp. 898–911, June 2009. [Article\(CrossRefLink\)](#)
- [30] Paulo Salvador, Antonio Pacheco, Rui Valadas, “Modeling IP traffic:Joint characterization of packet arrivals and packet sizes using BMAPS,” *Computer Networks, Elsevier*, vol 44, pp.335-352, October 2003. [Article\(CrossRefLink\)](#)
- [31] Ozel, K Tutuncuoglu, J Yang, S. Ulukus and A .Yener, “Transmission with energy harvesting nodes in fading wireless channels: optimal policies,” *IEEE Journal.sel.Areas communication*, vol 29, pp. 1732-1743, sep. 2011. [Article\(CrossRefLink\)](#)
- [32] E.Uysal-Biyikoglu,A.El Gamal,B. Prabhakar, “Adaptive transmission of variable-rate data over a fading channel for energy-efficiency,” in *Proc. of the IEEE global Telecommunications conference, GLOBECOM*, 21(1), 98-102, November 2002. [Article\(CrossRefLink\)](#)
- [33] K.Tutuncuoglu and A. Yener, “Optimum transmission policies for battery limited energy harvesting nodes,” *IEEE Transaction on wireless communication*, vol.11, no.3 pp.1180-1189, Mar 2012. [Article\(CrossRefLink\)](#)
- [34] Kirik Pruhs, “Competitive online scheduling for server systems,” *SIGMETRICS perform.Eval.Rev.*, 34(4), pp.52-58, 2007. [Article\(CrossRefLink\)](#)
- [35] N.Bansal,H.L.Chan,and K Pruhs, “Speed scaling with an arbitrary power function,” in *Proc. of SODA '09: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, 2009. [Article\(CrossRefLink\)](#)
- [36] M. Lin,Z.Liu,A.Wierman and L.L.H Andrew, “Online algorithms for geographical load balancing,” in *Proc. of IEEE IGCC*, 2012. [Article\(CrossRefLink\)](#)
- [37] Yujae Song, Ki Won Sung and Youngnam Han, “Impact of Packet Arrivals on Wi-Fi and Cellular System sharing unlicensed Spectrum,” *IEEE Transactions on Vehicular Technology*, vol.65, no.12, pp .10204 -10208, December 2016. [Article\(CrossRefLink\)](#)
- [38] Glenn Daneels, Bart, Steven, and Jeroen, “ReSF: Recurrent Low-Latency Scheduling in IEEE 802.15.4e TSCH networks,” *Ad Hoc Networks, Elsevier*, vol 69, pp.100-114, November 2017. [Article\(CrossRefLink\)](#)



Manjunath. G. Asuti is working as Assistant Professor, School of Electronics and Communication Engineering, REVA University, Bangalore, India. He has completed B. E. in Electronics and Communication Engineering and M.Tech in VLSI and Embedded systems from Visvesvaraya Technological University, Belgaum and presently pursuing Ph.D (Wireless Sensor Networks) in REVA University, Bangalore. He has 14 years of teaching experience. His areas of interest are Wireless Sensor Networks, VLSI, DSP, Embedded systems, and Communication systems.



Dr. Prabhugoud I. Basarkod, Professor, School of Electronics and Communication Engineering, REVA University. He has 32 years of teaching experience with many research publications in reputed national/international journals and conferences. Some of the works of his research are published in various international journals such as International Journal of Computer Science, Journal of Electronic and Telecommunications, Journal of Telecommunications and Information Technologies, Elsevier International Journal of Computer and Electrical Engineering, and IETE Journal of Education. His research areas of interests include Wireless Networks, Mobile Ad hoc Networks (MANETs), Agent technology, Multimedia communication and wireless sensor networks. Currently he is guiding 6 Ph.D students. He is a member of Indian society for Technical Education, Institute of Electrical and Electronics Engineers (IEEE), Member Institution of Electronics and Member of Institution of Electronics and Telecommunication Engineers (IETE).