

IEEE 802.15.4e TSCH-mode Scheduling in Wireless Communication Networks

Ines Hosni[†] and Ourida Ben boubaker^{††},

[†] Department of Information Systems

^{††} Department of Computer and Information Sciences

College of Computer and Information Sciences Jouf University, Sakaka, Saudi Arabia

Summary

IEEE 802.15.4e-TSCH is recognized as a wireless industrial sensor network standard used in IoT systems. To ensure both power savings and reliable communications, the TSCH standard uses techniques including channel hopping and bandwidth reserve. In TSCH mode, scheduling is crucial because it allows sensor nodes to select when data should be delivered or received. Because a wide range of applications may necessitate energy economy and transmission dependability, we present a distributed approach that uses a cluster tree topology to forecast scheduling requirements for the following slotframe, concentrating on the Poisson model. The proposed Optimized Minimal Scheduling Function (OMSF) is interested in the details of the scheduling time intervals, something that was not supported by the Minimal Scheduling Function (MSF) proposed by the 6TSCH group. Our contribution helps to deduce the number of cells needed in the following slotframe by reducing the number of negotiation operations between the pairs of nodes in each cluster to settle on a schedule. As a result, the cluster tree network's error rate, traffic load, latency, and queue size have all decreased.

Keywords:

Optimized scheduling function; IEEE 802.15.4e TSCH; cluster tree; scheduling.

1. Introduction

With the adoption of new technologies, the Industry 4.0 is appeared, employing IoT, Cyber-Physical Systems (CPS) and Cloud technologies [1]. Although there are multiple IoT architectures, which improve communication efficiency, the requirements and the urgency demand greater efficiency, speed and reliability, known as Quality of Service (QoS). To address this problem, Wireless sensor Networks (WSNs) have emerged and become one of the most important network infrastructures. These stringent quality of service requirements of communication protocols have traditionally been addressed by making modifications to the IEEE 802.15.4 standard and proposing external mechanisms. The main challenge of industrial WSNs is to minimize the latency time given the critical nature of the physical events detected by the sensor nodes. In other words, in order to evaluate their bandwidth needs, the challenge is

to decrease the rate of control packets exchanged between nodes. The TSCH mode is planned to permit sensor nodes to endure a wide range of applications, including industrial applications, by implementing the 802.15.4 protocol [2]. This mode consists of a technique for accessing the communication medium using time synchronization between nodes in order to attain an operational level with low power consumption. On the other hand, TSCH implements frequency hopping to allow the network to achieve an advanced level of reliability. The 802.15.4e amendment is the latest standard proposed by IEEE for low power WSNs. This standard is implemented in an industrial environment with high requirements in terms of reliability, availability and security. In this environment, deploying sensors in parallel with metallic equipment results in signal degradation due to interference, thus blocking the use of a single channel for communication. However, TSCH mode, implemented in the 802.15.4e standard, allows greater agility when using communication resources by providing greater reliability to the network. TSCH mode only focuses on MAC layer operation. As a result, the 6TiSCH Working Group was formed by the IETF Standardization Organization to link the IEEE802.15.4e TSCH-mode capabilities to previous standardization efforts and recommendations such as 6LoWPAN and ROLL (Routing Over Low-power and Lossy networks) [3]. The 6TiSCH group offers an architecture based on open-source standards with the objective of achieving high performance at industrial level with regard to latency, reliability and power consumption. However, the group does not define how to plan and schedule the sending and receiving of frames between nodes in the network and during the time slots that define the scheduling [4]. This work aims to develop a scheduling algorithm able to meet the needs of industrial WSNs in terms of latency and energy. In other words, it aims to reduce the amount of control traffic exchanged between nodes to determine their requirements of bandwidth. The proposed solution is based on a distributed algorithm that allows two nodes in a cluster to calculate the number of cells required to exchange their packets. Depending on the mode of operation of the network, the chain and sequence of events was modeled as a Poisson process. The algorithm is based on a probability calculation

that aims at predicting in a later stage the number of cells needed for a pair of nodes in cluster topology. This will minimize the control packets number exchanged and thus reduce the latency time. Also, clustering can reduce the energy consumption used for data collection. So, there will be a proposal for an energy-efficient data collection model focused on clustering and prediction. The remainder of this paper is structured as follows. We present an overview of 6TiSCH via TSCH 802.15.4e mode in Section II and III. In the two following sections, we discuss some context details and related work. Sections VI and VII explain the nature of the system. In section VIII, we present the results of the simulation, and conclude the article in section IX.

2. IEEE802.15.4e Standard

IEEE 802.15.4 standard presents the benchmark for wireless sensor networks. This describes the service of low-speed local area wireless networks (LR-WPAN) and identifies the physical layer and the layer of media access control [5]. These networks were very established, though retaining a simple and versatile protocol stack, for their ease of implementation, extremely low cost, short-range operation, reliable data transfer and fair battery life [6]. The efficiency of the IEEE 802.15.4 standard network in the WSNs has been analyzed by many studies. However, many drawbacks and flaws have been found, making this standard inadequate for sensitive applications. These, typically operating in harsh environments, have strict criteria in terms of reliability, latency, power efficiency, and scalability. Lack of reliability, infinite latency, a built-in frequency hopping technique, and poor power management are among these limitations [7]. These flaws render the IEEE 802.15.4 standard inappropriate for several systems, particularly where the reliability and latency of these systems are very demanding. In 2012, as an update to the IEEE 802.15.4 standard, the IEEE Standards Association Council accepted the IEEE 802.15.4e [8] standard in order to better serve the numerous industrial fields of implementation. Additional features such as low power consumption, information elements, improved tags, MAC performance metrics and quick pairing are supported by the new standard.

3. Scheduling in 802.15.4 TSCH Networks

As stated in the literature, the scheduling issue has indeed increased concern on TDMA networks [9]. Despite this, most current multi-channel scheduling schemes for TSCH networks are not appropriate. They have not been developed for nodes with limited bandwidth, do not enable packet channel hopping, and are not effective for channel use. New scheduling schemes developed for TSCH networks were

also developed by the researchers. There are different approaches, which could be used to establish the scheduling. The IEEE 802.15.4-2015 standard in TSCH mode allows the upper layers to create delays that all nodes must respect. This allows nodes to communicate with each other in multihop, thus making the information flow from the initial point to the collection point easier and faster. A node implements global and local scheduling by allocating cells to each respective flow to aid in the sharing of cells containing information between clients.

4. Related Works

The design of a schedule is unique to the application and many scheduling schemes for planning TSCH networks have been identified. In order to determine the schedule, creative approaches may be used. It is possible to identify them as centralized and distributed.

In a centralized approach, a single coordinator node is responsible for planning and building all communications, as well as maintaining network scheduling. A centralized machine called the Path Computation Element may be the scheduler (PCE) [10]. The Traffic Aware Scheduling Algorithm (TASA) is a centralized scheduling algorithm for IEEE802.15.4e TSCH-mode networks proposed in [11][7]. This approach considers the topology of a tree structure and reflects on a convergecast scenario where the coordinating node must be supplied with various amounts of data. TASA's primary objective is to establish the best schedule, reducing the amount of slots required to send all information to the coordinator. This scheduling may be achieved through the process of matching and coloring. The TASA method applies the matching algorithm at every phase to pick a collection of suitable schedule links in the same time slot. Then, for each connection selected in the previous step, a vertex coloring algorithm applies the various channel offsets. In addition, the authors found that the use of more channels would increase network efficiency, reduce latency and improve energy efficiency considerably. The authors of [12] planned the MODESA (Multi-channel Optimized Delay Time Slot Assignment) method. Unlike TASA, MODESA aims coherent traffic conditions where the same number of packets are generated by all nodes. Conflict-free scheduling in TASA is constructed using an iterative method. TASA picks a set of links at each step and arranges their transmissions in the same timeslot, using several channel offsets, if required [7]. The MODESA method selects a single node and chooses a single channel to support one of its mandatory transmissions. Additionally, by first scheduling nodes that provide more packets in their queues, MODESA decreases queue congestion, while TASA does not take queue congestion into consideration. This method was enhanced and built in [13], in order to maintain diverse traffic, as well as several coordinators. In

[14] an adaptive, centralized and multi-hop (AMUS) scheduling method based on the TSCH mode is proposed. At the PCE unit situated network, the authors introduced their approach and used a flexible application layer protocol (the CoAP protocol) to gather the data needed to measure the schedule. The AMUS approach enables a multihop planning sequence (MSS) to offer low latency and distributes extra resources to susceptible connections in order to significantly minimize the delay created by conflict or collisions. This approach exceeds TASA in improving contact efficiency and also achieving exceptionally low delay. Unlike centralized scheduling, distributed solutions appear to be more stable in the face of change, without creating a priori assumptions about the topology of the network or the amount of traffic to be transmitted. When implementing a distributed approach, each node needs to negotiate with neighboring nodes and defines locally which links to plan with them. Decentralized Traffic Aware Scheduling DeTAS is the distributed version of TASA method [15]. This method is intended for networks of multi-coordinators. Therefore, to build the overall schedule, it utilizes mixed micro-scheduling. All micro scheduling uses a collection of specific channels to prevent interference. TASA has been compared to the DeTAS approach and the results obtained indicate that the former offers better management of queues. Besides, DeTAS guarantees high duty cycle, end-to-end delay and packet loss ratio efficiency (Packet Loss Ratio) [16]. The DiSCA solution (Distributed Scheduling for Converge cast in Multi-channel Wireless Sensor Networks) recognizes two types of transmission: without an acknowledgment of receipt and with an acknowledgment of receipt [17]. On each iteration in this algorithm, a node schedules a transmission following a set of rules. Each iteration provides micro-scheduling. This algorithm is likely to interlock in order to minimize the total slot number. The authors compared DiSCA to [12] and to [18] and their results obtained indicate that DiSCA is very similar to optimal scheduling with a limited number of control messages.

5. Proposed system

5.1 Requirements

The 6top (6p) protocol enables a 6TiSCH network's neighboring nodes to add or delete cells from their schedules. It is part of the 6TiSCH IEEE802.15.4e sublayer of operations, which provides frameworks in this type of network for performing distributed orchestration. It is the scheduling feature that determines when cells should be added or removed, so 6p is used to efficiently assign resources.

The 6p protocol performs a so-called 6p transaction when new cells need to be added or removed, which includes the negotiation of adding or removing cells between a pair of nodes. The nodes hold their own orchestrations in the case of networks with distributed scheduling. In terms of signaling duration between nodes, this guarantees better performance. Less signaling also suggests, however, that nodes are less knowledgeable about the network, making it more difficult to generate efficient scheduling.

5.2 Design constraints

During the design phase of the scheduling algorithm, we defined some operating constraints, which are essential when running our solution. Each node keeps changing the quantity of resources assigned to its neighbor nodes depending on both its current allocation of resources and its own resource needs. The minimal scheduling function does not take into consideration the recurring traffic load, which means that each allocated cell reiterates at each slot frame and subsequently wastes assets in the event that the generation of packets in the network is not as frequent. In addition, during the phase of determining the bandwidth required for communication between a pair of nodes, the number of exchange messages must be controlled. This will reduce the waiting time during end-to-end transmission and also minimize the packets number in the memory queue.

5.3 Network model

In TSCH mode networks, defined by 802.15.4e, when a node senses a sudden fluctuation of a physical event, it creates and queues a large flow of data packets in the memory of the sensor node. This node checks to see if it has enough bandwidth to transmit these packets to its parent node. Otherwise, the node checks, in the next slotframe, if it has enough reserved cells with its parent and compares if it is able to forward those packets during those cells. Checking the bandwidth with the parent node triggers a high number of transactions per packet, thus generating a considerable increase in the use of energy resources. When the limited number of transmissions and retransmissions is exceeded, there will be a loss of packets which results in the depletion of the node's power resources. We suggest a new scheduling algorithm based on the minimal scheduling feature provided in the IETF draft [19] and applied on cluster topology to minimize communication failures, the number of packet losses and end-to-end latency. The developed algorithm consists of two principal processes: calculating the average number of packets each node produces and predicting the number of cells needed in the next slotframe. We present these descriptions:

6.2 Poisson process generation model

In our model, we consider TSCH-mode 802.15.4e sensor networks, which transmit packets only when they detect a change or deviation from a physical event taking into consideration the nature of the way packets is sent and received in the network. We may follow a Poisson process model to formulate data packet generation in a cluster tree network.

6.3 Mathematical formulation

First, we represent by $E(t)$ the number of events that happen in an interval of time $(0, t]$, and we undertake that $E(0) = 0$. The process $E(t); t \geq 0$ is the counting process and it verifies the following conditions: [20]

$\forall t \geq 0, E(t) \in \mathbb{N} \rightarrow E(t)$ is increasing $\forall 0 < a < b, E(b) - E(a)$ denotes the number of events that have taken place in time interval $]a, b]$. In our model, the counting process is considered to be incrementally independent because the events (transmission / reception of data packets) that occur in a disjoint time interval (the duration of a slotframe) are independent. In addition, we adopt the following conditions:

- Condition 1: Events that repeat in disjoint time intervals are independent; a couple of nodes begins the exchange of packets during slotframes if and only if an event is detected. Therefore, for any choice of real numbers $0 \leq t_1 < t_2 < \dots < t_n$, the random variables $E(t_2) - E(t_1), E(t_3) - E(t_2), \dots, E(t_n) - E(t_{n-1})$ are mutually independent.

- Condition 2: A pair of nodes transmits packets regardless of their state during the previous slotframe. We can then conclude that for any positive real number t and h , the number $E(t+h) - E(t)$ of events which occur during a time period $(t, t+h]$ is independent of the value t and depends only on the interval length h .

- Condition 3: In the 802.15.4e TSCH standard mode, the maximum length of a slotframe must not exceed 101 time slots with a duration of 15 milliseconds each. This means that the length h of the time interval over which we count the events is reduced, so we can deduce that the probability of observing more than one event is almost zero. $g(h)$ here is a small order function of h .

$$\Pr[E(t+h) - E(t) \geq 2] = g(h) \quad (6)$$

$$\Pr[E(t+h) - E(t) \geq 1] = \lambda h + g(h) \quad (7)$$

The allocation of the packets number, $E(t)$, designed by each network node along the slotframe length is calculated as follows:

$$\Pr(E(t) = n) = \frac{\lambda t^n}{n!} = e^{-\lambda t} \quad (8)$$

where λ represents the average value of the number of packets generated and transmitted between a pair of nodes in cluster since they have been synchronized to the network scheduling. Once the nodes are synchronized, the 802.15.4e standard implements the minimal scheduling function which ensures minimal network operation. Mathematically, λ is defined as follows:

$$\lambda(n) = \frac{\sum_{i=1}^T \text{pkt}_i(n)}{S_T - S_i} \quad (9)$$

where T represents the current time, $\text{pkt}_i(n)$ is the generated packets number by node n at time $T = i$, S_T represent the slotframe number at T and S_i represent the slotframe number at $T=i$. Let $\lambda = S_i - S_T$ is the total of numbers preceding past slotframes. In order to have a precise value of γ , the 802.15.4e standard implements the scheduling function until to slotframe number 10.

6.4 Add / Remove cells (6top)

The 6TiSCH Operation (6top) sublayer is the layer immediately above the IEEE Std 802.15.4 TSCH communication media access control layer. The roles of the 6top underlay are as follows:

- Allows neighbor nodes to communicate to add / remove cells from each other.
- Execute one or more 6top scheduling functions, which precise the measures determining when to add / delete cells. Once a node joins a 6TiSCH network, it will be able to add, delete or move cells with its preferred parent node for the following three reasons:
 - Adapt the connection layer's resources to the traffic between the node and its chosen parent.
 - Manage the change of preferred parent (triggered by the RPL protocol).
 - Manage a collision in the scheduling.

We only focus on where there will be a change in the scheduling to accommodate network traffic. The proposed algorithm uses the minimal scheduling function in the event of a collision or a change of parent. Our solution provides statistics on bandwidth usage at the 6top sublayer so that you can make decisions about adding or removing cells. From slotframe $\gamma = 10$, each node (parent or child) executes the algorithm described in the next paragraph. The aim is to estimate the packets number that will be created in the next slotframe.

In order to reduce the resources use made by every node, the algorithm terminates its execution when achieving a maximum probability of packets producing λ . By identifying the packets number that will be produced in the next slotframe, a node may estimate the number of cells required to share data with its FFD. A node may also do a

6p transaction with its parent to update (increase/decrease) cells to the TSCH schedule between the two nodes, depending on the performance of the algorithm. (Fig. 3).

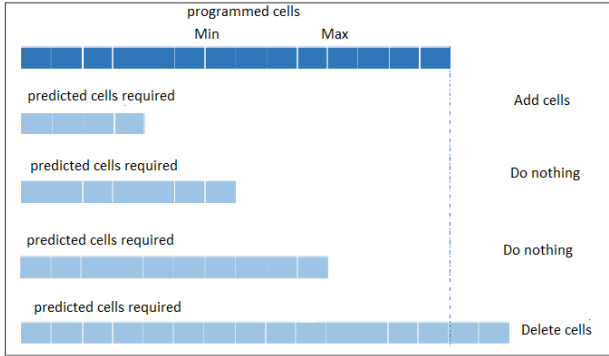


Fig. 3: Add/ delete cells

6.5 Scheduling Algorithm

In the algorithm below, each node will perform certain calculations based on statistics collected with its parent in order to predict how many cells it will need during the next slotframe. This prediction replaces the method used in the minimal scheduling function which is based on an exchange of control packets to calculate the bandwidth required for communication between a pair of nodes. As mentioned previously, this prediction takes place at a single node level and will therefore avoid a packet exchange.

Algorithm 1 Prediction algorithm applied on Slotframe S

Require: : cluster node n, $G(n)$: the number of packets sent by node n

Ensure:

```

for each cluster do
  for S=S11 to simulation time; Si++ do
    Calculate λ from Equation(9)
    λ ← n
    while prmax do
      pr ← Calculate(PrE(t), λ) : packet generation probability
      if (pr ≥ prmax) then
        Maxi ← pr
      end if
    end while
    return pr : Maximum probability
    if (pr < Pr(Cells(n))) : (Cells(n) the present allocated cells between the parent FFD(n) and the node RFD(n)) then
      6p_REMOVE(Cells)
    else
      if pr > Pr(Cells(n)) then
        6p_ADD(Cells)
      end if
    end if
  end for
end for

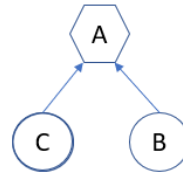
```

6.5.1 Calculating the mean

In order to have a precise value of the average generated packets between a pair of nodes, the algorithm uses Eq. 9 from the slotframe $S = \gamma = 11$. This manipulation is repeated at each start of a slotframe until the node becomes desynchronized from the network.

6.5.2 Predicting the number of packets

Take as a simple example, a cluster containing 3 nodes. After synchronization between nodes, Node B (RFD) is considered to have Node A as its preferred parent (FFD). Node A begins to calculate the probability of having 1 single packet based on the generated packets average number between the pair of nodes during the last 10 slotframes until a maximum probability value is found (Fig. 4). With the objective of saving computational resources and based on the probability mass function of a Poisson process, the algorithm stops the computation when it reaches this maximum probability.



$$Pr(E(10) = 1) = \frac{(\lambda(\gamma)T)^1}{1!} e^{-\lambda(\gamma)T}$$

$$Pr(E(10) = 2) = \frac{(\lambda(\gamma)T)^2}{2!} e^{-\lambda(\gamma)T}$$

$$Pr(E(10) = 3) = \frac{(\lambda(\gamma)T)^3}{3!} e^{-\lambda(\gamma)T}$$

$$\vdots$$

$$Pr(E(10) = i) = \frac{(\lambda(\gamma)T)^i}{i!} e^{-\lambda(\gamma)T} = Maxi$$

Fig. 4: Poisson prediction model

6.5.3 Add/delete cells

Once the prediction of the next number of packets, which will be generated between the child node and its preferred parent in a cluster, is calculated and based on the number of cells already allocated, a node can add or remove or even keep the same number of allocated cells. These results are sent to the 6top sublayer which will be responsible for sending the following requests.

- 6p-addCell(Max): If the number of cells already allocated for the pair of nodes is lower than the prediction calculated by the algorithm, the request takes as a parameter the number of cells (Max) that will be added during the next slotframe.
- 6p-removeCell(Max): If the number of cells already allocated for the pair of nodes is greater than the prediction calculated by the algorithm, the request takes as a parameter the number of cells (Max) that will be deleted during the next slotframe.

- Do not send anything if Max is equal to the number of cells already allocated

7. Performance Evaluation

We run our simulations on OpenWSN. It is an open source simulator for WSNs, which supports the IoT based protocol stack with support for IEEE802.15.4 TSCH, 6LoWPAN, RPL and CoAP. To evaluate our proposed solution, we have developed several simulation scenarios under different conditions. We have simulated a number of nodes that varies between 2 and 100 using a cluster tree topology. The packet delivery rate between each pair of nodes (PDRs) has been set at 100%. Each node generates a random number of packets during each slotframe. In the 802.15.4 physical layer configuration, we have considered that all communication channels are available and have the same physical characteristics.

7.1 Error rate

With each simulation, we increased the number of nodes by 10 distributed equitably between the different clusters. At each increase, we nested the responses from the 6p transactions, which contain an 8-bit sub-register containing an error code (defined in the 6tp protocol). We have taken into account all types of errors that a 6p transaction can return. Then, we calculated the average of these error transactions according to the number of nodes.

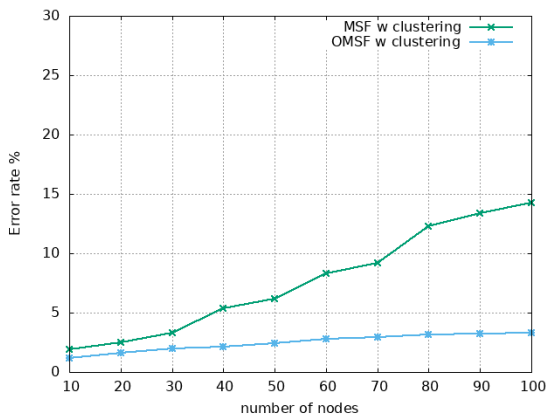


Fig. 5: 6p Error rate

From the fig. 5, we show the negotiation error rate increases with increasing network density, in case we implemented MSF, the error rate increases dramatically from 1.9% up to 14.3%, whereas by implementing OMSF the rate increases from 1.2% up to 3.3%. The proposed mechanism, OMSF, largely outperforms MSF and maintains a negotiation error rate of less than 3.3% for all network densities. This is attributable to the replacement of the mechanism for

measuring bandwidth, necessary for communication between a pair of nodes by the prediction calculation performed independently in each node at each cluster and which is implemented in OMSF. Otherwise, OMSF helps reduce the number of control packets exchanged over the network, allowing the node to send specific transactions of adding or removing cells.

7.2 Energy savings

A limited network duty cycle ratio (DCR) can be given by the OMSF algorithm, defined as the proportion of timeslots in which a node is operating. We observe that a very low duty cycle ratio (Fig. 6) is reached, often below 2%, even with 100 nodes. Clearly, when there is more traffic on the network for forwarding, the ratio of active time slots increases. This is related to the role of the distributed method to complete local data processing and local prediction. Within any cluster, the cluster head gathers data. Then, on data distribution, the cluster head should finish the local prediction. Predictions must be executed by the members of each cluster, and predicted data must then be transmitted to the head of the cluster. Therefore, each cluster head has a true vision on all sensor data, via the cluster. Thus, the energy consumption significantly decreased.

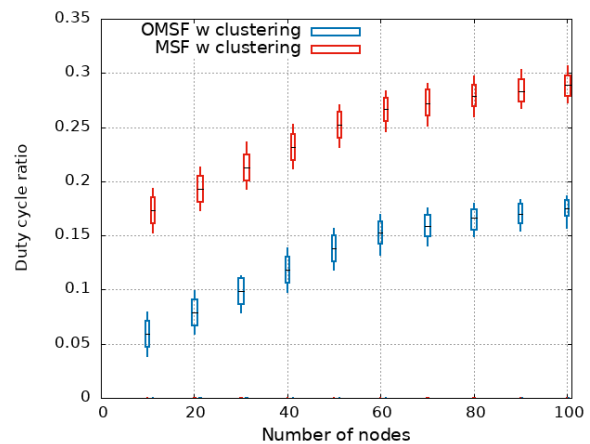


Fig. 6: 6p Error rate

7.3 Traffic load

In the following simulation, we calculated the number of control and measure packets that a pair of nodes transmits in order to decide the number of required cells in the next slotframe. We have gradually increased the number of nodes in the network from 10 to 100 deployed in a cluster tree topology, in intervals of 10. The MSF protocol implements the bandwidth estimation algorithm to translate the needs of the nodes into a number of cells. The latter monitors the amount of data sent between each pair of

nodes. When this quantity becomes large or small (reaches a determined threshold) compared to the number of cells already allocated, MSF asks 6top to add or delete cells with the designated node. This process generates the transmission of certain control packets, which leads to an additional traffic load.

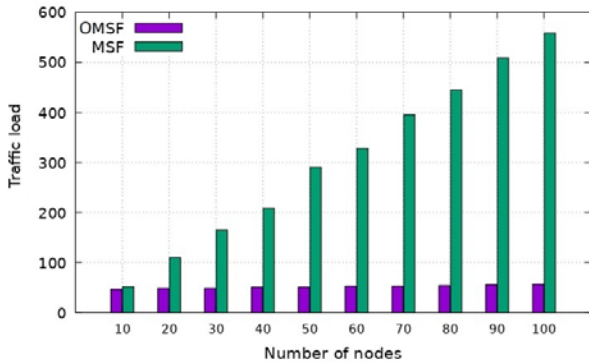


Fig. 7: Traffic load

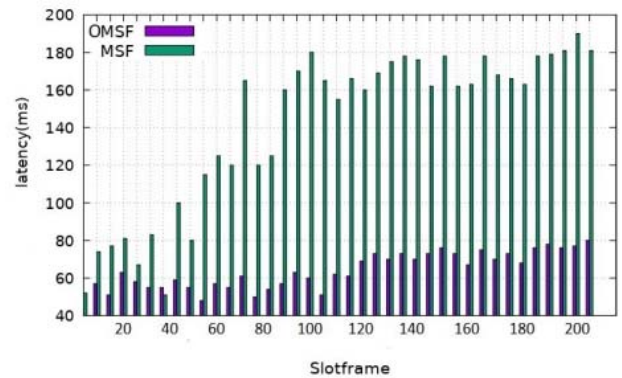
Fig. 7, displays the overhead load of traffic (measured in bytes) used by nodes to share network information. We note that the amount of messages exchanged increases with the number of nodes deployed. This is attributed to the negotiation phases made between nodes to decide which 6p transaction to deploy in scheduling. We also find that the OMSF protocol retains a nearly constant number of transmitted packets. This is because of predicting the number of cells needed during Fig. 7: Traffic load the next slotframe for each pair of nodes at each cluster. It eliminates the transmission of transactions exchanged between a pair of nodes in the MSF protocol to determine the necessary bandwidth. The OMSF protocol prevents overloads from being sent and maintains a constant average over the network life of control packets.

7.4 Latency

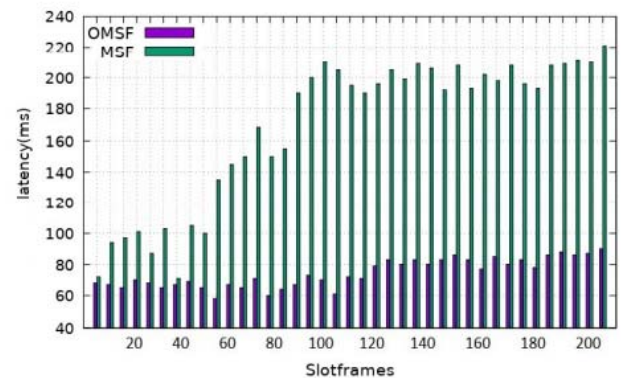
In order to study the end-to-end latency of our OMSF proposal, we simulated a network consisting of 50 nodes distributed over 5 clusters. Each node in the network generates, during the first slotframes, a constant data traffic equal to 2 packets for each slotframe. Then a transient data flow varying from 2 to up to 8 packets per slotframe is created by each node in each cluster. We simulated the same network under the same conditions by implementing the MSF protocol in order to trust OMSF in terms of latency.

Fig. 8a presents a comparison between MSF and OMSF in terms of latency, where the source node is chosen from a level 1 cluster which is the nearest to the DAGroot. Fig. 8b presents also the same comparison, but the source is chosen from a level 4 cluster. We calculated the latency as follows: each packet is time stamped from the time it was generated at the source node, until it hits the DAGroot node. When a packet reaches 4 retransmission attempts and the node's

queue memory becomes full, the packet will be significantly discarded. However, if a packet is transmitted on several occasions, an increase in the latency time may occur. Each attempt to retransmit a packet generates an increase in latency, since each packet takes longer than its allotted time to reach its destination. The latency was almost constant during the first slotframes due to stable data traffic flow that has been generated by the nodes of the network. Thereafter, the latency varies between cycles due to failed exchanges. In both figures, we note that the OMSF protocol maintains an end-to-end latency of less than that achieved by the protocol MSF. This is due to the reduction in negotiation errors as well as collisions, which the scheduling algorithm of the OMSF protocol guarantees based on the prediction calculation. We also note that for the two functions the latency is less when the source belongs to the cluster closest to the dagroot. This is due to greater variability in the neighborhoods of the clusters, especially at the lowest tree levels. OMSF protocol maintains an end-to-end latency of less than 80 milliseconds during all slotframes in case if level 1 cluster and an end-to-end latency of less than 90 milliseconds in level 4 cluster. So, it always presents better results even by changing the level of the cluster. Therefore, the added traffic load is also reduced, causing in a high PDR value when irregular data streams are generated.



(a) level 1 cluster

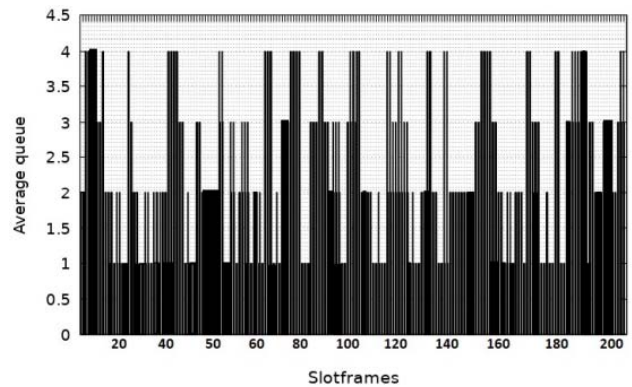


(b) level 4 cluster

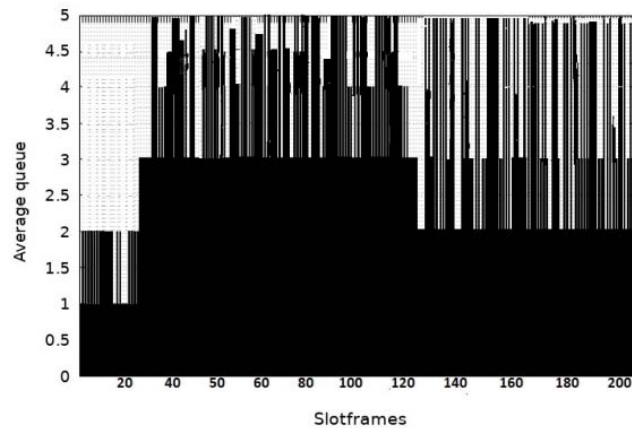
Fig. 8: Latency

7.5 Queue size

The traffic flow in a TSCH 802.15.4e network varies depending on the location of the cluster, to which the node belongs, in the DODAG. Nodes belonging to a cluster near to the root have a higher traffic load than nodes in end clusters. Following the detection of a sudden event by a group of nodes, a high traffic load will be transported in the network. When this traffic reaches nodes in cluster near the root, there will be an accumulation of packets in their memory queues. As a result, each of these nodes will need multiple cells in a slotframe to dump packets from its queue. This will cause unwanted delays, negatively affecting the cumulative communication delay. When the client requires to provide vital data such as alarms, this delay is extremely inappropriate. In order to study the impact of the prediction algorithm proposed in the OMSF protocol on the size of the memory queue, we simulated a network of 100 nodes distributed over 5 clusters and we compared the result obtained by implementing the protocol MSF. The memory size of each node can hold a maximum of 5 packets in the queue. Once this memory is full, the first arrived packet in memory will be discarded first. During each slotframe, we determined the queued packets average number of all nodes in the network. Fig. 9 shows the queued packets average number in the network. In this figure, it seems to be that, in the case of MSF protocol Fig.9b in some slotframes, the queues in the memories of the nodes are almost complete and reach their limits, contributing to the missing packets. The OMSF protocol exhibits better performance Fig.9a by keeping a virtually moderate queue, showing an average of 0 to 4 packets in the queue during all slotframes. In particular, the nodes which are located in clusters near to the DAGroot node, over-allocate a number of cells greater than those which are located in clusters distant to it. The high packet flow through these nodes allows the prediction algorithm, proposed in the OMSF protocol, to reserve more cells. Therefore, this ensures sufficient bandwidth, which meets the fluctuating needs of the pairs of nodes.



(a) OMSF



(b) MSF

Fig. 9: Queue size

8. Conclusion

In this work we have formulated the packet transmission mode in TSCH network based on clustering and prediction model. A probability calculation is made between each parent node and its child using poisson model, we were able to predict the number of packets that will be exchanged between this pair of nodes. This prediction allowed us to deduce the number of cells needed in the next slotframe. As a result, the traffic load circulating in the network, used to determine the scheduling, has been reduced considerably. In addition, data clustering and prediction methods can reduce the energy consumption of sensor nodes for data collection.

In a future perspective, we intend to integrate into our model a cell selection method, which gives priority to the nodes closest to the root node. This will provide more bandwidth to the nodes that have more traffic, thus the latency will be further reduced.

References

- [1] H. Kurunathan, R. Severino, A. Koubaa, "IEEE 802.15.4e in a Nutshell: Survey and Performance Evaluation", IEEE Communications Surveys and Tutorials, pp (99):1-1, 2018, doi: 10.1109/COMST.2018.2800898.
- [2] G. Z. Papadopoulos, T. Matsui, P. Thubert, G. Texier, T. Watteyne, N. Montavont, "Leapfrog collaboration: toward determinism and predictability in industrial- IoT applications", in 2017 Proc. Int. Conf. Commun, ICC, Paris, pp. 16, Jul. 2017.
- [3] P. Verma, Pawan Kumar, R. Verma, A. Prakash, A. Agrawal, K. Naik, R. Tripathi, M. Alsabaan, T. Khalifa, T. Abdelkader, A. Abogharaf, "Machine-to-Machine (M2M) communications: A survey", Journal of Network and Computer Applications, (66), 83-105, (2016), doi.org/10.1016/j.jnca.2016.02.016.
- [4] M. Palattella, T. Watteyne, Q. Wang, K. Muraoka, N. Accettura, D. Dujovne, L. Grieco, T. Engel, "On-the-fly bandwidth reservation for 6TiSCH wireless industrial networks", IEEE Sensors J., 16(2), 550-560, 2016, doi: 10.1109/JSEN.2015.2480886.
- [5] IEEE. (2006). IEEE Standard for Information technology–Local and metropolitan area networks– Specific requirements– Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs).
- [6] M. Mohamadi, B. Djamaa, M. Reda Senouci, "Industrial internet of things over IEEE 802.15.4 TSCH networks: design and challenges", International Journal of Internet Technology and Secured Transactions, 10, 61-80, 2018, doi:10.1504/IJITST.2020.10026294.
- [7] M. Mohamadi, M. Reda Senouci, "Chapter 2 Scheduling Algorithms for IEEE 802.15.4 TSCH Networks: A Survey", Springer Science and Business Media LLC, 2019.
- [8] IEEE802.15.4e. (2012). IEEE Standard for Local and Metropolitan Area Networks.Part15.4: Low-Rate Wireless Personal Area Networks Amendment 1: MACSublayer.
- [9] S. C. Ergen, P. Varaiya, "TDMA scheduling algorithms for wireless sensor networks". Wireless Networks, 16(4), 985-997, 2010.
- [10] Farrel,Vasseur, A, "A Path Computation Element (PCE)-Based Architecture (Rapport no 4655)", Network Working Group, 2006.
- [11] M. R. Palattella, N. Accettura, M. Dohler, L. A. Grieco, G. Boggia, "Traffic Aware Scheduling Algorithm for reliable low-power multihop IEEE 802.15.4e networks", IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC), 327-332, 2012, doi: 10.1109/PIMRC.2012.6362805
- [12] R. Soua,P. Minet, E. Livolant, "MODESA:An optimized multichannel slot assignment for raw data convergecast in wireless sensor networks", IEEE 31st International Performance Computing and Communications Conference (IPCCC), 91-100, 2012, doi: 10.1109/IPCCC.2012.6407742.
- [13] R. Soua, E. Livolant, P. Minet, "MUSIKA: A multichannel multisink data gathering algorithm in wireless sensor networks", International Wireless Communications and Mobile Computing Conference (IWCMC), 1370-1375, 2013, doi: 10.1109/IWCMC.2013.6583756.
- [14] Y. Jin, P. Kulkarni, J. Wilcox, M. Sooriyabandara, "A centralized scheduling algorithm for IEEE802.15.4e TSCH based industrial low power wireless networks", IEEE Wireless Communications and Networking Conference, 1-6,2016.
- [15] N. Accettura, M. R. Palattella, G. Boggia, L. A. Grieco, M. Dohler, "Decentralized Traffic Aware Scheduling for multi-hop Low power Lossy Networks in the Internet of Things" IEEE 14th International Symposiumon "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), 1-6, 2013, doi: 10.1109/WoWMoM.2013.6583485.
- [16] N. Accettura, E. Vogli, M. R. Palattella, L. A. Grieco, G. Boggia, M. Dohler, "Decentralized Traffic Aware Scheduling in 6TiSCH Networks: Design and Experimental Evaluation", IEEE Internet of Things Journal, 2(6), 455-470, 2015, doi: 10.1109/JIOT.2015.2476915.
- [17] R. Soua, P. Minet, E. Livolant, "DiSCA: A distributed scheduling for convergecast in multichannel wireless sensor networks", IFIP/IEEE International Symposium on Integrated Network Management (IM), 156-164, 2015, doi: 10.1109/INM.2015.7140288.
- [18] R. Soua, P. Minet, E. Livolant, "Wave: a distributed scheduling algorithm for convergecast in IEEE 802.15.4e TSCH networks", Transactions on Emerging Telecommunications Technologies, 27(4), 557- 575, 2016, doi.org/10.1002/ett.2991.
- [19] T. Chang, M. Vucinic, X. Vilajosana, S. Duquenooy, D. Dujovn, "6TiSCH Minimal Scheduling Function (MSF)" (Rapport no draft-ietf-6tisch-msf-01), 2018, Internet EngineeringTaskForce, <https://datatracker.ietf.org/doc/html/draft-ietf-6tisch-msf-01>.
- [20] W. Paul, J. Baschnagel,Stochastic processes, springer, 2013



Ines Hosni Ph. D in communication systems. received her engineering degree in January 2012 from the National Institute of Applied Sciences and Technologies (INSAT) of tunisia and his master's degree in october 2012 from the polytechnic school of tunis (EPT). She had her doctorate in communication systems from the national school of engineers of tunis (ENIT). Currently, she is an assistant professor at Al Jouf University Arabie Saudite. Her research interests are communication networks, wireless networks.



Ourida Ben Boubaker Saidi Ph. D in Computer Science from the Paul Sabatier University of Toulouse, France 1986. Ex-head of the Department of Computer Science in the ISG of Tunis, University of Tunis, a member of SOIE research Laboratory, and Coordinator of the Department of Computer Sciences, College of Science and Arts, Tabarjal, Jouf University, Kingdom of Saudi Arabia for four years. Actually Assistant Professor at College of Computer and Information sciences, Jouf University, sakaka, Kingdom of Saudi Arabia, Department of Computer Sciences. Her research interests are Artificial Intelligence and networks.