Hybrid Duty Cycle Algorithm for Industrial WSNs Using Machine Learning

Charbel Nicolas[1], Abdel-Mehsen Ahmad[2], Jose Rubio Hernan[3], Gilbert Habib[1]

[1] LaRRIS, Faculty of Sciences, Lebanese University, Fanar, Lebanon
[2] School of Engineering, Lebanese International University, Lebanon
[3] Electronics and Physics,TélécomSudParis. InstitutPolithechnique de Paris, Evry, France
{Charbel.Nicolas, Gilbert.Habib}@ul.edu.lb, abdelmehsen.ahmad@liu.edu.lb, rubio_he@telecom-sudparis.eu.

Abstract. Wireless Sensor Networks (WSNs) are used to monitor physical or environmental conditions. Due to energy and bandwidth constraints, wireless sensors are prone to packet loss during communication. To overcome the physical constraints of WSNs, there is an extensive renewed interest in applying data-driven machine learning methods. In this paper, we present a mission-critical surveillance system model for industrial environments. In our proposed system, a decision tree algorithm is installed on a centralized server to predict the wireless channel quality of the wireless sensors. Based on the machine-learning algorithm directives, wireless sensor nodes can proactively adapt their duty cycle to mobility, interference and hidden terminal. Extensive simulation results validate our proposed system. The prediction algorithm shows a classification accuracy exceeding 73%, which allows the duty cycle adaptation algorithm to significantly minimize the delay and energy cost compared to using pure TDMA or CSMA/CA protocols.

Keywords: Machine learning, J48 decision tree, prediction, PRR, LQI, TDMA, CSMA/CA.

1 Introduction

Fully automated factories use various types of systems to monitor the different components and parts of the production chain. Specifically, WSN (Wireless Sensor Network) has many advantages in monitoring the mobile parts that are difficult to monitor using fixed sensors. However, due to its limited resources, WSN node's energy depletion and hardware malfunctions can lead to node failures to form a robust network. Furthermore, these malfunctions may cause the node to fail in conveying the information and metrics about the production systems to the management server. Moreover, the industrial environment can also affect the wireless communication channels, leading to network reliability problems [6]. Usually in such environment, the traffic load flowing through the WSN can be characterized as minimal, only conveying periodically measured metrics.

There are different layers of protocols that manage the wireless module in the WSN nodes. In this paper, we focus on the data link layer. In this latter, different types of

protocols can be used, mainly opportunistic protocols like CSMA/CA, non-opportunistic protocols like TDMA, and hybrid protocols that can get the best of both worlds. In our previous work [4], we adapted the link layer dynamically between TDMA and CSMA/CA protocol based on the topology change without looking at the channel quality. In [9], we proposed a cognitive radio that analyzes the channel quality and adapts the MAC layer accordingly. The weakness of this method was the delay needed for adaptation. These solutions can be effective in specific environments. In the case of the industrial environment, the main issue is to minimize delays and avoid packet loss to minimize energy consumption. Traditionally, to reduce the energy consumption in WSN, duty cycling is used for scheduling by the link layer [11, 12]. The use of duty cycling is very efficient as the node goes to sleep when not active. Although duty cycling minimizes the energy cost, it does not minimize delays or manage the channel quality. To determine the channel quality, many parameters and metrics have been considered. The most used to characterize communication link stability and reliability is the packet reception ratio (PRR) [13]. By definition, the PRR is the number of packet received divided by the number of packet transmitted. In [3], authors utilize the PRR as the probability of successfully receiving a packet between two neighbor nodes. If the PRR is high means that the link quality is high and vice versa. PRR model in [3] shows a direct link with transmission power, receiver sensitivity, and distance between nodes. The transmission power and receiver sensitivity are evaluated using the Link Quality Indicator (LQI) [14]. The LQI represents the magnitude of the error between ideal constellations and a number of symbols immediately following the sync word in a transmitted packet. Moreover, the LQI presents a good correlation with PRR, however, it needs to be averaged over many packets (about 120 packets) [8].

Recently, various techniques form the machine-learning (ML) research area have lightened the interest of the wireless communication community specifically to improve the cognitive radio in wireless communication [15] and more precisely to improve the link layer in WSN [5, 16]. The survey done in [16] shows that most of the existing protocols use ML for reactive decisions, not proactive decisions, and only proactive algorithms have been used to adapt the routing algorithm at the routing layer. In [5], the proposed frame work classifies and chooses statically between applying TDMA or CSMA/CA.

In this paper, we propose an algorithm that predicts the channel quality and dynamically adapts the link layer using a hybrid TDMA-CSMA-sleep protocol. The proposed proactive adaption for the link layer uses the predicted PRR to have the time to adapt the physical layer (RF) to a hybrid TDMA-CSMA/CA mechanism, i.e., the time slot selection and organization.

The remainder of this paper is structured as follows. Section 2 positions our work in the current literature. In section 3, we describe the part of the system managed by the Machine Learning-based Hybrid Duty Cycle Algorithm (ML-HDCA). Section 4 presents the topology of the network and the simulation results. And section 5 concludes and discusses future work.

2 Related Work

The authors in [5] proposed a model to use ML in decision-making at the link layer to choose between TDMA and CSMA/CA. They do not propose dynamic protocol adaption or channel quality prediction to adapt preemptively to traffic throughput change. In [17], a hybrid TDMA and CSMA protocol is designed. It tackles the fire monitoring issue in buildings and consequently, the autonomous switching from energy-efficient normal monitoring to emergency monitoring to cope with heavy traffic. However, this paper does not address the issue of critical information loss due to poor channel quality. We assume, due to the nature of the industrial environment, that every aspect of the environment is usually monitored and kept by a centralized server. These logged data are usually analyzed by the network administrators to correct anomalies or upgrade the system. In this work, we will use these data as the dataset to create a reliable dynamically adaptable WSN. Moreover, we assume that the traffic load in the industrial WSN is minimal, due to the nature of information exchanged, which eliminates the need to manage the situations where we have high amount of traffic in the network.

To the best of our knowledge, our contribution is the first to introduce ML into the industrial environment to predict the state of the communication channel, and proactively protect the WSN critical information from being lost. We introduce the use of time as parameter to predict the channel quality. We argue that the particularity of monitoring the industrial topology (cf. Fig. 1), which is characterized by specific mobility patterns, the periodical mobility patterns will create a predictable state of the communication channels. Moreover, by using the time factor as a feature for the prediction, we show that the proactive adaptation of the protocols is better than only using ML in reactive protocol selection.

In the literature [16, 17], some studies used the traffic load to model or adapt the link layer mechanisms to increase the throughput. In this paper, we propose a mechanism to predict the PRR and adapt proactively the link layer scheme accordingly to prevent packet loss due to interference, hidden terminal and mobility. To predict the PRR, we require different characteristics and criteria, like the mobility state of a node, the link quality indicator (LQI) as there are a correlation between LQI and PRR [7], and the historical PRR values logged by the system periodically with their timestamps.

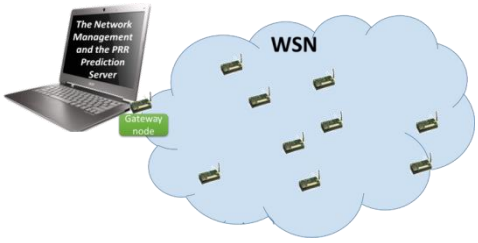

Fig. 1. A representation of the network management server, the gateway and the WSN.

3 System Description

Our system (cf. Fig. 1), called ML-HDCA, is composed of two parts. The first one is a centralized server containing the network management and the PRR prediction software. The second part is the hybrid duty cycle algorithm based on TDMA-CSMA/CA mechanisms. The sink node (gateway node) manages the newly arriving nodes by signaling and sending control messages from the server to the nodes. Usually, the server intervenes in two states of the nodes duty cycle: The initialization and the radio mechanism correction. During the initialization state, the new node arrives, and communicates to the server its mobility speed (fixed, low, high), the specific time where the predicted PRR is needed, and the LQI. The server, based on its ML algorithm (cf. Section 3.1), predicts the quality of the communication channel by calculating the predicted or estimated PRR. Then, the server replies to the sensor node by sending the predicted PRR for the specific requested time of day. Finally, the sensor adapts its data link layer accordingly. To classify the PRR, we use a terminology that is similar to the one used in [2]. The used PRR classes are *Bad* (<70%), *Good-* (70-80%), *Good+* (80-90%), and *Excellent* (>90%).

In the radio mechanism correction state, calculated PRR is sent periodically to the server and logged with the other data (cf. Table 1). Each hour, the node's radio adapts its configuration, either based on the predicted or the calculated PRR. In this paper, the scenario used focuses on only using the predicted PRR to update the radio configuration, and the calculated PRR to update the logging database on the ML server. Each sensor node selects the appropriate mode to communicate with its neighbors, depending on its neighborhood mode of configuration and its own configuration (cf. section 3.2).

Table 1. Logged data (dataset) by the ML server.

Time of day	Mobility speed (High, Low, Fixed)	PRR classes (Bad, Gd-. Gd+, Excellent)	QoS of a TS (Excellent, Good, Bad) - LQI	PRR %
0:00	Fixed	Bad	Good	56,698 %
1:00	Fixed	Excellent	Bad	0.8668 %
…	…	…	…	…
23:00	Fixed	Excellent	Excellent	100 %

3.1 The Network Management and the PRR Prediction Server

The server is composed of two parts. The first one is the software part that is composed of: a) The Network management server using SCADA (Supervisory Control and Data Acquisition) technology in the system, or TinyOs platform with a server side application that can be used for data collection form the WSN, depending on the technology used. b) The PRR prediction server (based on Weka [1] platform containing

the data set and the prediction model). The second part is the hardware part that is composed of the computer server and the sink node (the gateway).

To build the classification model, we study and develop the relationship between the decision-making criterion (PRR) and the WSN features from the existing studies [3, 7]. Based on these studies, we conclude that, the most influential features that affect the classification criterion are the time, the LQI, the mobility status and the previous PRR values. All these features are logged on the server and form the dataset (cf. Table 1) used by the classification platform.

To generate the dataset, we create our own simulator to simulate the wireless sensor network and the radio behavior of each node. The data is transmitted in the header of each packet to the sink node then to the server. The simulator topology is further described in section 4. We have trained and tested our classifier using the data generated by the simulator.

In order to validate the proposed model, we use two groups of datasets, one for training and another for evaluation. Afterwards, we import them into Weka for processing. Weka [1] is a java based platform that is composed of a collection of machine learning algorithms for data mining tasks; it processes the data set and generates the classification model. The classification model is validated by the second dataset. Then, Weka platform feeds the different features necessary for predicting the PRR from the requesting node. Finally, Weka generates the appropriate predicted PRR class to be used and the server sends it to the WSN node to adapt its physical layer. To do so, the channel quality forecast using the predicted PRR is used in different states of the WSN nodes: the initialization state and the steady state. During the WSN nodes state changes, the nodes update the information on the server by sending the LQI, the mobility state, the time of day, the existing PRR and the node ID to the server, and receive from the server the predicted PRR to use.

To choose the most suitable ML algorithm, we compared three classification algorithms: J48 decision tree, Sequential minimal optimization (SMO)-Support vector machine (SVM), and Random forest using our data set (cf. Table 1). To choose the best of the three classification algorithms we used the same data set of 5997 instances to test all the algorithms. The results showed that J48 decision tree is the best suited for our application (cf. Fig.2-4). It has the best delay to update the model (0.05 seconds), and the best time to apply classification (0.03 seconds), and a correct classification similar to the other algorithms with a percentage of 73.7%.

The resulting model and the confusion matrix of the Machine learning classification algorithm J48 decision tree that we chose to use by the server are shown in Fig. 5 and Table 2 respectively.

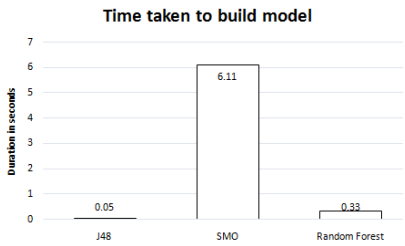

Fig. 2. Time taken to build the classification model.

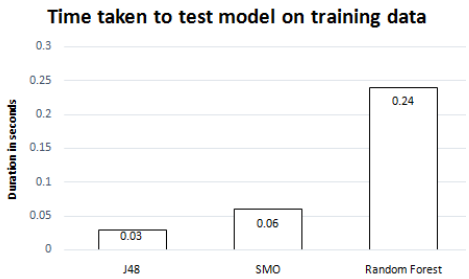

Fig. 3. Time taken to test the model on training data.

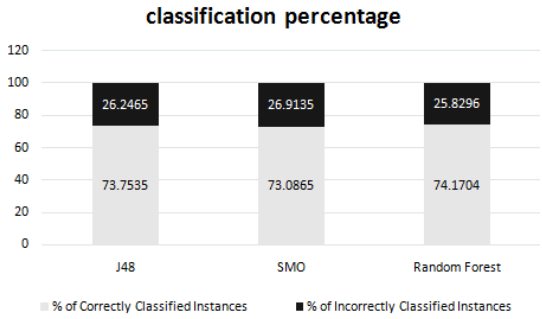

Fig. 4. Percentage of correctly classified instances.

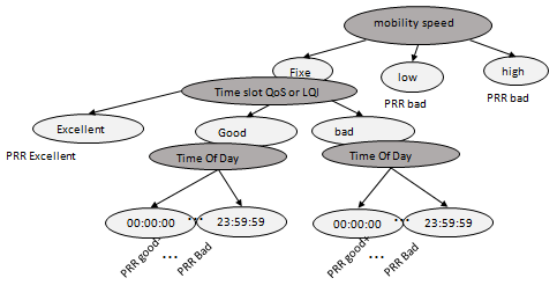

Fig. 5. The J48 decision tree model.

Table 2. Confusion Matrix of the J48 classification model.

Good+	Good-	Bad	Excellent	Classifiers
21	63	145	165	Good+
0	150	249	429	Good-
0	110	3038	392	Bad
0	0	0	1235	Excellent

3.2 Hybrid TDMA-CSMA/CA Duty Cycle Algorithm Based on the PRR classes

The standard CSMA/CA protocol is the initial state of the link layer mechanism for every new arriving node or any node that does not have any neighbor in TDMA mode. By receiving the predicted PRR from the server, as described earlier, the link layer of a sensor adapts to a specific configuration schema. Then, to calculate the actual PRR, the sensor node starts counting the number of correctly received packets from a neighbor. At the same time, this neighbor integrates in its packet header the total number of sent packets. This is done for all its neighbors. These numbers are used to calculate the real PRR value of the links, and then the PRR values for each node are averaged and sent to the server. This operation is done during the complete lifetime of a sensor and sent periodically through the header of the data packets to the server.

The node calculates the PRR and sends it to the server using this formula:

$PRR = \frac{\sum_{i=1}^{N} \frac{x_i}{X_i}}{N}$, where N is the total number of neighbors, X_i is the total number of packet sent by the neighbor i, $x_i$is the number of packets correctly received from the i^{th}neighbor.

The server predicts the PRR using the model shown in Fig. 6 and sends it to the node. The sensor nodes only use the predicted PRR to proactively adapt their duty cycle as the link layer adaptation takes time.

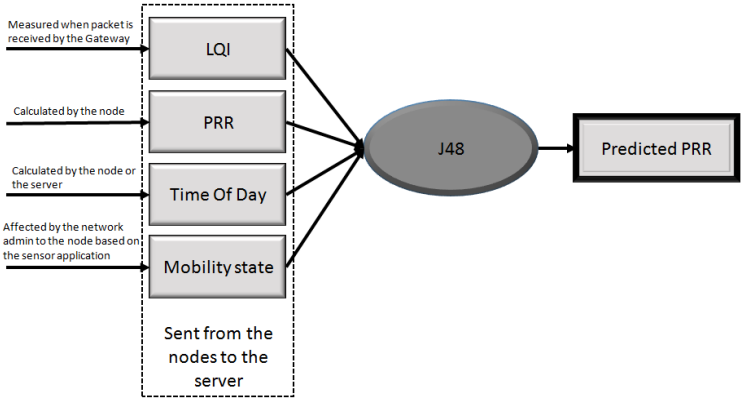

Fig. 6. The proposed PRR prediction model

3.3 Hybrid TDMA-CSMA/CA Algorithm and the Duty Cycle

To secure the links from packet loss and to create a temporal redundancy, we propose the Hybrid TDMA-CSMA/CA Algorithm where the TDMA slots are reinforced in case of low LQI by using CSMA/CA depending on the predicted PRR class. To optimize the lifetime of a WSN node, the sensor, if it is not active, enters a sleep mode [4]. This mode obliges the sensor's physical layer to turn off if not needed. If the physical layer was not well managed, the outcome can be a rapid reduction of the lifetime of a WSN

node. Moreover, since it is necessary to transfer information to the server, the node should compromise between waking up to send data and sleeping to conserve energy, and at the same time ensure that the sensor information is sent correctly to the server.

To solve this problem, we define a periodical duty cycle (cf. Fig. 7) that is composed of three time intervals: the TDMA cycle, the CSMA/CA cycle, and the sleep cycle.

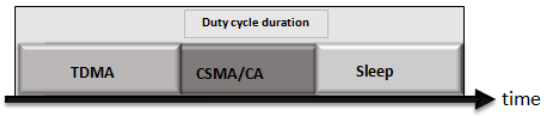

Fig. 7. WSN node duty cycle.

The *TDMA cycle* is the part where the nodes can communication synchronously without hidden terminal packet collisions and without concurrency. Each node can send its packet during its reserved time slot, and listen to all the neighbors during their transmission time slots. The TDMA weaknesses can be summarized as follows:

1- In the case of coexistence with Wi-Fi technology, the sensor radio is submitted to extensive interference, which can cause a high number of packet loss. This is because all these technologies (Wi-Fi, ZigBee, Bluetooth, etc.) share the same free frequency band without any coordination among them.
2- TDMA needs to synchronize the neighboring nodes among themselves, and at the same time, it needs to synchronize their clock rates due to the clock drift problem.
3- Mobility causes topology variation which causes nodes desynchronization.

These weaknesses are solved by using *CSMA/CA cycle*, because this protocol does not need any requirement similar to the TDMA protocol. However, CSMA/CA has its own weaknesses:

1- It consumes a lot of energy since each node needs to listen to the channel before sending packets, and during this time, all the neighboring nodes should stay awake to listen to their neighbors, without knowing when they are going to transmit.
2- In the case of a hidden terminal, two nodes can transmit at the same time causing collision and packet loss to their neighbors.

The *sleep cycle* is the time interval where a sensor turns off most of its components, keeping active only the necessary modules to collect the monitored data like temperature, pressure, speed, luminosity, etc. This is to conserve energy and to expand the life span of the node.

Finally, for ensuring that there are always some nodes awake for an emergency, our hybrid algorithm selects in rotation a random number of nodes to stay awake during all the CSMA/CA cycle. These nodes are interchanged at the end of each cycle to prevent the depletion of their batteries.

3.4 Hybrid TDMA-CSMA/CA Duty Cycle Algorithm

We map for each PRR class (as described in Section 3.1) a specific action at the link layer. Algorithm 1 summarizes the proposed model shown in Fig. 8.

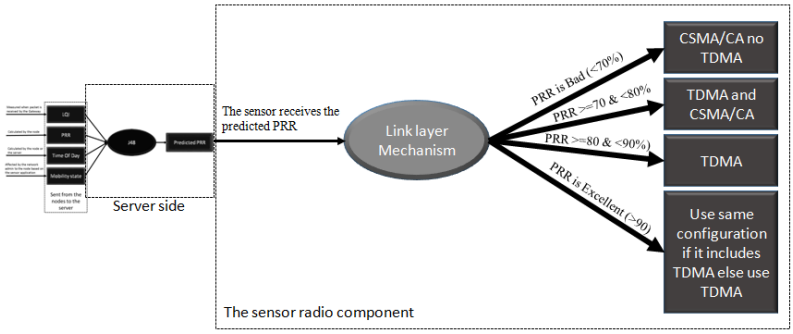

Fig. 8. For each PRR, the radio proactively adapts its configuration

Algorithm I: Hybrid TDMA-CSMA/CA Duty Cycle Algorithm

```
radio_initalisation()
while (True)
if (PRR = Bad)
  CSMA-CA()
  Sleep()
  Sleep()
else if (PRR = Good(-))
  TDMA()
  CSMA-CA()
  Sleep()
else if (PRR = Good(+))
  TDMA()
  Sleep()
  Sleep()
else if (PRR=Excellent)
  if (TDMA_not_used())
       TDMA()
   Sleep()
   Sleep()
  End if
End if
if (random_selection())
  if (CSMA-CA_Not_Used())
     CSMA-CA()
  End if
End if
End while
```

The corresponding functions are described as follows.

radio_initalisation(): This function initializes the radio CSMA/CA, and sets the node ID, the time of day, the mobility state for a node, etc.

Sleep(): By setting the parameter for this function, the node goes into a deep sleep for a specific duration based on the parameter value.

TDMA(): This function puts the node in TDMA mode for a specific duration, where the node receives its time slot dedicated for transmission and the time slots reserved to listen to the neighbors. The synchronization can be executed using two methods. The first method uses a distributed mechanism and is coordinated by the nodes themselves, where each node during the CSMA/CA cycle exchanges its time slots. In the second method, the synchronization can be centralized and coordinated by the server.

CSMA-CA(): This function applies the CSMA/CA protocol for a specific duration.

random_selection(): This function tests the variable of activation sent by the server in the packets header. The server knows the distribution of the nodes in the network and therefore knows the neighborhood of each node. For each cycle, it randomly selects a node from the neighborhood that was not selected previously to stay awake, and uses CSMA/CA by setting the variable of activation. The random waking up of specific nodes is to ensure that there are always some nodes awake for an emergency transmission. Moreover, the selected node activates a periodical beacon so that it can be detected.

TDMA_not_used(): to test if TDMA is used in the duty cycle.

CSMA-CA_Not_Used(): to test if CSMA/CA is used in the duty cycle.

4 Simulation Configuration and Results

To test our system and configure the simulator, we analyze the structure of different industrial and manufacturing facilities. The simplified standard topology is shown in Fig. 9.

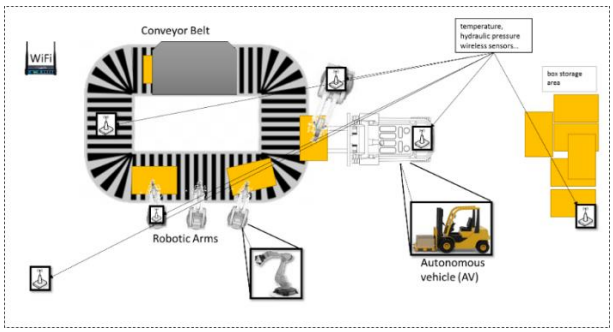

Fig. 9. Top view of the factory.

We classify three different mobility speeds for the WSN nodes that we categorize as fixed, low speed, high speed. For instance, sensors can be fixed in strategic places to

monitor the room temperature in different parts of the facility. Mobile sensors with relatively low speeds are attached to the conveyor belt to monitor its pressure and temperature. Moreover, they are attached to the robotic arms (cf. Fig. 11) to monitor the hydraulics, etc. The high mobility sensor is attached to the forklift truck to monitor the speed and load of the fully automated forklift. In addition, each palette can contain an integrated sensor characterized as highly mobile to monitor the temperature. The palette is assembled in the cargo bay area for shipping. This part is monitored by fixed sensors. The facility can be divided into four zones shown in Fig. 10. The scenario considered for the simulation is as follows:

- **Zone A:** The zone where WSN node are under interference by the Wi-Fi technology. This interference exists as long as the workers, supervisors and administrators are using Wi-Fi network. In the simulation, we modeled the Wi-Fi packet arrivals following the Poisson model and the Wi-Fi transmissions as active only between 8h00 and 17h00.
- **Zone B**: The zone contains a mix between fixed nodes and low speed mobile nodes with periodic movement. This situation creates a favorable circumstance for the machine-learning algorithm, as there are a limited number of mobility states that can be studied and adapted to.
- **Zone C**: The zone where the nodes have the highest mobility speed as it can contain a number of mobile forklifts, which can affect the LQI due to distance variations, and can disconnect the nodes from the network.
- **Zone D**: The fixed nodes zone where nodes remain stationary as long as they stay in the facility. This region is not affected by interference.

The factory is fully automated and the production process is continuously on, day and night, unless there is a request for maintenance. Moreover, it is worth noting that we use the Poisson Point Process to distribute each node in each zone. At each hour, the state of the WSN nodes is collected to verify the following features: the mobility, the LQI, the PRR and the Predicted PRR. Later the data is fed to the ML platform as described earlier. Noting that the *duty cycle* duration is equal to the sum of the CSMA/CA duration, TDMA duration and sleep duration. The sleep cycle is when most of the node's components go off to preserve energy (at this point the radio module is turned off). Moreover, the duration to transmit one packet is 0.5s which is equal to the length of one slot in TDMA, and the energy cost to send or receive one packet is considered one energy unit.

The physical layer of each node using CSMA/CA mode stays active for all the CSMA/CA defined cycle. In the simulation, the CSMA/CA duration is set to 10s. As the radio is always on, the energy consumption is constant and equal to the average energy for the duration of the cycle.

The cycle duration in seconds for TDMA is related to the number of neighboring nodes: *TDMA Cycle = (nb_of_neighbor nodes + 1)*Time_Slot_Length.*

To calculate the energy cost for TDMA:

E_{TDMA} *= (nb_of_ neighbor_ nodes + 1)*energy cost per packet.*

In our hybrid system, a part of the energy and delay is affected by the random selection of some nodes to stay awake in CSMA/CA mode.

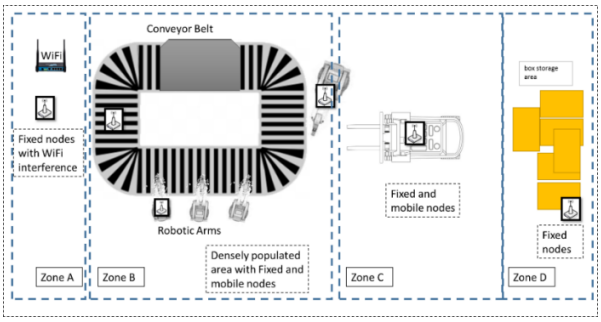

Fig. 10. Factory divided by type of mobility for the wireless sensor nodes

The energy consumption for our proposed system (cf. Fig. 12) evolves linearly but slower than the pure TDMA protocol. This is relative to the number of neighbors that affects the PRR (quality of the link) and due to the dynamic selection of the configuration. When the number of neighbors becomes higher than eight, our mechanism clearly becomes better in energy consumption then CSMA/CA and TDMA. Whereas, the average energy consumed when using CSMA/CA is constant and equal to 20 units of energy. This is because the physical layer is constantly on, listening to the channel or transmitting.

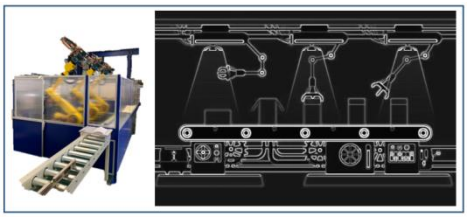

Fig. 11. On the left, a real view of the mobile robotic arms on top of a conveyor belt; on the right, a schematic view of the robotic arms.

The delay shown in Fig. 13 represents the average waiting time where a packet is queued before being transmitted. As our system has different mode of transmission, we calculated the average maximum for the worst-case scenario and the average minimum waiting duration in the best-case scenario.

For our proposed system, we conclude from Fig. 13 that the average maximum wait for the worst-case scenario (cf. Fig. 13, cross mark) is approximately 10s (the sleep duration) added to the average waiting time in TDMA mode in respect to the time slot position. This due to the dominant combination of TDMA-CSMA/CA-Sleep duty cycle. This represents in average the half of the waiting duration for the static pure TDMA (cf. Fig. 13, diamond shape). This is equal to all the duty cycle duration (10s sleep+10s sleep + duration of the TDMA cycle). Moreover, as the CSMA/CA protocol is part of the mechanism's duty cycle, a node can send its packet whenever it needs during CSMA/CA cycle unless there was a back off from transmission.

The average minimum waiting time (cf. Fig. 13, triangle shape) is estimated as being the average of the minimum between the need to transmit and the available transmission time slot (in case of TDMA), and the listening before transmission (in case of CSMA/CA). It is worth noting that, in the case of using the static CSMA/CA protocol, the node will need to sleep at least for two cycles (20s) before entering the CSMA/CA cycle for transmitting. Then, it is necessary to take into account the back offs and the listening time. For this reason, the delay is approximately constant near 25s as shown in Fig. 13.

Fig. 14 represents the time needed for a new node (added to the WSN) to calculate its PRR based on the number of neighbors. To accurately calculate the PRR (with a confidence interval of 10%), a node needs at least 100 packet [10].

To estimate the PRR (in hours), we calculate the delay as $D = N * (100 * B)/3600$, where N is the number of neighbors, and B is the duration of one duty cycle (10 + 10 + TDMA cycle) with a time slot length of 0,5s.

In contrast, the time needed to get the predicted PRR for initialization from the server will not exceed two sleep cycles independently from the number of neighbors.

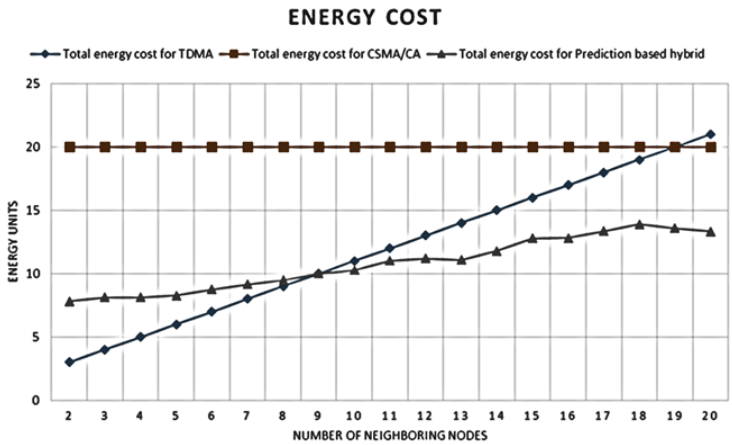

Fig. 12. The average energy consumed by a node during one duty cycle relatively to the number of neighbors

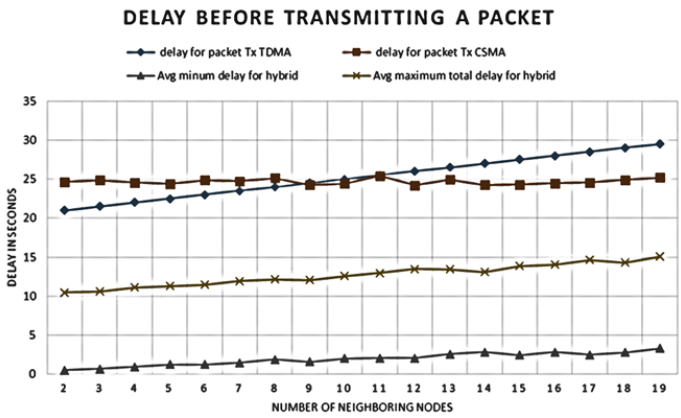

Fig. 13. The average delay needed to send one packet by a node relatively the number of neighbors

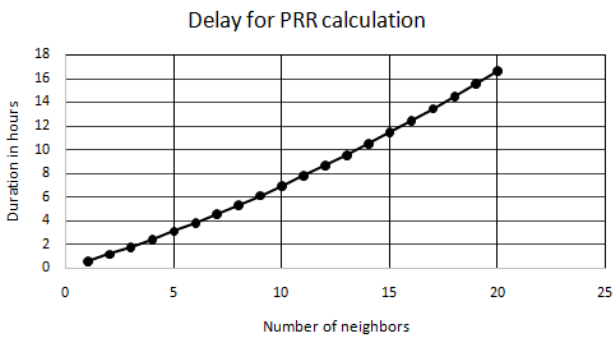

Fig. 14. Delay for PRR calculation

5 Conclusion and Future Work

In this paper, we propose a mission-critical surveillance system for industrial environments. The system assured time diversity for critical packet delivery with minimum delays. It is composed of two parts, a machine-learning server installed on centralized network management server, and a Hybrid TDMA-CSMA/CA Duty Cycle Algorithm installed on a WSN node. The system uses the PRR predictions to predict the channel quality and by that, minimizing the energy cost and delays of transmissions, and assure packet delivery to destination. Extensive simulation results validate our proposed system. The prediction algorithm of the system showed a classification accuracy exceeding 73%. Also, the radio adaption algorithm minimizes the energy cost, for a node of 20 neighbors, for one cycle by 37% relatively to TDMA. That is possible thanks to its distribution of the energy cost on multiple cycles and the avoidance of packet loss due to interference by the Wi-Fi technology. Moreover, it minimizes the energy cost compared to CSMA/CA by 35%. Furthermore, we show that calculating

the average PRR using the classical method along with slow link layer adaptation, is a time consuming task, and It results in critical transmission failures.

As future work, the system will be implemented on a real platform. Moreover, the PRR prediction will be used in proactive routing algorithm to choose the most stable and critical pathways in the network. By that, we will have a complete proactive system that adapts dynamically to the predicted quality of the radio channels.

References

1. Eibe Frank, Mark A. Hall, and Ian H. Witten (2016). The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016.
2. M. A. Kazandjieva, M. Jain, K. Srinivasan and P. Levis, “PRR Is Not Enough,” Technical Report SING-08-01.
3. A. A. Ahmed, N. Fisal, “Experiment Measurements for Packet Reception Rate in Wireless Underground Sensor Networks”, IJRTE, Vol 2, No. 2, November 2009.
4. C. Nicolas, M. Marot and M. Becker, “A Self-Organization Mechanism for a Cold ChainMonitoring System” , IEEE 73rd Vehicular Technology Conference 2011.
5. M. Qiao, H. Zhao, S. Huang, L. Zhou and S. Wang, "An Intelligent MAC Protocol Selection Method based on Machine Learning in Wireless Sensor Networks", Ksii Transactions On Internet And Information Systems VOL. 12, NO. 11, 2018.
6. Y. Duan, W. Li, X. Fu, Y. Luo, L. Yang ,“A methodology for reliability of WSN based on software defined network in adaptive industrial environment”, IEEE/CAA Journal of AutomaticaSinica, Volume: 5 , Issue: 1 , 2018.
7. K Srinivasan, P Levis, “RSSI is under appreciated”, EmNets 2006.
8. A. Bildea, O. Alphand, F. Rousseau, A. Duda. “Link Quality Metrics in Large Scale Indoor Wireless Sensor Networks”.,PIMRC 2013.
9. C. Nicolas and M. Marot, "Dynamic Link adaptation Based on Coexistence-Fingerprint Detection for WSN", 2012 The 11th Annual Med-Hoc-Net, 2012.
10. A. Woo and D. Culler, "Evaluation of Efficient Link Reliability Estimators for Low-Power Wireless Networks", Technical Report UCBCSD-03-1270, U.C. Berkeley 2003.
11. Y. Zhang, C.H. Feng, I. Demirkol, W. B. Heinzelman,"Energy-Efficient Duty Cycle Assignment for Receiver-Based Convergecast in WSN", GLOBECOM 2010.
12. A. Guidara, F.Derbel, G.Fersi, S.Bdiri, M. BenJemaa, "Energy-efficient on-demand indoor localization platform based on wireless sensor networks using low power wake up receiver", Ad Hoc Networks Volume 93, October 2019.
13. R.D. Komguem, R.Stanica, M. Tchuente, F. Valois, "Ground Level Deployment of WSN: Experiments, Evaluation and Engineering Insight",Sensors 2019.
14. CC2420 Datasheet Retrieved 2019, from TI: http://www.ti.com/lit/ds/symlink/cc2420.pdf
15. T. Erpek, T. O'Shea, Y. E. Sagduyu, Y. Shi, and T. C. Clancy, "Deep Learning for Wireless Communications" Development and Analysis of Deep Learning Architectures, Springer (to appear in 2020).
16. D.Praveen Kumar, T. Amgoth, C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey",Information Fusion Vol 49, 2019.
17. L.Sitanayah, C. J. Sreenan, K. N. Brown, "A hybrid MAC protocol for emergency response wireless sensor networks", Ad Hoc Networks 2014.