

Scalability Validation of the Posting Access Method through UPPAAL-SMC Model-Checker

Bethaina Touijer¹

LRIT, Rabat IT Center, Faculty of Sciences, Mohammed V University in Rabat, Morocco

Yann Ben Maissa²

Telecommunication Systems Networks and Services Laboratory, National Institute of Posts and Telecommunications, Rabat, Morocco

Salma Mouline³

LRIT, Rabat IT Center, Faculty of Sciences, Mohammed V University in Rabat, Morocco

Abstract—The standard IEEE 802.15.6 provides a new physical layer (PHY) and medium access control sublayer (MAC) specifications that support several challenges of wireless body area networks (WBANs). The posting is the access method of the IEEE 802.15.6 MAC protocol that is used by the hub to send data to the nodes. In this paper, we use a formal method to evaluate the posting access method under the WBANs stochastic environment. Based on the statistical model checking (SMC) toolset UPPAAL-SMC, we model and evaluate the behavior of the posting access method in terms of scalability. The evaluation results showed that according to the allocated time intervals, the energy consumption, and the throughput the scalability was validated.

Keywords—WBANs; IEEE 802.15.6 MAC protocol; posting access method; UPPAAL-SMC; energy consumption; throughput

I. INTRODUCTION

Wireless body area network (WBAN) [1] [2] [3] [4] is composed of biomedical sensors nodes that can be worn on or placed in the human body to measure several physiological parameters of the human body, such as temperature and pressure. These sensors nodes must wirelessly send their data to a control and monitoring device carried on the body. This device then delivers its data via a cellular or Internet network to an emergency center or a doctor room.

The WBANs [1] [2] [3] [4] [5] are designed to support a wide range of medical applications, such as asthma, cardiovascular disease, and cancer detection. They have enormous potential to revolutionize the future of healthcare by diagnosing many deadly diseases and providing remote and real-time monitoring of patients' health status without any restrictions, which improve their quality of life and reduce their costs of hospitalization. On the other hand, they impose several challenges related to the medium access control (MAC) protocols design in terms of energy efficiency, quality of service, reliability, priority, security, and scalability.

In November 2007, the IEEE 802 created a standard called IEEE 802.15.6 for WBANs, its final version [6] is published in February 2012. The main idea behind this standard is to define two new layers that are the physical (PHY) layer and the MAC sublayer dedicated to WBANs.

The IEEE 802.15.6 standard [6] [7] organizes the nodes into one- or two-hop star topology. A single control and monitoring device controls the entire operation of each WBAN.

The WBAN must have one control and monitoring device (i.e., the hub) and a number of sensors nodes, ranging from 0 to 64. The node(s) will refer, in the rest of this paper, to the sensor(s) node(s).

Following our earlier study [8] regarding the existing MAC protocols for the WBAN, we found that the MAC protocol of the IEEE 802.15.6 standard takes into consideration all the WBAN challenges. With this MAC protocol, the hub and the nodes can employ one or more access methods to transmit their data frames. These access methods are the carrier sense multiple access with collision avoidance (CSMA/CA), the slotted aloha, the type-I polling, the type-II polling, and the posting.

The posting is the most important access method for the hub. This latter uses it to grant itself a posted allocation for initiating one or more frame transactions. A posted allocation is a downlink allocation time interval, during which the hub can service and transmit unexpected or extra management and data traffic to the node. For example, it is used in the case of network management needs, data rate variations, and channel impairments.

Due to the important role of the hub in the WBAN, as a controller and monitor device, evaluating the scalability of its posting access method is important. Scalability represents the ability of the network to continue to operate with the same performance despite the addition of other nodes [1]. Validating this property allows the validation of the posting access method and, therefore, the performance of the WBAN.

The problem here is that the WBAN is considered as a stochastic environment, where the prediction of when the physiological parameters change their values is non-deterministic. This stochastic nature makes it difficult for the prediction of who it should allocate, between the hub and the nodes, the time interval to transmit their frames. Consequently, it makes it difficult to model and evaluate the behavior of the posting access method.

In this paper, we propose to use the statistical model checking (SMC) toolset UPPAAL-SMC [9] to investigate the posting access method under the WBANs stochastic environment, as depicted in Fig. 1. The UPPAAL-SMC has the ability to provide a stochastic interpretation of the stochastic behavior of complex and real-time systems, such as WBANs, and it is based on the statistical model-checking (SMC) [10]. The model-checking [11] has been used primarily in the verification

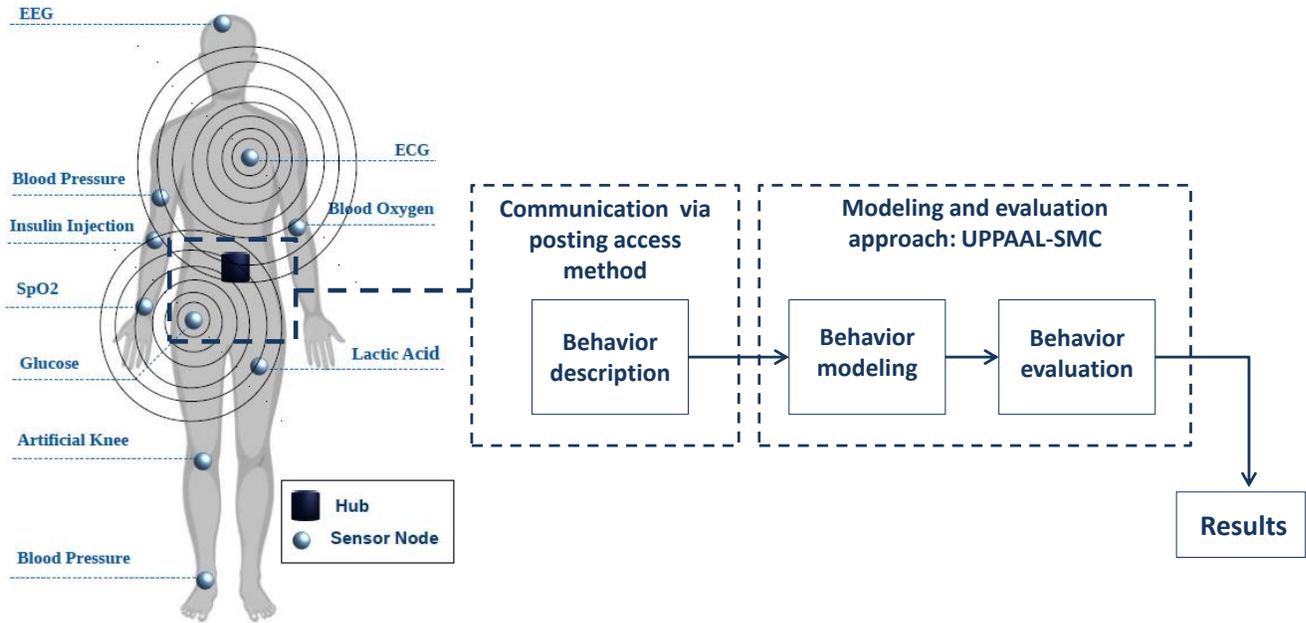


Fig. 1. The WBAN Illustration and the Used Approach for the Communication Method Modeling and Evaluation. EEG: ElectroEncephaloGraph. ECG: ElectroCardioGraph. SpO2: Blood Oxygen Saturation Level.

of the synchronization protocols, and it is considered as a useful method for the analysis and the evaluation of the communication protocols [11] [12].

Based on UPPAAL-SMC, we model and evaluate the behavior of the posting access method in terms of scalability. This is the first study of the posting access method through UPPAAL-SMC in our best knowledge. We first use the stochastic timed automata (STA) formalism provided by UPPAAL-SMC to construct a detailed model of this behavior. Then we use the metric interval temporal logic (MITL) specifications adopted by UPPAAL-SMC to evaluate the scalability of this behavior. In addition to evaluate the scalability according to the energy consumed by the hub and the throughput, which are, respectively, the dominant problem and the key performance properties that we should validate for WBAN. We should, also, evaluate the number of the posted allocation time intervals of the hub. This property shows the ability of the hub to still communicating with the nodes despite the growth of the network density.

The rest of the paper is organized as follows: the next section describes the behavior of the posting access method. Section 3 provides the posting access method behavior modeling. Section 4 presents the posting access method behavior evaluation. Section 5 presents the paper's conclusion.

II. POSTING ACCESS METHOD: BEHAVIOR DESCRIPTION

The IEEE 802.15.6 MAC protocol, as defined in [6], supports three access modes: the beacon mode with superframes, the non-beacon mode with superframes, and the non-beacon mode without superframes, as depicted in Fig. 2a, Fig. 2b, and Fig. 2c, respectively. The time axis of the latter access mode is divided into time intervals, where the node and the hub can employ one or more access methods. However,

the time axis of the other two access modes is divided into superframes of equal length. The superframe can be divided into one or more access phases, where the node and the hub can employ one or more access methods. The access methods of the IEEE 802.15.6 MAC protocol are divided into five classes that are scheduled, scheduled-polling, unscheduled, improvised, and contention, as depicted in Fig. 2a, Fig. 2b, and Fig. 2c. Accordingly, the posting is the only access method that is used by the hub to transmit its data frames in scheduled-polling, unscheduled, and improvised access methods classes. In this section, we describe the behavior of the posting access method before, within, and after the posted allocation time interval.

1) *Before the Posted Allocation Time Interval:* Based on the posting access method, to grant to itself a posted allocation time interval, the hub sends to the node a poll frame, as depicted in Fig. 3. This latter is a control frame addressed to the node to inform it about a future post. A post is a management or data frame sent by the hub to the node within a posted allocation time interval. While granting the posted allocation time interval, the hub can start sending posts after a pre-determined time.

2) *Within the posted allocation time interval:* When the posted allocation time interval starts, the hub can transmit one or more new frames and it can retransmit one or more old frames. These frames are separated by a short inter-frame spacing (*pSIFS*) time. The hub transmits the frame with a required immediate or block acknowledgment frame and with the more data (*M*) and the last frame (*L*) fields of the MAC frame header. The values of these fields can be:

- 1) Case 1, with $M = 0$ and $L = 0$: it means that no frames are waiting for transmission. Other than

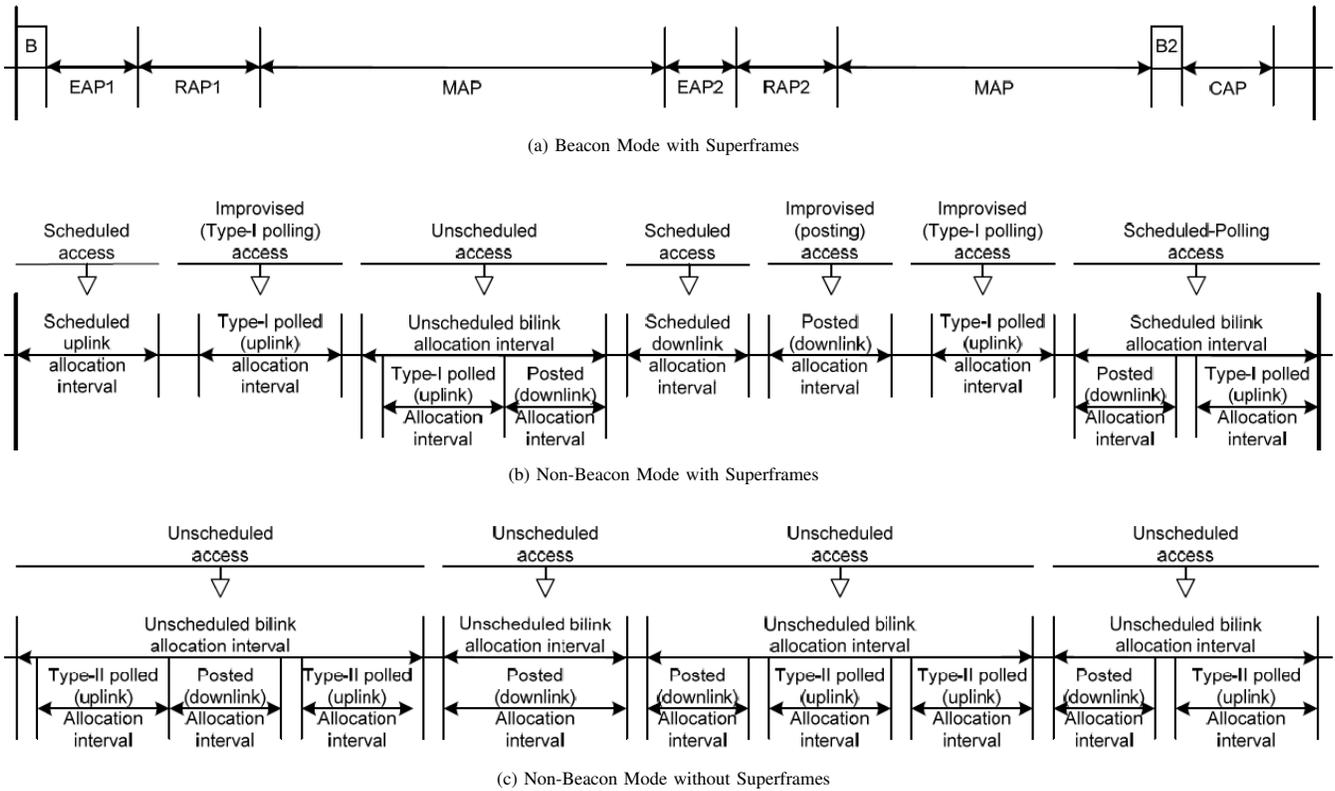


Fig. 2. Access Modes of the IEEE 802.15.6 MAC Protocol [6].

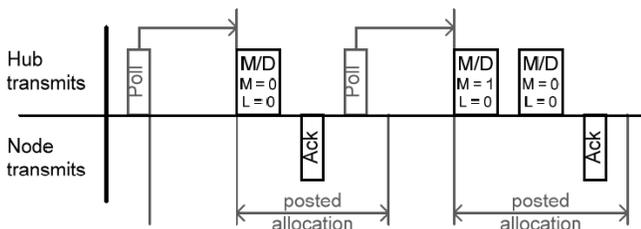


Fig. 3. Posting Access Method Illustration: The Future Posts Transactions [6].

probable retransmission of the last frame in the currently posted allocation time interval. That one is most likely due to the no reception of the required acknowledgment frame. Otherwise, the hub should relinquish and reclaim the posted allocation time interval if it has received the required acknowledgment frame from the node.

- 2) Case 2, with $M = 1$ and $L = 0$: it means that one or more frames are waiting for transmission or retransmission after a $pSIFS$ time in the currently posted allocation time interval.
- 3) Case 3, with $M = 0$ and $L = 1$: it means that no frames are waiting for transmission. Other than probable retransmission of the last frame in the next posted allocation time interval. That one is most likely due to the no reception of the required ac-

knowledgment frame. As well, is due to not enough time remaining in the currently posted allocation time interval for completing another frame transaction plus an appropriate guard time. Thus, the hub should relinquish and reclaim the currently posted allocation time interval.

- 4) Case 4, with $M = 1$ and $L = 1$: it means that one or more frames are waiting for transmission or retransmission in the next posted allocation time interval. That one is most likely due to not enough time remaining in the currently posted allocation time interval for completing another frame transaction plus an appropriate guard time. In this case, the hub should relinquish and reclaim the currently posted allocation time interval.

After sending the required acknowledgment frame to the hub, the node behaves according to the cases defined above. For the first case, it should be ready to receive the retransmission of the last frame after a $pSIFS$ time. As well, it should relinquish the currently posted allocation time interval after a time out ($mTimeOut$) if at this time it has not received it. As for the second case, it should be ready to receive the transmission or the retransmission of one or more frames in the currently posted allocation time interval. While for the third and fourth cases, it should relinquish the currently posted allocation time interval.

3) After the posted allocation time interval: After the end of the posted allocation time interval or after reclaiming it, the hub can send to the node a poll frame conveying an immediate

or a future new posted allocation time interval, extending the remaining or the existing one if there are other frames to transmit. Moreover, it can cancel the future posted allocation time interval by sending a poll frame to the node before the start of it.

Thereafter, we will model the detailed communication between the hub and the nodes through the posting access method and within the stochastic environment of WBAN.

III. POSTING ACCESS METHOD: BEHAVIOR MODELING

The UPPAAL-SMC, as defined in [9], is an alternative formalism to the timed automata limits. The timed automata formalism, as defined in [13] [14], is not flexible and expressive enough to model the stochastic behavior of systems. The UPPAAL-SMC formalism is based on the timed automata formalism and a stochastic interpretation, thus it generates stochastic timed automata (STAs). A model in UPPAAL-SMC consists of a network of STAs interacting components. The STAs components communicate via broadcast channels and share variables to generate networks of stochastic timed automata (NSTAs).

In this section and based on the STA formalism of the UPPAAL-SMC model-checker, we model the whole behavior of the posting access method before, within, and after the posted allocation time interval. As well, we model the parts of the node behavior that is associated with it. The resulting model is a network of stochastic timed automata (NSTA) that is composed of a couple of two templates, as depicted in Fig. 4. Moreover, we mention in Table I and Fig. 5 the intervals of the random functions, the values of the parameters used in this model, and the illustration of these parameters value determination.

1) *Behavior modeling of the hub and the nodes before the posted allocation:* Consider the Hub template, as depicted in Fig. 4(a). It starts with the stochastic interpretation of its non-deterministic choice between allocating time intervals for itself or to the node. Indeed, according to the data sensed by the node or the data received by itself, the hub determines its choice. The node allocated time interval starts at the location *ImmNodeAlloc*. The hub allocated time interval (i.e., the posted allocation time interval) starts after an (X) time-units. The X is selected randomly through the random function *rand* ($e : rand$). The posted allocation is considered as a future allocation compared to the allocation of the node. This latter is considered as an immediate or future allocation. In this section, we model the allocation of the node as an immediate allocation.

To inform the node about its allocated time interval, the hub sends to it a poll frame. In this case, it sends the signals ($IP[id]!$) and ($EP[id]!$). These signals indicating, respectively, the start and the end of the poll frame. While in the case of the hub allocated time interval, it sends to the node the signals ($FP[id]!$) and ($EP[id]!$). The same as the first signals, these signals indicating, respectively, the start and the end of the poll frame.

We model the communication between the hub and the nodes of the network by a random selection of a node (i)

from the network to communicate with the hub. We use the function *randid* ($eid : randid$) for this random selection. Then, we put in it the variable (id) ($id = eid$). This latter represents the identifier of the selected node. As well, we define the time interval (M) of the node allocation randomly through the function *randm* ($em : randm$).

We define the posted allocation time interval and the required time of the frame transmission by a random selection: ($ep:randp$) represents the random selection of the posted allocation time interval, and ($ef:randf$) represents the random selection of the frame transmission time. Then, we put them in the variables ($hp = ep$) and ($hf = ef$), respectively. The variable (MD) used in the Hub template, indicates to the Node template if the hub has more data to transmit ($MD == 1$) or not ($MD == 0$). This variable is determined randomly by the function *randmd* ($em: randmd, MD = em$). The clock (c) is used in the Hub template to compute the current time of the posted allocation time interval. It starts after the locations *PostedAllocationStart* or *HpSIFS2* and resets at the end of the posted allocation time interval, in the location *PostedAllocationEnd*. The clock (h) is used locally to compute the time in the locations.

2) *Behavior modeling of the hub and the nodes within the posted allocation:* Once the posted allocation time interval starts (i.e. immediately after the locations *PostedAllocationStart* or *HpSIFS2*), the Hub starts the transmission of its frame by sending to the Node the synchronization signal ($IS[id]!$). After the transmission time of the frame ($h == hf$), the Hub sends to the Node the synchronization signal ($ES[id]!$) indicating the end of the frame transmission. Then, it moves to the location *HWack* waiting for the reception of the acknowledgment frame:

- 1) If the Hub receives from the Node the synchronization signals ($Ak[id]?$) and ($EAK[id]?$) that are indicating, respectively, the start and the end of the immediate acknowledgment frame transmission, it moves to:
 - a) The transmission of a new frame with new frame transmission time ($ef:randf$), if it has more data to transmit ($MD == 1$) and if there is enough time in its posted allocation time interval ($c \leq hp - (ef + tg)$). This transmission will start after staying in the location *HpSIFS4* for the pSIFS time $h == 1$. The guard ($c \leq hp - (ef + tg)$) represents the L state (in this case $L = 0$) and its value is determined as depicted in Fig. 5.
 - b) The location *PostedAllocationEnd*, if it has more data to transmit ($MD == 1$), but it has no more time to complete its frame transaction ($c > hp - (hf + tg)$, in this case $L = 1$).
 - c) The location *PostedAllocationEnd* if it has no more data to transmit ($MD == 0$). The Hub resets the clock c before moving to the location *PostedAllocationEnd*, which means the relinquishment of the posted allocation time interval.
- 2) If the Hub has not received during ($h == at$) time-units the synchronization signal ($Ak[id]?$), it should

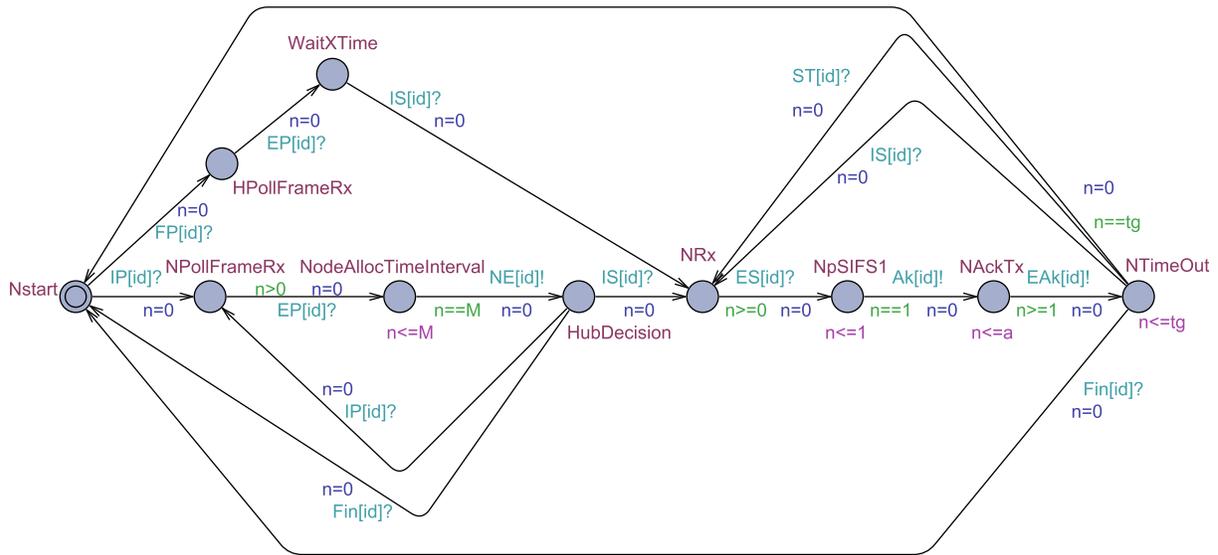
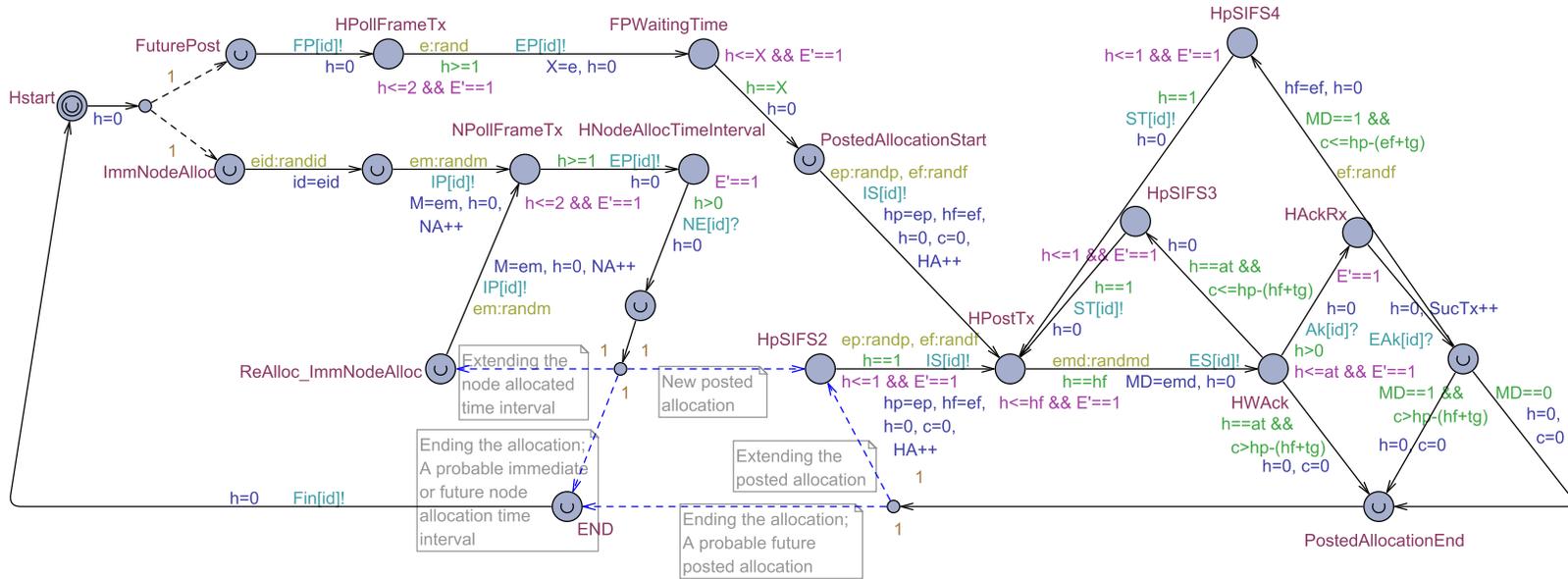


Fig. 4. NSTA Model of the Posting Access Method.

retransmit, after staying in the location HpSIFS for ($h == 1$), its last frame if it has enough time in its current posted allocation ($c \leq hp - (hf + tg)$). Otherwise, it resets the clock c and moves to the location *PostedAllocationEnd*.

Consider the Node template, as depicted in Fig. 4(b). After receiving from the Hub the synchronization signals ($IS[id]?$) and ($ES[id]?$), the Node waits for a *pSIFS* time ($n == 1$) in the location *NpSIFS1*. Then, it moves to the transmission of the acknowledgment frame, which starts and ends after sending to the Hub the synchronization signals ($Ak[id]!$) and ($E Ak[id]!$), respectively. After that, the Node moves to the location *NTimeOut*, where it waits for (tg) time-units the reception of the signals ($IS[id]?$), ($ST[id]?$), or ($Fin?$). The

value of tg is determined as depicted in Fig. 5. If at this time it has not received any signal from the Hub (e.g., the hub is broken down), the Node relinquishes the current allocation time interval by returning to the location *Nstart*. The signal ($ST[id]?$) indicates the transmission of a new frame or the retransmission of the last frame. The signal ($IS[id]?$) indicates the start of a newly posted allocation time interval and the start of the first Post. While, the signal ($Fin[id]?$) indicates the end of the allocation time interval of the hub.

The acknowledgment frame: missing situation: we consider that the hub and the nodes are connected. As mentioned in Fig. 3, the hub sends to the node the poll frame without a required acknowledgment frame. The problem is that in the case of the transmission loss of the poll frame, the hub

can not synchronize with the node when the posted allocation starts. Therefore, this situation allows the data frame transmission loss. We suppose that the posting access method treats this situation through the requirement of an acknowledgment frame transmitted with the data frame, as explained before in Section II. However, the posting access method has not treated the case of the transmission loss of this acknowledgment frame. To prevent that the hub stays blocked until the end of the posted allocation time interval waiting for the acknowledgment frame, we proposed in our model a bounded time (*at*) during with the Hub automaton can stay in the location *HWAck*. Its value is determined as depicted in Fig. 5.

3) *Behavior modeling of the hub and the nodes after the posted allocation:* The blue lines represent the Hub decision after the end of the allocation time interval of itself or for the node. After the end of the allocation time interval, we use another stochastic interpretation, with the same probability weight (i.e., the probability weight as called in UPPAAL-SMC $Pw = 1$), to model the non-deterministic choice of the hub between extending the existing allocation and granting a probably immediate or a future new allocation time interval.

Based on this accurate behavior model of the posting access method and according to the values of the parameters used in this model, we will evaluate in the next section, the performance of this access method.

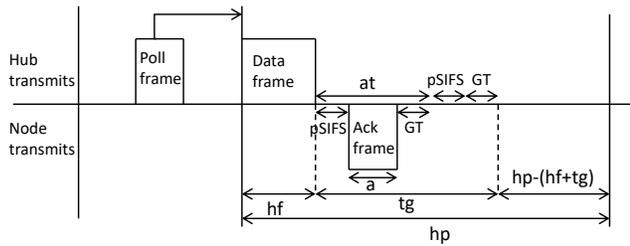


Fig. 5. Illustration of the Parameters Value Determination. GT: Guard Time. Ack: Acknowledgment.

TABLE I. PARAMETERS VALUES AND RANDOM FUNCTIONS INTERVALS.

<i>rand</i>	[5, 10]	<i>randp</i>	[10, 25]
<i>randf</i>	[3, 5]	<i>randm</i>	[10, 25]
<i>randmd</i>	[0, 1]	<i>a</i>	3
<i>tg</i>	7	<i>at</i>	5
<i>pSIFS</i>	1	<i>GT</i>	1

IV. POSTING ACCESS METHOD: BEHAVIOR EVALUATION

To specify properties over NSTAs, UPPAAL-SMC uses a weighted extension of the MITL [15]. Besides, to analyze these properties, UPPAAL-SMC uses the SMC [10]. This latter uses the Monte Carlo simulation to respond to the quantitative questions (i.e., probability estimation), and it uses sequential hypothesis testing to respond to the qualitative questions (i.e., hypothesis testing and probability comparison). Additionally, UPPAAL-SMC provides the simulation of the system behavior and the evaluation of the expected values of the min or the max expression. In this section, to evaluate the scalability of the posting access method, we use the following property:

- The evaluation of the expected values of max:

$$E[bound; N](max : expr) \quad (1)$$

Where *bound* is a time-bound in the evaluation, *N* is the number of runs, and *expr* is the expression to evaluate.

In a network of nodes ranging from 4 to 64 and through $N = 10000$ runs of stochastic scenarios generated by UPPAAL-SMC, we evaluate the scalability. This latter, we evaluate it in terms of the number of the allocated time intervals of the hub and nodes. As well, the energy consumed and the successful transmitted frames by the hub during a determined period of time.

In the experiments, we use three networks and three periods of time (T_1 , T_2 , and T_3) to evaluate and visualize the scalability of the posting access method. The first network is composed of one hub and 4 nodes. The second network is composed of one hub and 16 nodes. As for the third network is composed of one hub and 64 nodes. Moreover furthermore, the first period of time $T_1 = 3600$ time-units, the second period of time $T_2 = 7200$ time-units, and the third period of time $T_3 = 10800$ time-units.

A. Allocated Time Intervals

Using Equation 1, we evaluate the average of the maximum number of allocated time intervals of the hub and the nodes within the three networks and during the three periods of time:

$$E[\leq T; N](max : Hub.HA) \quad (2)$$

$$E[\leq T; N](max : Hub.NA) \quad (3)$$

These formulae compute, in the interval of time T (i.e., the T can be T_1 , T_2 , or T_3) and using N runs, the average of the maximum value of the counters (HA) and (NA).

In our NSTA model, as depicted in Fig. 4, the Hub template uses the HA and NA counters to compute the number of allocated time intervals of the hub and the nodes, respectively. The Hub automaton increments the counter ($HA++$) when the posted allocation time interval starts. As well, it increments the counter ($NA++$) when the node allocated time interval starts.

UPPAAL-SMC estimates the averages of Equations 2 and 3 to be in the confidence intervals, as depicted in Tables II and III, respectively. As well, Fig. 6 and 7 present the visualization of the results for the hub and the nodes, respectively. The results show that within the three networks, the hub and the nodes retain the same number of the allocated time intervals during each period of time. Along with this, we remark that the hub allocates the double of the time intervals number compared to the nodes.

TABLE II. CONFIDENCE INTERVALS OF THE ESTIMATED AVERAGES OF THE MAXIMUM NUMBERS OF ALLOCATED TIME INTERVALS FOR THE HUB.

Parameter evaluated: <i>HA</i>			
<i>T</i>	4 Nodes	16 Nodes	64 Nodes
<i>T1</i>	161.860 ± 0.248666	161.857 ± 0.249261	161.824 ± 0.246545
<i>T2</i>	323.919 ± 0.347292	323.716 ± 0.347743	323.389 ± 0.351441
<i>T3</i>	485.783 ± 0.431237	485.471 ± 0.433604	485.899 ± 0.430501

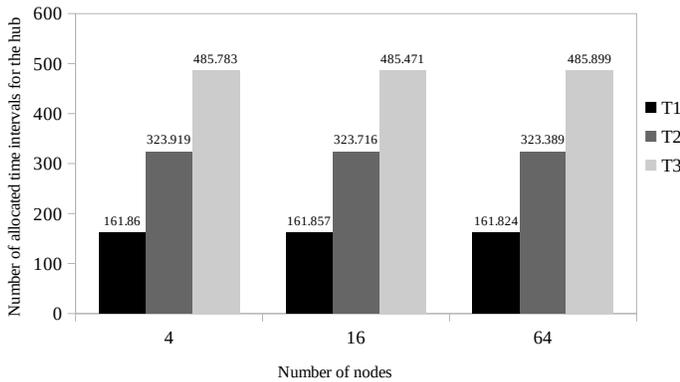


Fig. 6. Averages of the Maximum Numbers of Allocated Time Intervals for the Hub within Three Networks and During Three Periods of Time.

TABLE III. CONFIDENCE INTERVALS OF THE ESTIMATED AVERAGES OF THE MAXIMUM NUMBERS OF ALLOCATED TIME INTERVALS FOR THE NODES.

Parameter evaluated: <i>NA</i>			
<i>T</i>	4 Nodes	16 Nodes	64 Nodes
<i>T1</i>	81.2289 ± 0.163605	81.3295 ± 0.163182	81.4821 ± 0.162648
<i>T2</i>	162.254 ± 0.23184	162.237 ± 0.231811	162.305 ± 0.229111
<i>T3</i>	243.392 ± 0.285667	243.014 ± 0.285128	243.268 ± 0.281951

B. Energy Consumption

Using Equation 1, we evaluate the average of the maximum value of the energy consumed by the hub within the three networks and during the three periods of time:

$$E[\leq T; N](max : Hub.E) \quad (4)$$

This formula computes, in the interval of time *T* (i.e., the *T* can be *T1*, *T2*, or *T3*) and using *N* runs, the average of the maximum value of the variable (*E*).

In our NSTA model, as depicted in Fig. 4, the Hub template uses the variable *E*. This latter computes the energy consumed

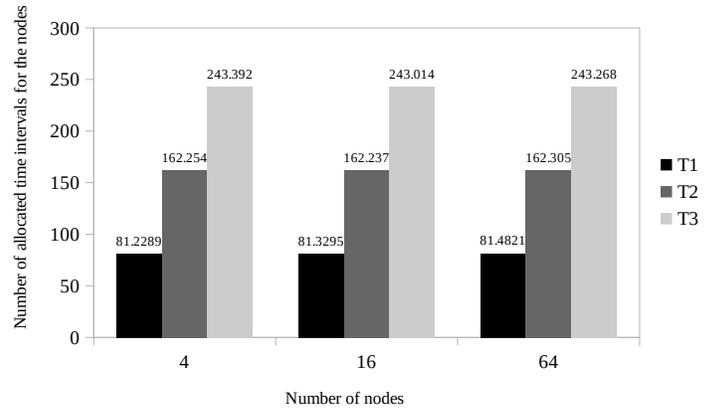


Fig. 7. Averages of the Maximum Numbers of Allocated Time Intervals for the Nodes within Three Networks and During Three Periods of Time.

proportionally to the time spent by the hub when it passes by the locations that have ($E == 1$).

UPPAAL-SMC estimates the averages of Equation 4 to be in the confidence intervals, as depicted in Table IV. As well, Fig. 8 presents the visualization of the results. These latter show that within the three networks, the hub retains the same value of the energy consumed during each period of time.

TABLE IV. CONFIDENCE INTERVALS OF THE ESTIMATED AVERAGES OF THE MAXIMUM VALUES OF THE ENERGY CONSUMED BY THE HUB.

Parameter evaluated: <i>E</i>			
<i>T</i>	4 Nodes	16 Nodes	64 Nodes
<i>T1</i>	3595.17 ± 0.107094	3595.17 ± 0.105837	3595.10 ± 0.107957
<i>T2</i>	7195.17 ± 0.106899	7195.15 ± 0.107162	7195.21 ± 0.105604
<i>T3</i>	10795.2 ± 0.106891	10795.2 ± 0.106597	10795.2 ± 0.105469

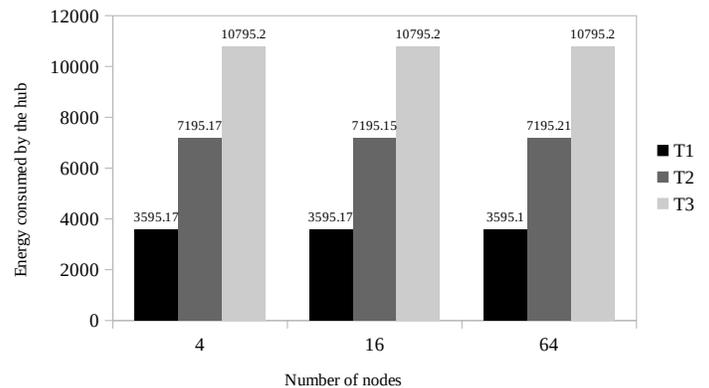


Fig. 8. Averages of the Maximum Values of the Energy Consumed by the Hub within Three Networks and During Three Periods of Time.

C. Throughput

Using Equation 1, we evaluate the average of the maximum number of the successful transmitted frames by the hub within the three networks and during the three periods of time:

$$E[\leq T; N](max : Hub.SucTx) \quad (5)$$

This formula computes, in the interval of time T (i.e., the T can be $T1$, $T2$, or $T3$) and using N runs, the average of the maximum number of the counter ($SucTx$).

In our NSTA model, as depicted in Fig. 4, the Hub template uses the counter $SucTx$ to compute the number of successful transmitted frames by the hub to the nodes. The Hub automaton increments the counter ($SucTx++$) when it receives the acknowledgment frame from the node.

UPPAAL-SMC estimates the average of Equation 5 to be in the confidence intervals, as depicted in Table V. As well, Fig. 9 presents the visualization of the results. These latter show that within the three networks, the hub almost retains the same number of throughput during each period of time.

TABLE V. CONFIDENCE INTERVALS OF THE ESTIMATED AVERAGES OF THE MAXIMUM NUMBERS OF SUCCESSFUL TRANSMITTED FRAMES BY THE HUB.

Parameter evaluated: $SucTx$			
T	4 Nodes	16 Nodes	64 Nodes
$T1$	203.968 ± 0.31096	203.873 ± 0.312695	203.880 ± 0.312903
$T2$	408.322 ± 0.452698	408.563 ± 0.442229	408.371 ± 0.445739
$T3$	613.109 ± 0.554558	612.956 ± 0.545412	612.775 ± 0.540818

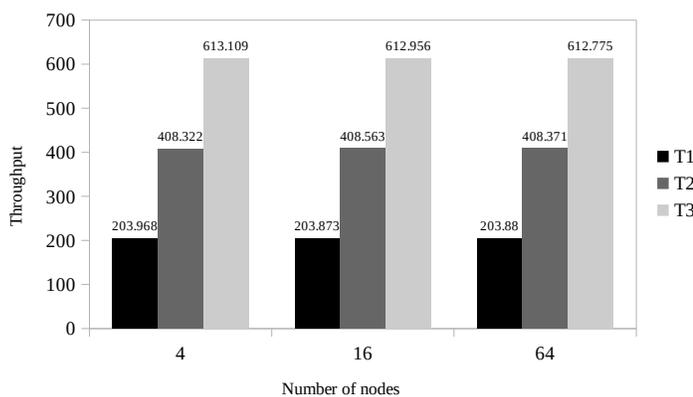


Fig. 9. Averages of the Maximum Numbers of Successful Transmitted Frames within Three Networks and During Three Periods of Time.

D. Results

From these evaluation results, we interpret the constant values obtained as that during each period of time T the hub allocates a fixed number of time intervals for itself and for

the nodes. Accordingly, the values of the energy consumed and the throughput remain constant, even with the increase of the network density. Hence, extending the period of time T , we can get more time intervals, as well as more throughput rate and more energy consumption. As we interpret the double number of the allocated time intervals of the hub compared to the nodes as that the posting access method, as modeled in Section III, provides the priority to the hub to allocate more time intervals to still communicating with the nodes despite the growth of the network density. This is due to its important role in the network as a controller and a monitor device.

As a result, with the posting access method, the hub works with the same performance, even with the increase of the network density. Thus, validating the scalability of the posting access method, and therefore, validating the scalability of the WBAN.

V. CONCLUSION

In this paper, we used the statistical model checking toolset UPPAAL-SMC to investigate the posting access method under the WBANs stochastic environment. Based on UPPAAL-SMC, we modeled and evaluated the behavior of the posting access method in terms of scalability. We first used the stochastic timed automata formalism to construct a model of this behavior. Then we used the metric interval temporal logic specifications to evaluate it. The evaluation results showed that, with the posting access method, the hub works with the same performance in terms of the allocated time intervals, the energy consumed, and the throughput, even with the increase of the network density. The thing that validated the scalability of the posting access method and, therefore, the scalability of the WBAN.

Supplement to the first conclusion, we came to learn from this case study that with UPPAAL-SMC, we can model and evaluate accurately the behavior of the MAC protocols. Nevertheless, modeling and evaluating the behavior of MAC protocols through the stochastic timed automata and the metric interval temporal logic specifications, adopted by UPPAAL-SMC, necessitate a certain level of expertise in this side. The thing that is not available for many MAC protocol designers.

As future work, we will combine the posting access method with other access methods to construct a new MAC protocol for WBANs. Furthermore, to facilitate the use of Uppaal-SMC powerful modeling and analyzing algorithms, we propose to define a model-driven engineering approach that uses a domain-specific modeling language (DSML) as a start and the UPPAAL-SMC as a target and back. This DSML should be dedicated to the MAC protocols of WBANs.

ACKNOWLEDGMENT

This work is supported by the Moroccan Ministry of Higher Education Scientific Research and Professional Training and the CNRST under the project "Réseaux de capteurs sans fil biomédicaux" [grant number PPR/2015/43].

REFERENCES

- [1] S. Movassaghi et al: Wireless body area networks: A survey. IEEE Communications Surveys & Tutorials, 1658–1686 (2014)

- [2] B. Latré et al: A survey on wireless body area networks. *Wireless Networks*, 1–18 (2011)
- [3] Y. Ben Maissa et al: Modeling and analyzing wireless sensor networks with verisensor: An integrated workflow. *Transactions on Petri Nets and Other Models of Concurrency VIII*, Springer, 24–47 (2013)
- [4] Z. Mohammadi et al: New high-rate UWB scheme for WBAN-based healthcare systems. *Progress In Electromagnetics Research, EMW Publishing*, 125–139 (2014)
- [5] M. Chen et al: Body area networks: A survey. *Mobile Networks and Applications*, 171–193 (2011)
- [6] IEEE Standards Association and others: IEEE Standard for Local and Metropolitan Area Networks—Part 15.6: Wireless Body Area Networks. *IEEE Standard for Information Technology, IEEE*, 1–271 (2012)
- [7] S. Ullah et al: A review of IEEE 802.15. 6 MAC, PHY, and security specifications. *International Journal of Distributed Sensor Networks*, 950–704 (2013)
- [8] B. Touijer et al: MAC protocols for Wireless Body Area Networks: An overview. *Wireless Communications and Mobile Computing Conference (IWCMC)*, 1227–1232 (2017)
- [9] A. David et al: Time for statistical model checking of real-time systems. *International Conference on Computer Aided Verification*, Springer, 349–355 (2011)
- [10] E.M. Clarke et al: Statistical model checking for cyber-physical systems. *International Symposium on Automated Technology for Verification and Analysis*, Springer, 1–12 (2011)
- [11] E.A. Emerson et al: The beginning of model checking: A personal perspective. *25 Years of Model Checking*, Springer, 27–45 (2008)
- [12] E.M. Clarke et al: Formal methods: State of the art and future directions. *ACM Computing Surveys (CSUR)*, 626–643 (1996)
- [13] R. Alur et al: A theory of timed automata. *Theoretical Computer Science*, Elsevier, 183–235 (1994)
- [14] A. Alur et al: Decision problems for timed automata: A survey. *Formal Methods for the Design of Real-Time Systems*, Springer, 1–24 (2004)
- [15] R. Alur: The benefits of relaxing punctuality. *Journal of the ACM (JACM)*, 116–146 (1996)