



Modeling and Performance Evaluation of Energy Consumption in S-MAC Protocol Using Generalized Stochastic Petri Nets

M. YadollahzadehTabari*^a, P. Mohammadizad^b

^a Department of Computer Engineering Babol Branch, Islamic Azad University, Babol, Iran

^b Department of Computer Engineering Mazandaran Institute of technology, Babol, Iran

PAPER INFO

Paper history:

Received 07 January 2020

Received in revised form 17 February 2020

Accepted 06 March 2020

Keywords:

Wireless Sensor Network

Modeling

Performance Evaluation

Energy Consumption

Generalized Stochastic Petri Net

ABSTRACT

One of the specifications of wireless sensor networks is the limited power sources for their nodes. Therefore, assessment of energy consumption in these networks is very important, and using traditional simulators to evaluate the energy consumption of network nodes has become common practice. Simulators often have problems such as fluctuating output values in different implementations of the same scenario, lack of a visual structure to understand the system, and lack of theoretical background for the gained results. In contrast, analytical modeling methods such as Petri nets do not have the aforementioned problems. Also, they have the necessary tools to model and evaluate the performance of investigated system. The Sensor-MAC (S-MAC) protocol in these networks, which operates in MAC layer, is a competition-based protocol designed to reduce power consumption. This paper presents an analytical model based on a Generalized Stochastic Petri Net (GSPN) to evaluate the power consumption of nodes in an S-MAC-based wireless sensor network. The presented Petri model was implemented in PIPE software, and by applying mathematical calculations from the model, we were able to derive the equations for calculating its energy consumption. The validity of the proposed model is measured by implementing similar scenarios in the Castalia simulator. The experiments were conducted in terms of the number of nodes, duty cycle rate, upper layer data flow, and packet size. The results of the presented model are extracted in a much shorter time than the simulator with the same gained values.

doi: 10.5829/ije.2020.33.06c.08

1. INTRODUCTION

Advancements in wireless communications have enabled the development of wireless sensor networks (WSNs), which include small devices that collect information through collaboration. Currently, WSNs are widely used in commercial, industrial and military applications. The sensor nodes in these networks usually perform in a way that they sense the environment and transmit information towards the base station by communicating with other nodes [1]. WSNs have multiple resource constraints that include limited amount of power, short communication range, low bandwidth, and limited processing and storage capacity per node [2]. Also the sensor nodes usually have a limited power source which is difficult to replace. The performance of each node in this network is highly

dependent on its energy consumption and this has become a major issue in WSNs. [1]. The radio transmitter/receiver unit is a major energy consumer in these networks, especially considering that the radio receiver is always on. Therefore, a large amount of energy savings can be achieved through energy-efficient media access control (MAC) mechanisms [3]. One way to optimally utilize the limited energy resources of sensor nodes is to use the MAC protocol that follows a periodic sleep and wake-up schedule to reduce energy consumption. The idea behind these protocols is that its nodes often have no new data to send to the network. Therefore, there is no need for them to be permanently active and to monitor the network regarding their energy sources.

One of the protocols that use this workflow is the

*Corresponding Author Institutional Email: m_tabari@baboliau.ac.ir
(M. Yadollahzadeh Tabari)

Sensor-MAC (S-MAC) protocol. The S-MAC protocol is a competition-based protocol designed to reduce power consumption in WSNs. This protocol works on a periodic work cycle. There are two main components in S-MAC to achieve data-sharing goals in the media context and to minimize energy consumption, periodic sleep and wake-up, and Four-step handshake for collision avoidance. Network developers should always investigate the best protocols to reduce energy consumption as much as possible. Developers must evaluate the energy consumption of the applications before deploying the network. By evaluating the energy consumption, the lifetime of the node can be estimated, and we will adopt strategies to increase the network lifetime and anticipate the time to replace the sensor node. The traditional approach towards network analysis is to use the existing simulation tools or simulators, such as OPNET, NS-2 and etc. For using these tools, we must consider that the result of simulating a scenario may differ depending on the selected tool, because of important divergences between simulators [4]. Another problem encountered using these simulators is the time it takes to run them. In most cases, in order to achieve a sustainable result, it is necessary to run multiple scenarios across the network to achieve a stable average. An alternative to simulation is to employ formal modeling and analysis techniques. Using these techniques, both performance evaluation and model checking can be performed. It also allows for flexible and intuitive modeling to evaluate complex network scenarios without having to implement the actual environment [5, 6]. Besides, the Petri net formalism is one of the most widely used analytical tools for modeling and performance analysis of systems like wireless networks [4]. Generalized Stochastic Petri Net as a development of Petri Nets can be used for modeling and analyzing systems with properties such as synchronization, distribution and parallelism. Given these properties, the Generalized Stochastic Petri Network is an ideal tool for modeling S-MAC mechanisms and can be used to analyze its energy consumption [4, 7]. Therefore, in this paper, we intend to present an analytical model based on Generalized Stochastic Petri net to evaluate energy consumption in S-MAC protocol based sensor network.

2. RELATED WORKS

There are not many studies that utilize Petri Nets and its developed variants to model and analyze protocols in WSNs. But below we consider some of the main research investigated in this area. Trin et al. [8] presented a color-based Petri Net approach for modeling WSNs and congestion detection. The model utilizes the hierarchical modeling capability of Coloured Petri Nets (CPNs), including different levels of abstraction. This facilitates

the study and development of the model. The proposed model consists of five sensor nodes and five channels consisting of four components (source, middle nodes, base station, channel).

Zairi et al. [9] presented a CPN model for the general behavior of WSNs. They focused specifically on the two components of WSN as the radio and the Mac protocol. In fact, the proposed formalism is based on a component-based modeling approach, so each subnet is developed separately and then communicates with other components. The generated model is used to estimate the energy consumption and lifetime of the network.

Mokdad et al. [3] used the color Petri net model to model and evaluate the EQ-MAC protocol. The performance of the protocol was evaluated in terms of mean delay rate and packet delivery ratio.

Gupta [10] proposed an analytical approach to predict the collision probability and estimate the available bandwidth for accurate admission control in wireless ad hoc networks. The author used Markov Chain for modeling purposes. They also claim that their work does not take into consideration any heterogeneous conditions to resemble real networks. The work is verified through extensive simulations to validate the proposed model. Abdullahi Azgmi and Khalili [11] presented a CPN model for modeling and evaluating the performance of a media access protocol in WSNs called S-MAC. Performance measures include power consumption and packet delivery delay. The proposed model uses hierarchical CPNs.

Tabari and Pouyan [12] presented a framework for performance analysis of the network layer in Mobile Ad-Hoc Networks (MANET). They use Stochastic Reward Nets for modeling that is a variation of Generalized Stochastic Petri Nets. The produced results show that the obtained values from the SRN model well matched to the values generated from the NS-2 simulator with considerable shorter execution time.

3. S-MAC Protocol

The S-MAC protocol is a competition-based protocol designed to reduce power consumption in WSNs. Previously, the CSMA/CA technique required the nodes to continuously monitor the channel to ensure it was empty. This results in energy consumption even when the node has no transmission packets. To address this challenge, the S-MAC protocol has been introduced [13-16]. This protocol works on a periodic work cycle and is competition-based. There are two main components in S-MAC to achieve data-sharing goals in the media context and to minimize energy consumption, including periodic sleep and wake-up and four-step handshake for collision avoidance[15].

3.1. Periodic Sleep and Wake-up

In the S-MAC protocol, time is divided into periods of constant T length. Each period comprises two distinct parts as wake-up and

sleep. The Duty Cycle of the nodes in this protocol is defined as the ratio of the waking period T_{Listen} to the total length of the period T [11, 17]:

$$Duty\ Cycle = \frac{T_{Listen}}{T} \quad (1)$$

The nodes are free to choose their sleep and wake-up schedules, but prefer to adjust with the neighboring nodes to reduce overhead [16]. Each node needs to select its own time table and exchange it with its neighbors before it can begin its sleep and wake-up cycle [16]. Nodes that have the same sleep and wake schedule are called a virtual cluster. Therefore, part of the awakening period of the nodes in this protocol is dedicated to the coordination of the nodes in order to perform their sleep and wake-up periods. The nodes in the rest of the wake-up period will send their data packets. The synchronization process in this protocol is performed periodically by sending *SYNC* messages. In the sync time frame that is indicated by (T_{sync}), one of the neighboring nodes sends a *SYNC* packet and the rest of the nodes listen to it. In each N_{sync} cycle, each node is once responsible for transmitting the *SYNC* packet and receiving response from other nodes, and in the other $N_{sync}-1$ cycle it receives a *SYNC* packet from another transmitter node [17, 18].

3. 2. Four-step Handshaking S-MAC uses RTC / CTS / DATA / ACK four-step handshake sequence to avoid collision with DATA packets [19]. The procedure is that the nodes that intend to send data packets to the channel first monitor the channel for DIFS period to ensure that it is empty. The beginning of the four-step handshake phase delayed as random slot numbers. This is called the Contention Window. A winning node has fewer slots to compete in the window and thus completes the process sooner. It then sends the RTS packet. The receiver also sends the CTS packet to the sender if it is ready, and the sender sends the DATA packet. Finally, the receiver sends the ACK packet alongside receiving the DATA packet. All transmitter and receiver neighbors when they hear RTS and CTS messages should sleep until the end of the current transmission according to the NAV vector [14, 15]

4. GENERALIZED STOCHASTIC PETRI NET

Formally a Generalized Stochastic Petri net is octet set as $\{P, T, \rho, I, O, H, M0, W\}$ in which (P, T, I, O, M0) contains the basic definition of a Petri net, and are respectively for showing Places, Transitions, Transition Input Set, Transition Output Set, and Start state. The set (T) here is divided into two subsets T_1 and T_2 , which represent the set of immediate transitions and timed transitions, respectively. H represents the set of inhibitory arcs and the transients' priority represented by

ρ and their weight is represented by $W = (w_1, w_2, \dots, w_n)$. GSPN provides the same modeling capability as a reward-based Markov chain model and its solution will be accomplished by matching to the corresponding continuous Markov chain model. The relationship of the reward function varies for each system and is usually a combination of evaluation criteria that can be used to evaluate a system that is modeled by GSPN [4, 20]. The derivation of each evaluation criterion is obtained from Equation (2), where r_i is the reward function corresponding to each evaluation criterion, and η_i is the steady state probability related to the condition of the Markov chain for which the condition r_i is satisfied [21]. The reward criterion can include the probability of a particular condition in the Petri net, the expected number of tokens in a given location $\#P_i$ or the average number of transitions per unit time $Thr(T_j)$.

$$E[R] = \sum_{i \in \Omega_T} r_i \eta_i \quad (2)$$

5. ANALYTICAL MODEL OF S-MAC PROTOCOL USING GSPN

In the presented model, we will detail the most important components of the S-MAC protocol including: the sleep and wake-up process, the 4-step handshake avoidance process and the data transfer process. The proposed model is compatible with any number of network nodes that are in competition for channel acquisition. In the following we will derive the mathematical equations for calculating the energy consumption of the nodes from the proposed model. The overall model was presented in Figure 1.

To model the sleep and wake-up process that is shown in Figure 1, two transitions T_{WakeUp} and T_{Sleep} are used, whose firing rates depend on the work cycle and the frame length of the nodes. During the delay time intended for the T_{Sleep} transition, the node is assumed to be in sleep mode, and when the T_{Sleep} transition is activated, the node passes the sleep period and the node wake-up period begins, and also this supposed for transition T_{WakeUp} .

At the beginning, the nodes start synchronizing before starting the operations related for sending data to share their sleep and wake schedules with other nodes, as well as the schedule of other nodes in the network. The T_{Sync} transition represents the protocol synchronization period. The T_{Sync} transition refers to the synchronization time during a frame. After the end of the wake-up period, the node sleep period begins with the activation of the T_{WakeUp} transition. In the next step, the nodes compete for access to the channel using the collision avoidance mechanism. The number of tokens in the place PN represents the number of nodes in a virtual cluster.

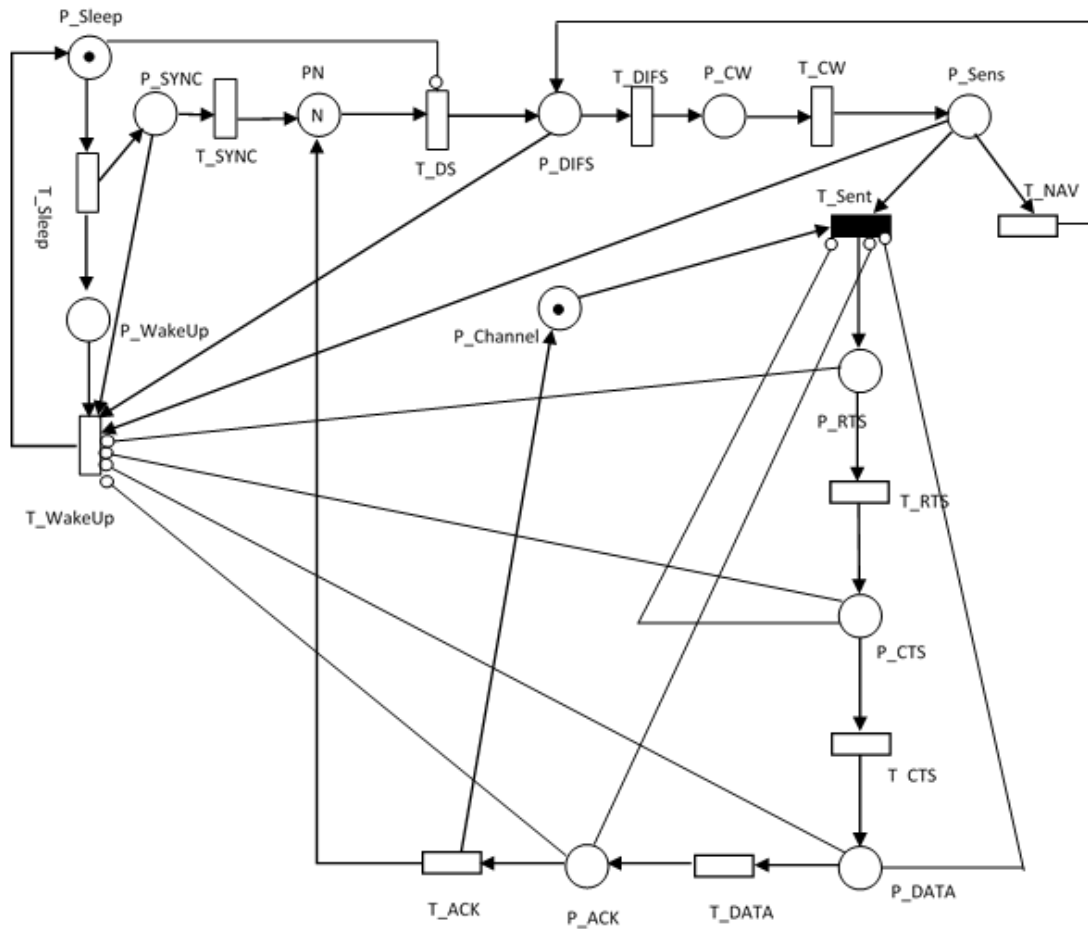


Figure 1. The general GSPN Model of S-MAC .Protocol protocol

After synchronization, nodes that share a common schedule are known as a virtual cluster. After the activation of T_{DS} transition, each node will monitor the channel as DIFS time period that is shown by the T_{DIFS} transition. After monitoring the channel as DIFS time period, each node needs to defer its sending as random slots before beginning each transmission. This action is indicated by the T_{CW} transition. At place P_{Sense} , the situation of successful transmission or collision are investigated in the channel as indicated by the selection structure in model, which means that one of these two transitions is executed. The T_{Sent} instant Transition indicates a state where the node has succeeded to capture the channel and is ready to send it. The T_{NAV} transition is executed when the channel is busy and the node fails to send data. The execution time of this transition can be calculated according to Equation (3).

$$T_{Nav} = T_{CTS} + T_{DATA} + T_{ACK} \quad (3)$$

After the NAV interval is over, the node waits again for the DIFS interval and then monitors the channel for free.

This is shown in the model by the arc connected from the T_{NAV} transition to the place P_{DIFS} . If the channel is free, the T_{Sent} transition is activated and a four-step handshake operation is performed.

The first step of the four-step generator is to execute the T_{RTS} transition. The node that wants to send data, first sends an RTS packet, after receiving the RTS from the receiving node and sending the CTS packet back to the sender, the sender node sends the Data packet. After completing the data delivery, the receiver sends an ACK message back to the sender node. These steps are shown using the T_{RTS} , T_{CTS} , T_{DATA} and T_{ACK} transitions, respectively.

Along activating the T_{ACK} transition, a mark is added to the place $P_{Channel}$ to indicate that the channel has been released and can be forwarded to other nodes. A token returns to the place PN so that the node can again compete for channel acquisition and data transmission. The inhibitor arc between the places P_{RTS} , P_{CTS} , P_{DATA} , P_{ACK} , and the T_{Sleep} transition represents the fact that the transmitter node does not change to the

sleep mode until it is in the process of data exchange and the Ack message has not been reached back. As long as a node is in the processing of sending and receiving one of the *RTS*, *CTS*, *DATA*, and *ACK* packets, the other node will not be able to transmit data. This is shown by the inhibitor arc between the places P_{RTS} , P_{CTS} , P_{DATA} , P_{Ack} and the transition T_{Sent} . The inhibitor arc between T_{DS} and P_{Sleep} indicates that the nodes have no transmissions during sleep mode.

5. 1. S-MAC Protocol Energy Consumption Assessment

For deriving the mathematical equations for energy consumption from the model, it should be noted that the cycle time of the wireless sensor nodes consists of two sleep modes and an active one. The amount of network energy consumption also depends on the mode of operation of the sensor nodes in either of the two states. Therefore, the total energy consumed is divided into two parts of the energy consumed in the wake cycle (E_{awake}) and the energy consumed in the sleep cycle (E_{sleep}) as shown in Equation (4).

$$E = E_{active} + E_{sleep} \quad (4)$$

The energy consumption in each mode also depends on the node modes as, transmission, reception, channel listening and sleep, which are represented by txp , rxp , $idelp$ and sp , respectively. The energy consumed by the node when listens to the channel is the same as the energy consumed in receiving data. Energy consumption in the transmission mode depends on the used power level. The energy analysis of the S-MAC protocol assumes the use of sensor nodes based on the CC2420 standard whose energy consumption in different nodes modes is determined according to Table 1. Node Awakening Energy (E_{active}) also includes energy consumed during synchronization (E_{sync}), energy consumed during the competition and data transfer period (E_{data}), and energy consumed to stay awake after competition and data transfer ($E_{noSleep}$) as shown below:

$$E_{active} = E_{sync} + E_{data} + E_{noSleep} \quad (5)$$

To calculate the energy consumption of the nodes during the synchronization period based on what is described in Section 3, in each synchronous frame represented by (T_{sync}), one node from the set of neighboring nodes sends a closed *SYNC packet* and the rest of the nodes listen to it. Therefore, in each N_{sync} cycle, each node is once responsible for transmitting the *SYNC packet* and receiving response from other nodes, and in the next $N_{sync}-1$ cycle, it receives a *SYNC packet* from another transmitter node. Therefore, in each N_{sync} cycle, each node is once responsible for transmitting the *SYNC packet* and receiving response from the other nodes, and in the next $N_{sync}-1$ cycle, it receives a *SYNC packet* from the other neighbor nodes. This is calculated

using Equations (6)-(8). Here, $m(P_{sync})$ is the average number of nodes that have gone through the synchronization step and is equal to the average number of tokens in place P_{sync} :

$$E_{sync} = [(E_{sync}^{sender} + E_{sync}^{receiver}) / N_{sync}] \times m(P_{sync}) \quad (6)$$

$$E_{sync}^{sender} = (t_{sync} \times txp + (T_{sync} - t_{sync}) \times rxp) \quad (7)$$

$$E_{sync}^{receiver} = T_{sync} \times rxp \times (N_{sync} - 1) \quad (8)$$

In the wake-up state, a node will send a *DATA* packet after caching the channel. Based on what is described in Section 3, the transmitter node will perform *RTS-CTS* 4-way handshake to send data. Depending on being in one of the states of the transmitter or receiver, a node may be transmitting or receiving one of the *RTS* or *DATA* packets, or receiving/transmitting the *CTS* /*ACK* packets. Therefore, the amount of power consumed by the node in the data transfer process and the four-step *RTS* / *CTS* handshake process will be based on Equation (9). Each *RTS*, *CTS*, *DATA*, and *ACK* packets spend time on transmission as t_{Rts} , t_{Cts} , t_{Data} , t_{Ack} , respectively.

$$E_{data} = N_{succ} \times [t_{Difs} \times (rxp) + t_{cw} \times (rxp) + (t_{Rts} + t_{Data}) \times txp + (t_{Cts} + t_{ack}) \times rxp] + N_{succ} \times [t_{Difs} * (rxp) + t_{cw} \times (rxp) + (t_{Rts} + t_{Data}) \times rxp + (t_{Cts} + t_{ack}) \times txp] + 2 \times N_{fail} \times [t_{Difs} \times (rxp) + t_{cw} \times (rxp) + t_{Nav} \times (sp)] \quad (9)$$

To calculate Equation (9), we need to obtain from the model the number of nodes that successfully send data (N_{succ}) as well as the nodes that failed to send (N_{fail}), and the length of time the nodes wake up and listen to the channel.

These will be calculated using Equations (10) and (11) of the GSPN model in Figure 7. The term thr in this equation, is the symbol for calculating the throughput of the corresponding transition.

$$N_{succ} = p_{sense} \times \frac{thr(T_{sent})}{thr(T_{sent} + T_{Nav})} \quad (10)$$

$$N_{fail} = p_{sense} \times \frac{thr(T_{Nav})}{thr(T_{sent} + T_{Nav})} \quad (11)$$

TABLE 2. Power consumption for different nodes' modes according to the CC2420 Radio satandard

Modes	Symbol	Energy Consumption
Receive data	Rxp	62 mj
Send data	Txp	46.2 mj
Idel	Idelp	62 mj
Sleep	Sp	1.4 mj

During each T -Active wake-up cycle in the S-MAC protocol, a node will not sleep when it is not synchronized and has no data to send. But they listen to the channel until their sleep period reaches, which consumes as much energy as the time it takes to listen to the channel. This time will be different for the nodes that have successful or unsuccessful sending, depending on the process they are doing. The channel listening time and related energy consumption are calculated using Equations (12)-(14). $idelp$ is the energy consumption of a node in *wake-up* and *idle* state.

$$T_{idel_s} = T_{active} - (T_{sync} + t_{Difs} + t_{Cw} + t_{Rts} + t_{Cts} + t_{Data} + t_{Ack}) \quad (12)$$

$$T_{idel_f} = T_{active} - (T_{sync} + t_{Difs} + t_{Cw}) \quad (13)$$

$$E_{nosleep} = 2 \times N_{succ} \times [T_{idel_s} \times idelp] + 2 \times N_{fail} \times [T_{idel_f} \times idelp] \quad (14)$$

The duration of the sleep period and its associated energy consumption are obtained from Equations (15)-(16).

$$t_{sleep} = T_{frame} - (T_{sync} + t_{Difs} + t_{Cw} + t_{Rts} + t_{Cts} + t_{Data} + t_{Ack}) \quad (15)$$

$$E_{sleep} = t_{sleep} \times sp \times m(P_{sleep}) \quad (16)$$

6. MODEL IMPLEMENTATION AND VALIDATION

We evaluate the model by comparing the energy consumption obtained from Equation (4) and the energy consumption obtained by the simulator. To this end, we derive a specific scenario from the WSN network and implement it in the *Castalia* simulator to extract the energy values of the nodes in the network. Then the relevant network scenario parameters are given to the presented analytical model and the energy results are calculated using the equations presented in the previous section. *PIPE* software was also used to implement the proposed analytical model. *PIPE* is a Java-based utility with a simple GUI to build and analyze generalized random Petri net models. The implemented scenario assumes 10 fixed sensor nodes. The nodes were randomly assigned to a limit of 50 by 50 m. The packet size was assumed to be 100 bytes in the MAC layer. The RTS/CTS and ACK/Sync packet size, as IEEE 802.11 standard, are set to 13 and 11 byte respectively. Frame and Sync time are also 610 and 6 ms.

6. 1. Implementation and Results of the Protocol

In order to comprehensively evaluate the energy consumption in the S-MAC protocol, the scenario was implemented using four parameters as, *Duty Cycle ratio*,

number of sensor nodes, *data entry rate from the upper layer*, and the *packet size*. According to the observed fluctuation in the output of the *Castalia* simulator software for each scenario, the values of the energy consumption of the nodes were obtained and averaged from 10 distinct runs of the simulation. To run each scenario in *PIPE* analytical simulation, the value of each parameter is interpreted to the model. Also the value of confidence level for the model is considered to be 95%. Then the energy consumption values are calculated according to the equations presented in Section 5-1. Figure 2 shows the relationship between energy consumption and the *Duty Cycle* rate of wireless sensor nodes. It is possible to change the value of the *Duty Cycle* in the model by changing the firing rate of the transitions T_{Sleep} and T_{Wakeup} in the model, according to the formula presented in Equation (1). For example, for 10% and 20% *Duty Cycles* with a total frame length of 610 seconds, the *wake-up periods* will be equal to 61 and 122 seconds, respectively. The results show that as the *Duty Cycle* increases, the energy consumed by the sensor nodes increases linearly. The results also show the match between the simulator and the model. Figure 3 shows the energy consumption per different number of nodes in different cycles of activity. Results show less than 5% difference between model and simulator values. The parameter of the *number of sensor nodes* is set by adjusting the number of tokens in place PN as the figure shows. As the number of nodes in the cluster increases, the power consumption gradually increases until the number of nodes exceeds eight nodes.

This may be due to the fact that with the small number of nodes in the network, there are lower number of nodes competing with each other over channel acquisition, with almost every node listening to the channel in each cycle. This will increase the idle listening energy consumption. But with increasing in node density, more nodes have more packets to send per cycle and hence most of the nodes hear the unwanted RTS. As a result, these nodes go to sleep until the neighboring nodes are transmitted,

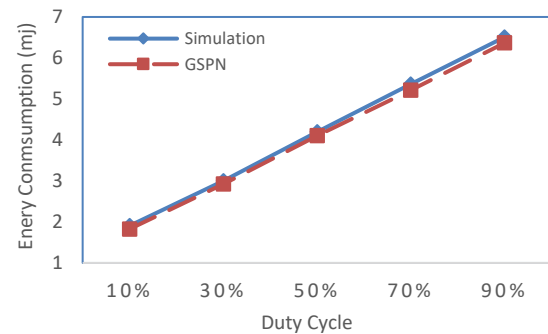


Figure 2. Results of the energy consumption for different values of Duty Cycle

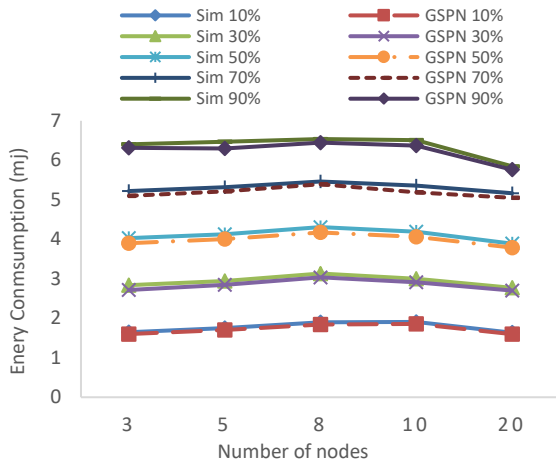


Figure 3. Results of the energy consumption for different number of nodes and Duty Cycle

which reduces overall power consumption. In the next experiment, we investigated the energy consumption in terms of different *data entry rate* of the network layer at different wake-up cycles of the sensor nodes. The data entry rate values of the network layer are adjusted by varying the value of the T_{DS} transition firing rate in the model. Figure 4 shows that the average energy consumption increases with an increase in the *Duty Cycle*. This is because, in the lower *Duty Cycle*, since the nodes spend most of their time in the sleep mode, increasing the data entry rate has no significant effect on the amount of energy consumed. Also in the high *Duty Cycle* (90%), because of the high number of RTS and CTS messages, the nodes quickly fall asleep by adjusting their NAV timer, which increases the not-so-high energy consumption per increasing in data entry rate. Figure 5 also illustrates the energy consumption of different data

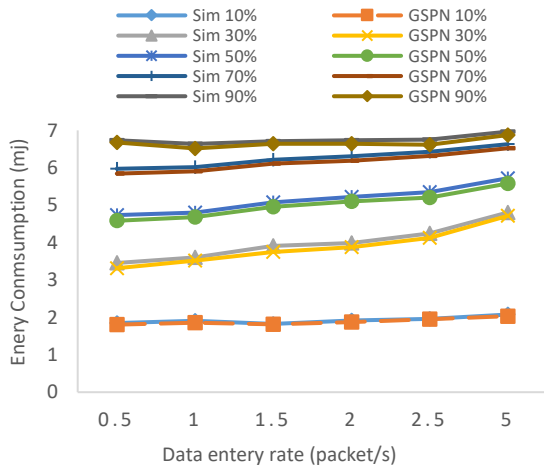


Figure 4. Results of energy consumption with different amounts of Data entry rate traffic

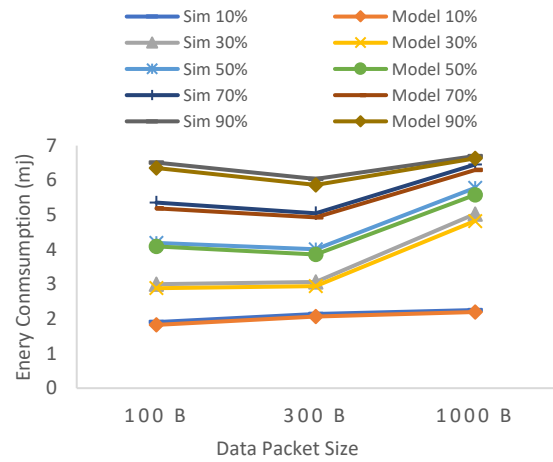


Figure 5. Results of energy consumption of different data pack sizes

packet sizes in both simulator and model. The results indicate the little difference between the two methods. A larger packet, increases energy consumption. Also, a larger packet requires more transmission time. Then as mentioned earlier, if the time needed to send and receive a packet exceeds the node's wake-up period, the node adds its wake-up time. This will consume more energy for the nodes.

7. CONCLUSION

In this study, we analyzed and modeled the S-MAC, as an efficient media access control protocol, in wireless sensor network. The modeling was performed using Generalized Stochastic Petri net. Then, the energy consumption of the nodes was calculated analytically, using the proposed model. To determine the accuracy of the proposed model, the energy consumption values were compared with the results of the simulator which execute the same scenario. The experiments conducted to investigate the effects of different number of nodes, network traffic, packet size and Duty Cycles on energy consumption of the sensor nodes. The results earned from the two environments well matched to each other.

8. REFERENCES

1. Romer, K. and Mattern, F., "The design space of wireless sensor networks", *IEEE Wireless Communications*, Vol. 11, No. 6, (2004), 54-61.
2. Rawat, P., Singh, K.D., Chaouchi, H. and Bonnin, J.M., "Wireless sensor networks: A survey on recent developments and potential synergies", *The Journal of Supercomputing*, Vol. 68, No. 1, (2014), 1-48.

3. Mokdad, L., Ben-Othman, J., Yahya, B. and Niagne, S., "Performance evaluation tools for qos mac protocol for wireless sensor networks", *Ad Hoc Networks*, Vol. 12, (2014), 86-99.
4. Yadollahzadeh Tabari, M. and Pouyan, A.A., "Misbehavior analysis of ieee 802.11 mac layer in mobile ad hoc network using stochastic reward nets", *International Journal of Communication Systems*, Vol. 30, No. 16, (2017), e3385.
5. Dâmaso, A., Rosa, N. and Maciel, P., "Using coloured petri nets for evaluating the power consumption of wireless sensor networks", *International Journal of Distributed Sensor Networks*, Vol. 10, No. 6, (2014), 423537.
6. Dâmaso, A., Freitas, D., Rosa, N., Silva, B. and Maciel, P., "Evaluating the power consumption of wireless sensor network applications using models", *Sensors*, Vol. 13, No. 3, (2013), 3473-3500.
7. Verma, A., Singh, M., Singh, J.P. and Kumar, P., "Survey of mac protocol for wireless sensor networks", in 2015 Second International Conference on Advances in Computing and Communication Engineering, IEEE., (2015), 92-97.
8. Trinh, G., Le, K., Bang, T., Tram, Q., Bui, T. and Quan, T., "Modelling and congestion detection of wireless sensor networks: A concurrent-based approach using coloured petri nets", *International Journal of Applied Information Systems*, Vol. 11, No. 7, (2016), 1-9.
9. Zairi, S., Mezni, A. and Zouari, B., "Formal approach for modeling, verification and performance analysis of wireless sensors network", in International Conference on Wired/Wireless Internet Communication, Springer., (2015), 381-395.
10. Gupta, N., "Analytical approach towards available bandwidth estimation in wireless ad hoc networks", *Wireless Networks*, (2020), 1-26.
11. Azgomi, M.A. and Khalili, A., "Performance evaluation of sensor medium access control protocol using coloured petri nets", *Electronic Notes in Theoretical Computer Science*, Vol. 242, No. 2, (2009), 31-42.
12. Tabari, M.Y. and Pouyan, A.A., "An sm based approach for performance evaluation of network layer in mobile ad hoc networks", *Information Systems & Telecommunication*, Vol., No. 228.
13. Cabezas, A.C. and Patron, J.d.J.S., "A performance evaluation of the smac protocol", *ARN Journal of Engineering and Applied Sciences*, Vol. 12, No. 1, (2017), 41-46.
14. Ye, W., Heidemann, J. and Estrin, D., "Medium access control with coordinated adaptive sleeping for wireless sensor networks", *IEEE/ACM Transactions on Networking*, Vol. 12, No. 3, (2004), 493-506.
15. Samant, T., "Analysis and comparison of smac and tmac protocol for energy efficient dynamic topology in sensor network", *International Journal of Electrical and Computer Engineering*, Vol. 6, No. 5, (2016), 2331.
16. Ye, W., Heidemann, J. and Estrin, D., "An energy-efficient mac protocol for wireless sensor networks", in Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE. Vol. 3, (2002), 1567-1576.
17. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E., "Wireless sensor networks: A survey", *Computer Networks*, Vol. 38, No. 4, (2002), 393-422.
18. Yang, O. and Heinzelman, W., "Modeling and performance analysis for duty-cycled mac protocols with applications to s-mac and x-mac", *IEEE Transactions on Mobile Computing*, Vol. 11, No. 6, (2012), 905-921.
19. Kochhar, A., Kaur, P., Singh, P. and Sharma, S., "Protocols for wireless sensor networks: A survey", *Journal of Telecommunications and Information Technology*, (2018). DOI: 10.26636/jtit.2018.117417
20. Agarwal, S., Jain, V. and Goswami, K., "Energy efficient mac protocols for wireless sensor network", *International Journal on Computational Sciences & Applications (IJCSA)*, Vol. 4, No. 1, (2014), 153-160.
21. Demirkol, I., Ersoy, C. and Alagoz, F., "Mac protocols for wireless sensor networks: A survey", *IEEE Communications Magazine*, Vol. 44, No. 4, (2006), 115-121.

Persian Abstract

چکیده

یکی از ویژگی‌های منحصر به فرد شبکه‌های حسگر بی‌سیم این است که گره‌های آنها منابع انرژی بسیار محدود و معمولاً غیرقابل جایگزینی دارند. پروتکل S-MAC در این شبکه‌ها که در لایه کنترل دسترسی رسانه به منظور جلوگیری از اتلاف انرژی در این شبکه‌ها طراحی شده است. با توجه به اهمیت این پروتکل در عملکرد شبکه‌های حسگر بی‌سیم در این مقاله به ارائه یک مدل تحلیلی بر اساس شبکه پتری تصادفی تعمیم‌یافته برای ارزیابی میزان مصرف انرژی گره‌ها در یک شبکه حسگر بی‌سیم مبتنی بر S-MAC پرداخته شده است. مدل پتری ارائه شده در نرم افزار PIPE پیاده سازی شد و با اعمال محاسبات ریاضی از خروجی مدل، موفق به استخراج روابط مربوط به محاسبه میزان مصرف انرژی از آن شده ایم. اعتبار مدل ارائه شده با پیاده سازی سناریوهای مشابه در شبیه ساز Castalia و مقایسه میزان انرژی مصرفی در آن اندازه گیری شده است. نتایج حاصل از مدل ارائه شده بصورت پایدار، در زمان بسیار کمتری نسبت به شبیه ساز استخراج شده و در اکثر موارد با آن منطبق می باشد.
