Modified Adaptive Mechanism For Optimising IEEE 802.15.4 WPANs For Wireless Sensor Networks

Sukhvinder Singh Bamber (ss.bamber@pu.ac.in)
UIET, PUSSGRC Hoshiarpur https://orcid.org/0000-0002-1749-2940
Naveen Dogra
UIET,PUSSGRC Hoshiarpur
Mohit Angurala
Chitkara University

Research Article

Keywords: Access control, GTS mechanism, Wireless personal area network, Wireless sensor networks

Posted Date: August 10th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-755827/v1

Abstract

Different applications of Wireless Sensor Networks (WSNs) have different expectations from the working of Medium Access Control (MAC) protocols. Some value reliability more than delay incurred while some demand a fair trade-off for the factors like: Throughput, Bit Error Rate (BER) etc. This paper evaluates the performance of Wireless Personal Area Networks (WPANs) from 802.15.4 group for WSNs with modified algorithm which helps in reducing the Medium Access Delay and Delay in reaching of the packet from one end to another end. In this paper certain modifications to existing algorithm have been proposed for reducing the Medium Access Delay and to reduce the number of packets dropped. The result comparisons on the performance parameters like: network output load, generated acknowledged traffic, media access delay, battery consumed and delay in packet transmission from one end to another end that the back-off number & exponent values used for transmission play vital role for improving the performance of WSNs as they directly affect the number of packets dropped, successfully acknowledged and Medium Access Delay.

Introduction

WPANs are becoming part and parcel of our life. They find usage in home automation, industrial sensor and control networks, smart grids, environment sensing, and in internet of things to name a few. These applications generally have numerous embedded devices running on batteries and communicating with each other via wireless networks. These hundreds or thousands of nodes work continuously to monitor and gather data, but it's quite cumbersome to change batteries every now and then. So, they require low battery consumption. Also, unlike WLANs which provide more bandwidth for better throughput and low delay for file and multimedia transfers, WPANs need lower data rate of order of few kbps.

The working group of IEEE for WPANS defines the physical layer as well as and medium access layer for WPANs with low power consumption & lower cost. Firstly, the physical layer - It performs modulation and demodulation on outgoing and incoming signals and above all performs hop to hop communication; Secondly, the MAC layer helps to access the medium/network by using CSMA/CA technique in order to provide reliable transmission. In this research work the beacon enabled mode of the protocol for WSN enabled IEEE 802.15.4 WPANs has been used where nodes are synchronised with a coordinator for a smother communication. Numerous studies have been done in order to increase efficiency for better performance and low energy usage, it has been seen that performance metrics like dropped packets, Medium Access Delay, battery consumed, successfully acknowledged packets and delay (end to end), the back-off number, back-off exponent, Superframe Order (SO) and Beacon Order (BO) values used for transmission directly affect performance of the network. These metrics have been considered for boosting the performance of WPANs.

Many algorithms for efficient functioning of WSNs have been proposed in the past. [1] used PEDAMACS is a Time Division Multiple Access (TDMA) scheme which is an extension of common single hop TDMA to a multi-hop sensor network that to manage transmission and reception in WSNs. In [2] the simulation and analytical results are compared to emphasise on the value of SO for efficient GTS allocation strategies and lower delays.

The KEB algorithm in [4] increases or decreases back-off exponent depending upon level of collision relative to a pre-determined value. The scheme for avoiding collision of packets proposed in [5] using DBA offers suitable back-off period. The longer the back-off period more the energy consumption but shorter it is more is the collision. To resolve the trade off and to reduce randomness, the DBA coordinator assigns different tailored back-off to different nodes competing for medium. The authors in [7] propose an efficient CSMA-CA algorithm EBA which modifies contention window size in accordance with the probability of collision parameter. This parameter depends on the number of nodes in network and divides the back-off periods for maximum utilisation.

DSAA in [6] adjusts duty cycle and super frame order dynamically depending upon collision probability and channel availability. [8] also talks about the optimum (SO, BO) pair for analysis of performance of IEEE 802.15.4 duty cycles. DBSAA [10] changes both the SO and BO concurrently giving better results. The role of values of SO and BO along with the nature of traffic flow is considered to propose Adaptive duty cycle algorithm (ADCA) in [11]. In the ABSD algorithm proposed in [12], the coordinator predicts trends in incoming traffic and calculates a variation rate in the traffic flow. It then calculates values of BO and SO accordingly. SUDAS scheme [13] allots adjustable GTS slot length depending on packet size in order to optimise bandwidth utilisation in contention free period. Also, other nodes which fail to have had GTS access can use length of CAP for packet transmission. The authors of [14] also worked on similar idea of modifying SO and BO values based on certain parameters but it assumes that GTS is not used. [16] proposed an adaptive MAC-frame payload tuning technique for wireless body area networks (WBANs) to exploit the ratio of successfully delivery of data packets based on real-time analysis. [17] proposed an algorithm known as network equivalent adaptive parameter tuning (NEAPT) which is of distributed type and works without any predefined information or acknowledgement. Further, in [18], the authors proposed a software-defined WSN architecture (Soft-WSN) for supporting application-oriented service provisioning in Internet of Things. Authors in [19] investigated the Hybrid Directional CR-MAC based on Q-Learning with Directional Power Control in cognitive radio (CR) systems in which the authors highlighted that the channel utilization is done in more appropriate manner in comparison to the existing techniques. There are other algorithms also which modified Ad hoc OnDemand distance vector routing protocol (AODV) to improve energy efficiency of WSNs [20–21]. [22] propose a priority-based algorithm for adaptive superframe adjustment and GTS allocation (PASAGA) for the superframe structure to improve goodput and delay in WSNs.

Although all the previous studies provide solid optimisation techniques for working of WSNs but none has worked it in an integrated fashion in which both the media allocation using CSMA-CA algorithm and the (SO, BO) pair value is optimised in a simpler yet effective manner. This lack brings forth a research gap which our study aims to fulfil.

This paper is organised as follows. Firstly, the simulation model used and algorithms used have been explained, then the next section explains the proposed simple adaptive algorithm/mechanism. The next section compares results and followed by conclusion and references. This paper evaluates the performance of original algorithm and compares it with the modified algorithm. Here certain modifications to the available simulation model using slotted CSMA-CA back-off exponent updating algorithm and adaptive SO and BO have been proposed.

Working Of Existing 802.15.4 Algorithm

This protocol is flexible enough and provides the GTS feature to support time sensitive WSN applications. This protocol has two types of devices: FFDs (Fully Functional Devices) &RFDs (Reduced Function Devices). FFDs act as coordinators and are responsible for the communication with the cluster. The RFD shave limited resources relying on the coordinator for communication.

When operating in beaconed mode, beacon frames are sent periodically by the coordinator to synchronise the devices associated with it. The time gap between two consecutive beacons is called Superframe. The Superframe structure comprises of active & inactive time periods [2]. The active period is divided in sixteen time slots of equal sizes and this active period corresponds to the Superframe Duration (SD) during which transmission occurs. Each active period is further split into CAP (Contention Access Period) &CFP (Contention Free Period) consisting of GTSs. The CAP implements slotted CSMA/CA protocol. Further, the Superframe structure is defined with two parameters: BO (Beacon Order) and the SO (Superframe Order), which determines the length of the Superframe and its active period respectively. Also, BO and SO must satisfy the relationship $0 \leq SO \leq BO \leq 14$. Superframe length is equal to Beacon Interval (BI) and length of its active period equals to the Superframe Duration (SD)which can be defined as follows:

$BI = SuperframeDuration_{*}2^{BO}$ (1)

$SD = SuperframeDuration_{*}2^{SO}$ (2)

In CAP, CSMA/CA (slotted) works by initializing the three variables:

- Contention Window Size (CW = 2 at the time of initialization and each time the channel is found to be busy).
- Back-off Stages (NB = 0).
- Back-off Exponent (BE set to standard parametric value: minBE).

Then the node that wants to transmit data, introduces the delay for a random Back-off Period (BP) chosen from the range: [0, 2^BE – 1] slots. After the BP has expired, node starts sensing the channel in the form of CCA1 (Clear Channel Assessment). If the channel is free, then again, the channel is accessed for its idleness in the form of CCA2. Now at this stage, if the channel is still idle, then the node starts transmitting the data and waits for the acknowledgement from the coordinator. But if after CCA1 or CCA2 the channel is found busy, then increase the values of BE & NB by 1. On the other hand, the maximum values that NB & BE can have are: maxCSMABack-offs & aMaxBE respectively (both values are assigned manually). Now if the BE reaches its maximum value, then BE is again reassigned the value in the range: [0, 2BE + 1–1]. Also, if the NB reaches its maximum value, then the data transmission fails and the packets are discarded.

2.1 Suggested Changes to CSMA/CA Mechanism:

CSMA/CA (slotted) makes use of random Back-off delay in the form of BP before transmitting a data packet on the medium. BP is implemented by the use of Inter Frame Space (IFS) time slots. This is done to ensure that the nodes that are almost equidistant from the coordinator get ample time to ascertain that the channel is still idle and no collisions take place. In other words, the IFS time slots allow the nodes that are at the distant locations to resolve the collision conflicts with the nodes that are nearer to the coordinator. Now after this IFS time slot if the channel is still idle, even then the end device waits for the time slot equal to CW. Finally, if the channel is found to be idle then the end device transmits. But IFS time slot (variable) can be used for ascertaining the priority of transmitting station or the data frame to be transmitted.

Next, if number of retries for a packet to be sent by a node increases beyond 2, then it needs to access the channel more vigorously. BE is reduced, so that the minimum time slots to wait, reduces in value and it can access the channel quicker than other nodes contesting for the channel. The modified algorithm as explained below and shown in Fig. 2 can be used for optimising IEEE 802.15.4 WPANs for WSNs.

Implementation of Algorithm:

//Modified Slotted CSMA/CA

1. def NB = 0 // initialise Back-off Number
2. def CW = 2 // initialise Contention Window
3. def BE = minBE // initialise Back-off Exponent to its minimum value
4. def IFS = get(beacon_MSDU); // get inter frame space
5. def back-off_time
6. defback-off_max;
7. back-off_max = 2^*BE*-1
8. back-off_time = (double) back-off_unit *aUnitBack-offPeriod*WPAN_DATA_RATE
9. back-off_time = back-off_time + ifs_time; // calculate back-off time
10. Perform CCA
11. if channel NOT idle{

CW = 2,

NB = NB + 1,

check if (2 > = number of retries > = 5)

/*prioritise the packet with more retries that is if number of retries is more than two but less than five the algorithm will give more priority for this transmission by assigning lesser back off exponent */

{

BE = minimum of (BE, maxBE-1);

/* assign less back-off exponent to packet with higher number of retransmissions so that random delay is lower */ }

else

BE = minimum of (BE + 1, maxBE);

/* if retries are lower or have exceeded five, the original values for back off exponent and its maximum value are assigned */

check if NB < = MAX_BACK-OFF

redo another back-off go to step 5 }

12. If channel idle
CW = CW-1 // reduce the contention window by one if channel is idle.

If CW > 0 goto 10 // perform CCA again if contention window is greater than zero.
Else

Report success

2.2 Modified Adaptive Superframe Order (SO) Value:

As can be observed from (1) and (2), the SO and BO values play key role is the beacon enabled networks. In the original algorithm [8] these values are constant. But the real time situations demand the delay to be less and reduced bandwidth wastage. So as to improve the performance, modifications have been proposed in the existing algorithm. Modified algorithm considers the updating of SO based on the arrival data rate. The arrival Data Rate (R) is calculated as given by the equation below:

R = MSDU (Maximum Size Data Unit) size/ MSDU inter arrival time (bps) (3)

Based on the observation the value of arrival data rate is used to determine the SO while keeping other parameters constant. It is observed that keeping r equal to 200 bps as threshold one can change the values of SO and BO. If the arrival rate increases the threshold, the SO is incremented by 1 and vice versa.

Algorithm Implementation:

// update the Superframe parameters

def *IFS* = get(beacon_MSDU); //get Inter Frame Space (IFS)

def *Frame_Len* = get(beacon_MSDU);

def *Rate* = *Frame_Len*/ *IFS*; // get data rate

check if *Rate* > = threshold && SO < = BO

/* condition to check of rate is less than the threshold of 200bps and SO is less than BO.

{

SF.SD = *SuperframeDuration* $*2^{SO+1}$

/* increase Superframe Duration when traffic increases so to allocate more resourcesby increasing the active period*/

SF.BI = *SuperframeDuration* $*2^{BO}$

}

else if *Rate* < = threshold and SO < = BO

/* if the rate is more than the threshold and SO is less than BO and thus decreases the active period.*/

{

SF.SD = *SuperframeDuration* $*2^{SO-1}$

/*decrease Superframe Duration when traffic decreases*/

SF.BI = *SuperframeDuration* $*2^{BO}$ }

Simulation Model Structure

The basic setup is common to all scenarios is shown in the Fig. 4. It consists of 20 End Devices or sensor nodes. These only communicate with the PAN Coordinator (PANC) at the centre which is the bridge between different nodes which need to communicate. PANC broadcasts beacon frames at fixed intervals to synchronise all nodes for communication. The nodes listen to the beacons and then apply slotted CSMA/CA algorithm for medium access. In simulation, the size of buffer is fixed at the default value of 1000 bits. The MSDU size is kept same in same distribution (exponential) with same mean value of 400 bits and the IFS is also kept in same distribution (exponential) range with a mean of 2 seconds.

The nodes wait for beacons from PANC. Once it receives a beacon it tries to access the channel after waiting for random Back-off Period & performing CCA. i.e. it applies the CSMA/CA (slotted) algorithm for channel access in first scenario and the modified algorithms in the following scenarios.

Slotted CSMA/CA algorithm is modified in the next scenario with an adaptive BE. Keeping other parameters same, the contention window and back off exponent are made adaptive in accordance with the number of retries that a packet undergoes. If the retries lie between two and five then they are assigned a higher priority by decreasing their wait time.

In the third scenario, the SO and BO are made adaptive to the Data Rate flow and Modified Adaptive BE-SF algorithm is implemented. Here, the incoming Data Rate (R)is taken into consideration and a threshold is chosen. If Rate increases beyond the threshold, then more bandwidth is assigned and vice versa.

In nutshell, the three scenarios are taken into consideration as shown in Table 1. The first scenario uses the original code without any changes. The second scenario uses only the adaptive Back-off Exponent (BE) algorithm. The third scenario takes into account the Superframe updating changed algorithm as well as the new CSMA-CA BE proposed algorithm.

Table 1
Various scenarios and their description

Scenario	Description
Scenario 1	Original Algorithm
Scenario 2	Adaptive BE Algorithm
Scenario 3	Modified Adaptive BE-SF Algorithm

Results And Discussion

Here implementation of 802.15.4 (IEEE) WPANs for WSNs are explained. Performance of mechanism is compared with the model derived in [1]. It is based on (b, R) model which considers the linear arrival curve created by sensor nodes with GTS traffic. The cumulative arrival curve from application layer is upper bounded by a (t) = b + R*t where 'b' is the maximum burst size and 'R' denotes average arrival rate. The results and equations obtained from the simulations are compared with numerical model [1].

4.1 Medium Access Delay is the extra Back-off time that the packet at MAC layer has to wait before being successfully transmitted or dropped after number of failed retransmissions in a specified time interval.

Figure 5 displays the medium access delay for three scenarios. The maximum average delay was shown by the scenario using original algorithm and is found to be 12.7 ms and least was shown by simple adaptive BE-SF algorithm and came out to be 5.65 ms. Adaptive BE algorithm has average Medium Access Delay equal to 9.59 ms. This implies that the proposed algorithm helps reduce the Medium Access Delay by forty five percent approximately. According to the numerical model Ts given by (T_{data} +

T_{idle}) denotes the time slot duration in Superframe with SD and BI as Superframe duration and beacon interval then delay latency is given by

D_{max}= (K * BI / μ*SD- T_{idle}) + (BI-T_{data}- T_{idle}) (4)

Where D_{max} is maximum delay, K is a constant, T_{data} is the maximum duration for transmission of frame (data) inside GTS and T_{idle} is combination of IFS, acknowledgement time and other overheads.

Since the transaction has to complete before the GTS duration ends; So when SO and BO are made adaptive the bandwidth usage becomes better. The Superframe duration increases after certain threshold of arrival data rate. The CSMA algorithm reduces wait time (BI – Tdata – Tidle). The reason for this can be attributed to the fact that Medium Access Delay depends on the random BE which increases delay factor as the number of retries increase. But our algorithm prioritises the packets having retries more than two and allots them a lower BE until their retries become five. Thus, they are provided access faster thereby reducing the overall Medium Access Delay. This results in efficient use of CFP of the GTS. Now removing the wait time part from the equation we find:

D_{max} = (K * BI / SD – T_{idle}) (5)

Since SD changes at run time and keeping BI same, more SD will result in lesser delay and hence the graphs obtained. The approximation equations obtained for the resultant graphs were as follows:

d = 0.0074*t + 4E-05,R^2 = 0.7275 (6)

d = 0.0061*t − 0.001, R^2 = 0.7344 (7)

d = 0.0052*t − 0.002,R^2 = 0.7381 (8)

Generalizing the above (6–8) equations, ignoring k2and replacing t from the equation of arrival curve (a = b + R*t), where b is maximum burst size and R is average arrival data rate, we get:

(K * BI / SD – Tidle) = k1*(a-b)/R (9)

This leads us to the conclusion that SD – T_{idle} α Rate (R). So as the rate increases if SD also increases, it contributes in giving better results and hence lower delays.

4.2 End to End Delay: It is the extra time taken (excluding the specified time period) by the packet to be transmitted from the original source to original destination. It depends on transmission, propagation, processing and packetization delays. It comprises of sender delay, network delay and receiver delay.

Figure 6depictsDelay (End to End) of three scenarios. The least Delay (End to End) is in Modified Adaptive BE-SF Algorithm. It is observed that Simple Adaptive BE-SF Algorithm reduces the Delay (End to End) by about three percent. The reasons for this can be estimated from (9) which shows that as the rate increases if SD also increases it mitigates the delay. The difference is less as compared to that in Medium Access Delay as End-to-End Delay is governed by other factors as given by following relation:

$d_{end\text{-}end}$= N[d_{trans}+d_{prop}+d_{proc}] (10)

where:

$d_{end\text{-}end}$= Delay (end-to-end)

d_{trans}= Delay (transmission)

d_{prop}= Delay (propagation)

d_{proc}= Delay (processing)

N = Number of links.

Since propagation delay is majorly dependent on bandwidth which is 2.4 GHz for all the three scenarios and it depends on material medium of travel. Next is the transmission delay which may be defined as: d_{trans} = packet_length/data_rate, with data rate being constant at 250kbps for 2,4GHz frequency band. Since packet length and data rate are same for all scenarios, we come to processing delay. The d_{proc} involves checking packet headers for errors and to check the next destination address for packet, accessing the medium etc. In this way the difference in Delay (End to End) can be explained.

Similar results are also given by the approximated equations obtained from simulation results as plotted on the graph as well. The equations (11–13) are respectively for three scenarios:

$y = -0.00003x^2 + 0.6029x - 1.9079, R^2 = 0.7531$ (11)

$y = 0.5031x - 1.5882, R^2 = 0.7545$ (12)

$y = 0.4886x - 1.5458, R^2 = 0.7531$ (13)

Generalizing these equations (11–13) as in (9) simulation results have a very similar behavior as in case of analytical model.

4.3 Successfully Acknowledged Packets: Those packets out of the total sent towards the destination that successfully reach the destination and are acknowledged back to the source.

Figure 7 represents the successfully acknowledged packets. Maximum packets in numbers are successfully acknowledged in case of Simple Adaptive BE-SF Algorithm. The graph result shows an increase of thirty six percent (approx.) in number of packets successfully acknowledged in the same time frame.

The maximum packets in numbers that can be transmitted are be given by:

$Np = T_{data} + T_{idle} / Ps + IFS$ (14)

Where Np is number of packets sent and Ps is maximum packet size and IFS being inter frame space.

Since IFS and Ps are constants for three scenarios then Np depends directly on Ts (Tdata + Tidle). This value is least in simple adaptive BE-SF algorithm as delay for this scenario is least as proved in (9).

The approximation for the simulation results give following variation for the packets successfully transmitted for three scenarios respectively.

$p = 0.1341t^2 - 1.1504t + 3.2393$ $R^2 = 0.9345$ (15)

$p = 0.0916t^2 - 0.7139t + 1.8196$ $R^2 = 0.9172$ (16)

$p = 0.0954t^2 - 0.9147t + 2.6661$ $R^2 = 0.8464$ (17)

where ‘p’ denotes number of packets transmitted successfully and ‘t’ is the time elapsed.

The results in Fig. 7 and the approximated equations are in congruence with the numerical analysis and give better results for adaptive BE-SF Algorithm.This observation can be attributed to the fact that the changed CSMA-CA algorithm gives higher priority to the node which has tried twice or more than twice to transmit the packet but has failed to do so. Also, it ensures a smaller number of collisions and hence more acknowledged packets in same time duration by waiting for IFS time in addition to back-off-delay. This ensure that the nodes at the same distance from coordinator may sense the carrier to be idle when one of the node has already sent a packet. By waiting for IFS time, it is ensured that the packet gets successfully delivered to destination.

4.4 Dropped Packets: Number of packets that are dropped (those which are unable to reach the destination) from the total sent to the destination due to unavailability of medium or on exceeding the permitted number of retries.

The Fig. 8 above shows the dropped packets. The maximum number of packets were dropped in case of scenario 1 with original algorithm in play. The absolute number of packets dropped in original algorithm were two whereas in Modified Adaptive BE-SF Algorithm there were none. This observation can be attributed to the fact that the revised Simple Adaptive BE-SF algorithm avoids collision in a better way and prioritises the access by giving the access to node which has failed twice or more to send the data packet. Since the delay for channel access is reduced in new scheme the probability of packet drop reduces. Also, the Superframe duration is adjusted according to the incoming traffic which ensures proper utilisation of bandwidth.

Probability of packet drop α Medium Access Delay (18)

Since from (9) the Medium Access Delay is least for Simple Adaptive BE-SF algorithm and hence from (5) probability of packet drop is also least. The same results are also seen in the approximate equations from the simulations:

$y = 0.0303x + 0.0902$, $R^2 = 0.7741$ (19)

$y = 0.0183x - 0.0761$, $R^2 = 0.6463$ (20)

$y = 0$ (21)

with Eq. (19–21) for three scenarios respectively.

4.5 Battery Energy Consumed: The amount of energy consumed by the particular node for the transmission or reception.

From the Fig. 9 above it is observed that maximum battery consumption is in case of original algorithm and least in case of Simple Adaptive BE-SF Algorithm.

It should be noted that a low duty cycle conserves energy by putting device to sleep. But, a low value of duty cycle also reduces the bandwidth and increases delay. So, a fine balance of SO and BO values is needed which decide the duty cycle. The adaptive algorithm tweaks the value of SD in real time and increased bandwidth utilization.

Also in simpler terms the equation for energy consumption may be given by:

Energy = Power * Time (22)

The 'Power' includes the transmission or reception power and 'Time' represents the 'Ts'.

In (9), it was found that the 'Ts' is least in Modified Adaptive BE-SF Algorithm. Since all motes considered are same, so their power values are also same. This leads us to an obvious conclusion of the energy consumption pattern.

The results obtained on running regression analysis of the data for battery consumption we get following best fit mathematical equations:

$e = 3E\text{-}05t^2 - 0.0001t + 0.0001$, $R^2 = 92\%$ (23)

$e = 4E\text{-}06t^2 + 8E\text{-}06t - 2E\text{-}05$, $R^2 = 95\%$ (24)

$e = 3E\text{-}06t^2 + 3E\text{-}05t - 5E\text{-}05$, $R^2 = 90\%$ (25)

where 'e' is the energy consumed and 't' is the time elapsed in seconds.

As the time elapses the data rate arrival increases thus increasing the battery consumption. This result is in accordance with the numerical analysis of the model. The equations (23–25) also show that on a longer period of time the best performance shall be given by Modified Adaptive BE-SF Algorithm which is in harmony with the numerical analysis as well.

Conclusion

It is concluded from this research that the Simple Adaptive BE-SF Algorithm makes the network better performing. Prioritising of the nodes with higher retries and waiting for IFS time reduces the overall delays. The average medium access delay is reduced by 7.05 ms and shows a three percent decrease in average delay (end to end). The adaptive SO increases bandwidth utilisation and increases efficiency. The number of packets dropped are nil in modified algorithm. Also, the battery energy consumption difference is significant.

In comparison to original algorithm, the Adaptive BE Algorithm shows improvement with regard to medium access delay, number of packets dropped and acknowledged. It shows a little increase in end-to-end delay but overall consumes less energy than original algorithm.

In case of Modified Adaptive BE-SF Algorithm, battery consumption is least and no packets were dropped. It shows a significant increase in number of packets acknowledged which were nearly twice in number as compared to original one. There is a reduction in medium access delay and a small decrease in delay (end to end) as well.

Although the factors considered in the adaptation of the BE, SO can be expanded but that may add to complexity. Finally, it can be concluded that the Simple Adaptive BE-SF Algorithm can be used in WSNs for better performance.

Declarations

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Code availability

The authors declare the availability of working code of OPNET simulator.

References

1. S. C. Ergen and P. Varaiya, "PEDAMACS: Power efficient and delay aware medium access protocol for sensor networks", *IEEE Trans. Mobile Comput.*, vol. 5, no. 7, pp. 920-930, 2006.
2. P. Jurcik, A. Koubaa, M. Alves, E. Tovar, Z. Hanzalek, “A Simulation Model for the IEEE 802.15.4 Protocol:Delay/Throughput Evaluation of the GTS Mechanism”, MASCOTS 2007, *15th International Symposium*, 24 – 26 Oct. 2007.
3. S. Feng ,S. Taka ,Z. Moshe ,L. Hai, “Packet Loss Analysis of the IEEE 802.15.4 MAC without Acknowledgements,” *IEEE Communications Letters*, vol. 11, no.1, 2007.
4. S. Woo, W. Park, S. Ahn, S. An, and D. Kim, "Knowledge-Based Exponential Back-off Scheme in IEEE 802.15. 4 MAC,” *Information Networking Towards Ubiquitous Networking and Services*, pp. 435-444, 2008.
5. B. H. Lee and H. K. Wu, "Study on a delayed back-off algorithm for IEEE 802.15.4 low-rate wireless personal area networks," *Communications, IET,* vol. 3, pp. 1089-1096, 2009.
6. B. Lee and H. Wu, "Study on a Dynamic Superframe Adjustment Algorithm for IEEE 802.15.4 LR-WPAN," *2010 IEEE 71st Vehicular Technology Conference*, 2010, pp. 1-5, doi: 10.1109/VETECS.2010.5493884.
7. Dahham, Zahraa & Sali, Aduwati & Mohd Ali, Borhanuddin & Jahan, Md. (2012). An efficient CSMA-CA algorithm for IEEE 802.15.4 Wireless Sensor Networks. *2012 International Symposium on Telecommunication Technologies*, ISTT 2012, pp.118-123. 10.1109/ISTT.2012.6481575.
8. F. Despaux, Y. Song, A. Lahmadi, “Measurement based Analysis of the effect of duty cycle in IEEE 802.15.4 MAC performance”, *IEEE International Conference on mobile Ad-hoc and Sensor Systems*, 2013.
9. V.B. Swati, K. Mahesh, P. Raviraj, “Performance Analysis of IEEE 802.15.4,” *International Journal of Advanced Research in Computer Science and Software Engineering*, March 2013.
10. O. H. S. Camila, G-D, Yacine, and L. Stephane, “A duty cycle self- adaptation algorithm for the 802.15. 4 wireless sensor networks”, *In IEEE Tran. On Global Information Infrastructure Symposium,* pp 1-7, Oct. 2013.
11. H. Rasouli, Y. S. Kavian, and H. F. Rashvand, “ADCA: Adaptive Duty Cycle Algorithm for Energy Efficient IEEE 802.15.4 Beacon-Enabled Wireless Sensor Networks,” *IEEE Sensors Journal*, Vol. 14, no. 11, 2014.
12. T. D. Nguyen, J. Y. Khan and D. T. Ngo, "An energy and QoS-aware packet transmission algorithm for IEEE 802.15.4 networks," *2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2015, pp. 1255-1260, doi: 10.1109/PIMRC.2015.7343491.
13. B. Lee, E. Yundra, H. Wu and M. Udin Harun Al Rasyid, "Analysis of superframe duration adjustment scheme for IEEE 802.15.4 networks," *2015 Seventh International Conference on Ubiquitous and Future Networks*, 2015, pp. 355-360, doi: 10.1109/ICUFN.2015.7182564.
14. A. Farhad, S. Farid, Y. Zia and F. B. Hussain, "A delay mitigation dynamic scheduling algorithm for the IEEE 802.15.4 based WPANs," *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, 2016, pp. 1-5, doi: 10.1109/ICCSII.2016.7462430.

15. IEEE 802.15.4 OPNET Simulation Model, available online at: http://www.open-zb.net/.
16. S. Moulik, S. Misra and D. Das, "AT-MAC: Adaptive MAC-frame payload tuning for reliable communication in wireless body area networks", *IEEE Trans. Mobile Comput.*, vol. 16, no. 6, pp. 1516-1529, Jun. 2017.
17. Y. Wang, W. Yang, R. Han, K. You, “A Network Equivalent-Based Algorithm for Adaptive Parameter Tuning in 802.15.4 WSNs,” *Sensors (Basel, Switzerland)*, vol. 18, no. 7, pp. 2031, 2018. https://doi.org/10.3390/s18072031
18. S. Bera, S. Misra, S. K. Roy and M. S. Obaidat, "Soft-WSN: Software-defined WSN management system for IoT applications", *IEEE Syst. J.*, vol. 12, no. 3, pp. 2074-2081, Sep. 2018.
19. A. Carie, M. Li, C. Liu, P. Reddy and W. Jamal, "Hybrid directional CR-MAC based on Q-learning with directional power control", *Future Gener. Comput. Syst.*, vol. 81, pp. 340-347, Apr. 2018.
20. M. Angurala, M. Bala and S. S. Bamber, "Performance Analysis of Modified AODV Routing Protocol With Lifetime Extension of Wireless Sensor Networks," in *IEEE Access*, vol. 8, pp. 10606-10613, 2020, doi: 10.1109/ACCESS.2020.2965329.
21. Angurala, M., Bala, M. & Bamber, S.S. Energy efficient CLB approach to find optimum modulation scheme in wireless rechargeable sensor networks. *Int. j. inf. tecnol.* **13,** 269–276 (2021). https://doi.org/10.1007/s41870-020-00561-2
22. B.-H. Lee, H.-K. Wu and N.-C. Yu, "A priority-based algorithm for adaptive superframe adjustment and GTS allocation (PASAGA) in IEEE 802.15.4 LR-WAN", *Proc. IEEE Int. Conf. Appl. Syst. Invention (ICASI)*, pp. 318-320, Apr. 2018.

Figures

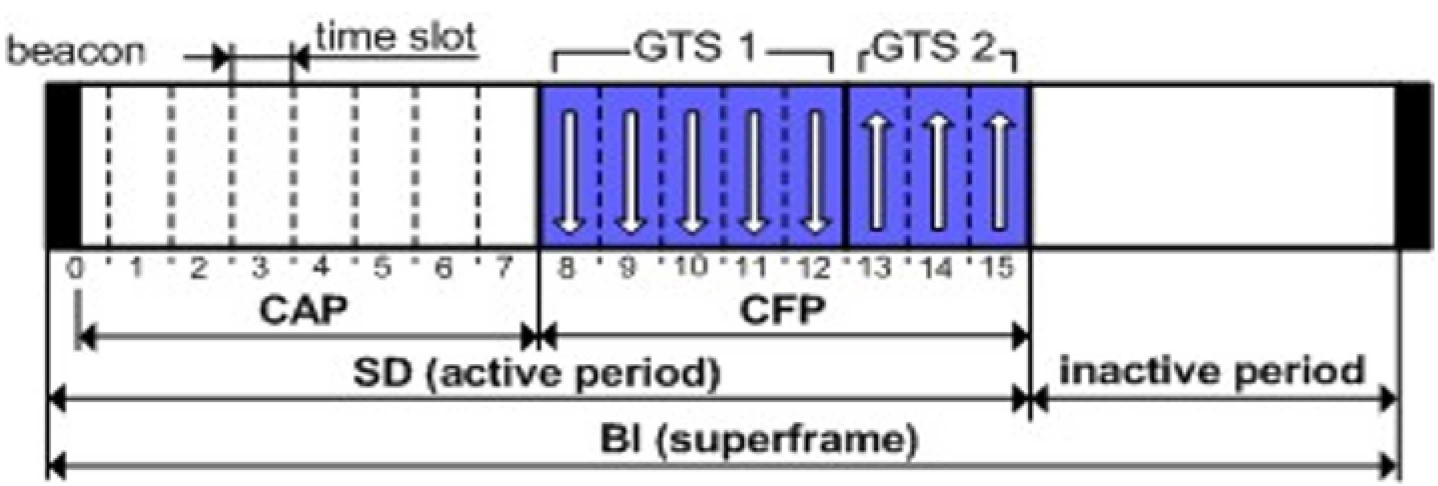

Figure 1

IEEE 802.15.4 Superframe Structure [2].

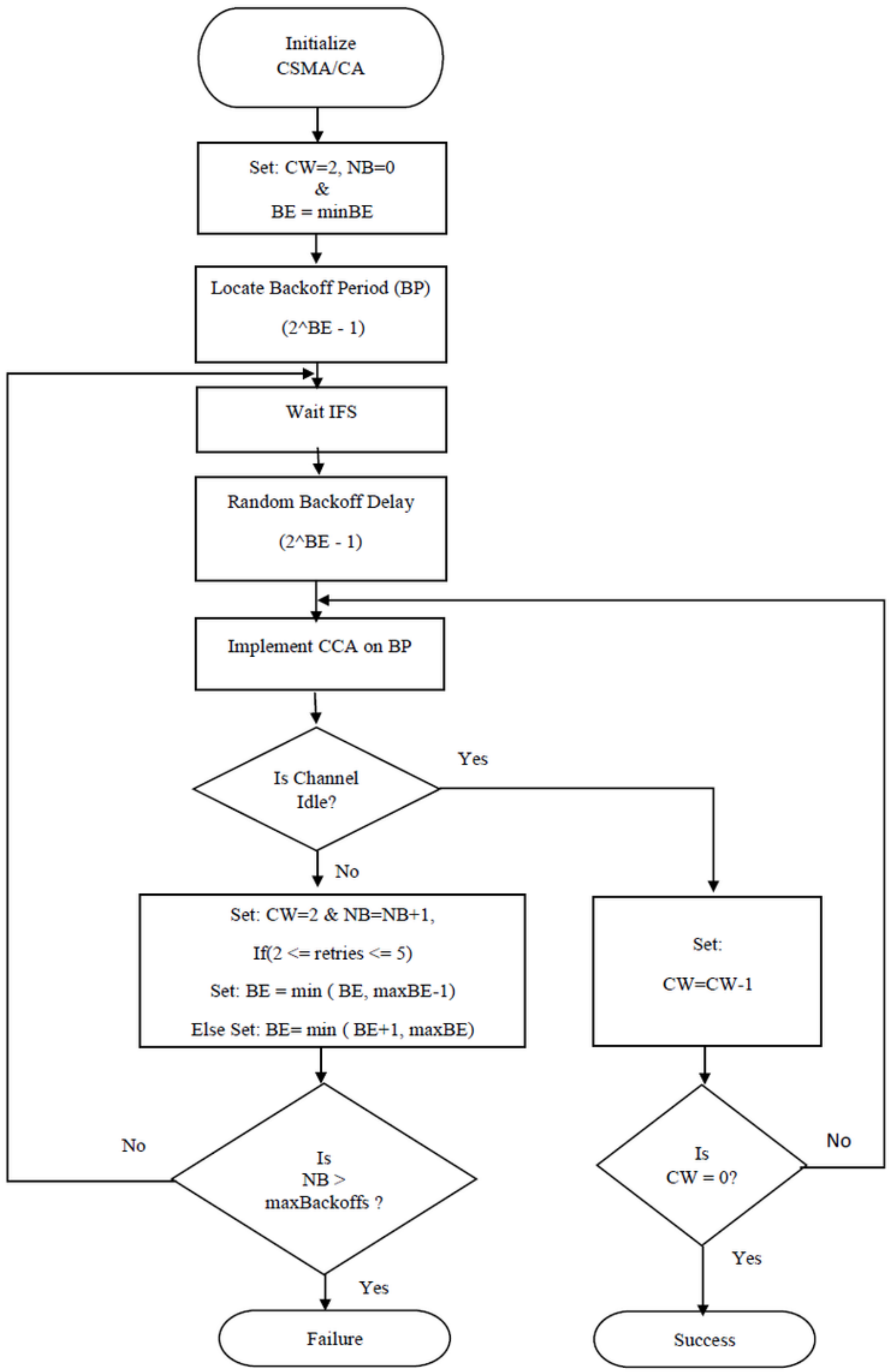

Figure 2

Modified Slotted CSMA/CA Mechanism.

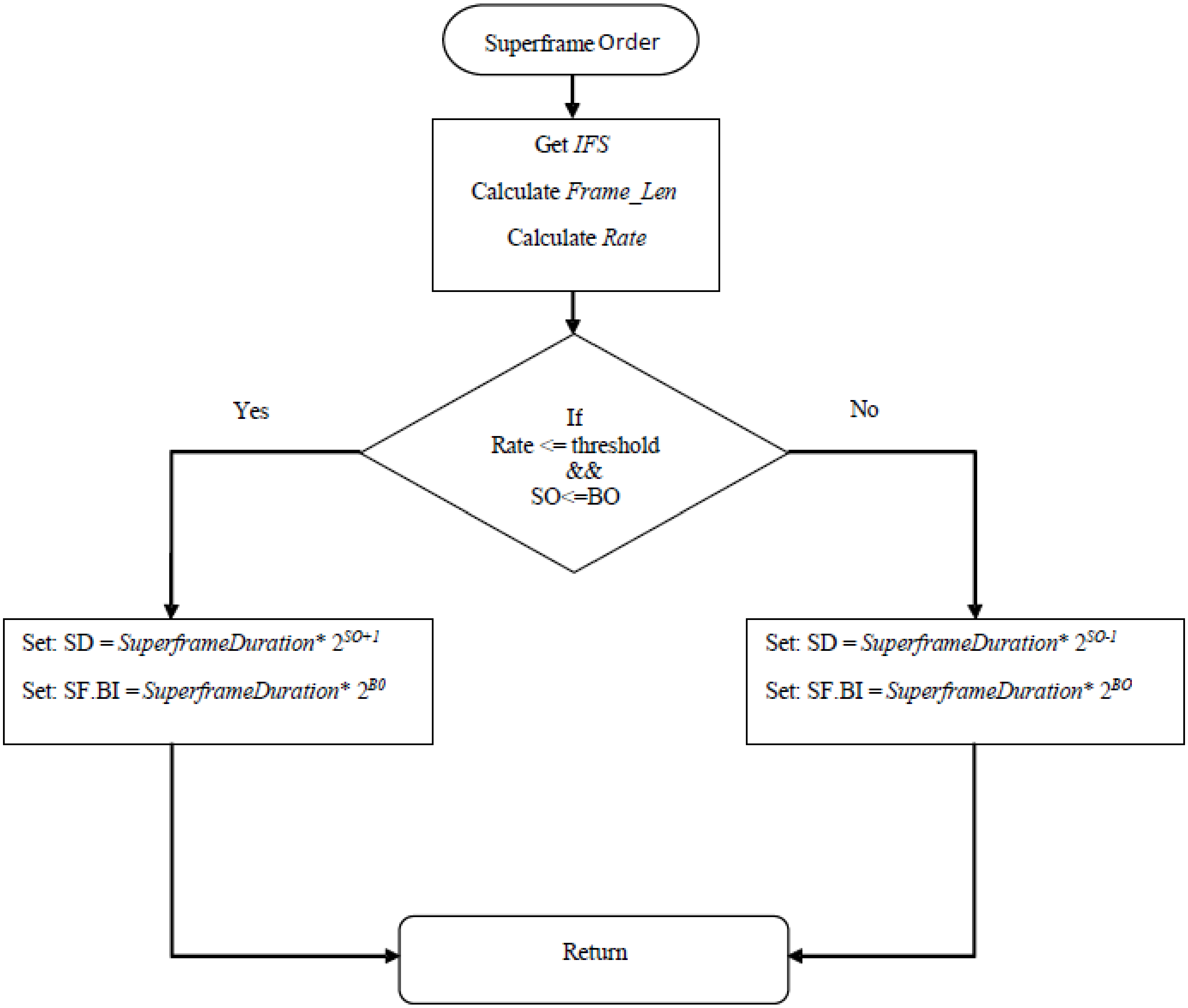

Figure 3

Modified Adaptive Superframe Mechanism.

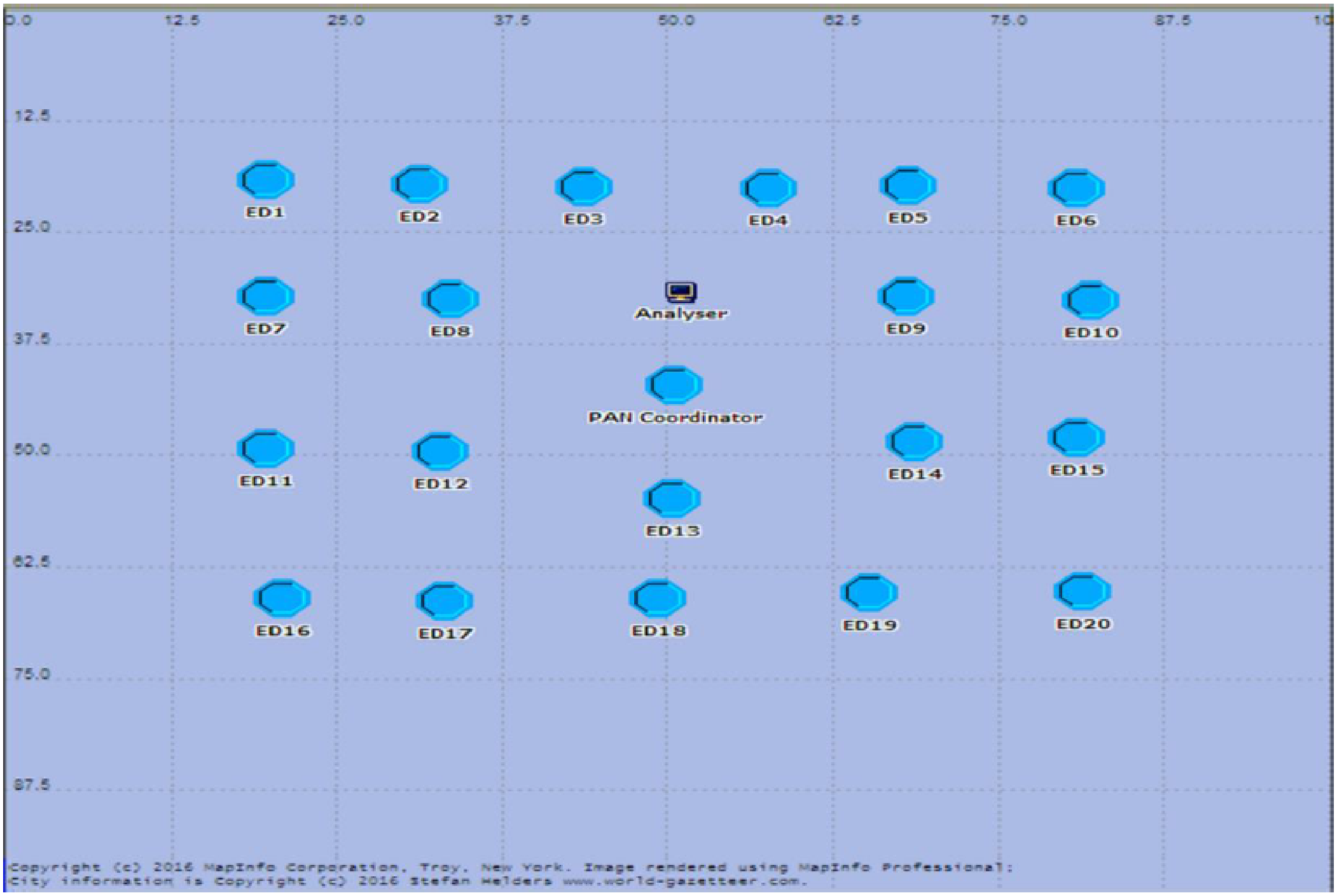

Figure 4

Scenario Setup

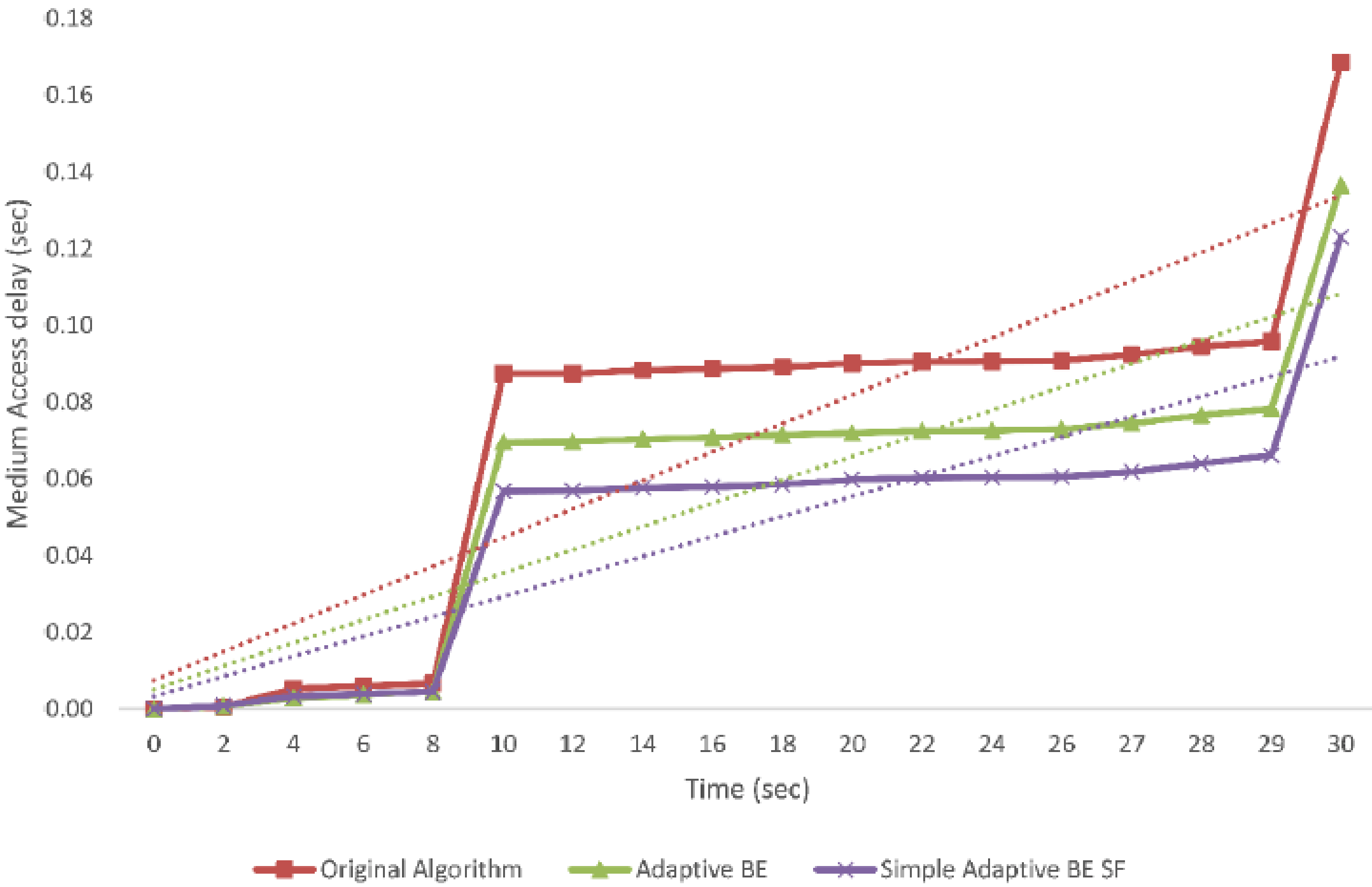

Figure 5

Medium Access Delay.

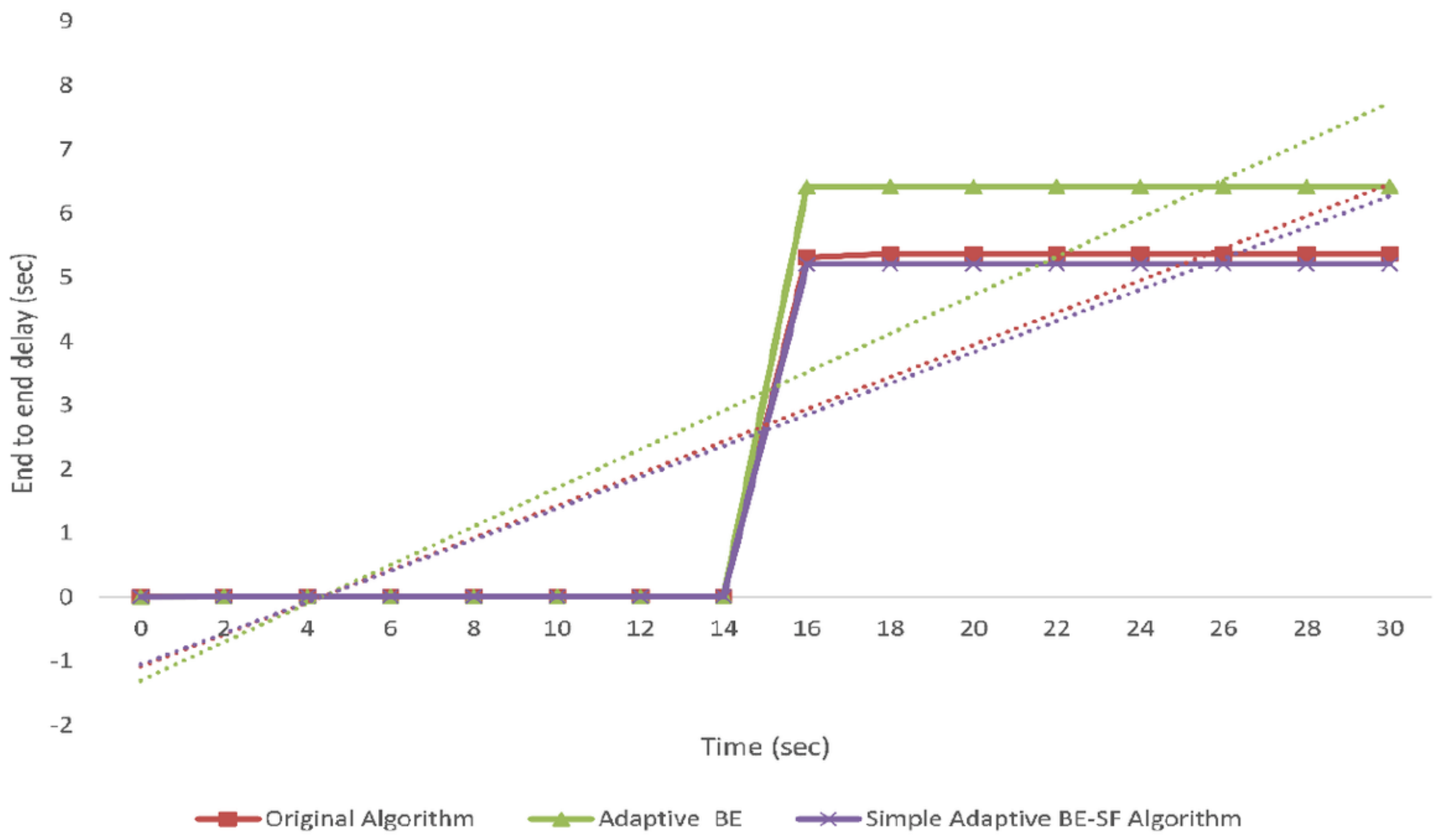

Figure 6

Delay (End to End)

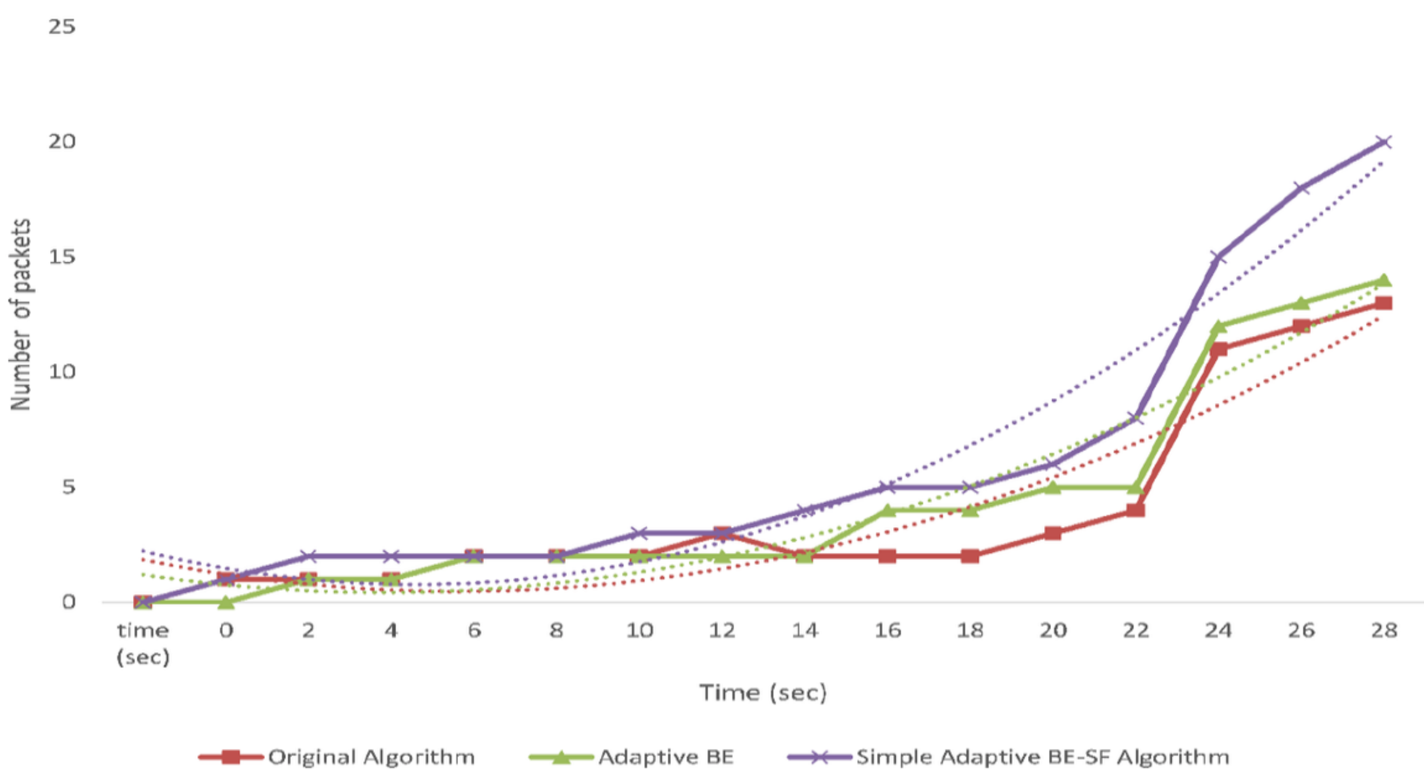

Figure 7

Successfully Acknowledged Packets

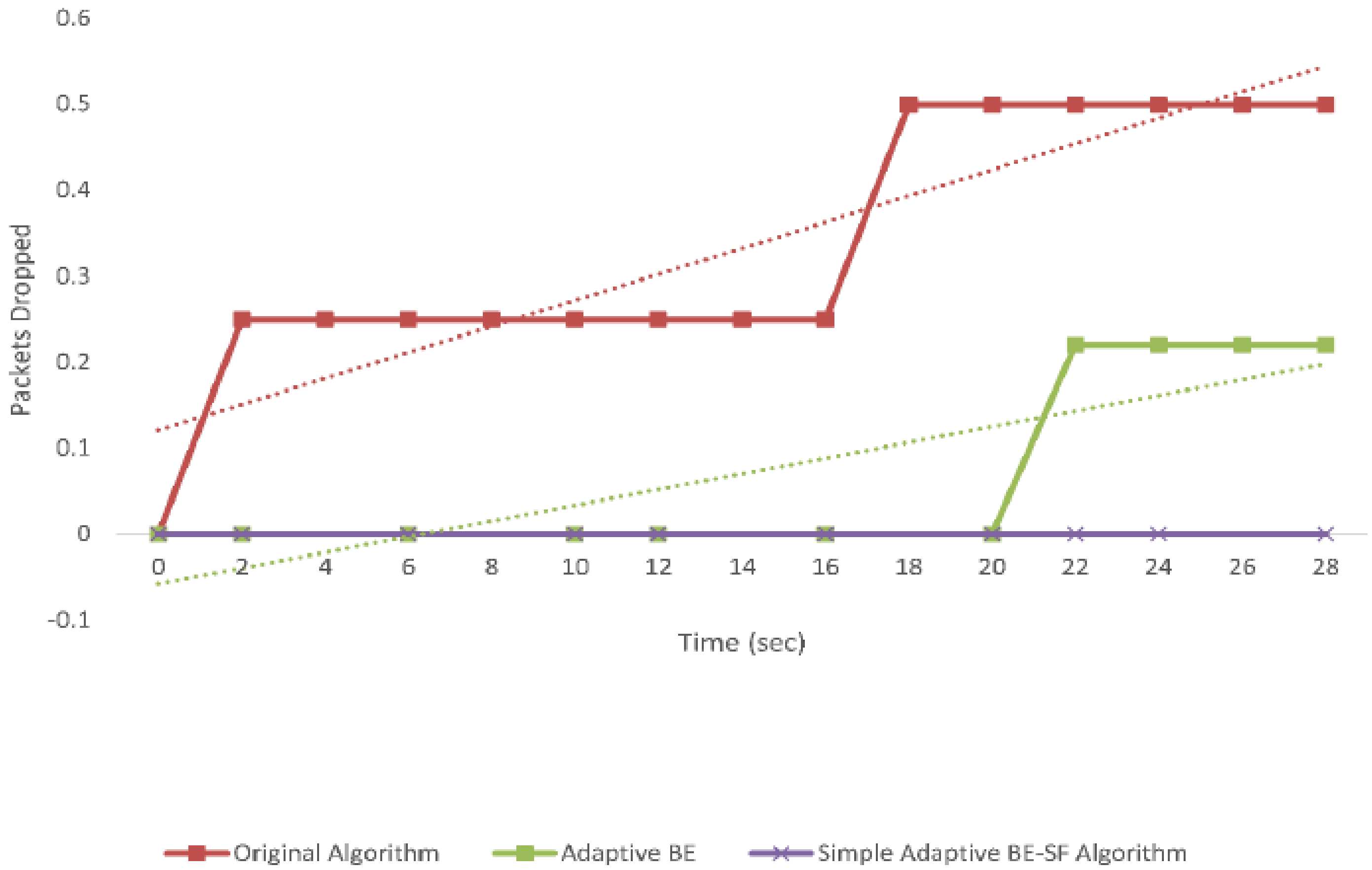

Figure 8

Packets Dropped

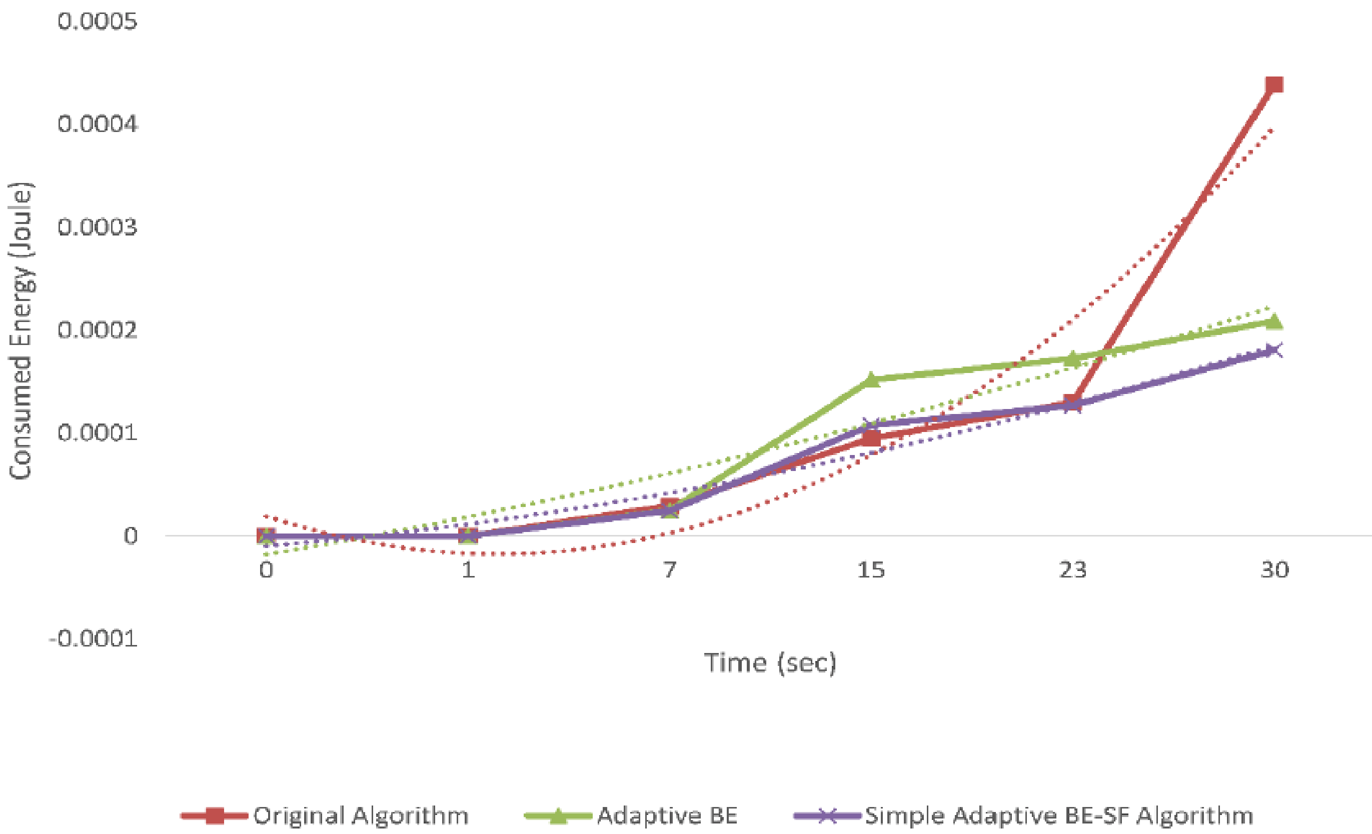

Figure 9

Battery Energy Consumed