

RESEARCH

Open Access



Adaptive Sleep Efficient Hybrid Medium Access Control algorithm for next-generation wireless sensor networks

Sheikh Tahir Bakhsh^{1*}, Rayed AlGhamdi², Abdulrahman H. Altalhi², Sabeen Tahir² and Muhammad Aman Sheikh³

Abstract

The scheduling algorithm is a fundamental design problem to allocate resources amongst different entities in distributive wireless sensor networks (WSNs). These sensor nodes have limited power and non-replenishable energy resources. In WSNs, the duty cycling mechanism is commonly used to save energy due to idle listening. On the other hand, a fixed duty cycling mechanism increases transmission latency in WSNs. Therefore, in order to ensure the prolonged network-life of WSNs, the medium access control (MAC) protocol should be tackled in an efficient manner to improve energy efficiency by minimizing idle listening, maximizing sleep duration, and eliminating data collision. This paper proposes a practical Adaptive Sleep Efficient Hybrid Medium Access Control (AEH-MAC) algorithm in which the key idea is to dynamically adjust nodes' sleep time to improve the scheduling in WSNs. The AEH-MAC allows nodes to adjust sleep time dynamically according to the traffic load and coordinate wakeup time with neighbour nodes. A series of short taken packets are transmitted to wake the receiver, and a prediction field is introduced in the ACK packets (GRANT/RELEASE) to reduce the waiting time of the source node. In the proposed algorithm, each node maps a conflict-free time slot for itself up to two-hop neighbouring nodes. The simulation results show that the AEH-MAC algorithm achieves high performance in terms of runtime, number of rounds, energy consumption, and slot reservation.

Keywords: Ad hoc networks, Dnyamic, Synchronization, Network performance

1 Introduction

Wireless sensor networks for last few decades have become a dynamic research area due to its widely used potential applications such as industrial process monitoring, target tracking, environmental monitoring, and tactical systems. In WSNs, after the deployment, the sensor nodes are unattended such as in the hazardous, nuclear plants, hostile, or remote areas. WSNs provide a seamless bridge to permeate the gap between distance and movement [1–3]. Typically, in WSNs, the sensor nodes are not isolated rather they are geographically-distributed and may significantly vary up to thousands of sensor nodes depending upon the application [4, 5]. In WSNs, each node is capable of sensing environmental variations and responding

accordingly to those changes [6]. Each sensor node immediately reacts on the environmental variations and routes the environmental changes to the sink node [7, 8]. Finally, the sink node translates and routes the aggregated data of all of the sensor nodes to the end user via a wireless radio interface.

Most of the existing scheduling techniques are either centralized or topology dependent. Therefore, it is a complex and inefficient approach to managing several nodes by only one centralized controller with limited memory and battery power. Moreover, these techniques cannot efficiently adapt to the dynamic wireless environment [9, 10]. The progressive nature and exploration in WSNs have experienced unprecedented development and is considered to be the most preferable medium for communication [11, 12]. Current WSNs differ a lot from traditional communication networks in many aspects. The distributed environment, little or no predetermined infrastructure, mobility, lack of bandwidth and scalability

* Correspondence: stbakhsh@kau.edu.sa

¹Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University, 21589 Jeddah, Makkah, Saudi Arabia
Full list of author information is available at the end of the article

are challenges and issues that distinguish the existing WSNs from traditional WSNs [13]. In the past, many researchers have focused on designing tiny sensors, limited power consumption processors, cost-effective and energy-efficient protocols to minimise energy consumption and prolong network lifetime. However, in wide-spread distributive WSNs, the single, most important challenge of energy-efficient protocols has remained a mystery for the last four decades.

The MAC layer is responsible for the allocation of resources among different entities. It helps nodes to access a channel for data transmission. Furthermore, the performance and lifetime of WSNs can be improved by resolving the contention among all of the conflicting nodes within the network. It has been observed that the distributed interactions and self-organizing nature of WSNs bring about challenges in predicting the performance of current network technologies. Whenever conflicting nodes simultaneously start transmission it results in a collision. Collisions give rise to extra control packet overhead, overhearing and idle listening; these are all anomalies that have a direct influence on network lifetime. Thus, to deal with these challenges, this current research has required the presentation of distributed scheduling (MAC) algorithms. Therefore, to design a large and scalable network, the scheduling algorithms should be computationally distributed as well as simple to resolve contention by providing collision-free scheduling.

In this paper, the main focus is on energy efficiency through dynamic sleep method. The proposed AEH-MAC algorithm reduces energy consumption by dynamically adjusting node's sleep time according to traffic load. A threshold is dynamically adjusted to overcome frequent switching between active and sleep state for saving energy consumption. When traffic load increases in the network, node's sleep time is shortened to reduce energy wastage due to switching. On the contrary, when the traffic load is low sleep time is increased to save energy, which is wasted due to idle listening. The proposed algorithm minimizes packet collisions as well as overhead for monitoring and controlling a network operation. In particular, the proposed algorithm has the ability to minimize the effects of problems resulting from hidden and exposed nodes. It guarantees fairness in its bandwidth allocation to each and every node. The proposed AEH-MAC also possesses scalability capable of forming a large network.

The remainder of this paper is organized as follows. In Section 2, review of related works is provided. The proposed Adaptive Sleep Efficient Hybrid Medium Access Control algorithm is discussed in Section 3. Simulation results are presented in Section 4 based on the facts and figures carried out in this research. Finally, conclusions are drawn in Section 5.

2 Background and related MAC protocol

The medium access control protocol handles issues regarding efficiency, fairness, and reliable access to the medium that is simultaneously used by the variety of devices. The role of MAC layer protocols in wireless networks is of particular importance as these networks are quite different from their wired counterparts in many ways. It is due to the nature of wireless communication medium itself; this difference is that devices do not need to be connected directly to communicate with each other [14, 15]. Rather, it is enough for the devices to be within the same range of the radio transmission. The best case would be that the unwanted packets are handled as noise that hinders the reception of the intended packet but could possibly be filtered out. The worst case would be that the wanted packet could be damaged in such a way that prevents repair causing the receiver to be unable to understand it; this refers as a collision [16].

Variations of carrier sense multiple access (CSMA) techniques are utilized in contention-based protocols [17]. With CSMA, the basic characteristic is that the nodes listen to the shared transmission medium of the network before trying to transmit any packet. When a transmission in progress is detected the node will wait for the end of that transmission, and only then attempt to transmit its packet. Four sources of energy consumption are minimized by contention-based protocols, such as sensor-MAC (S-MAC) and timeout-MAC (T-MAC). *Idle listening* is the first source; this is where a node is kept in a state wakefulness so that it can actively listen for any traffic that may or may not be present. *Overhearing* is the next source and happens when a broadcast packet, addressed to another node, is picked up by an ideally listening node. *Collisions* are another source; they make the node transmit its data again which uses up at least two times the amount of the energy for the same data [18]. The last source is *protocol overhead*. This source uses up not only energy but also resources because of the transmission and reception of large control packets.

Time-division multiple access (TDMA) is the basis on which schedule-based protocols are developed [19]. They conserve energy by making use of both scheduling and reservations. In this way, collision-free [20, 21] communication is guaranteed without the contention-introduced overhead. This is because slots are scheduled for each node. Idle listening is also decreased which results in considerable energy savings [22, 23]. However, making use of a TDMA protocol necessitates that nodes create real communication clusters instead of the virtual ones normally present in CSMA protocols. It is not an easy job to manage inter-cluster communication and

interference job. Determining which slot is assigned to which node, the high initial overhead when setting up and distributing a schedule for throughout in the WSN, and accurate time synchronization that can prevent clock drifts in order to keep the time slots of the nodes from overlapping are some of the challenges when using the TDMA protocol [24, 25]. Furthermore, it is not easy for a TDMA-based protocol to change its schedule without resending overhead packets when the number of nodes in a cluster is altered. Therefore, scalability is better in contention-based protocols.

Distributed randomized scheduling algorithm (DRAND) [26] enhances the scheduling deficiencies by utilizing the strengths of TDMA and CSMA MAC protocols. Consequently, to the level of contention, it switches between TDMA and CSMA. All the nodes attempt to reserve the medium for future data exchange. Therefore, each node will toss a coin with one-half head probability. The nodes with a head will try to attempt for a lottery, with certain preset probability. If a node fails the head and the lottery phase, it will reattempt for this phase after T_a time. Figure 1 show the winning cycle where H after winning lottery phase moves to REQUEST state and broadcast this message to one-hop neighbouring nodes to find out reserved slots by one-hop neighbouring nodes. If one-hop of G are in IDLE state, then node G will receive the record of reserved slots through GRANT message. The GRANT message will update node G about the reserved slot by its two-hop nodes. To avoid the collision node G will schedule a conflict-free for itself. Node G will schedule its slot and will broadcast RELEASE message, to update its neighbouring node about the reserved slot.

The query starts with the unsuccessful cycle of DRAND when nodes are not in IDLE or RELEASE state. Figure 2 shows, if in one-hop of H , suppose node E has already GRANT F prior to G , upon receiving a REQUEST

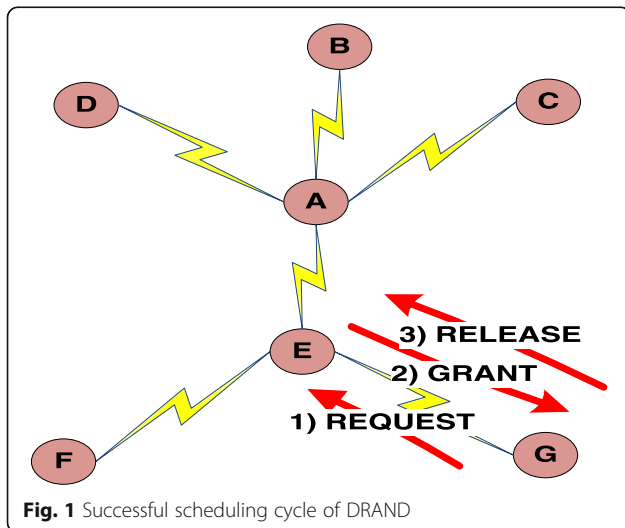


Fig. 1 Successful scheduling cycle of DRAND

message for node G , node E will send a REJECT. The REJECT message indicates the unsuccessful cycle of DRAND. Because after each REJECT message node G will defer the current cycle and will broadcast FAIL message. After T_a Time node G will reattempt the coin toss and lottery. If again node G receives the REJECT message, then it again broadcasts FAIL and each time it will differ its cycle. Although, DRAND presents distributive scheduling process, but each time the unsuccessful cycle lead to excessive message overhead, which consume system resources and decrease network performance and lifetime.

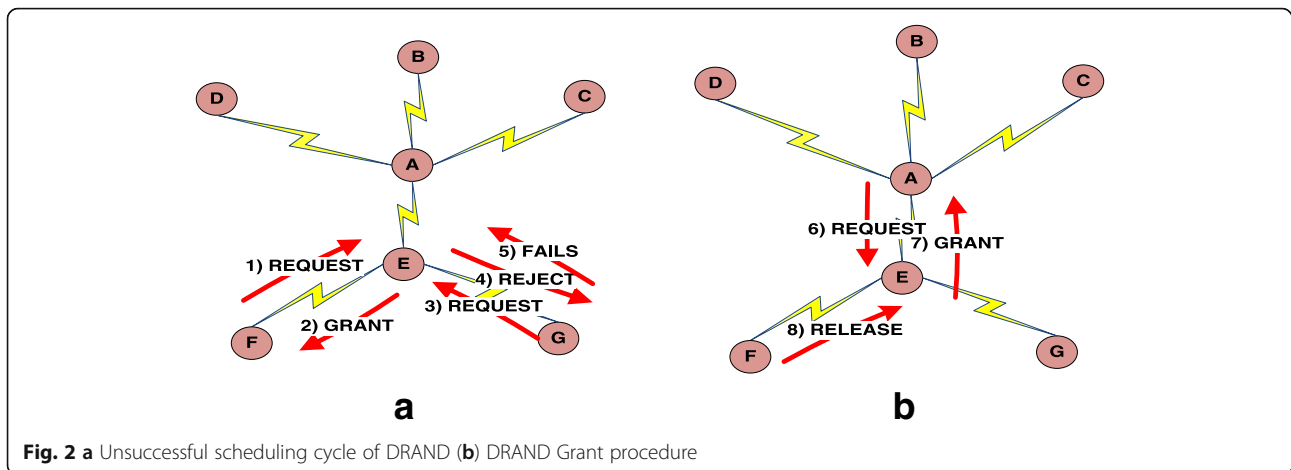
The self-learning receiver-initiated MAC (L-MAC) [27] protocol works on low data rate. It follows a tree structure to collect data within a network, where a child node adjusts its sleep time according to its parent. The child node wakes up before or closely earlier than its parent through beacon messages. A staggered wakeup scheduler is also designed by L-MAC in which sink node receives packets regardless of a wakeup interval. Figure 3 shows the traffic model of the L-MAC; where, a parent (P) has multiple child nodes (C), and C neighbour nodes 1 and 2. The network follows the shortest route; where, incoming traffic, outgoing traffic, and interfering traffic are represented by F_{in} , F_{out} , and F_{inter} respectively. It has been observed that when numbers of neighbour nodes increase in the L-MAC, the slot utilization is decreased due to a bottleneck problem. Moreover, the parent node exhausts its energy sharply, which reduces the network lifetime and increases maintenance cost.

3 The proposed adaptive scheduling algorithm

This section consists of two subsections. In subsection 3.1, the network model and notations are given for the proposed algorithm. Subsection 3.2 describes how the AEH-MAC builds a scheduler based on the self-adaptive procedure.

3.1 System model for slot assignment

Total N nodes were randomly deployed in an $A \times A$ m^2 area equipped with equal radio transceivers. The transmission range (TR) of all the nodes was the same, which helped the nodes to identify the set of nodes in its communication range. The symbols used in the paper are defined in Table 1. Typically, a multi-hop (MH) WSN is represented by a graph $G = (V, E)$; whereas, $E = \{e_1, e_2, e_3, e_4, \dots, e_n\}$ represents the distance (edges) and $V = \{v_1, v_2, v_3, v_4, \dots, v_n\}$ represents the set of nodes [28]. Nodes u and v are adjacent if both are within TR of each other and their distance is less than the Euclidean distance, where the edge $e = (u, v) \in E$ and $v = (u, v) \in V$. The set of links are defined in Eq. 1:



$$E = \{(u, v) \in V.V \mid dist_{EU}(u, v) \leq TR\} \tag{1}$$

Two nodes, u and v , are in TR of each other (one-hop ($H1$)) if both u and v have a direct link, $(u, v) \in E$.

$$\forall (u, v) \in H1, \text{ there exists a link, } (u, v) \in E \tag{2}$$

If two nodes, u and v , do not have a direct link $(u, v) \in E$ but alternatively, there exists a link, $(u, w), (w, v) \in E$, it means the nodes are in a two-hop ($H2$) neighbourhood.

$$\forall (u, v) \in H2, \text{ there exists a link, } (u, w), (w, v) \in E \tag{3}$$

Similarly, in this case, nodes u and v are connected through node w , which plays the role of a bridge between nodes u and v .

It is ensured that none of the nodes can reserve a slot that is already reserved by its two-hop neighbourhood node, $N2(u)$.

$$N2(u) = H1(u) \cup H2(u) \tag{4}$$

where $H1(u)$ represents a one-hop and $H2(u)$ represents a two-hop neighbour.

$$N2(u) \supseteq N1(u) \tag{5}$$

where $N1(u)$ is in the one-hop and neighbourhood of node u .

Correspondingly, k -hop neighbours of node v can be represented by $\Delta_k(v)$, which is a set of all of the sensor nodes whose distance to node v is at most k . Thus, the number of k -hop neighbours of node v is denoted by $\delta_k(v)$, i.e., $\delta_k(v) = |\Delta_k(v)|$, and the maximum k -hop neighbours are given as $\delta_k(v) = \max_k \delta_k(v)$. Each sensor node has a fixed transmission (TR) and interference (IR)

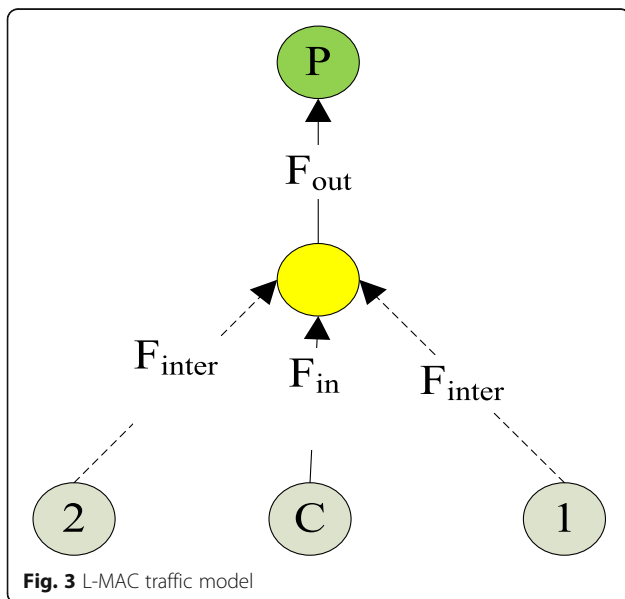


Table 1 Symbols and definitions

Definition	Symbol
Edge	E
Energy consumption in transmitting, listening, wake-up, and sleep state respectively	E_t, E_l, E_w, E_s
Group of free nodes	G
One-hop and two-hop node	$H1$ and $H2$
Interference range	IR
K -hop neighbour	Δ_k
Ratio between interference range and transmission range	γ
Total number of nodes	N
Slot	S
Time, additional time, sending time, and acknowledge time, respectively	$\tau, \tau_a, \tau_s^i, \tau_{ack}^i$
Remaining time, sensing time, and calculation time, respectively	τ_r, τ_s, τ_c
Transmission range	TR
Vertex	V
Adjacent neighbour nodes	u, v, w

range, usually, $TR < IR$ and there is $\gamma = IR / TR$. Practically, $2 \leq \gamma \leq 4$, two nodes, v_i and v_j , can communicate at time τ , if there is no node, v_k , within the same TR at τ . It can be expressed as below:

$$v_i(\tau) = \begin{cases} 1 & \text{if none of the nodes from } H1(v_i) \\ & \text{and } H2(v_i) \text{ is transmitting at time } \tau \\ 0 & \text{if any of the nodes from } H1(v_i) \\ & \text{and } H2(v_i) \text{ are transmitting at time } \tau \end{cases} \quad (6)$$

Consequently, the two nodes, u and v , are adjacent and conflicting, i.e., $v \in N_u$ and $u \in N_v$, where N_v and N_u are neighbouring nodes of u and v .

3.2 The proposed scheduling algorithm

The proposed Adaptive Sleep Efficient Hybrid Medium Access Control algorithm is dynamic and topology independent. The AEH-MAC permits nodes to decide their schedule based on their local record, instead of being dependent on neighbouring nodes and requesting for GRANT messages. Each node in the network maintains one-hop and two-hop neighbours' IDs and their reserved slot. To increase spatial reuse, nodes reserve lower slot number. Energy efficiency in WSNs is achieved through asynchronous MAC algorithms through the wakeup-sleep method. This mechanism increases end-to-end delay due to the unavailability of node(s). In order to fix the latency problem, the proposed algorithm adjusts the nodes' sleep time, dynamically. A sender predicts the receiver's wakeup time through acknowledgement packets. The proposed AEH-MAC divides the preamble into short tokens shown in Fig. 4, which contain the next hop target address and

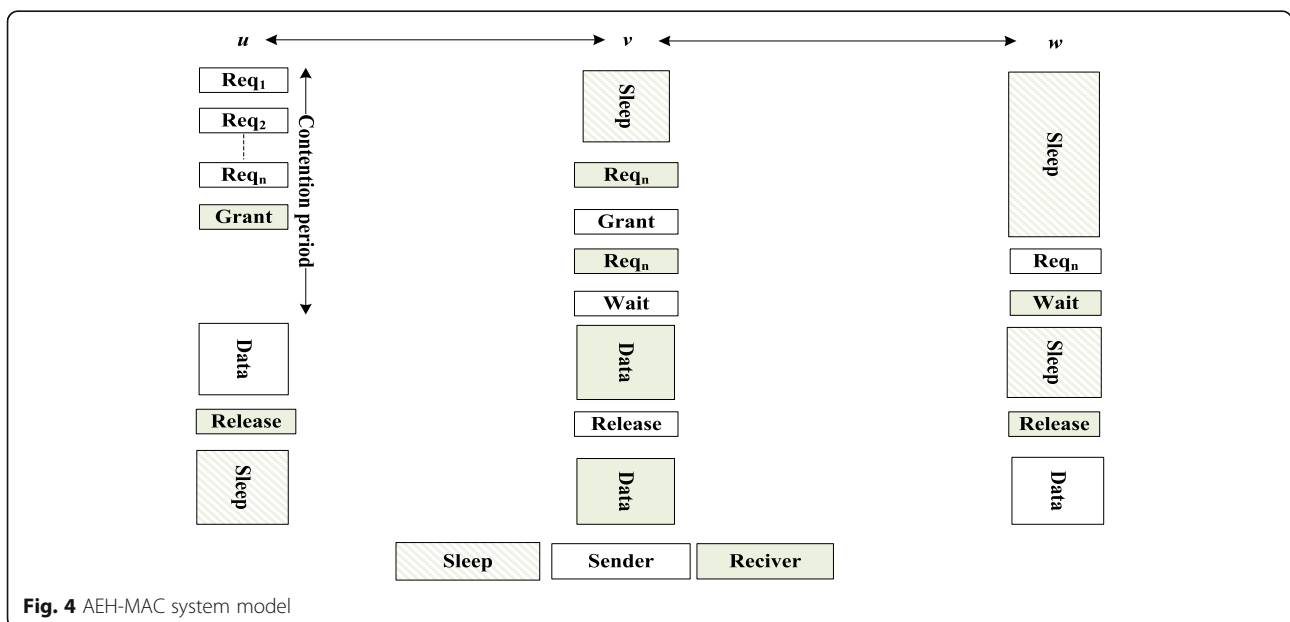
the sequence number of the packet. When node u needs to reserve a slot, it transmits short token packets until it receives an ACK from the destination node. On receiving the token packets, node v verifies the destination address; if it does not match, v enters into the sleep state for saving energy. Otherwise, v adjusts its sleep time according to the amount of traffic it is receiving.

A. Slot scheduling procedure

Initially, all the nodes in the AEH-MAC are in the IDLE state. Unscheduled node will contend to reserve a conflict-free slot. The nodes which successfully pass through the initial phase will move to the REQUEST state and broadcast a message to $H1$ neighbours. In the REQUEST message, the node broadcasts its intended reserved slot. The nodes receive the REQUEST message update their 1-hop information with the reserved slot and reply a GRANT message. During the contention period, if another node in $H2$ broadcasts a REQUEST message, the node that already sent a GRANT message will reply with a WAIT message instead of a REJECT message as shown in Fig. 2. Actually, the WAIT message is an acknowledgement that the request has been received, therefore, there is no need rebroadcast the message. In the next cycle, a GRANT message is sent to the node that has the highest priority in the QUEUE. It is ensured that a reserved slot should not conflict with one and two hop neighbours.

$$S_i \notin (H1 \cup H2) \quad (7)$$

where S_i is the i th which can be assigned to a node, v_i if it is not reserved by $H1$ and $H2$ neighbours.



In the proposed AEH-MAC, the nodes independently choose their duty cycle without exchanging synchronous information. The synchronous wakeup time nodes add their information in the ACK packet. The ACK packet also contains the communication duration field, which is used by the neighbours to set the network allocation vector (NAV) value. When the NAV reaches 0, the node starts competing for the channel. As shown in Fig. 5a, *E* and *A* are communicating, *F* and *G* are adjacent to *E*. Therefore, *F* and *G* overhear the REQUEST and note the communication duration. Both nodes set the value of the NAV; until the value is greater than 0, both nodes stay in the sleep state. During the contention period, *A* receives REQUESTS from *C*; instead of a FAIL message, *A* replies with a WAIT message as shown in Fig. 5b. On the other hand, nodes *B*, *C*, and *D* listen for the GRANT of *A* and set their NAV value. After decoding the GRANT packet, all the nodes will turn to the sleep state. Meanwhile, *E* and *A* will start collision free communication. At the end of the communication, *E* will turn to the sleep state; whilst, before the next duty cycle, *F* and *G* will wake up and start the contention for a new slot.

B. Update procedure

To reduce control overhead, both scheduling and update policies are concurrently executed. Figure 6 shows that *G* reserves slot (0) based on its one-hop (*H1*) and two-hop (*H2*) record, i.e., $S(H) = \{0\}$ (*S* is the slot) and broadcasts the REQUEST message. The REQUEST message is the information of the slot reserved by node *G*. On receiving REQUEST message, each node updates one-hop record that there is a node in one-hop that reserved $S(0)$, i.e., $H1(E) = \text{Error ! Bookmark not defined.}$ and slot scheduled $S1(E) = (0)$. Further, upon updating the 1-hop record, nodes will broadcast an ACCEPT message. Those receiving an ACCEPT message, will

update their two-hop record, i.e., $H2(A) = \{G\}$ and $H2(I) = \{G\}$ with the reserved slot $S2(0)$. So, every unscheduled node will look up its own record, i.e., $H1(u)$, $S1(u)$, $H2(u)$, $S2(u)$, instead of relying on the neighbouring nodes to schedule a slot.

C. Adjusting active and sleep time

In the AEH-MAC, nodes dynamically adjust sleep time, based on the traffic load. Next, the duty cycle is acknowledged in the ACK packet and priority is set for source nodes, which decreases control overhead and reduces unnecessary energy consumption. The operations of the nodes in the proposed algorithm are discussed as below:

i. Active state

When an event occurs and node *u* finds that there is data to be transmitted, it samples the channels in τ_s . Node *u* transmits token packets to node *v* or a sink node, inserting a constant interval. On receiving the token packet, the destination node replies with an ACK packet. The waiting time of node *u* can be calculated as:

$$\tau_{u,v}^w \leq \sum_{i=0}^n (\tau_t^i + \tau_{ack}^i) \tag{8}$$

where τ_t^i is the sending time and τ_{ack}^i is the acknowledge time of the i^{th} packet. Since token packets consume the same amount of time, the sending time of a data packet for one-hop (*H1*) is:

$$Delay_{H1} = \tau_{u,v}^w + \tau_{data} \tag{9}$$

After the sleep time, each samples the channels, τ_s . If the node receives a packet and verifies that it is not intended for it, once again it enters to the sleep state after waiting an additional time, τ_a .

ii. Sleep time

In the proposed AEH-MAC, the nodes' sleep time is dynamically adjusted based on the traffic load it receives. A node consumes energy when

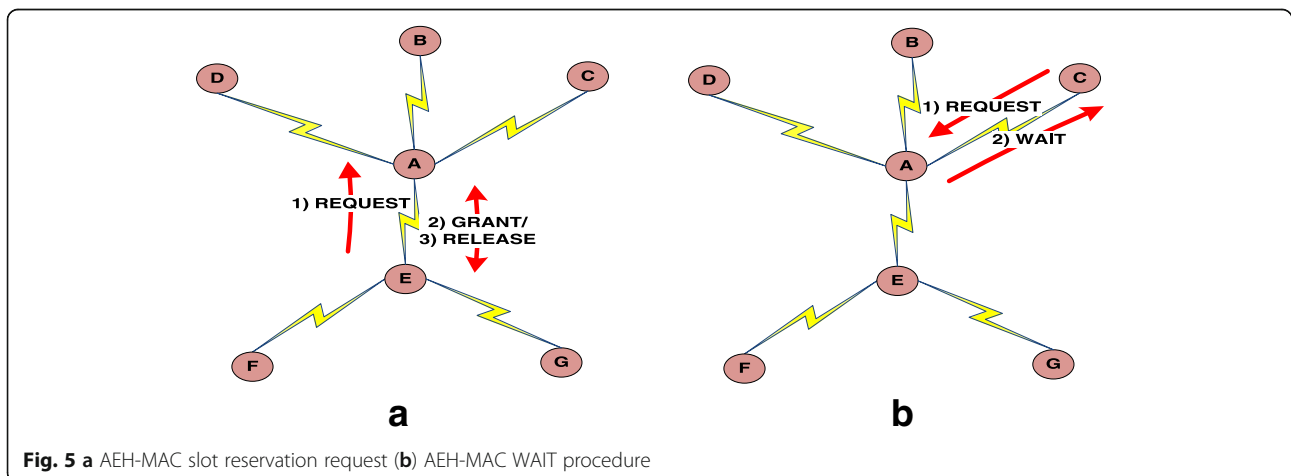
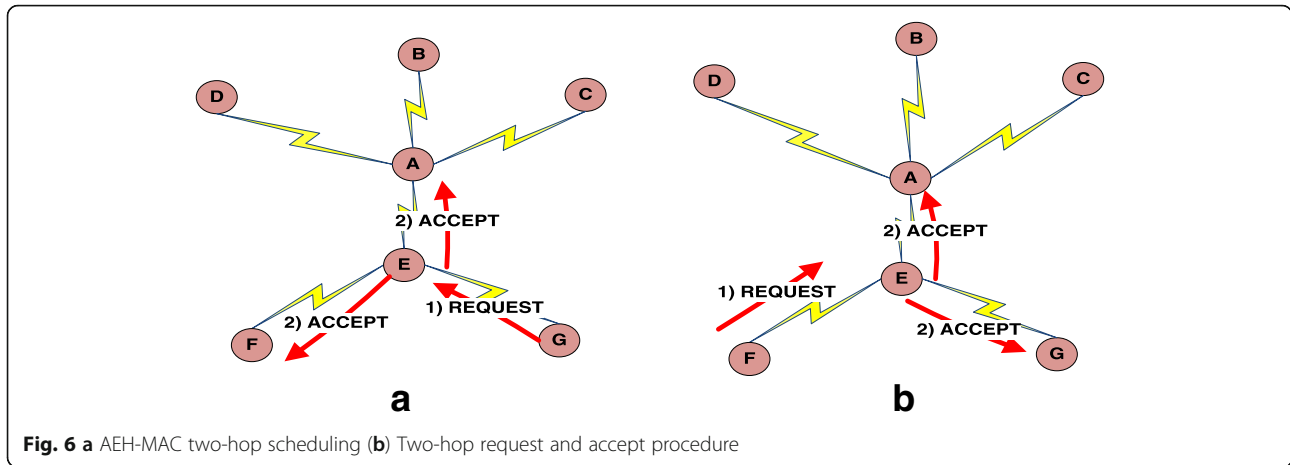


Fig. 5 a AEH-MAC slot reservation request **(b)** AEH-MAC WAIT procedure



switching between the sleep and active states; therefore, a node will not unnecessarily go to the sleep state. The energy consumption of each state is shown in Fig. 7. Where E_t , E_l , E_w and E_s are the transmitting, listening, wakeup, and sleep states, respectively, and the energy consumption relation in each state is $E_t > E_l > E_w > E_s$. If switching between the active and sleep state is less than the saved energy consumption of the active state, the node will go to the sleep state.

iii. Adjusting sleep time

In the proposed AEH-MAC, each node has a counter to calculate the received data. On receiving a data packet, the counter is incremented by one, and the node computes the data packets' receiving time.

$$\tau_t = \tau_1 \times n \tag{10}$$

where τ_1 represents a single data packet receiving time and n denotes the total number of received packets, The τ_t represents total time.

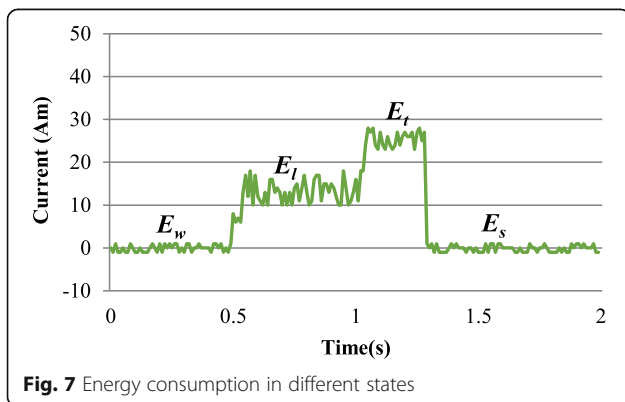


Fig. 7 Energy consumption in different states

The length of a duty is represented by τ and the remaining time of a duty cycle is τ_r . Thus, the remaining time can be calculated as:

$$\tau_r = \tau - \tau_s - \tau_c \tag{11}$$

where τ_c is the calculation time of a duty cycle; on the bases of the remaining time, the node decides whether to stay active or go to the sleep state (τ_s).

The sleep time in the proposed AEH-MAC is dynamically selected as shown in Fig. 8. If a node finds during sampling the channel, that there is no data to forward, it goes to the maximum sleep time, τ_{sleep}^M , which is equal to $\tau - \tau_l$. The maximum sleep time is almost one duty cycle. When $\tau_{sleep}^M > \tau_r > \tau_s$, it indicates the maximum sleep time is greater than remaining sleep time. It means that there is a medium traffic load so the sleep time is set as $\tau_s = \tau_r$. When the minimum sleep time is greater than the rest time it indicates that the traffic load is heavier and energy consumption between switching is higher than saving energy. Therefore, the node will remain active and $\tau_r = 0$. In the last case, when there is no traffic load, the maximum time, τ_{sleep}^M , is less than τ_r ; so, the node will immediately go to the sleep state. Therefore, the maximum sleep time, $\tau_s = \tau_{sleep}^M$.

iv. Sleep time fault tolerance

In the AEH-MAC, each node synchronizes wakeup time with neighbour nodes without increasing control overhead. The next wakeup time is set, dynamically, according to neighbour nodes wakeup time, so that the node wakes up slightly earlier than its neighbour nodes. A node measures its wakeup time with short token messages; when a node wakes up before its

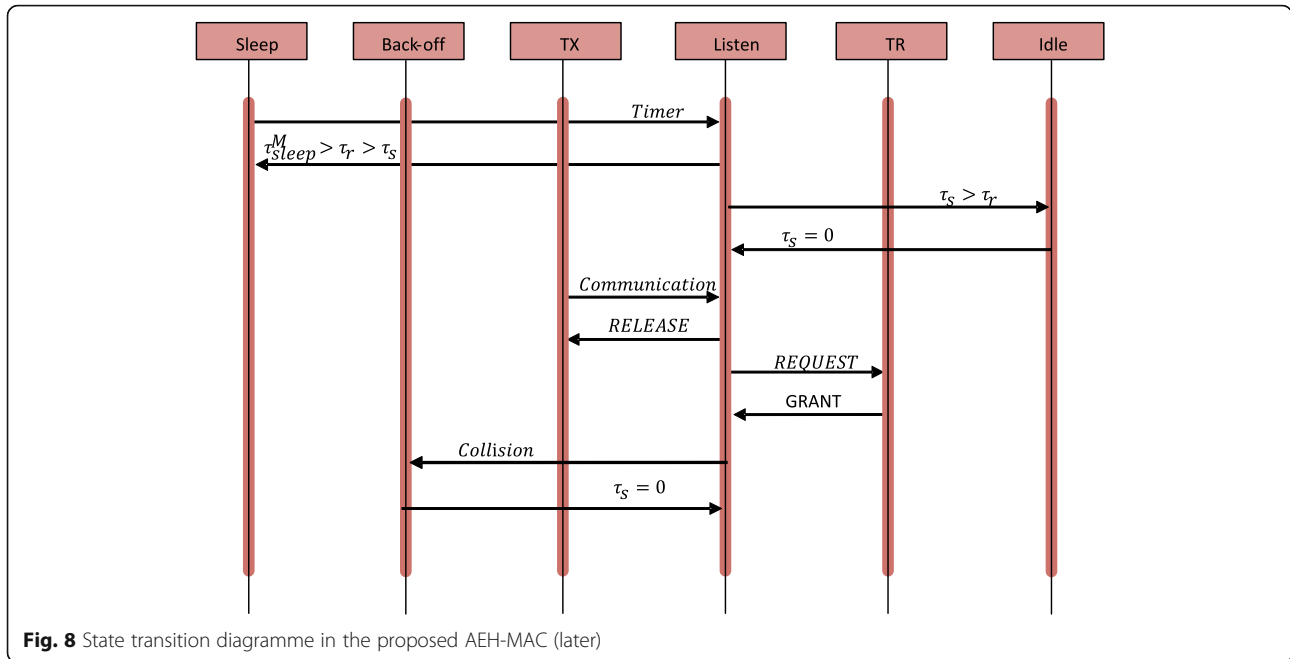


Fig. 8 State transition diagramme in the proposed AEH-MAC (later)

neighbour nodes, it measures the offset amongst its wakeup time and the neighbour nodes. After each wakeup interval, a node readjusts its sleep time according to the traffic load in the network. There are chances that a node may not completely detect the short messages due to a late wakeup, interference, or collision. Therefore, the node divides sleep time by 2 ($\tau_{sleep}^M/2$) and returns to the wakeup state, earlier. If a node intends to transmit, it always wakes up earlier than its neighbour nodes. Once the error is removed, the node wakes up according to the normal sleep time.

v. Collision avoidance

Collisions not only waste the bandwidth of the network but also the energy consumption of the individual devices (receivers and transmitters). Collision detection is not a wise solution in WSNs, where not many devices possess the necessary ability. Packet collision may happen due to hidden and exposed terminal problems. In the AEH-MAC, nodes dynamically adjust their sleep time and transmit at different times to avoid the collision. If multiple neighbour nodes transmit simultaneously, there is a chance that a collision may occur; therefore, before sending a packet, each node executes a random backoff to avoid the collision. During the backoff period, if a sender detects transmission from another node, it stops its transmission. The sender enters into the sleep state and retries in the subsequently contention period. The AEH-MAC detects a collision through beacon by measuring the power level of the channel. If it

detects a single packet header, the receiver rebroadcasts a beacon to notify the sender and requests for retransmission of the data packets.

4 Performance metrics/parameters and the results' evaluation

This section presents a comparison between the AEH-MAC, L-MAC, and DRAND. For the performance evaluation, the proposed algorithm was simulated in a network simulator (NS-2) [29] using constant bit rate (CBR) and variable bit rate (VBR) traffic loads [30, 31]. The AEH-MAC is dynamic and topology independent because it adapts the schedule in response to the topology without rescheduling the whole transmission frame. The transmission schedule of the AEH-MAC is dynamically executed, as a result, it is preferable as it adjusts to the workload more effectively and efficiently as compared to the traditional algorithms. During the setup phase, the scheduling was carried out; therefore, the setup parameters were measured during the setup phase until the network reached a steady state.

The nodes were randomly deployed in an area of $500\ m \times 500\ m$. Whereas, the *TR* of all the nodes was kept constant at $40\ m$ with a link capacity of $2\ Mbps$. The energy consumption in each state was computed by the power consumed by a sensor node in that particular state to the proportions of time that the node spent in that state. Therefore, the total power and energy consumption was the sum of the power and energy consumed by the network in each mode of operation. In the proposed work, the TwoRayGround was selected as the propagation model as it has a minimum control

overhead and consumes less energy. Moreover, the TwoRayGround provides the best neighbouring information [32, 33]. The CC1101 provides reliable wireless communication functions for the MAC layer and the PHY layer, and its radio operates at a low power. The bandwidth and data rate were set to be the same, which were equal to 250 kHz and 250 kbps, respectively. The transmission time was computed from the data rate and packet lengths, which were used to compute the energy dissipation in each state of the node. The simulation time was set to 1000 s, and the initial energy of each node was set to 1000 joules. The results were obtained after 20 repetitions of trials. The simulation confidence was 95%, and the relative error was less than 5%. The results were compared in terms of energy consumption, running time, number of rounds, and slot reservation. Table 2 summarises the parameters and the model used in the AEH-MAC, which were acquired from the standard CC1101 radio transceiver.

A. One-hop scenario for the CBR simulation results

The one-hop performances of the DRAND, L-MAC, and AEH-MAC were evaluated as shown in Fig. 9. The number of neighbour nodes were varied from 3–30 and deployed in a circular topology. It was analyzed that the AEH-MAC had the lowest number of rounds due to its control mechanism. The DRAND's and L-MAC's one-hop neighbours reserved slots, sequentially. On the other hand, the AEH-MAC allowed the nodes to parallel participate in slot reservation. As a result, the proposed algorithm reduced unnecessary attempts for slot reservation, which significantly reduced the number of rounds

compared to the DRAND and L-MAC. It was analysed that the DRAND and L-MAC transmitted more numbers of control packets that caused a higher number of rounds for scheduling as shown in Fig. 9a. It is important to note that, one-hop scheduling is not collision free. Since the CSMA is used to access the channel access when multiple nodes transmit concurrently collision occurs. Therefore, increasing the contention window size, the number of transmissions can be reduced. Figure 9b shows that the AEH-MAC had a lower runtime due to the lower number of rounds. In the proposed algorithm, the nodes did not need to resend slot reservation requests. Once the transmission was over, the neighbour nodes broadcast a release message. The node in the wait state started its transmission without increasing control overhead.

B. Multi-hop scenario for the CBR simulation results

The simulation for the multi-hop scenario was based on different number of nodes and neighbour nodes, and by adding new nodes. Runtime requires establishing a scheduling process in a network. Wireless is a shared and unreliable medium. A receiver within the interference range of multiple nodes, whilst trying to receive from one node, may experience a collision; as a result, all active nodes waste resources. In the DRAND, the nodes woke up periodically to verify if there was any message and every node independently selected its schedule. Before transmitting a message, each node generated a preamble to ensure that the destination had received it. It was analyzed that, the runtime of the DRAND and L-MAC protocols increased since the sleep period of the sensor nodes were extended. The proposed algorithm had a lower runtime than the DRAND and L-MAC by dynamically adjusting the sleep time. Therefore, the AEH-MAC had a reduced transmission overhead and idle listening, and it enabled nodes to learn their neighbours' wakeup schedules. The existing protocols increased the control overhead by exchanging the schedule information. The proposed AEH-MAC had less runtime due to reducing the wait time and idle listening. Figure 10 shows the runtime, it is observed that when the network size was small, the runtime of the AEH-MAC was almost equal to the L-MAC, but when the network became denser, the DRAND and L-MAC techniques faced more challenges and their runtime rapidly increased. The number of rounds against the varying topology size is shown in Fig. 11. Since the DRAND, L-MAC, and AEH-MAC ran in rounds to schedule a slot, when the nodes increased, the number of rounds of the existing protocols also increased due to the large wakeup interval. As a result, a higher number of

Table 2 CBR simulation parameters

Parameters	Values
Number of nodes	10–200
Simulation area	500 m × 500 m
Transmission range	40 m
Sensor node deployment	Random deployment
Propagation model	TwoRayGround
Initial contention window (slot)	128
Congestion contention window (slot)	64
Packet size (Request, Release, Grant)	50 bytes
Data packet size	100 bytes
Bandwidth	250 kHz
Data rate	250 kbps
Listening time	0.025 s
Switching time	0.005 s
Initial energy	1000 J
Simulation time	1000 s

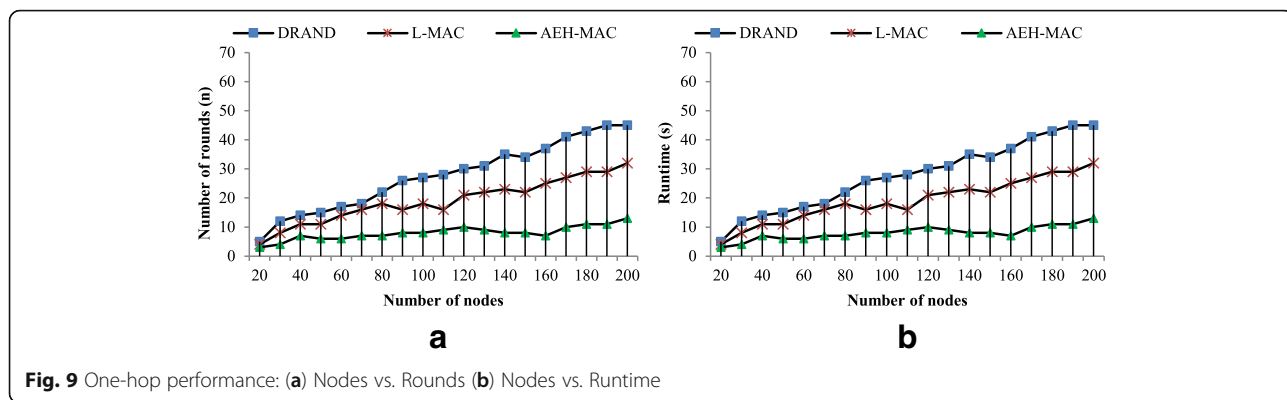


Fig. 9 One-hop performance: (a) Nodes vs. Rounds (b) Nodes vs. Runtime

packets were forwarded that caused a higher probability of collision and increased the scheduling time. On the other hand, for the runtime of the AEH-MAC, the number of rounds was fairly good due to the neighbours' adaptive wakeup time. It has been analyzed that, the AEH-MAC scheduled the network with fewer numbers of rounds as compared to the DRAND and L-MAC. This advantage came

through avoiding the unsuccessful cycles of the DRAND; because, after every unsuccessful cycle, the node had to differ its round, and had to pass the whole coin toss and lottery phase, again. Whilst in the AEH-MAC, each node had to decide its own slot based on the local information within its one-hop and two-hop neighbouring nodes. The proposed AEH-MAC, instead of a FAIL message, maintained a

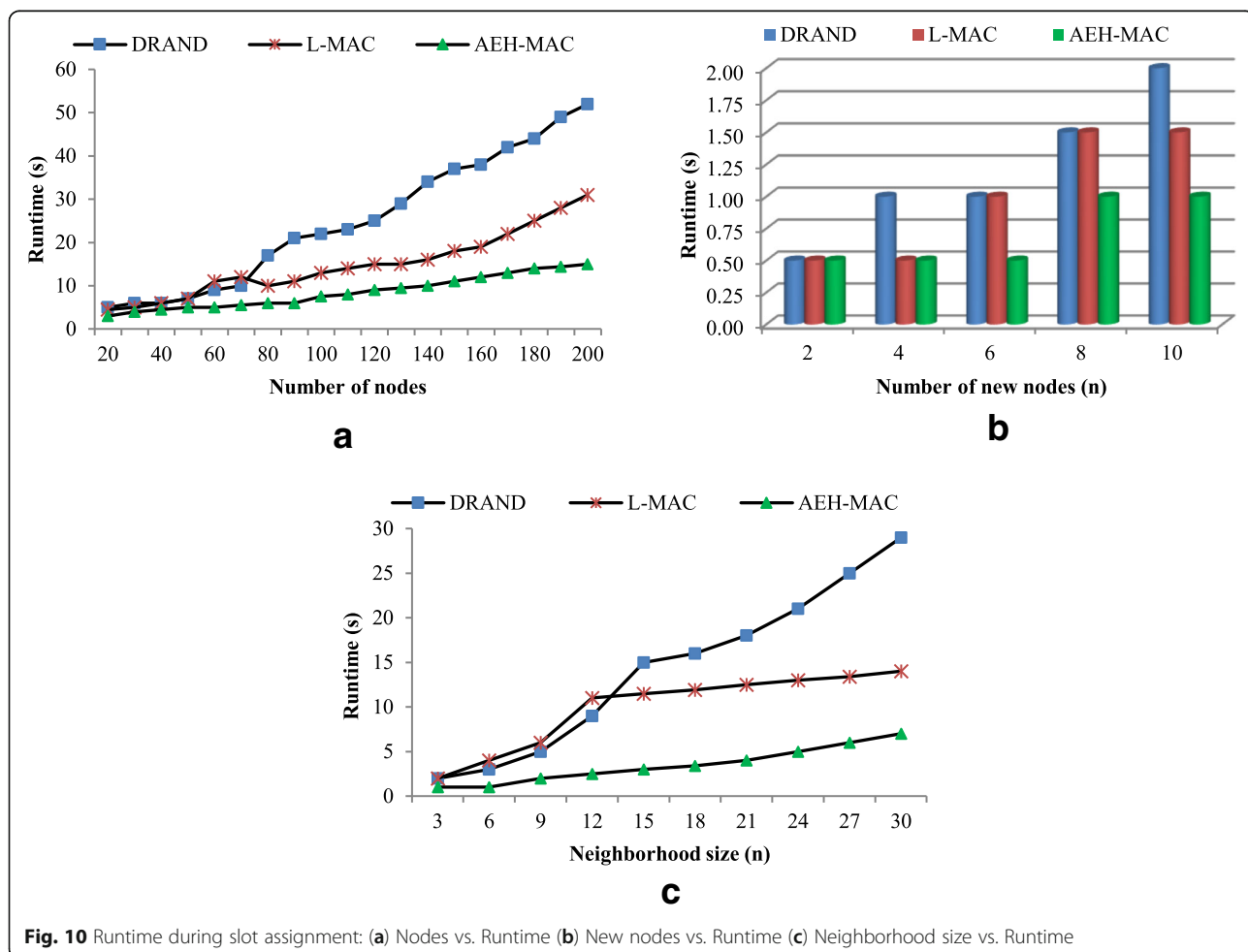


Fig. 10 Runtime during slot assignment: (a) Nodes vs. Runtime (b) New nodes vs. Runtime (c) Neighborhood size vs. Runtime

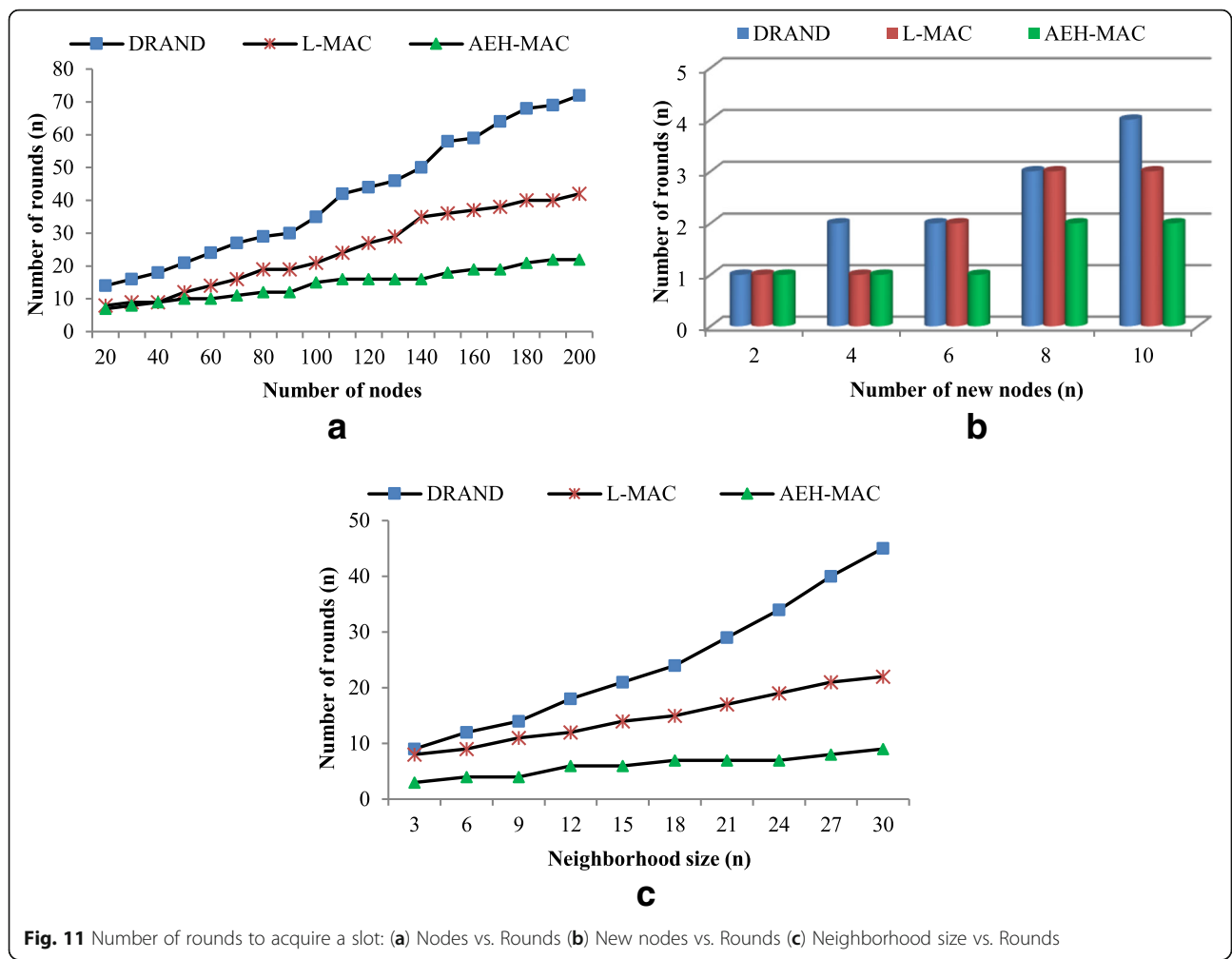


Fig. 11 Number of rounds to acquire a slot: (a) Nodes vs. Rounds (b) New nodes vs. Rounds (c) Neighborhood size vs. Rounds

priority queue of the received request messages. The AEH-MAC assigned the new slot to the node having the highest priority (FIFO) in the queue. The energy consumption was calculated as the total energy consumed during receiving, transmitting, listening, and switching from the sleep to idle mode and vice versa. When the network density increased the energy consumption also increased. However, the nodes scanned the channel to ensure that it was free and they could utilize it. It was analyzed that, unnecessary energy was consumed during idle listening and overhearing. All the techniques ran in rounds and each node had to pass through all the primitive operations (idle listening, receiving and transmitting a byte, and sleep mode). Hence, the energy consumed in each of these operations and the total energy consumed during scheduling was calculated as the sum of all the operations. Figure 12 relates the scheduling algorithm to the amount of energy consumed; the AEH-MAC had the lowest energy consumption compared to the existing

protocols. The control overhead increase caused more energy consumption. Each node broadcast packets to find its one-hop and two-hop neighbours. The DRAND and L-MAC consumed more energy due to higher collisions and packet loss. It was observed that, when the network topology changed by adding new nodes to the network, the nodes updated their slot reservation. The DRAND and L-MAC needed more time and energy to reschedule the network compared to the AEH-MAC. With increasing the neighbourhood size, the AEH-MAC outperformed the DRAND and L-MAC. The advantage of the AEH-MAC came due to avoiding an unsuccessful cycle that directly trimmed down the number of messages. The fewer number of messages exchanged, the less the energy will be consumed; as in the case of the AEH-MAC. When a node wanted to transmit data, it first reserved a slot and established a link with other nodes. A node had to stay active to ensure that the channel was free and that there was no

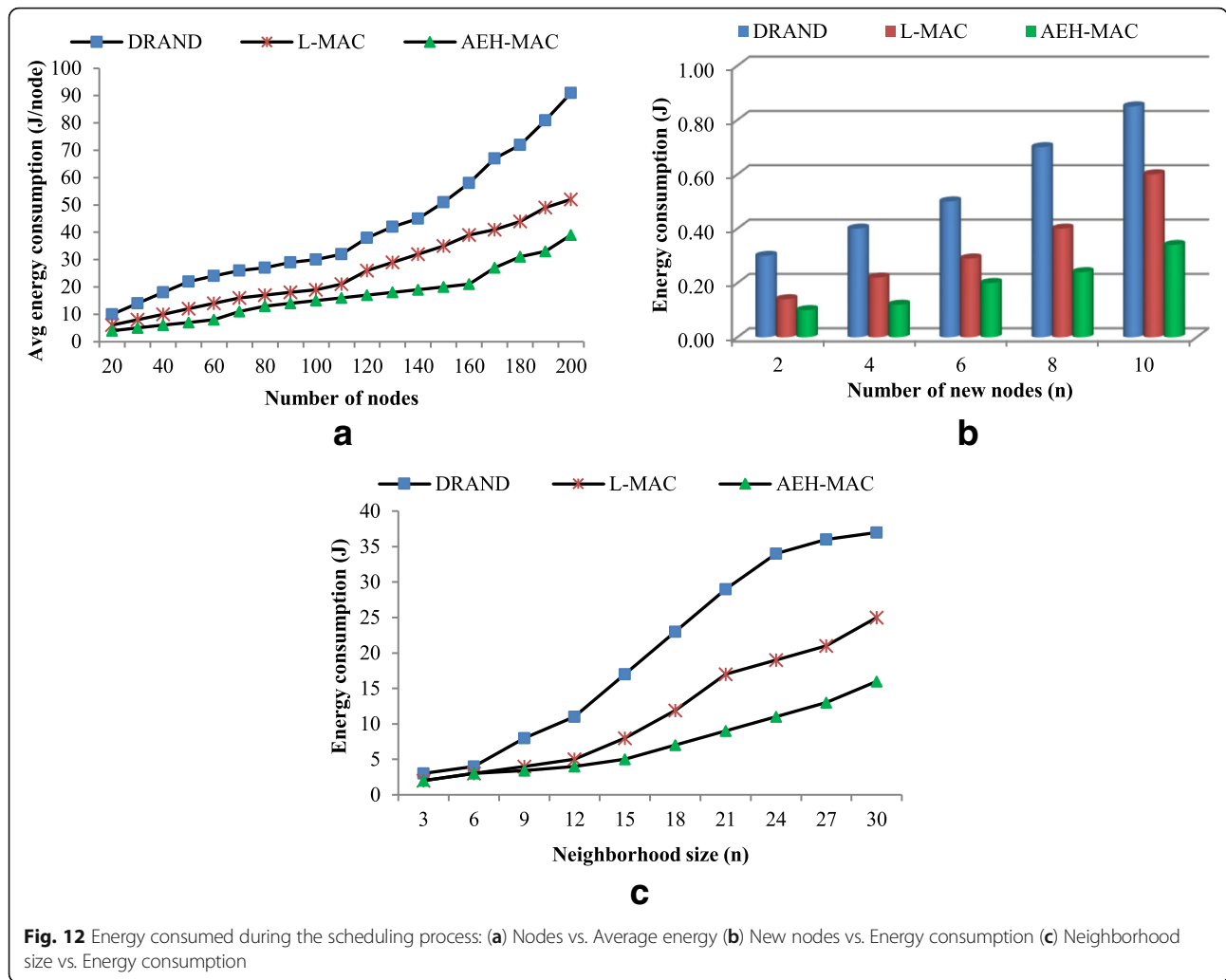
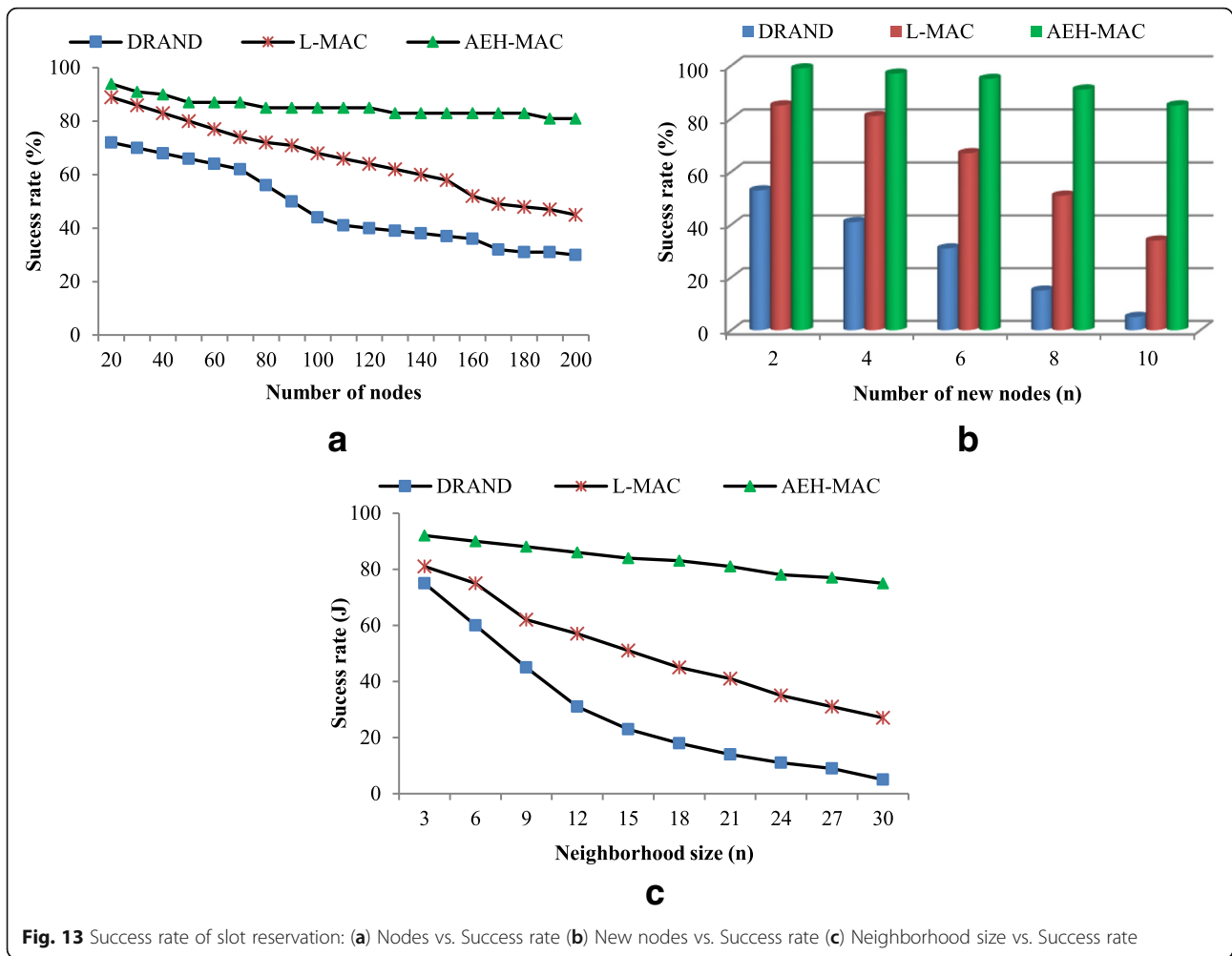


Fig. 12 Energy consumed during the scheduling process: (a) Nodes vs. Average energy (b) New nodes vs. Energy consumption (c) Neighborhood size vs. Energy consumption

transmission on the channel. When a node was in the transmission range of the other nodes, it overheard all the messages for a long period of time. Nodes in the sleep mode had to wake up before starting of a new slot through the exchange of control messages. Since control messages consume energy, bandwidth, and time, therefore, the control messages had to be reduced. As control messages were exchanged during the network phase to reserve a slot, therefore, these control messages were considered as overhead. Figure 13 shows the success rate of the slot reservation during the scheduling phase. The proposed AEH-MAC adjusted the node sleep time, dynamically; if the traffic load was less in the network, the nodes increased the sleep time. Therefore, the AEH-MAC had a higher success rate compared to the DRAND and L-MAC. In the DRAND, excessive messages were exchanged whilst the AEH-MAC limited the number of messages exchanged during scheduling. In the proposed algorithm, a node

reserved a slot if it was not reserved by any of its neighbour nodes. If a node reserved a forbidden slot, it was declined after not receiving a response from the neighbour nodes. It was observed that increasing the number of nodes reduced the slot reservation attempts and affected the scheduling accuracy. The results show that the increasing number of neighbourhood sizes rapidly decreased the DRAND and L-MAC slot reservation percentage. In the proposed algorithm, a collision due to a hidden-node was significantly reduced as a node could reserve the same slot only if it was not in the two-hop neighbours. The AEH-MAC utilized the highest number of slots for network scheduling, which was spatial reuse efficiency. The proposed algorithm had a 20–30% higher slot reservation rate compared to the L-MAC and DRAND, respectively. This was because it reduced the collision probability by adding to the wakeup time of each node.



C. Multi-hop scenario for VBR simulation results
 To evaluate the performance of both algorithms, a simulation was executed in a VBR traffic model under different traffic loads. The traffic followed a Poisson process with the parameter λ , Table 3 summarises the parameters used for VBR traffic. Figure 14 shows the performance of all the algorithms by varying the λ value between 1 and 5. During the simulation, the traffic load was varied

Table 3 Variable bit rate simulation parameters

Parameters	Values
Send rate during on time	250 kbps
Size of random variation on rate	0.25 bps
Mean time of ordinary rate period	2 s
Mean time of burst rate period	1.5 s
Number of changes during burst time	3 s
Deviation from time periods	0.5 s
Maximum rate of burst / constant variation	450 kbps

every 50 s from 1 to 5 packet/node/s. The traffic load was varied to check the network condition or response of all three techniques. It has been analyzed from the results that the performance of DRAND and L-MAC decreased compared to the AEH-MAC when the value of λ increased. Since the AEH-MAC adjusted its sleep time, dynamically, according to the received traffic interval, if the arrival duration was long, the proposed algorithm considered a light traffic load and increased its sleep time to reduce energy consumption. On the contrary, when the traffic load was higher, the AEH-MAC reduced the sleep time to overcome the bottleneck problem. As shown in Fig. 14, when λ increased, the DRAND and L-MAC performance decreased due to sleep time. On the other hand, the AEH-MAC, using a prediction mechanism, synchronized its sleep time with its neighbours; therefore, it performed better than the DRAND and L-MAC. It has been analyzed that, when the λ value was small, fluctuations in the

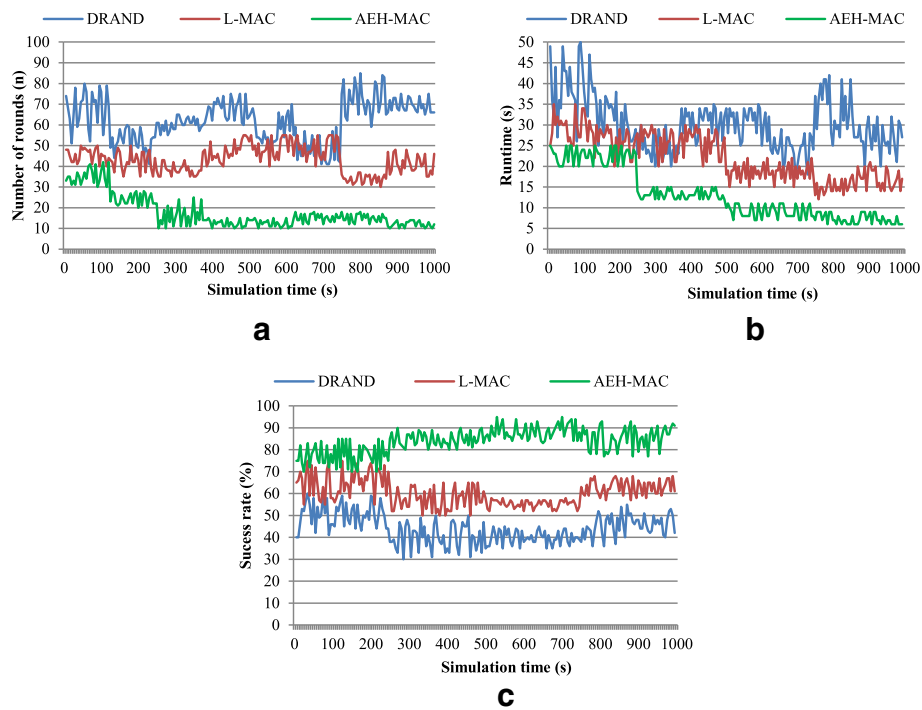


Fig. 14 Multi-hop Scenario for variable bit rate: (a) Simulation time vs. Rounds (b) Simulation Time vs. Runtime (c) Simulation time vs. Success rate

performance of all the algorithms were fewer. Figure 14b shows the DRAND and L-MAC is rapidly increasing when the λ value increased; whilst, the AEH-MAC runtime and number of rounds remained almost constant. When the transmission rate changed, the DRAND and L-MAC failed to adopt to the environment; therefore, it decreased the overall network performance.

5 Conclusions

A hybrid MAC algorithm improves scheduling in WSNs using the advantages of asynchronous and synchronous algorithms, such as energy efficiency, runtime, and slot reservation. The AEH-MAC reduces control overheads by sharing scheduling information and efficiently utilizing system resources. The proposed algorithm is an efficient solution for a dynamic environment that allows nodes to concurrently reserve a slot without increasing control overhead. The AEH-MAC dynamically adjusts nodes' sleep time according to the traffic load in the network. If a node predicts traffic in a channel it increases its sleep time and predicts the wake up of its neighbours. Simulation results were obtained under various network scenarios, varying the number of nodes and neighbourhood size, and adding new nodes. It has been analyzed that the AEH-MAC outperforms existing protocols in terms of energy consumption, runtime, number of rounds, and successful slot reservation.

The proposed work will be enhanced in the future for high mobility networks to overcome the control overhead problem. Moreover, energy is a critical issue in WSNs; therefore, the node energy consumption will be investigated for network lifetime.

Acknowledgements

This research work was financially supported by the Deanship of Scientific Research (DSR) at King Abdulaziz University, under grant No. (G-575-611-37).

Authors' contributions

All authors contributed equally to this work. STB wrote the main paper, and RAG wrote the introduction and literature. AHA performed analytical analysis and system design. ST and MAS carried out the implementation of the simulation model, and edited the manuscript.

Competing interests

I declare that authors have no significant competing financial, professional or personal interests that might have influenced the performance or presentation of the work described in this manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

- ¹Computer Science Department, Faculty of Computing and Information Technology, King Abdulaziz University, 21589 Jeddah, Makkah, Saudi Arabia.
- ²Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, 21589 Jeddah, Makkah, Saudi Arabia.
- ³Department of Electrical and Electronic Engineering, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia.

Received: 20 September 2016 Accepted: 23 April 2017

Published online: 08 May 2017

References

- S Chouikhi et al., "A survey on fault tolerance in small and large scale wireless sensor networks," *Comput. Commun.* **69**, 22–37 (2015)
- MA Mahmood et al., "Reliability in wireless sensor networks: A survey and challenges ahead," *Comput. Netw.* **79**, 166–187 (2015)
- M Khanafer et al., "A Survey of Beacon-Enabled IEEE 802.15.4 MAC Protocols in Wireless Sensor Networks," *IEEE Commun. Surv. Tutorials, IEEE Trans. Power Electron.* **16**, 856–876 (2014)
- YZL Shu, LT Yang, Y Wang, M Hauswirth, N Xiong, TPGF "Geographic routing in wireless multimedia sensor networks," *Telecommun. Syst.* **44**, 79–95 (2010)
- NXY Liu, Y Zhao, AV Vasilakos, J Gao, Y Jia, "Multi-layer clustering routing algorithm for wireless vehicular sensor networks," *IET Commun.* **4**, 810–816 (2010)
- F Khoshnoud, CW De Silva, "Recent advances in MEMS sensor technology-mechanical applications," *IEEE Instrum. Meas. Mag. IEEE Trans. Power Electron.* **15**, 14–24 (2012)
- MA Sheikh et al., "Fair scheduling algorithm for wireless sensor networks," (National Postgraduate Conference (NPC), Kuala Lumpur, 2011), pp. 1–5
- B Dezfouli et al., "DICSA: Distributed and concurrent link scheduling algorithm for data gathering in wireless sensor networks," *Ad Hoc Netw.* **25**(Part A), 54–71 (2015)
- M. Roth, et al., "Flexible and modular low power wireless networks," *Procedia Computer Science*, vol. 52, pp. 695–699, 2015. <http://www.sciencedirect.com/science/article/pii/S1877050915008777>
- ME Haque et al., "Performance Assessment of tree topology sensor network based on scheduling algorithm for overseeing high-rise building structural health information," *Optik - Int. J.Light Electron Optics* **126**, 1676–1682 (2015)
- AWN Xiong, LT Yang, L Song, Y Pan, R Kannan, Y Li, "Comparative analysis of quality of service and memory usage for adaptive failure detectors in healthcare systems," *IEEE J. Sel. Areas Commun.* **27**, 495–509 (2009)
- H. S. Y. Sang, Y. Inoguchi, Y. Tan, N. Xiong, "Secure data aggregation in wireless sensor networks: a survey, parallel and distributed computing," *Approp. Technol.* 4–7 (2006). <http://ieeexplore.ieee.org/document/4032199/>
- G Aldabbagh et al., "QoS-Aware Tethering in a Heterogeneous Wireless Network using LTE and TV White Spaces," *Comput. Netw.* **81**, 136–146 (2015)
- G. Aldabbagh, et al., "Distributed dynamic load balancing in a heterogeneous network using LTE and TV white spaces," *Wirel. Netw.* **21**(7), 2413–2424 (2015). <https://links.springer.com/article/10.1007/s11276-015-0921-4>
- S. T. Bakhsh, et al., "Self-schedule and self-distributive mac scheduling algorithms for next-generation sensor networks," *Int. J. Distrib. Sens. Netw.* 2015. <http://dl.acm.org/citation.cfm?id=2930078.2930086>
- R Adhikari, "A meticulous study of various medium access control protocols for wireless sensor networks," *J. Netw. Comput. Appl.* **41**, 488–504 (2014)
- S. A. Chowdhury, et al., "Performance study and simulation analysis of CSMA and IEEE 802.11 in Wireless Sensor Networks and limitations of IEEE 802.11," in *Computers and Information Technology, 2009. ICCIT'09. 12th International Conference on*, 2009, pp. 431–436. <http://ieeexplore.ieee.org/document/5407277/>
- S. Bakhsh, "A self-organizing location and mobility-aware route optimization protocol for bluetooth wireless," *Journal of King Saud University-Computer and Information Sciences*, pp. 239–248, 2016. <http://thesai.org/Publications/ViewPaper?Volume=7&Issue=6&Code=ijjcsa&SerialNo=31>
- J. Ma, et al., "Energy efficient TDMA sleep scheduling in wireless sensor networks," in *INFOCOM 2009, IEEE, 2009*, pp. 630–638. <http://ieeexplore.ieee.org/document/5061970/>
- DT Le et al., "LABS: Latency aware broadcast scheduling in uncoordinated Duty-Cycled Wireless Sensor Networks," *J. Parallel Distrib. Comput.* **74**, 3141–3152 (2014)
- V Galvao Guimaraes et al., "Dynamic Timed Energy Efficient and Data Collision Free MAC Protocol for Wireless Sensor Networks," *IEEE Lat. Am. Trans., IEEE (Revista IEEE America Latina)* **13**, 416–421 (2015)
- P Wiatr et al., "Energy efficiency versus reliability performance in optical backbone networks [invited]," *IEEE/OSA J. Opt. Commun. Networking* **7**, A482–A491 (2015)
- L Wei, Z Longmei, "A Novel Multi-Channel MAC Protocol for Cluster Based Wireless Multimedia Sensor Networks," *Phys. Procedia* **25**, 2203–2210 (2012)
- V. Zibakalam, "A new TDMA scheduling algorithm for data collection over tree-based routing in wireless sensor networks," *ISRN Sensor Networks*, **2012**, 1–7 (2012). Article ID 864694. <http://dx.doi.org/10.5402/2012/864694>.
- EJ Kim et al., "Latency bounded and energy efficient MAC for wireless sensor networks," *Commun. IET* **6**, 2120–2127 (2012)
- I Rhee et al., "DRAND: Distributed Randomized TDMA Scheduling for Wireless Ad Hoc Networks," *IEEE Trans. Mob. Comput.* **8**, 1384–1396 (2009)
- T Dinh et al., "L-MAC: A wake-up time self-learning MAC protocol for wireless sensor networks," *Comput. Netw.* **105**, 33–46 (2016)
- S. Gobriel, et al., "TDMA-ASAP: Sensor Network TDMA Scheduling with Adaptive Slot-Stealing and Parallelism," in *Distributed Computing Systems, 2009. ICDCS '09. 29th IEEE International Conference on*, 2009, pp. 458–465.
- E. Hossain, "The network simulator (NS-2) " pp. 2014. <http://www.isi.edu/nsnam/ns/ns-build.html>. Accessed 03 Jan 2016.
- S. Ullah, M. Imran, M. A. Alnuem, "A Hybrid and Secure Priority-Guaranteed MAC protocol for Wireless Body Area Network", *Int. J. Distrib. Sens. Netw.*, vol. 2014 (2014).
- M Imran, MA Alnuem, MS Fayed, A Alamri, "Localized Algorithm for Segregation of Critical/Non-critical Nodes in Mobile Ad Hoc and Sensor Networks". *Procedia Comput. Sci.* **19**, 1167–1172 (2013)
- World Academy of Science, Engineering and Technology International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering **7**(5), 562–576 (2013)
- H. Lee and H. Kwon, "Improving estimation of the nearest neighbor in mobile wireless sensor networks," ed: Citeseer, 2009.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com