International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8, Issue-1, May 2019

ES-MAC: A Sink-aware Beacon Scheduling Transmission for Receiver-Initiated MAC Protocol for Wireless Sensor Network

P Rachelin Sujae, S Arulselvi

Abstract*: In Wireless Sensor Network energy is a vital parameter as the sensor nodes are low power devices. Many energy efficient protocols have been proposed to efficiently utilize the battery-life of the sensor nodes. Hence it is vital to forward the data packet from the sensor nodes to the sink node without any delay. In contention-based Receiver-Initiated MAC protocols this delay is caused by the idle listening which makes the data packet to stay in the nodes for a longer duration. Also due to collision and retransmission it further introduces delay in the network. As it is not aware of the sink node the data packet may end up in looping. In this paper an Enhanced Scheduling-MAC (ES-MAC) protocol is proposed which a sink-aware beacon is scheduling algorithm applied over receiver-initiated MAC protocol, guaranteeing a delay-efficient communication enhances network lifetime. The ns-2 performance evaluation shows the proposed ES-MAC has high throughput and less network latency. Moreover the comparative simulation result indicates that the proposed protocol outperforms the contention-based RI-MAC protocols.

***Index Terms*: Wireless sensor network, beacon, scheduling, idle listening, looping, throughput, latency, receiver-initiated MAC.**

I. INTRODUCTION

A wireless sensor network (WSN) is a spatially distributed wireless network device incorporated with multiple sensors to monitor the physical or environmental affairs. A WSN network incorporates a gateway sink that provides the wireless connectivity to the real world and measures a number of physical conditions like humidity, temperature and stress. Also, the wireless protocol equipped is based on the application requirement. The available standard for the transceiver radio is 2.4 GHz based on IEEE 802.15.4 or IEEE 802.11 (Wi-Fi) standards, which usually operates in 900MHz. WSN has been deployed in various applications mainly where human intervention is prohibited, e.g., military [1-2], environmental monitoring [3-4], health monitoring [5-10], smart energy management system [11-15] and various other [16-20] industries.

The major components of a sensor node are microcontroller, external memory, sensors [21], battery, and transceiver. The transceiver radio [22] in the sensor node consumes majority of the battery power as it has dual activity for transmission/reception of control/data packets. Duty cycling is the most efficient method to control the transceiver power in which it controls the radio's sleep/wake state. The power consumption in radio model, IEEE 802.15.5 TelosB TRP2420CA [4] consumes 1µA, 21µA and 23 mA in sleep mode, idle mode and receive mode, respectively. Therefore, to put the transceiver radio efficiently in sleep mode and avoid unnecessary data transmission is vital to save energy. This is done by the MAC layer, which is the second layer of the IEEE 802 OSI reference model. The MAC layer is a foremost reason to provide the reliability and efficiency in a shared medium for a WSN. The MAC layer is responsible for channel access policies, scheduling, error control, buffer management, framing, addressing and flow control. A WSN requires a MAC protocol to consider energy efficiency, reliability, and latency and high throughput as major priorities to in-house with sensor's limited resources and to avoid superfluous power consumption.

There are two schemes in which the data packets can be ported in a sensor network: (i) Synchronous scheme (ii) Asynchronous scheme. The receiver-initiated MAC (RI-MAC) protocol [5] in the asynchronous scheme is more promising and serves better throughput, less latency and bandwidth utilization compared to the sender-initiated MAC protocols [6-8]. The RI-MAC protocol [5] is mainly designed to reduce the idle listening duration and protocol overhead which caused more latency in sender-initiated MAC protocols. In RI-MAC protocol [5] only the receiver node is responsible to initiate the data communication. Whenever a sensor node has data packet in the transmission_queue, the sensor node (sender node) transits to active mode and waits for the intended receiver node to transit to its active mode. On the other hand, the receiver node transmits a beacon frame for a shorter duration to its neighboring sender nodes that it is "READY" to receive the data packets. So, when the sender node receives the beacon frame, it immediately sends an ACK packet and start to transfer the data packets after a random delay to avoid collision with neighboring sender nodes to the receiver node. After the timer expires in the receiver node it goes to the sleep mode.

Revised Manuscript Received on 30 May 2019.
***** Correspondence Author
P Rachelin Sujae*, Research Scholar/ECE, Bharath Institute of Higher Education and Research, Chennai, India.
S Arulselvi, Associate Professor/ECE, Bharath Institute of Higher Education and Research, Chennai, India.

Hence by efficiently manipulating the beacon duration, the efficiency of the RI-MAC protocol can be improved in terms of energy efficiency and spatial reusability. In this paper, a novel beacon scheduling mechanism named ES-MAC is implemented over the traditional RI-MAC protocol. This scheduling mechanism reduces the packet latency by avoiding the packet staying at each sensor node, eliminates sender schedule conflict in case of multiple senders and also the nodes are sink-aware. A random route discovery mechanism is implemented as a part of the setup. Through this, the neighboring nodes are made to efficiently coordinate with each other using this novel beacon scheduling mechanism in RI-MAC, which adaptively increases the throughput with less latency and eventually increase the network lifetime. The comparative study between the traditional RI-MAC, REA-MAC, R-MAC and the proposed MAC is made using the ns-2 simulator clearly reflect that the proposed MAC outperforms the existing protocols [5,3,3].

The rest of the paper is organized as follows: In Section 2, problem statement is given. In section 3, the related work about the RI-MAC protocol is discussed. In Section 4, the design of the RI-MAC is implemented with the novel scheduling algorithm. In Section 5, the power modeling of the sensor node is formulated and Section 6 presents the experimental setup, comparison evaluation of traditional RI-MAC, REA-MAC, R-MAC and proposed MAC using ns-2 simulation. Lastly, in Section 6, the conclusion of this research work is given.

II. RELATED WORK

A very limited Receiver-Initiated MAC protocols that have been proposed [7-8] in view of the data aggregation-delay. In this paper, the attempt is to propose a RI-MAC protocol implemented with duty cycling combined with sink aware protocol with different traffic conditions in WSN. The receiver-initiation mechanism was first implemented for the sensor networks in PTIP (Periodic Terminal Initiated Polling) [9], but only for infrastructure-based set-up sensor networks, where the access point is fixed and energy unconstrained. In PTIP, the sensor node wakes up periodically and polls a poll-packet to an access-point to which the sensor node is associated with. Upon receiving the poll-packet, the access-point starts forwarding all the buffered data packets. But the PTIP infrastructure is fixed and is very different from ad-hoc WSN, where multi-hop routing, topology customization is common and the sensor battery capacity is limited. Another receiver-initiated system known as Low-Power Probing (LPP) introduced by Koala System [3] was designed for bulk data downloads from different sensor nodes, for non-real time applications. In LPP, the downloads are initiated by the gateway, which allows the node to be in sleep-state until it receives the gateway’s download initiation. With LPP each sensor node periodically broadcasts a short probe request and upon receiving the acknowledgement, the node remains active to receive the data packets or else it will go to sleep-state. Garcia-Luna-Aceves et al. in [3] proposed a RI-collision avoidance scheme for general wireless networks during the time when collision was a major concern and power efficiency was a low key area of research. Yajun Sun et al. [5] applied the mechanism of RI-MAC to duty cycle MAC protocols for ad-hoc sensor networks by coordinating the neighbor nodes using a beacon frame [refer section III].

In REA-MAC [2] and R-MAC [3], the authors extend RI-MAC [5] by implementing a topology discovery protocol and scheduling beacon transmission based on its depth. Here a node selects its slot using TDMA approach to broadcast its beacon according to its depth. The data packets are lifted up to the sink node in a pipelined fashion in a single operational cycle. Few data packets remain in the queue at closure of the corresponding slot and it can transfer in succeeding operational cycle. R-MAC [3] is an enhanced protocol version of REA-MAC [2] in correspondence with the avoidance of beacon collision among the neighboring nodes. REA-MAC [2] evades the beacon collision by randomizing the beacon timings among the neighboring nodes within the selected TDMA slot. While, in [3] the neighboring nodes coordinate the beacon timings as the receiver node predicts the wake-up time of its child nodes. More fairness is achieved in RC-MAC [3] by reducing the scheduling interference using a tree topology. It [4] follows a parent-child paradigm where the parent acts as the coordinator node for gathering the data packets. Hence these protocols efficiently utilize the bandwidth by avoiding retransmission of data packets.

IRI-MAC[3] is an improved version of RI-MAC[5] and RC MAC[4]. A Clear Channel Assessment (CCA) is generated by an equation in IRI-MAC [2] to identify the channel clear status. It timestamps non-repetitive schedule for different nodes and hence avoids collision by intimating the other nodes within the same interference range. This [3] performs a CCA and wait for a random time for the next beacon signal to transmit the data packet. The hidden terminal problem in RI-MAC [5] is overcome by IRI-MAC [3].

Introduction of Wakeup Adapting Traffic (WAT) [6] out-performs the conventional RI-MAC[5] and X-MAC[37] to achieve high packet delivery ratio, low packet delivery latency with minimum duty cycle under wide range of traffic. It adopts an adaptive traffic policy which enables the receiver to choose its own wakeup interval depending on the previous traffic arrived by extending the linear congruential generator used in PW-MAC [6] and O-MAC [7]. Also, it adopts an efficient predictive wakeup mechanism in successive cycle in the WAT Beacon to fetch the next wake-up time of the source and the receiver.

By closely studying the protocol designs of [5,2,3], a research gap is identified in the data gathering mechanism of RI-MAC protocol. This includes minimization of idle listening period when the traffic load is dense, latency caused by looping of data packet in network and the cost of latency due to collision and re-transmission of data packet. The proposed ES-MAC protocol is designed to overcome these drawbacks.

III. PROBLEM STATEMENT

The WSN nodes can perform a full duplex data transfer operation meaning, it can send/receive data at the defined fixed intervals.

The MAC layer is primarily responsible for clearing the channel to initiate the data transmission between source and the sink. In a sensor network, the intermediate node will bypass or forward the data traffic to the destined destination. In RI-MAC protocol [5] the data transmission is initiated once the beacon frame is exchanged between the receiver node and any intended sender node over a common time schedule. When the radio is ON, the receiver node(s) transmit the beacon frame for a short duration and neighboring sender nodes if in active-state respond back to the intended receivers else the radio will go to sleep_state. As the receiver broadcasts the frame to multiple senders, it leads to collision without knowing its peer. The limitations of the RI-MAC are: (i) Each node sends/receives/forwards beacon frame and data packet, unaware of the sink node position. (ii) Collision and re-transmission when multiple sender nodes try to establish the communication. (iii) Data scheduling is carried over whenever the CPU collects the data from the sensor.

Based on the understanding, data transmission is initiated whenever the sensor kicks off with the data. In this case, nodes unnecessarily use the communication channel for transmitting a small chunk of data. Also, in the case of adverse traffic conditions, due to unawareness of the sink node path leads to packet looping in the ad-hoc network. Thereby the data communication occupies more network bandwidth which leads to poor network performance in terms of data delivery delay which in turn increases the total network delay and more number of packet collisions. Therefore, in RI-MAC protocol, delay is caused in scheduling the beacon packets, re-transmission and also while traversing the data packet to the sink node.

To resolve the beacon collision issue and to improve the delay, ES-MAC is proposed, which is an enhancement of RI-MAC protocol. The key idea of this novel architecture is to schedule the beacon control packet with the schedule timings based on the data buffer size. It also addresses the sink node path with the hop_count parameter from the source node to the sink node to avoid the collision and retransmission. Due to this technique, the model proposed above is not aware of data sensitive scheduling though the system is capable of scheduling earlier arrival-based node scheduling.

IV. OVERVIEW OF RECEIVER-INITIATED MAC PROTOCOL MECHANISM

This paper proposes a sink-aware beacon scheduling mechanism implemented over a RI-MAC paradigm as it tends to decrease the latency caused by idle listening and thereby increase the network life-time. The RI-MAC paradigm of communication is shown in figure 2. Here each node wakes-up periodically to check for incoming data packets. Node B (receiver node at instance) announces its neighboring nodes in the sensing region (node A & node C) that it is READY to accept the data packets by broadcasting a base beacon and waits for a short duration to receive the data packets. This base beacon frame acts like a trigger to establish data communication between two nodes. If no data packet arrives during this period, the receiver node A will go to sleep_state.

On the other hand, if the sender node has a data packet in the transmission queue, the sender node enterinto active_state and listens to the channel for the base beacon frame from the intended receiver node B. Once it receives the base beacon, the sender node forwards the data packets in the transmission queue until the timer expires in node B and the receiver node sends an ACK frame. This ACK frame plays a dual role as acknowledgement to the previous data packet, as well as request to initiate the next data packet transmission incase the sender node has pending data packets to send in the transmission queue. Meanwhile, if a contending node (node C) tries to access the channel at the same time, the data packets collide with each other and the node transmits the frame after a random back-off. Hence the transmission challenges of the contending nodes lead to maximization at the cost of collision. In addition, the RI-MAC paradigm visibly shows that the idle listening period of the sender nodes can be significantly reduced if the sender node predicts the receiver node's wake-up schedule and wakes up slightly before the receiver node. Another property of the RI-MAC is that the sender can request a base beacon frame from an intended receiver. This request from sender is a beacon frame with Back-off window (BW), which acts like RTS and if the intended receiver is awake, it replies with a base beacon frame. In Fig[b], sender node A sends a beacon_req to its contending receiver node B to initiate the data_transmission by sending a base beacon. Rest of the mechanism follows the normal mechanism of fig[a].

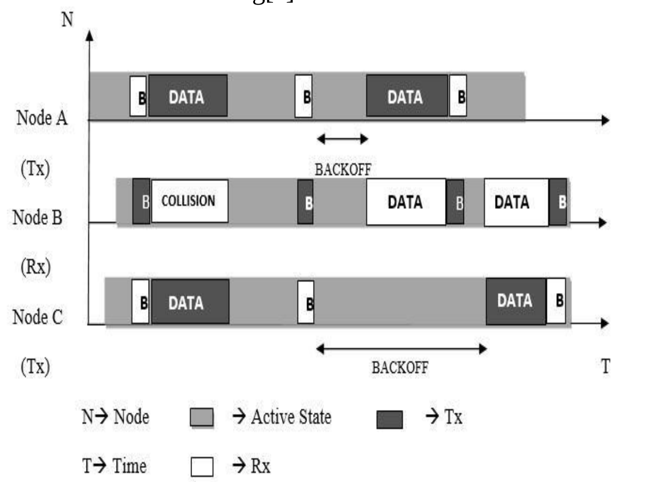


Fig.1a RI-MAC Beacon Transmission with Back-off window

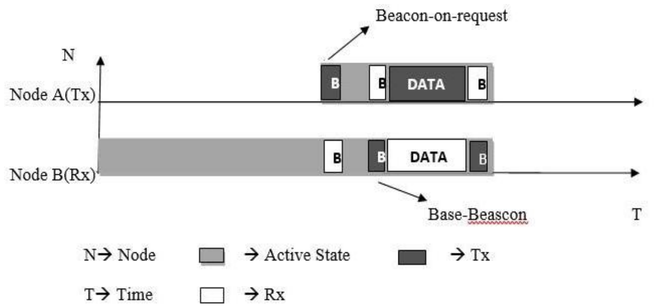


Fig.1b RI-MAC Transmission with Beacon-on-request

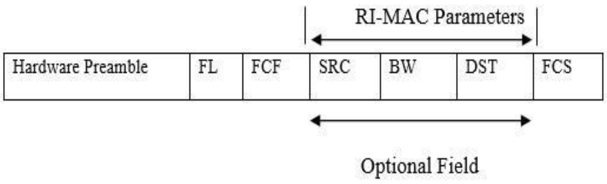


Fig.2 RI-MAC Beacon Frame Format for IEEE 802.15.4 transceiver. The Standard fields of IEEE 802.15.4 – FL (Frame Length), FCF (Frame Control Field), FCS (Frame Check Sequence)

V. PROPOSED ES-MAC PROTOCOL DESIGN

ES-MAC protocol is an enhancement of RI-MAC protocol. ES-MAC follows all the other mechanism similar to RI-MAC except the scheduling phase. In RI-MAC, each sensor node wakes up periodically to check for its incoming data packets. In active state of the sensor nodes it keeps broadcasting the beacon frame to notify its neighboring sensor nodes that it is in active state and ready to receive the data packets. Meanwhile the neighboring sensor node that has data in the transmission queue, continuously listens to the channel to forward the data packets. Upon receiving the beacon frame from the receiver sensor node the neighboring sensor node in active state starts forwarding the data packets.

The ES-MAC protocol differs from the RI-MAC protocol as the sensor nodes wakes up in its operational cycle according to its cross-layer routing information [**REA**]. Due to its cross-layer design it reduces the number of multihop to reach the sink node. Given below are the main functions of the proposed ES-MAC:

1. Establishing a route discovery mechanism with sink node as the root node and extract the neighborhood connectivity,
2. Prioritizing the sender nodes schedule according to the MAX data_payload in data buffer, after receiving the beacon frame from the receiver.
3. Utilizing the time slot based on the senders Data_Payload and the receiver Data_Threshold,
4. Managing the clock drifts by managing a local synchronization time,
5. Changing the MAC behavior according to network prioritization,
6. Identifying the next immediate peer to reach the sink.

The proposed MAC initially establishes the communication as contention-based scheme where it uses the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) with random back-off to avoid collision, where each transmission is initiated by Request-to-Send (RTS) followed by Clear-to-Send(CTS) and DATA/ACK**.** During the initialization phase, the Base station sends HELLO packet to build up a route discovery database. The ES-MAC protocol is implemented over the existing RI-MAC protocol as shown in figure 3 which reduces unnecessary traffic overhead in the network by keeping the data payload size as a parameter. The receiver node extracts the sender's data size after sending the beacon frame and prioritizes which sender node receives the data packet first. A detailed discussion is done in section 5.

The proposed system architecture is categorized into four phases. They are,

1.Network learning phase
2.Scheduling phase
3.Data sensitive slot assignment phase
4.Cross-layer based Sink aware data routing phase

The modified header structure of ES-MAC is given in figure 3. The changes proposed in the existing RI-MAC beacon frame with additional decision parameters like senders Packet size and sink router information are included.

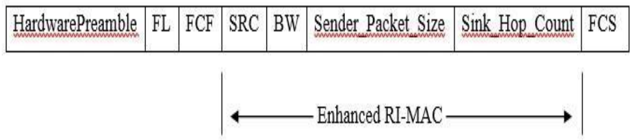
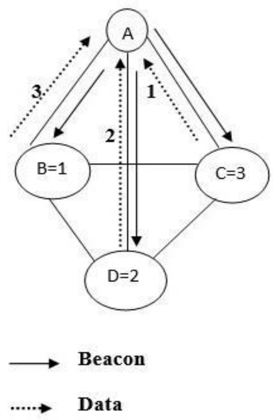


Fig.3 ES-MAC protocol frame structure

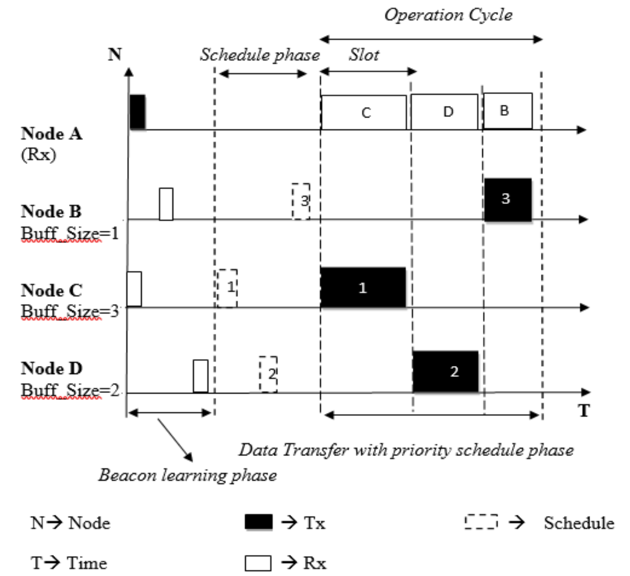
A. Frame Transmission

Figure 4 illustrates the behavior of the frame transmission while forwarding the data packet to the sink node. Here, a scheduling phase is introduced in between the beacon_Tx and data_Tx phase to avoid unnecessary delay caused by collision and re-transmission of the data packet between different nodes. The node shares a common operation cycle to forward the data packets to the sink node. It is assumed that period of the operation cycle corresponds to the bandwidth. The operation cycle is divided into N slots which is adaptive according to the sender data_size.

When a receiver node transmits a beacon, it waits for its neighboring nodes to send an ACK. The ACK packet contains the information about the node_Id and hop_count from the sink and the data_size. The receiver node stores all the information in its database in descending order according to the data_size i.e., the sender node with the highest data_size gains the priority to send the data first. The receiver node generates and broadcasts a schedule packet. The schedule packet notifies which node to start the data transmission. After forwarding the data packet, the sender node will go to sleep_state. Hence, data collision is avoided and there is no need for retransmission of the data-packet.



(a) Data Transmission



(b) Data Transfer with priority schedule phase

Fig.4 Proposed model for frame transmission

The slot size adaptively calculates and varies with different data_size. A detailed discussion is about slot assignment is done is ***section [5.4].***

The illustration in fig.4(a)(b) shows a simple mesh topology in which sensor node C wins the channel first as it has maximum buffer size of 3bytes. The sensor node D has a buffer size of 2bytes and sensor node B has a buffer size of 1byte. The significant mechanism of the ES-MAC is schedule according to the buffer size forward the data packet without any collision with its neighboring sender nodes. After scheduling , the receiver nodes sends a CTS from to each node to forward the data packet. If a node did not receive a CTS frame from the receiver node, it is understood that the data forwarding has been scheduled in the next operational cycle and it can go back to sleep_state. The RTS/CTS frame transmission is not shown in the illustration for understanding purpose. In the above illustration it is shown that all the nodes participates in data transmission as the data_size can accommodate in a single operational cycle.

B. Network learning phase

The proposed ES-MAC operates in 70:30 ratio of sleep-state and wakeup-state, respectively, of the existing RI-MAC protocol, changes are only made in the 30% of the wake-state of the sensor node. Each sensor node has two states: wake-state and sleep-state. In the wake-state, the sensor node can operate either as a transmitter or a receiver node depending on the protocol scheduling policy.

Algorithm 1: Network learning phase

Let total number of nodes be N & let n be the number of nodes in the sensing region, varying from n1 ....ni.

$$\text{Accumulation of nodes } N_i = \sum_{i=1}^{N} Ni$$

Let T be the total time slot

Wait until Data_Time (DT$_i$) ¥ each node varying from n1 .....ni

if event e_i receive at the dataframe Dt_i,

if event e_i is control frame then

construct neighbor database NB_i varies from NB_i.....NB_N.

update NBD_i....NBD_N with Src_Id, Hop_Count, Packet_Size

$NB_i = T_i + T/NT_i * \Delta\ T$

$$NBDi = \sum_{i=1}^{N} NBDi * T/NTi * \Delta\ T$$

else if

if Data_Time (DT$_i$) expires then

Sleep_State until next frame

End if

End if

The format of the ES-MAC beacon header is given in fig 5. The maximum number of nodes that participates in the network built be N. The sink initializes the Sink_hop_count as zero and broadcasts the packet to its neighbors within its transmission range. This accumulation of nodes within the sensing region is given by

$$\text{Accumulation of nodes } N_i = \sum_{i=1}^{N} Ni \qquad (i)$$

This broadcast expires at T_{INIT} secs.

In the Data_Time (DTi) if any event is detected by the sensor node is received in the transmission queue. If the event is a control frame then a neighboring database NBi is constructed and updated with the Src_Id, Hop_Count and Packet_Size.

$$NB_i = T_i + T/NT_i * \Delta\ T \qquad (ii)$$

$$NBDi = \sum_{i=1}^{N} NBDi * T/NTi * \Delta\ T \qquad (iii)$$

Equation (i) and (ii) formulates the neighboring database for a single node and for the entire network respectively. If DTi expires, the sensor node will go to sleep-state until reception of the next frame.

The pseudocode for the network learning phase in given in Algorithm 1.

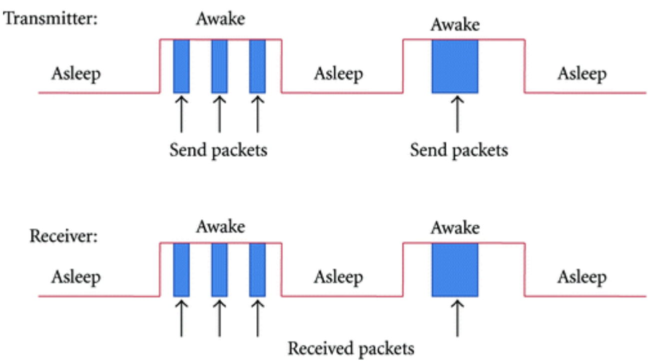


Fig.5 Packets Transceiver

C. Scheduling Phase: Duty Cycle Operation

During the duty operation cycle of the node, the relay node and the source node undergo beacon scheduling and forwarding mechanism to schedule the packet to the destination. The selection of peer is estimated based on the highest available data from the peer with the shortest path towards the sink based on the cross-layer routing information.

This enables the sender to schedule highly available data forwarded to the sink node to ensure the maximum data held by each node. Each node can act as a sender or a relay/intermediate node, which participates in the active topology network. This mechanism ensures the data transmission on the network to reach the sink based on highly available data payload from the sender or from the peer node. Additionally, the core logic is to schedule the nearest node with maximum data load, which is selected by the ASYNC based RI-MAC rather than a peer with a beacon arrival timing. This ensures data scheduling with the maximum available information. The Peer selection is purely based on the sender_packet_size information available in the received frame if the node is a relay node. As it is a duty cycle-based protocol, each node has data-time and sleep-time, to save power consumption. However, the sensor nodes are capable of sensing data during sleep_time and data collection will not be preempted during data frame transmission. This means that the sensor nodes are operational during the sleep-time slot, polling the data in the available sensors and storing it in the data buffers. The CPU operational cycle will be independently calculated by the sleep-time and data-time. In short, data collection is done in the sleep-time when the node radio is OFF and the data scheduling is available when the node radio is ON.

D. Data Sensitive Slot Assignment Mechanism in ES-MAC

To improve the utilization of the time slot given for a data_time, maximum data payload is scheduled first from the sender. If the receiver completes the sender's data transmission within the operational scheduling cycle then it would be (T/1) time slots, where the sender's transmission time (T) is directly proportional to the size of the data (D). The time of transmission is estimated based on size of the Data and link speed between the sender and receiver.

Where,

Time of Transmission (T) = Data_Size /Speed of transmission (iv)

If equation (iv) is applied for all sender nodes and receiver nodes, then the whole operational cycle slot is used by one node. If T >= (Data Size/Speed of Tx) for all sender nodes and receiver nodes then, the whole operational cycle slot is used by more than one peer if and only if the number of peer nodes total transmission time is equal to the total operational cycle of the node i.e, T/N time slots. If, T <= (Data Size/Speed of Tx) for all sender and receiver, then the whole operational cycle slot used by one peer and remaining data is scheduled in the next time slot ie., $(T + \Delta T)/N$. To summarize, in order to schedule the sender to receiver data transmission, a data-sensitive scheduling ensures that a MAX data payload sender wins the receiver's channel.

The pseudocode for the scheduling and slot assignment is given in **Algorithm 2.**

Algorithm 2: Data slot assignment algorithm

Let T_i be the total time of transmission for the slot S_i,
Let Mi be the data_size of the receiver frame Ri,
Let BWi be the bandwidth of the node,
Let Rsi be the residual size and i varies from 1….N,
Wait until Data_Time DTi
Let Y be the Data_Buffer threshold ¥ receiver nodes,
ni € N ¥ n varies from n1…..ni
Proceed,
If ($R_i > Y$)
  $Rs_i = M_i - Y_i$
    Loop until all frames in the time slot DT_i,
End
If $R_{si} > 1$ then
$T_i = [(M_i * \Delta T_i) + (Rs_i * \Delta T_i)] * S_i/N_i$
Else
$$T_i = \sum_{i=1}^{N} S_i * (M_i + \Delta T_i)/N_i$$
else if
  sleep until Data_Time
end

E. Sink aware data routing Phase

Since wireless sensor network is a mesh network, the destination path to the sink has multiple routes. Though all the sensor node runs with a routing stack, it finds its route information with a DSDV link state routing-based routing. For designing this novel architecture, the problem exists in identifying the next immediate peer to reach the sink node. To ensure the shortest path to the sink, the initial phase is to schedule the sender that wins the receiver's path. To make this a proactive and predominant decision, a cross-layer routing information is utilized to forward the frame to the next nearest hop. Referring to the frame structure above [fig], the sink_hop_count is added with the received frame.

With this information, during the beacon request the receiver collects all the sender information and maintains a neighbor information database ordered from the minimum to maximum sink_hop_count. Based on the neighbor information, receiver node changes its receiver_state to sender_state and waits to receive the beacon frame from the receiver intended node and route to the sink node. This process repeats until it reaches the sink node. The pseudocode for the sink-aware data routing is given in **Algorithm 3**.

Algorithm 3: Sink-aware Data routing

Let NB be the total number of nodes in the neighbor database.
Let h_i, NID_i, DS_i be the tuple values with Hop_Count, Node_ID and Data_Size
Assume,
  ¥ Nodes n1,n2,n3……n_i € NB
   Where NB is the neighboring database.
 Tuple any TP = {h_i, NID_i,DS_i}
 If dti is true then,
  ¥ nodes in NB, TP
  If($n_i \rightarrow h_i < n_j \rightarrow h_i$) &&
   ($n_i \rightarrow ds_i > n_j \rightarrow ds_i$) then
   Wait until Tx slot;
   Send data frame at n_i
  End if
End if
Else
Sleep till Data_Time

F. Power Modeling of the Node

In the RI-MAC protocol for diverse dense traffic conditions, the low power consumption model is not only defined by the control packet exchange.

RI-MAC Total power consumption = E(t) at slots +E(t)Carrier Sense at each slot (1)

The power consumption of RI-MAC protocol is a mixture of hardware overhead (HWOH) and software overhead (SWOH), because this protocol talks about the low power MAC modeling with the data traffic situations, i.e.,

Total power consumption=$E(t)_{HWOH}+E(t)_{SWOH}$ (2)

With respect to SWOH, the modeling carries delay during context switching, execution process as well as protocol parameters including delay with respect to DIFS, SIFS, BO, Overhearing delay and Idle listening delay On the other hand, HWOH carries delay with respect to clock jitter, clock drifting parameters and race conditions. These parameters are identified for modeling the power consumption in the RI-MAC protocol. An expression is derived using these parameters to identify the power consumption when the node is in idle_state. Although working of the power consumption model using different parameters is an idea, the Poisson distribution for modeling the diverse traffic conditions cannot be considered as the rate of traffic is not stable. It is possible that control traffic or data traffic could be exponentially increasing or decreasing. Hence, an Exponential distribution function is applied for diverse traffic conditions to calculate the power as it follows the Poisson process as well as the random process. In this condition, the assumption is that events occurring are independent of time and rate. The existing RI-MAC protocol is capable of assigning the channel allocation. Though 80% of the power is consumed by the radio unit of the sensor node, unnecessary power is consumed by the MAC protocol by the idle listening and by overhearing of the neighboring nodes. Taking the RI-MAC mechanism as the reference model, this paper proposes the analytical model of the control packet for power consumption and also for modeling MAC with diverse switching conditions, which is the key idea for modeling a MAC with runtime traffic conditions. The system overhead includes MAC protocol parameters and software overhead. Therefore, the total power consumption is given by,

$P(t) = P_{MACOH}(t) + P_{SWOH}(t)$ (3)

Here, the power consumption of node is measured with respect to time (t), the mac overhead (MACOH) is measured by transmission $T_{(x)}t$ and reception of packets $R_{(x)}t$. An abstract model for MAC overhead is given by,

$P_{MACOH}(t) = P\ T_{(x)}t + P\ R_{(x)}t$ (4)

The actual model solely depends on $T_{(X)}t$ and $R_{(X)}t$ which is varied with time factor T. The modified equation can be written as,

$P_{MACOH}(t) = [P\ T_{(x)}t + P\ R_{(x)}t] * T$ (5)

The hidden parameters affecting the power consumption of MAC model is idle listening and over hearing time. Including parameter $P_it(t)$ & $P_ot(t)$ in equation (5) derives

$P_{MACOH}(t) = [P\ T_{(x)}t + P\ R_{(x)}t + P_it(t) + P_ot(t)] * T$ (6)

Where T is the time factor.

When a sensor node has a packet for the data transmission it includes various delays like the carrier sense delay D_{CS}, Back-off delay D_{Bo}, processing delay, queuing delay, transmission delay and sleep delay combine as total delay D_{TL}. The total power consumption of the transmission is given by,

$P\ T_{(x)}t = [D_{CS}+D_{BO}+D_{TL}+D_{SIFS}+D_{DIFS}]*P_t$ (7)

The total power consumption of the reception is given by

$PR_{(x)}t=[D_{SIFS}+D_{DIFS}+BACK_{(t)}]*P_t$ (8)

Where P_t is power consumption

Substituting equation 7 & 8 in equation 6,

$P_{MACOH}(t)=[(D_{CS}+D_{BO}+D_{TL}+D_{SIFS}+D_{DIFS})*P_t+(D_{SIFS}+D_{DIFS}+BACK_{(t)})*P_t+P_it(t)+P_ot(t)]*T$ (9)

Power consumption of the SWOH depends on CPU process or network process.

$P_{CPU} = [P_{CPU}\ processing_{(t)} + P_{CS}(t) + P\ cache\ eff\ (t) + P\ stall\ time\ CS\ (t)]$ (10)

Where, P_{CPU} processing $_{(t)}$ is actual processing time of CPU when the system is in high traffic condition

P_{CS} (t) is power consumption of CPU at context switching over head, P cache eff (t) is CPU stall time due to cache memory delay and P stall time cs (t) is delay in time due to context switching.Substituting equation 9 & 10 in 1 & the total power consumption will be

$P(t)=[(D_{CS}+D_{BO}+D_{TL}+D_{SIFS}+D_{DIFS})*P_t+(D_{SIFS}+D_{DIFS}+BACK_{(t)})*P_t+P_it(t)+P_ot(t)]*T + [P_{CPU}\ processing_{(t)} + P_{CS}(t) + P\ cache\ eff\ (t) + P\ stall\ time\ CS\ (t)]$ (11)

Let equation (11) be the total power consumption of the MAC model in diverse traffic conditions. This power consumption model P(t) is validated with the exponential Probability Density Function because of the events, where event is considered as packet reception R_X and the transmission T_X with respect to time T.

The Probability Density Function of MAC model is given by,

$$f(P(t), t) = \begin{cases} P(t)e^{-P(t),t}, & t >= 0 \\ 0, & t < 0 \end{cases} \quad (12)$$

Where t is time varies 0,1,2…,N

P(t) is the total power consumption by the MAC model is represented in equation 11.

VI. PERFORMANCE EVALUATION

The performance metrics of the proposed MAC is evaluated using ns-2 simulator. To evaluate, multiple comparison is made under sparse network as well as a dense network with different protocols like R-MAC and REA-MAC in terms of energy consumption and data collection delay. In "Random" mechanism each node can randomly select its beacon transmission time over a whole operational cycle instead of choosing a specific time-slot.

A. Scenario and Simulation Setup

A simulation environment is created in NS-2 for a dense network using a mesh topology. Each sensor node is capable of communication within its sensing region equipped with omnidirectional antenna. The radio propagation model used is the two-Ray ground model. The wireless communication parameters are displayed in **Table 1**. Mostly CC2420CA radio parameters are exploited for the simulation purposes which is used in popular motes like MICAz and TelosB.

The sensor nodes are randomly deployed over [1,000 x 1,000]m square area. In a dense network, 200-600 nodes are placed for the simulation purpose. A cross-layer architecture for routing is utilized to find the shortest path to find the sink node. The nodes deployed are considered immobile in this research work. The simulation parameters and proposed protocol parameters are displayed in **Table 2** and **Table 3**. The simulation parameters utilized are exploited from RI-MAC as ES-MAC is an enhancement of RI-MAC.In the proposed MAC protocol, the slot-size and the sub-slot size is kept variable. The slot adjusts according to the packet size.

Table 1 Simulation Parameters [22]

Parameter	value
Channel Bandwidth (BW)	250kbps
SIFS	192µs
Slot Time	320µs
Tx Range	250m
Data_Size	50 Bytes
Beacon_Size	6-9 Bytes
ACK Size	5 Bytes
CCA Check Delay	128µs
Carrier Sensing Range	550m

Table 2 Proposed MAC Parameters

Parameter	Value
Operation Cycle Length	Variable
Number of Slots N	Variable
Min Beacon Gap duration	50ms
Data_Size	Variable
Proposed Protocol Slot Size	Variable
Proposed Protocol Sub-Slot Size	Variable

B. Protocol comparison

The performance metrics of Enhanced Scheduling MAC (ES-MAC) are evaluated and compared against traditional R-MAC and REA-MAC.

1. Latency Evaluation

The latency is evaluated for a dense network consists of 200 to 600 nodes where, latency is defined as the time taken by the generated packets in each node to traverse to the sink node during the simulation time. Fig [ ] compares the latency per hop in a dense network against the proposed ES-MAC, R-MAC, and REA-MAC. The proposed ES-MAC maintains a minimum average latency of 1.5s. Notice that the latency slightly increases when the node density increases more than 450 due to multi-hop network. The proposed ES-MAC outperforms the other conventional MAC due to its cross-layer architecture and elimination of re-transmission technique between the nodes. Secondly, the average latency is calculated for a sensor node with it packet flow rate. The packet flow rate varies according to the network traffic. In fig 6, shows that proposed ES-MAC performs better than the R-MAC. In [R-MAC], it has been already proved that R-MAC outperforms REA-MAC as R-MAC has cooperative scheduling for data transmission. The RI-MAC comparison is omitted as it uses a random forward mechanism whereas in R-MAC, REA-MAC and the proposed ES-MAC forwards the data in operational cycle.

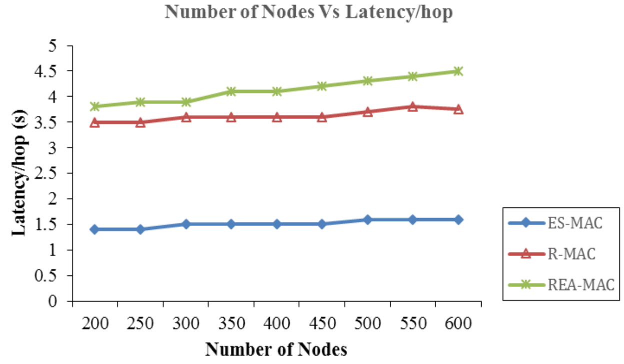


Fig.6 Latency per hop with increasing nodes

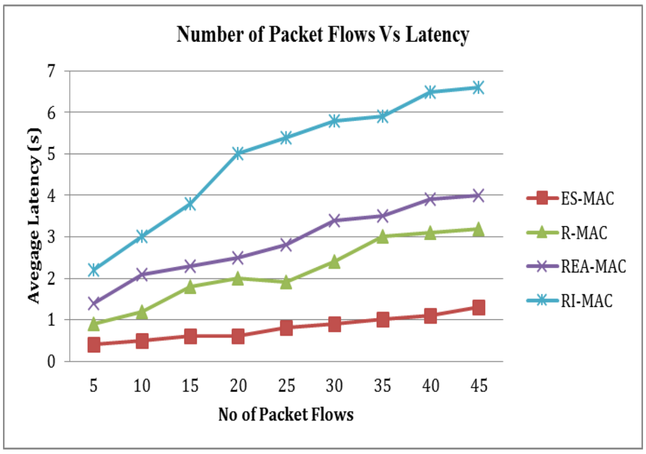


Fig.7 Number of Packet Flows Vs Latency

2. Throughput Evaluation

Throughput is the total number of successful packets received at the sink node during the simulation time measured in bits per second (bps). Throughput evaluation in a sensor network is crucial due to variation in the traffic loads. The graph in fig 7 shows the varying packet flow in the sensor nodes produces different throughputs in the conventional protocols and the proposed ES-MAC protocol. The ES-MAC has a high throughput compared to other protocols.

3. Evaluation based on Data Size

Average latency and throughput calculated over adaptivie data_size and fixed data_size. Unlike the traditional RI-MAC protocols, the sensor node will come to wake state only if it is scheduled in the current operational cycle.

The Protocol evaluation shows that the proposed MAC delivers low latency and high throughput. The following metrics has been calculated which measures the performance of this novel protocol.

VII. CONCLUSION

In this paper, an enhanced scheduling RI-MAC protocol is presented which is data sensitive based on its buffer size.

The collision of data packets are prevented as the node receives its schedule packet prior to the data transmission. And thereby, re-transmission of packets is not required. As this novel RI-MAC is a cross-layer sink aware protocol, unnecessary looping inside the network is eliminated. In emergency network this protocol may not be efficient due to scheduling of data packets in the next coming operational cycle.

REFERENCES

1. A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas,M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita, \A Line in the Sand: A Wireless Sensor Network for Target Detection, Classi_cation, and Tracking," Elsevier Computer Networks, vol. 46, pp. 605{634, 2004.
2. S. H. Lee, S. Lee, H. Song, and H. S. Lee, \Wireless sensor network design for tactical military applications: remote large-scale environments," in Proc. IEEE 28th Conf. on Military communications, MILCOM'09, pp. 911{917, 2009.
3. P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, \Energy-E_cient Computing for Wildlife Tracking: Design Tradeo_s and Early Experiences with ZebraNet," in Proc. ACM 10th Intel. Conf. on Architectural Support for Programming Languages and Operating systems (ASPLOS-X 2002), pp. 96{107, Oct 2002.
4. A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, \Wireless sensor networks for habitat monitoring," in Proc. 1st ACM International Workshop On Wireless Sensor Networks And Applications, WSNA '02, pp. 88{97, 2002.
5. S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, \Health monitoring of civil infrastructures using wireless sensor networks," in Proc. 6th Intl. Symposium on Information Processing in Sensor Networks, IPSN 2007, pp. 254{263, April 2007.
6. G. Hackmann, F. Sun, N. Castaneda, C. Lu, and S. Dyke, \A holistic approach to decentralized structural damage localization using wireless sensor networks," Computer Communications, vol. 36, no. 1, pp. 29 { 41, 2012.
7. M. Ceriotti, L. Mottola, G. P. Picco, A. L. Murphy, S. Guna, M. Corra, M. Pozzi, D. Zonta, and P. Zanon, \Monitoring heritage buildings with wireless sensor networks: The Torre Aquila deployment," in Proc. 8th Intl. Conf. on Information Processing in Sensor Networks, IPSN '09, pp. 277{288, IEEE Computer Society, 2009.
8. K. Chebrolu, B. Raman, N. Mishra, P. K. Valiveti, and R. Kumar, \Brimon: a sensor network system for railway bridge monitoring," in Proc. ACM 6th Intl. Conf. on Mobile systems, applications, and services, MobiSys '08, pp. 2{14, 2008.
9. M. Wang, J. Cao, M. Liu, B. Chen, Y. Xu, and J. Li, \Design and implementation of distributed algorithms for wsn based structural health monitoring," Int. J. Sen. Netw., vol. 5, pp. 11{21, Feb 2009.
10. Y. Hao and R. Foster, \Wireless body sensor networks for health-monitoring applications," Physiolog- ical Measurement, vol. 29, pp. 27{56, Nov. 2008.
11. J. Taneja, R. Katz, and D. Culler, \De_ning CPS Challenges in a Sustainable Electricity Grid," in Proc. IEEE/ACM 3rd International Conference on Cyber-Physical Systems, ICCPS '12, pp. 119{128, 2012.
12. V. Gungor, B. Lu, and G. Hancke, \Opportunities and Challenges of Wireless Sensor Networks in Smart Grid," IEEE Trans. on Industrial Electronics, vol. 57, pp. 3557{3564, Oct. 2010.
13. Z. Fadlullah, M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, \Toward intelligent machineto- machine communications in smart grid," IEEE Communications Magazine, vol. 49, pp. 60 {65, April 2011.
14. Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng, \Occupancy-driven energy management for smart building automation," in Proc. 2nd ACM Workshop on Embedded Sensing Systems for Energy-E_ciency in Building, BuildSys '10, pp. 1{6, 2010.
15. K. Randy, C. David, S. Seth, A. Sara, C. Yanpei, D.-H. Stephen, D. Prabal, H. Mike, J. Xiaofan, K. Laura, K. Andrew, L. Ken, O. Jorge, M. Prashanth, R. Evan, T. Jay, H. Je_, and S. Sushant, \An information-centric energy infrastructure: The berkeley view," Journal of Sustainable Computing: Informatics and Systems, vol. 1, pp. 7{22, March 2011.
16. A. Ilic, T. Staake, and E. Fleisch, \Using Sensor Information to Reduce the Carbon Footprint of Perishable Goods," IEEE Trans. on Pervasive Computing, vol. 8, pp. 22{29, Jan 2009.
17. N. Wang, N. Zhang, and M. Wang, \Wireless sensors in agriculture and food industry{Recent development and future perspective," Computers and Electronics in Agriculture, vol. 50, pp. 1{14, Jan.2006.
18. K. Langendoen, A. Baggio, and O. Visser, \Murphy Loves Potatoes: Experiences from a Pilot Sensor Network Deployment in Precision Agriculture," in Proc. 14th Int. Workshop on Parallel and Distributed Real-Time Systems (WPDRTS), pp. 1{8, apr 2006.
19. T. Wark, P. Corke, P. Sikka, L. Klingbeil, Y. Guo, C. Crossman, P. Valencia, D. Swain, and G. Bishop- Hurley, \Transforming Agriculture through Pervasive Wireless Sensor Networks," IEEE Trans. On Pervasive Computing, vol. 6, pp. 50{57, April-June 2007.
20. J. M. Lees, J. B. Johnson, M. Ruiz, L. Troncoso, and M. Welsh, \Reventador Volcano 2005: Eruptive activity inferred from seismo-acoustic observation," Journal of Volcanology and Geothermal Research, vol. 176, pp. 179{190, Sep 2008.
21. Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci,"A Survey on Sensor Networks"-Aug 2002, IEEE Communications Magazine. Stefan Mahlknecht, Michael Bock, On the use of High Bit Rate Transceivers for Low Duty Cycle Wireless Sensor Networks, IEEE AFRICON.
22. A. Bachir, M. Dohler, T. Watteyne, K. Leung, MAC Essentials for Wireless Sensor Networks, IEEE Commun. Surveys Tutorials 12 (2010) 222–248. CC2420 Single-chip 2.4 GHz rf transceiver, http://focus.ti.com/lit/ds/ symlink/cc2420.pdf.