A Receiver Initiated Low Delay MAC Protocol for Wake-up Radio Enabled Wireless Sensor Networks

Ripudaman Singh and Biplab Sikdar, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a broadcast-based receiver-initiated low-delay wake-up radio (RI-LD-WuR) Medium Access Control (MAC) protocol for wireless sensor networks. RI-LD-WuR MAC improves delay and energy-efficiency by reducing the packet collisions. We reduce the packet collisions by reducing the contention during channel access. We design a new cycle structure and divide the nodes into K groups to decrease the contention during channel access. We propose an algorithm to determine the value of K that provides the lowest delay. To evaluate the performance, we develop a discrete-time Markov-chain (DTMC) model and drive expressions for packet delivery ratio, average energy consumption per transmitted data packet, and delay. Numerical results demonstrate the superiority of the RI-LD-WuR MAC protocol.**

***Index Terms*—Delay, energy-efficiency, medium access control (MAC), throughput, wake-up radio, wireless sensor networks.**

I. INTRODUCTION

THE advent of 5G has brought high data rates and massive connectivity in wireless communication [1]. As a result of this, the Internet-of-Things (IoT) coupled with wireless sensor networks (WSNs) are being used for developing smart applications, e.g., smart grid, smart cities, intelligent automation, etc. [2]. In such applications, sensor nodes detect the events of interest and report the sensed data to the sink node. Then the collected data is processed and analyzed to make a decision or to extract valuable information [2]. In many IoT applications, energy-efficiency and low event-reporting delay are the two crucial requirements. Therefore, due to low energy consumption in idle listening and overhearing, a wake-up radio (WuR) enabled sensor node is the most suitable choice as a detection station. A WuR enabled sensor node contains two radios: one WuR and one main radio (MR). The function of WuR is to wake up the MR on wake-up call (WuC) reception for data packet exchange. A WuR consumes 1000 times lower power than that of the MR [3]. Idle listening occurs when the WuR/MR is active in the receive mode but the channel has no ongoing transmissions. Overhearing occurs when a node picks up packets that are destined to other nodes.

In broadcast-based receiver-initiated WuR MAC protocols, a receiving node broadcasts an unaddressed WuC to wake up its neighboring sensor nodes for data packet transmission [4]–[6]. A sensor node receives WuC through its WuR. In receiver-initiated WuR (RI-WuR) and receiver-initiated consecutive packet transmission WuR (RI-CPT-WuR) MAC protocols [6], on WuC reception, a sensor node wakes up its MR for channel contention if it has a data packet to send. In RI-WuR MAC, a node needs to win the channel access competition before the transmission of each of its data packets. Unlike RI-WuR MAC, in RI-CPT-WuR MAC [6], a node can send multiple data packets one-by-one consecutively after wining the channel access competition. As a result of this feature, in RI-CPT-WuR MAC, packet collisions reduce due to a reduction in channel access competition. With a decrease in packet collisions, both delay and energy-efficiency improve due to a reduction in packet retransmissions. RI-CPT-WuR MAC provides lower delay and higher energy-efficiency in comparison to RI-WuR MAC. In RI-WuR and RI-CPT-WuR MAC protocols, on WuC reception, all the sensor nodes that have a data packet contend for channel access at the same time. It leads to packet collisions due to high channel access competition.

In this paper, we propose a broadcast-based receiver-initiated low-delay WuR MAC (RI-LD-WuR MAC) protocol for WSNs. RI-LD-WuR MAC improves both delay and energy-efficiency by reducing the contention during channel access. For this, we design a cycle structure that contains K equal-size time segments (TSs) for data transmission and one TS for WuC. We then partition the sensor nodes into K (almost) equal-size groups and allow the nodes of the i^{th} group to transmit their data packets only in the i^{th} TS. Further, as in RI-CPT-WuR MAC [6], we allow a node to send multiple data packets one-by-one consecutively after wining the channel access competition. We propose an algorithm to determine the value of K that provides the lowest delay. To evaluate the performance, we develop a discrete-time Markov-chain (DTMC) model and derive the expressions for packet delivery ratio (PDR), average energy-consumption per transmitted data packet (AEC), and delay (D). Numerical results show that RI-LD-WuR MAC outperforms RI-WuR MAC and RI-CPT-WuR MAC with a significant margin in terms of PDR, D, and AEC. We believe this is the first attempt to apply the idea of node grouping to reduce the channel access competition for a WuR enabled WSN.

A summary of the main contributions is as follows:

1) A cycle structure and a synchronization technique are proposed to utilize the idea of node grouping for reducing packet collisions in a broadcast-based receiver-initiated WuR enabled WSN.

Manuscript received May 23, 2020; accepted June 16, 2020. Date of publication June ..., 2020; date of current version October ..., 2020. (*Corresponding author: Ripudaman Singh.*)

Ripudaman Singh is with the Department of Electronics and Communication Engineering, Indian Institute of Information Technology Guwahati, Guwahati-781015, India (*E-mail: ripudaman@iiitg.ac.in*).

Biplab Sikdar is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore-119077 (*E-mail: bsikdar@nus.edu.sg*).

2) For performance evaluation, a DTMC model is developed and closed-form expressions are derived for PDR, D, and AEC.
3) An algorithm is developed to determine the number of groups (K) corresponding to the minimum delay.

The rest of the paper is organized as follows. In Section II, we briefly discuss related work in this area. We describe the RI-LD-WuR MAC in detail in Section III. In Section IV, we describe our proposed DTMC model in detail and obtain expressions for PDR, AEC, and D calculation. In this section, we also describe the algorithm proposed for optimum K determination. In Section V, we compare the performance of RI-LD-WuR MAC with RI-WuR and RI-CPT-WuR MAC protocols in terms of PDR, AEC, and D. Finally, Section VI concludes this paper.

II. Related Work

In recent years, WuR enabled WSNs are increasingly gaining popularity due to lower energy consumption in idle listening and overhearing than traditional duty-cycled (DC) WSNs [3], [7]. However, overhearing and idle listening still happen in the existing WuR MAC protocols when two or more nodes compete with each other for channel access. In addition to this, delay and energy-efficiency of existing WuR MAC protocols degrade due to packet collisions during channel contention. The existing WuR MAC protocols can be divided into two parts: receiver-initiated (RI) [6], [8]–[11] and transmitter-initiated (TI) [12]–[18]. Receiver-initiated MAC protocols can be further divided into two parts: broadcast-based [6], [11] and address-based [9].

RI-MAC [11] is the most representative receiver-initiated MAC protocol, designed to improve the throughput and minimize the power consumption in a duty-cycled WSN. When the RI-MAC is applied to the WuR enabled WSNs, it is referred to as RI-WuR MAC [6]. In RI-WuR MAC, a node can send at the most one data packet on success in channel contention. Unlike RI-WuR MAC, in RI-CPT-WuR MAC [6], a node can send multiple data packets on success in channel contention. RI-CPT-WuR MAC provides low delay and high energy-efficiency than RI-WuR MAC. This indicates that delay and energy-efficiency improve with a reduction in channel access competition. Further, in RI-WuR and RI-CPT-WuR MACs, a node is allowed to transmit data packets whenever it finds the medium idle. Therefore, in a single-hop WSN, all N sensor nodes can compete for channel access at the same time to transmit their data packets. This results in performance degradation due to high channel access competition. Also indicates that packet collisions increase with an increase of N.

Unlike RI-WuR and RI-CPT-WuR MAC protocols, DoRa/DC-DoRa [8] uses separate channels for WuC broadcast and data transmission. Similar to DoRa/DC-DoRa, CMAC [19] also uses a separate channel to send WuC and follows backoff (BO) before a WuC transmission. In [9], Aoudia *et al.* studied multi-hop WuR networks and proposed a relay selection technique to reduce the false wake-up. For a similar reason, ZeroMAC [20] utilized a radio frequency (RF) watchdog to wake up only the nodes on the communication path by sending an unaddressed WuC in a hop-by-hop manner. In [13] and [21], CSMA/CA based schemes are proposed to avoid WuC collisions. Further, in [22], WuCs and data packets are transmitted using two distinct data rates to avoid collisions.

In literature, receiver-initiated MAC protocols are evaluated through discrete-event simulations and analytical models. In [23], Duan *et al.* developed an analytical model to study the system saturation throughput and power consumption of RI-MAC under a star topology. This model is developed assuming that a node can transmit only one data packet on success in channel contention. In [24], Guntupalli *et al.* developed a DTMC model to analyze RI-MAC's performance, considering that a node can transmit multiple data packets on success in channel contention. In [6], Guntupalli *et al.* developed DTMC models for performance analysis of RI-WuR MAC and RI-CPT-WuR MAC protocols. They used the solutions obtained from DTMC models to derive closed-form expressions for throughput, delay, packet reliability ratio, and energy consumption analysis. In [25], Aoudia *et al.* developed an absorbing Markov chain model to analyze TI-WuR, considering that the number of failures follows a geometric distribution. In [12], an M/G/1/2 model was developed for performance analysis of CCA-WuR, CSMA-WuR, and ADP-WuR MAC protocols.

III. RI-LD-WuR MAC Description

In this section, we describe our proposed protocol RI-LD-WuR MAC in detail. Table I describes the notations used in this paper.

TABLE I: DESCRIPTION OF NOTATIONS

Notation	Description
SIFS/DIFS	Short/distributed inter frame space
T_X	Transmission duration of $X \in$ {DATA, ACK, WuC, GROUP}
W	Number of time slots in contention window (CW)
σ	Duration of one CW slot
$\tilde{S}_t$	t^{th} sensor node where $t \in [1, N]$)
P_{tx}^{MR}	Power consumption of MR in transmission
P_{rx}^{MR}	Power consumption of MR in reception
P_{sleep}^{MR}	Power consumption of MR in sleep
P_{idle}^{MR}	Power consumption of MR in idle listening
P_{rx}^{WuR}	Power consumption of WuR in reception
P_{idle}^{WuR}	Power consumption of WuR in idle listening

A. Network Model and Assumptions

In typical IoT data reporting and collecting scenarios, a sink node receives sensed data from multiple sensor nodes. Therefore, as in [6], [12], [14], [26]–[29], we consider a cluster of N sensor nodes that send packets towards a common one-hop away sink node (S). The network model considered in this paper is shown in Fig. 1 where $\tilde{S}_t$, $1 \leq t \leq N$, denotes the t^{th} sensor node. In Fig. 1, N sensor nodes can be viewed as child nodes of S.

RI-LD-WuR MAC is developed assuming that 1) each sensor node contains one WuR and one MR; 2) a unique number is assigned to each sensor node as its address from $[1, N]$; and 3) there is only one channel through which both data and WuC are transmitted by switching the antenna configuration dynamically.

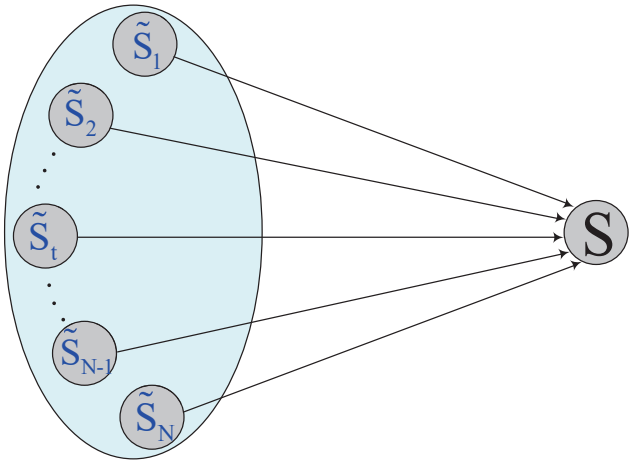


Fig. 1: Illustration of a WuR enabled WSN cluster with N contending nodes plus one sink node as the common receiver.

B. Grouping of Nodes

In this section, we propose a technique to partition the sensor nodes into K equal-size groups in a WuR enabled WSN. Our proposed technique is motivated by the technique adopted in [30] to partition the sensor nodes into K equal-size disjoint sets (DSs) in a WSN. In our proposed technique, we maintain equal sensor nodes of each group in the neighborhood of every sensor node. To achieve this, we consider that each sensor node maintains a Lookup Table (LT) that initially contains its address and group number (GN). GN of each sensor node is initialized with a randomly chosen integer number from the interval $[1, K]$. We use a packet named GROUP that contains the sender's address and GN.

Initially, S broadcasts a GROUP packet containing GN = 0. A sensor node receives the first GROUP packet through its WuR. After that, it wakes up its MR and broadcasts a GROUP packet once every $\widetilde{T}_G$ seconds. A sensor node follows back-off (BO) and clear channel assessment (CCA) before transmitting a GROUP packet. Each sensor node broadcasts α GROUP packets. Further, a sensor node follows Algorithm 1 to update its LT whenever it receives a GROUP packet. In Algorithm 1, we describe the process assuming that node S_u received the GROUP packet broadcast by node S_v where $1 \le u, v \le N$ and $u \ne v$. S_u follows the following four steps to update its LT.

Step 1: In **Step 1**, S_u first checks whether S_v exists in its LT or not. S_u adds S_v's GN in its LT if S_v does not exist in the LT. Further, S_u updates S_v's GN in its LT if S_v exists in the LT and $x \ne$ Rx_GN. Here, Rx_GN denotes S_v's GN in the most recently received GROUP packet. In case of $x > 0$, x denotes S_v's GN in S_u's LT before reception of most recent GROUP packet.

Algorithm 1 Following steps are followed by a node S_u to update its LT on GROUP packet reception from node S_v.

Definitions:
Rx_GN : GN in the received GROUP packet
GN$[S_u][S_v]$: S_v's GN in S_u's LT
node_GN$[S_u][z]$: Number of nodes of GN z in S_u's LT
Initializations: $x \leftarrow 0$, and $y \leftarrow 0$
% **Step 1:** Add/update S_v's GN in the LT %
if $S_v \notin$ LT **then**
  GN$[S_u][S_v] \leftarrow$ Rx_GN
else
  $x \leftarrow$ GN$[S_u][S_v]$
  if $x \ne$ Rx_GN **then**
    GN$[S_u][S_v] \leftarrow$ Rx_GN
  end if
end if
% **Step 2:** Update number of nodes of GNs x and Rx_GN in the LT %
if $x == 0$ **then**
  node_GN$[S_u]$[Rx_GN] $\leftarrow$ node_GN$[S_u]$[Rx_GN]+1
else if $x \ne$ Rx_GN **AND** $x > 0$ **then**
  node_GN$[S_u][x] \leftarrow$ node_GN$[S_u][x] - 1$
  node_GN$[S_u]$[Rx_GN] $\leftarrow$ node_GN$[S_u]$[Rx_GN]+1
end if
% **Step 3:** S_u updates its GN in the LT %
Find l such that node_GN$[S_u][l] \le$ node_GN$[S_u][m]$ for all $m \in \{1, 2, \ldots, K\}/\{l\}$
$y \leftarrow$ GN$[S_u][S_u]$
if node_GN$[S_u][y]$ – node_GN$[S_u][l] > 1$ **then**
  GN$[S_u][S_u] \leftarrow l$
end if
% **Step 4:** S_u updates number of nodes of GNs l and y in the LT %
if node_GN$[S_u][y]$ – node_GN$[S_u][l] > 1$ **then**
  node_GN$[S_u][y] \leftarrow$ node_GN$[S_u][y] - 1$
  node_GN$[S_u][l] \leftarrow$ node_GN$[S_u][l] + 1$
end if

Step 2: In **Step 2**, S_u updates the number of nodes of GNs Rx_GN and x in the LT. S_u increases the value of node_GN$[S_u]$[Rx_GN] by 1 if $x \ne$ Rx_GN. It decreases the value of node_GN$[S_u][x]$ by 1 if $x \ne$ Rx_GN and $x > 0$. Here, notation node_GN$[S_u][z]$ denotes the number of nodes of GN z in S_u's LT where $1 \le z \le K$.

Step 3: In **Step 3**, S_u first determines the GN l that has the minimum number of nodes in the LT. S_u sets its GN equal to l if node_GN$[S_u][y]$ – node_GN$[S_u][l] > 1$. As a result of this, the difference in the number of nodes of GNs l and y reduce in the LT. In this way, S_u maintains an equal number of nodes in each GN in its LT. (Here, y denotes S_u's GN in its LT before reception of the most recent GROUP packet.)

Step 4: In **Step 4**, S_u updates the number of nodes of GNs y and l in the LT. S_u increases value of node_GN$[S_u][l]$ by 1 and decreases value of node_GN$[S_u][y]$ by 1 if node_GN$[S_u][y]$ – node_GN$[S_u][l] > 1$.

Thus, using Algorithm 1, node S_u maintains an equal number of nodes of each GN in its LT. In this way, one-hop neighbors of each sensor node are partitioned into K (almost) equal-size groups. As a result of this, N sensor nodes are partitioned into K (almost) equal-size groups. Further, as in [30], we set $\widetilde{T}_G > N \cdot (\mathrm{T_{BO}} + \mathrm{T_{CCA}} + \mathrm{T_{GROUP}})$ where $\mathrm{T_{BO}}$ and $\mathrm{T_{CCA}}$ are BO and CCA durations, respectively. Furthermore, we performed extensive ns-2.35 based simulations to estimate α in a WuR enabled WSN. As in [30], it is noted that the sizes of all K groups are almost equal in case of $\alpha \geq 6$. Therefore, we choose $\alpha = 6.0$ to minimize the energy-consumption in the group number assignment process. Hereafter, we use the notations G_i, $\tilde{N}_i$, and S_j^i to denote the i^{th} group, total number of nodes in G_i, and j^{th} sensor node of G_i, respectively, where $i \in [1, K]$ and $j \in \left[1, \tilde{N}_i\right]$. It can be noted that $N = \sum_{i=0}^{K} \tilde{N}_i$.

C. Cycle Structure and Synchronization Technique

In this section, we describe the cycle structure that a node follows in RI-LD-WuR MAC whenever it starts transmitting/receiving a WuC. We also discuss the technique adopted to synchronize the wake up time instants of G_i nodes on WuC reception.

Cycle Structure: As shown in Fig. 2, we divide the cycle duration $\mathrm{T_{CYCLE}}$ into $K+1$ TSs. For $0 \leq l \leq K$, we use the notations TS_l and T_l to denote the l^{th} TS and its duration, respectively, where $\mathrm{T}_i = T = (\mathrm{T_{CYCLE}} - \mathrm{T}_0)/K$ for all $1 \leq i \leq K$. Further, we assign TS_i to the nodes of G_i. A node of G_i can transmit on an average n data packets in TS_i where $n = \lceil (\mathrm{T}_i - \mathrm{DIFS} - W \cdot \sigma/2) / (T_{DATA} + 2 \cdot \mathrm{SIFS} + T_{ACK}) \rceil$.

Synchronization Technique: At the beginning of TS_0, node S starts broadcasting an unaddressed WuC. All the N sensor nodes receive this WuC through their WuR. For $1 \leq i \leq K$, on WuC reception, each node of G_i sets a timer for T_i' seconds where $T_i' = \sum_{j=0}^{i-1} \mathrm{T}_j - \mathrm{T_{WuC}}$. Note that T_i'[1] denotes the time interval between the time instant when TS_i starts in the current cycle and the time instant when the node received the WuC. It ensures that the timer of all G_i nodes expires at the beginning of TS_i. In this way, we synchronize the timer expiration instant of each G_i node, with the beginning of TS_i, using the WuC broadcasted by S in the TS_0.

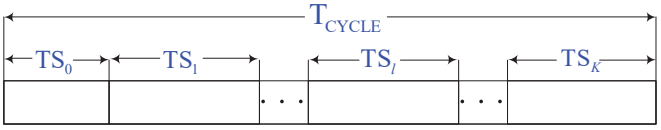


Fig. 2: Cycle structure that the nodes follow in RI-LD-WuR MAC.

D. Data Transmission Process (DTP)

In this section, we describe the steps that a node of G_i follows for data packet transmission in the TS_i where $1 \leq i \leq K$.

[1]Here, we do not consider the propagation time [31], reception time [31], and receive time [31] as their sum is very small and almost same for all the N nodes. Moreover, the sum of propagation time, reception time, and receive time is very small in comparison to SIFS and DIFS.

Step 1: It wakes up at the expiration of its T_i' seconds timer and sets a new timer for $rand(W) \cdot \sigma + DIFS$ seconds. Here, $rand(W)$ denotes the randomly chosen number from the interval $[0, W-1]$.

Step 2: It starts sending its data packets to sink S if it did not sense any ongoing communication till the expiration of its timer. After sending data packets, it goes in the sleep state. (Note that the wake up and sleep states apply only to MR.)

Step 3: It goes in sleep state and follows the same process in the next cycle if it failed in channel contention or its data packet collided with any other G_i node.

Note that nodes of G_i can transmit their data packets only in the TS_i of a cycle. As a result of this, packet collisions reduce due to a reduction in channel access competition. Moreover, a node of G_i wakes up at the expiration of its T_i' seconds timer only when it contains a data packet to send. Otherwise, it remains in the sleep state during the entire cycle. This ensures that there is no false wake up.

DTP is illustrated in Fig. 3 for the case of $N = 15$, $K = 5$, $n = 2$ and $\tilde{N}_i = 3$ for all $1 \leq i \leq K$. We assume that S_1^3, S_2^4, and S_1^5 do not have a data packet to send. (Recall that notation S_j^i denotes the j^{th} sensor node of G_i where $1 \leq i \leq K$ and $1 \leq j \leq \tilde{N}_i$.) In Fig. 3, DTP is illustrated for the nodes of G_3. As can be noted in Fig. 3, both S_2^3 and S_3^3 wake up at the expiration of their T_3' seconds timer for channel contention. Node S_1^3 remains in sleep state on its T_3' seconds timer expiration due to its empty queue. Node S_3^2 failed in channel contention and goes in sleep state. On the other hand, on success in channel contention, node S_3^3 sends its two data packets to node S. After that, node S_3^3 goes into the sleep state.

Note that only two nodes $\left(S_2^3 \text{ and } S_3^3\right)$ compete for channel access at the same time. On the other hand, in the case of RI-WuR and RI-CPT-WuR MAC protocols, twelve $(N-3)$ nodes compete for channel access at the same time. In this way, RI-LD-WuR MAC reduces channel access competition. As a result of this, both delay and energy-efficiency improve due to a reduction in packet collisions.

IV. Modeling and Performance Analysis

In this section, we develop a DTMC model to evaluate the performance of RI-LD-WuR MAC. For this, we arbitrarily select a node of G_i as the reference node (RN) where $1 \leq i \leq K$. As in [32]–[34], in our DTMC model, we assume that 1) each sensor node generates data packets independently following a Poisson process with a rate of λ packets/second, 2) a sensor node can store at the most Q packets in its first-in-first-out (FIFO) queue, and 3) the channel is ideal without channel fading and capture effect. However, our model can be easily extended for an error-prone channel following the process given in [6].

A. Analysis of the MAC Process of RI-LD-WuR MAC

Let $p_{b,i}$, $p_{t,i}$, $p_{s,i}$, and $p_{c,i}$ denote the probabilities that the node RN senses the channel busy, transmits data (with or without collision), transmits data without collision, and

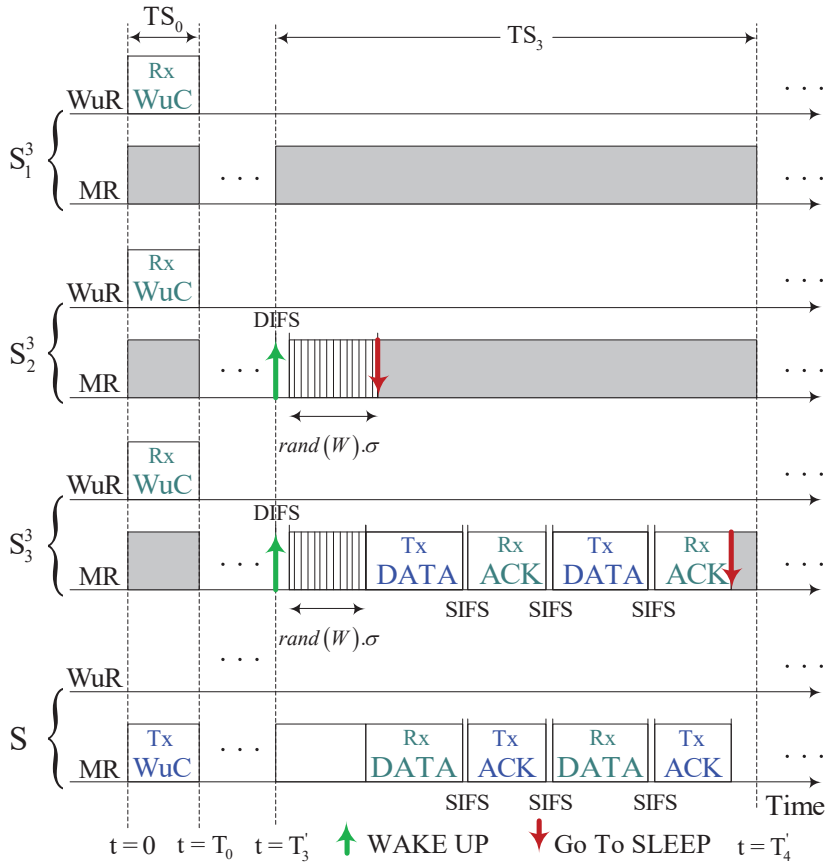


Fig. 3: Data transmission process of RI-LD-WuR MAC.

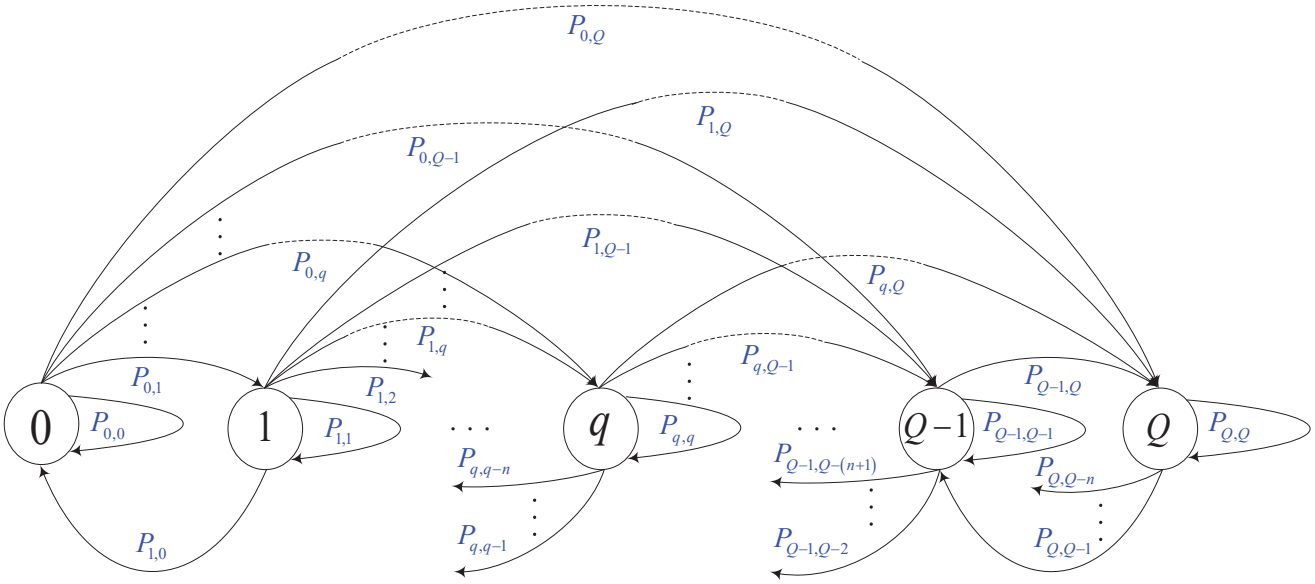


Fig. 4: Markov model for the queueing behavior of RN.

transmits data with collision, respectively. As shown in Fig. 3, at the most one node of G_i can send its data packets in a cycle when two or more nodes of G_i have data packets to send at the beginning of TS_i of the cycle. Let γ_{h_i} denote the probability that RN and $h_i \in \left[1, \tilde{N}_i - 1\right]$ other nodes of G_i have data packets to send in the same TS_i. Then,

$$\gamma_{h_i} = \binom{\tilde{N}_i - 1}{h_i} \cdot (1 - \pi_{0,i})^{h_i} \cdot {\pi_{0,i}}^{\tilde{N}_i - h_i - 1}$$

where $\pi_{0,i}$ is the probability that the queue of a node in G_i is empty when it enters its TS_i in every cycle.

Given that h_i $\left(\in \left[1, \tilde{N}_i - 1\right]\right)$ nodes are contending together with the RN, the probability that the RN wins the contention is $t_{h_i} = \sum_{l=0}^{W-1} \frac{1}{W} \cdot \left(\frac{W-l}{W}\right)^{h_i}$, the probability that the RN successfully transmits a packet is $s_{h_i} = \sum_{l=0}^{W-1} \frac{1}{W} \cdot \left(\frac{W-l-1}{W}\right)^{h_i}$, and the probability that the RN senses the channel busy is $b_{h_i} = \sum_{l=0}^{W-1} \frac{1}{W} \cdot \left(1 - \left(\frac{W-l}{W}\right)^{h_i}\right)$. Therefore, $p_{t,i} = \sum_{h_i=0}^{\tilde{N}_i - 1} \gamma_{h_i} \cdot t_{h_i}$, $p_{s,i} = \sum_{h_i=0}^{\tilde{N}_i - 1} \gamma_{h_i} \cdot s_{h_i}$, $p_{b,i} = \sum_{h_i=0}^{\tilde{N}_i - 1} \gamma_{h_i} \cdot b_{h_i}$, and $p_{c,i} = p_{t,i} - p_{s,i}$.

B. **DTMC Model for RN with Infinite Retransmissions**

Let random variable q $(\in [0, Q])$ denotes the stationary queue length of the RN when it enters its TS_i in every cycle. Then, as shown in Fig. 4, the DTMC model formed by q has $Q + 1$ states in total, each of which represents the number of data packets in the queue of the RN at the beginning of TS_i.

RN's state may change in between the beginning of successive TS_is, corresponding to the state transitions in the DTMC model, due to following events: RN receives k packets with probability $B_k = \frac{e^{-\lambda T_{\text{CYCLE}}} (\lambda T_{\text{CYCLE}})^k}{k!}$ and transmits $min\,(q, n)$ packets with probability $p_{s,i}$ if its queue is not empty. Let $\text{P} = [P_{l,m}]$ be the transition probability matrix of RN's DTMC, where $P_{l,m}$ is the probability that m data packets are found in the queue at cycle $\tilde{\alpha} + 1$, conditioned on finding l data packets in the queue at cycle $\tilde{\alpha}$. These transition probabilities are given as,

$$P_{0,m} = B_m;\ m \in [0, Q-1]$$

$$P_{0,Q} = \hat{B}_Q$$

$$P_{l,m} = p_{s,i} \cdot B_m + (1 - p_{s,i}) \cdot B_{m-l};\ l \in [1, n],\ m \in [l, Q-1]$$

$$P_{l,m} = p_{s,i} \cdot B_{m-l+n} + (1 - p_{s,i}) \cdot B_{m-l};\quad l \in [n+1, Q-1],\ m \in [l, Q-1]$$

$$P_{l,m} = p_{s,i} \cdot \hat{B}_Q + (1 - p_{s,i}) \cdot \hat{B}_{Q-l};\ l \in [1, n],\ m = Q$$

$$P_{l,m} = p_{s,i} \cdot \hat{B}_{Q-l+n} + (1 - p_{s,i}) \cdot \hat{B}_{Q-l};\quad l \in [n+1, Q-1],\ m = Q$$

$$P_{Q,Q} = p_{s,i} \cdot \hat{B}_n + (1 - p_{s,i}),$$

$$P_{l,m} = p_{s,i} \cdot B_m;\ l \in [1, n],\ m \in [0, l-1]$$

$$P_{l,m} = p_{s,i} \cdot B_{m-l+n};\ l \in [n+1, Q],\ m \in [l-n, l-1]$$

$$P_{l,m} = 0;\ l \in [n+1, Q],\ m \in [0, l-(n+1)]$$

where $\hat{B}_j = 1 - \sum_{x=0}^{j-1} B_x$. Let $\pi_i = [\pi_{0,i}\ \pi_{1,i}\ \ldots\ \pi_{Q,i}]$ denote the stationary distribution of $\pi_{l,i} = \Pr\{q = l\},\ (l \in [0, Q])$. The solution of this DTMC is obtained by solving $\pi_i \text{P} = \pi_i$ and $\pi_i e = 1$ where e is a column vector of ones.

C. **Packet Delivery Ratio (PDR)**

We define PDR as the ratio of the number of data packets received by the sink per cycle to the number of data packets generated in the network per cycle. Hence,

$$PDR = \frac{\sum_{i=1}^{K} \tilde{N}_i \cdot (1 - \pi_{0,i}) \cdot p_{s,i} \cdot \left(\sum_{j=1}^{n-1} j \cdot \pi_{j,i} + \sum_{j=n}^{Q} n \cdot \pi_{j,i}\right)}{N \cdot \lambda \cdot \text{T}_{\text{CYCLE}}} \tag{1}$$

where $(1 - \pi_{0,i}) \cdot p_{s,i} \cdot \sum_{j=n}^{Q} n \cdot \pi_{j,i}$ denotes that a node of G_i can transmit at the most n packets in a cycle if it wins the contention for channel access.

D. **Delay Analysis**

We define the delay (D) of a data packet as the time spent by the data packet in the queue of its source node. Let D_i be the average time spent by a data packet in the queue of the RN. D_i depends on the queuing delay $(D_{Q,i})$ and the contention delay $(D_{C,i})$, and is given as

$$D_i = D_{Q,i} + D_{C,i}. \tag{2}$$

Here, $D_{C,i}$ is defined as the time interval from when the packet is one of the first n data packets from the head of the queue to when the packet is transmitted to the sink. $D_{C,i}$ depends on $p_{s,i}$ and T_{CYCLE}, and is given as,

$$D_{C,i} = \text{T}_{\text{CYCLE}} \cdot \sum_{l=0}^{\infty} (l+1) \cdot (1 - p_{s,i})^l \cdot p_{s,i}.$$

$D_{Q,i}$ is defined as the time interval from when the packet enters into the queue to the packet becoming one of the first n packets from the head of the queue. Hence, $D_{Q,i}$ is given as,

$$D_{Q,i} = D_{C,i} \cdot \sum_{j=0}^{\delta_1} j \cdot \sum_{l=max(j \cdot n, 0)}^{\delta_2} \pi_{l,i} / (1 - \pi_{Q,i}),$$

where $\delta_1 = \left\lfloor \frac{Q-1}{n} \right\rfloor$ and $\delta_2 = min\left((j+1) \cdot n - 1, Q-1\right)$. Thus, D_i can be obtained by plugging $D_{C,i}$ and $D_{Q,i}$ into (2). Hence, D is given as

$$D = \frac{\sum_{x=1}^{K} D_x}{K}. \tag{3}$$

E. **Energy Consumption Analysis**

In each cycle, the RN receives WuC during the TS_0. The RN wakes up at the beginning of TS_i in a cycle only when it has a packet to send. Further, the RN can send up to n packets to S if it succeeds in channel contention. Otherwise, the RN goes in the sleep state and tries for the same in the next cycle. Let E_i^s, E_i^c, and E_i^b denote the average energy consumed by the RN's MR in case of successful transmission, collision, and

failure in channel contention, respectively, in a cycle. Then, E_i^s, E_i^c, and E_i^b are given as

$$E_i^s = p_{s,i} \cdot \left(\Delta_1 \cdot P_{tx}^{MR} + \Delta_2 \cdot P_{rx}^{MR} + (\Delta_3 + \Delta_4) \cdot P_{idle}^{MR}\right),$$

$$E_i^c = p_{c,i} \cdot \left((\Delta_4 + \Delta_5) \cdot P_{idle}^{MR} + \Delta_6 \cdot P_{tx}^{MR}\right),$$

and

$$E_i^b = p_{b,i} \cdot \Delta_7 \cdot P_{idle}^{MR},$$

respectively. Here $\Delta_1 = (1-\pi_{0,i}) \cdot T_{DATA} \cdot \sum_{j=1}^{n} j \cdot \pi_{j,i}$, $\Delta_2 = (1-\pi_{0,i}) \cdot T_{ACK} \cdot \sum_{j=1}^{n} j \cdot \pi_{j,i}$, $\Delta_3 = (1-\pi_{0,i}) \cdot SIFS \cdot \sum_{j=1}^{n} (2j-1) \cdot \pi_{j,i}$, $\Delta_4 = (1-\pi_{0,i}) \cdot \left(\frac{W}{2} \cdot \sigma + DIFS\right)$, $\Delta_5 = (1-\pi_{0,i}) \cdot (SIFS + T_{ACK})$, $\Delta_6 = (1-\pi_{0,i}) \cdot T_{DATA}$, and $\Delta_7 = (1-\pi_{0,i}) \cdot \left(DIFS + \sum_{j=1}^{W-1} \frac{1}{W-1} \cdot \sum_{l'=1}^{j} \frac{1}{j} \cdot (j-l'+1) \cdot \sigma\right)$.

Let E_i be the total energy consumption of RN in a cycle. Then, E_i is given as

$$E_i = T_{WuC} \cdot P_{rx}^{WuR} + (T_{CYCLE} - \Delta_8) \cdot P_{sleep}^{MR} + \Delta_8 \cdot P_{sleep}^{WuR} + E_i^s + E_i^c + E_i^b + (T_{CYCLE} - T_{WuC} - \Delta_8) \cdot P_{idle}^{WuR} \quad (4)$$

where $\Delta_8 = p_{s,i} \cdot (\Delta_1 + \Delta_2 + \Delta_3 + \Delta_4) + p_{b,i} \cdot \Delta_7 + p_{c,i} \cdot (\Delta_4 + \Delta_5 + \Delta_6)$ is the average time spent by the RN's MR in the active state per cycle. In (4), terms $T_{WuC} \cdot P_{rx}^{WuR}$, $(T_{CYCLE} - T_{WuC} - \Delta_8) \cdot P_{idle}^{WuR}$, and $\Delta_8 \cdot P_{sleep}^{WuR}$ denote average energy consumed by RN's WuR in WuC reception, idle listening, and sleep, respectively, per cycle. Terms $(T_{CYCLE} - \Delta_8) \cdot P_{sleep}^{MR}$ and $E_i^s + E_i^c + E_i^b$ denote the average energy consumed by the RN's MR in sleep and active state, respectively, per cycle. Thus, using (1), AEC is given as

$$\text{AEC} = \frac{\sum_{j=1}^{K} \tilde{N}_j \cdot E_j}{PDR \cdot N \cdot \lambda \cdot T_{CYCLE}}. \quad (5)$$

F. Optimum K Determination

We define the optimum K (K^*) as the value of K that provides the lowest D for the given values of T_{CYCLE}, λ, Q, and N. We propose Algorithm 1 to determine K^*. In Algorithm 1, we vary K from 1 to N in steps of 1. Note that $K = 1$ and $K = N$ indicate $\tilde{N}_i = N$ and $\tilde{N}_i = 1$, respectively, for all $1 \le i \le K$. For each value of K, we first calculate $p_{s,i}$, $\pi_{j,i}$, and D_i where $1 \le i \le K$ and $0 \le j \le Q$. Then we determine D using (3). We use variables γ and K^* to store the minimum D and corresponding K, respectively. Both γ and K^* are initialized with 0. For $K = 1$, we update K^* and γ with 1 and corresponding D, respectively. After that, we update γ and K^* only when the D corresponding to $K = j$ is less than γ where $2 \le j \le N$. In this way, Algorithm 1 provides the value of K corresponding to the lowest D as K^*.

Algorithm 2 Optimum K (K^*) determination.

1: **Initializations:** $\gamma \leftarrow 0$, and $K^* \leftarrow 0$
2: **Input:** T_{CYCLE}, λ, Q, and N
3: **for** $K \leftarrow 1:1:N$ **do**
4: $D \leftarrow 0$
5: Calculate $\text{T}_i = (\text{T}_{\text{CYCLE}} - \text{T}_0)/K$
6: Calculate $n = \lceil (\text{T}_i - \text{DIFS} - W \cdot \sigma/2)/u \rceil$ (where, $u = T_{DATA} + 2 \cdot \text{SIFS} + T_{ACK}$)
7: **for** $i \leftarrow 1:1:K$ **do**
8: Calculate $p_{s,i}$ and $\pi_{j,i}$ $\forall j \in [0, Q]$ (using DTMC model)
9: Calculate D_i
10: **end for**
11: $D \leftarrow \frac{\sum_{i=1}^{K} D_i}{K}$
12: **if** ($\gamma > 0$ **AND** $D < \gamma$) **OR** $\gamma == 0$ **then**
13: $\gamma \leftarrow D$ and $K^* \leftarrow K$
14: **end if**
15: **end for**
16: **Output:** K^*

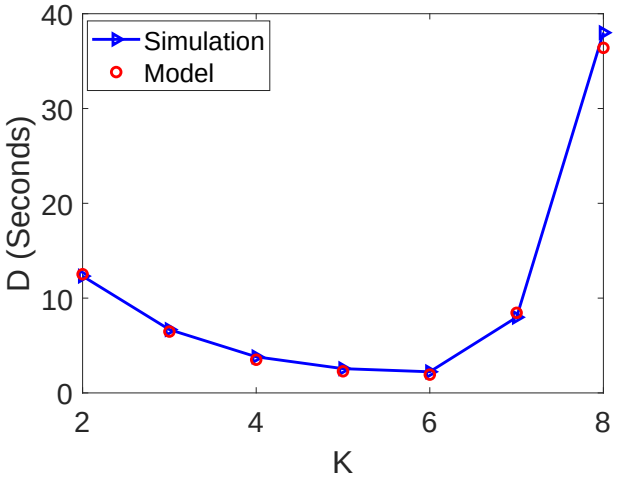


Fig. 5: Variation in D with K.

Fig. 5 shows variation in D with K in case of T_{CYCLE} = 1.0s, Q = 10.0, λ = 1.0 packet/second, and N = 30. It is to be noted that K^* = 6. In Fig. 5, for $K \le K^*$, D reduces with an increase in K. For $K > K^*$, D increases with an increase in K. Note that, for $K \le K^*$, D reduces with an increase in K due to reduction in channel access competition. For $K > K^*$, the D increases with an increase in channel access competition due to significant reduction in n.

V. Results

In this section, we compare the performance of RI-LD-WuR with RI-WuR and RI-CPT-WuR. We use the ns-2.35 simulator for PDR, D, and AEC calculation of each protocol. In Figs. 6-9, we have also shown the results obtained from the DTMC models. We use the DTMC models given in [6] with required changes for PDR, D, and AEC calculation of RI-WuR and RI-CPT-WuR. In Figs. 6-9, results are averaged over 60 simulations with different seeds, each lasting for 6000.0

seconds. The networking parameters used for PDR, D, and AEC calculation are given in Table II where MR and WuR parameters are the same as those in [6] and [35], respectively.

TABLE II: NETWORKING PARAMETERS [6], [35]

Parameter	Value	Parameter	Value	Parameter	Value
P_{tx}^{MR}	31.2 mW	P_{rx}^{WuR}	24 μW	Bandwidth	20 kbps
P_{rx}^{MR}	22.2 mW	P_{idle}^{WuR}	3.5 μW	W	64.0
P_{idle}^{MR}	22.2 mW	ACK/B	10 Bytes	T_0	12.2 ms
P_{sleep}^{MR}	3.0 μW	DATA	50 Bytes	σ	1.0 ms
SIFS	5.0 ms	DIFS	10.0 ms	K^*	6.0

A. **Performance with Varying** N

Figure 6 shows the variation in performance with N for the case of λ = 1.0 packets/second, Q = 10, K^* = 6.0, and T_{CYCLE} = 1.0 second. In Fig. 6 (a), it can be observed that the PDR of all the three protocols reduces with an increase in N. This happens because an increase in N increases both the channel access competition and the number of data packets generated per cycle. However, the decrease in PDR of RI-LD-WuR, with an increase in N, is lesser than RI-WuR and RI-CPT-WuR. The reason is that, in RI-LD-WuR, sensor nodes are partitioned into K (almost) equal size groups and the nodes of G_i $(i \in [1, K])$ are allowed for channel access only in the TS_i. As a result of this, with an increase in N, channel access competition in RI-LD-WuR grows slower than RI-WuR and RI-CPT-WuR. In Fig. 6 (a), at N = 24, RI-LD-WuR provides (almost) 45.0% and 36.0% higher PDR than RI-WuR and RI-CPT-WuR, respectively.

Figure 6 (b) shows the variation in D with N. It can be noted that the D of each protocol increases with an increase in N. The reason is that both D_Q and D_C increase with an increase in the channel access competition. It can also be noted that, with an increase in N, the D of RI-LD-WuR increases lesser than the other two protocols. The reason for this is once again that, in RI-LD-WuR, the channel access competition grows slower than RI-WuR and RI-CPT-WuR. In Fig. 6 (b), at N = 24, RI-LD-WuR provides (almost) 49.0% and 35.0% lower D than RI-WuR and RI-CPT-WuR, respectively.

Figure 6 (c) shows the variation in AEC with N. It can be noted that the AEC of each protocol increases with an increase in N. The reason is that an increase in channel access competition increases the idle listening, overhearing, and packet retransmission. It can also be noted that, with an increase in N, AEC of RI-LD-WuR increases lesser than RI-WuR and RI-CPT-WuR. It happens because, in RI-LD-WuR, channel access competition grows slower than RI-WuR and RI-CPT-WuR. In Fig. 6 (c), at N = 24, AEC of RI-LD-WuR is (almost) 86.0% and 43.0% lower than RI-WuR and RI-CPT-WuR, respectively.

B. **Performance with Varying** λ

Figure 7 shows the variation in performance with λ for the case of N = 30.0, Q = 10, K^* = 6.0, and T_{CYCLE} = 1.0 second. Figure 7 (a) shows the variation in PDR with λ. It can be noted that PDR of all the three protocols reduces with an increase of λ. The reason is that an increase in λ increases both the channel access competition and the number of data packets generated per cycle. For each value of λ, in RI-LD-WuR, channel access competition is lesser than RI-WuR and RI-CPT-WuR. As a result, RI-LD-WuR provides higher PDR than RI-WuR and RI-CPT-WuR. In Fig. 7 (a), at λ = 0.6, RI-LD-WuR provides (almost) 49.0% and 39.0% higher PDR than RI-WuR and RI-CPT-WuR, respectively.

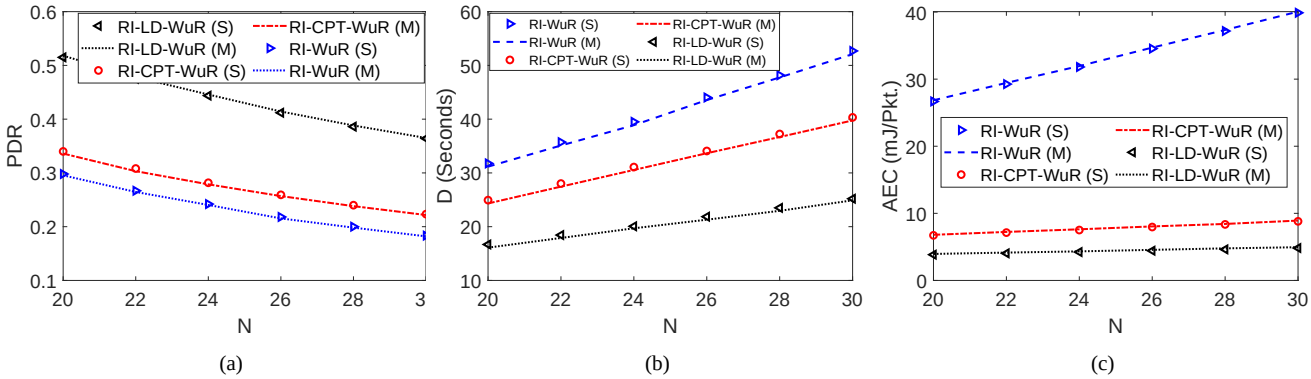
Figure 7 (b) shows the variation in D with λ. As can be noted in Fig. 7 (b), for each value of λ, RI-LD-WuR provides a lower D than RI-WuR and RI-CPT-WuR. The reason is that both D_Q and D_C increase with an increase in the channel access competition. Further, in all the three protocols, the rate of increase of D reduces with an increase of λ. The reason is that the rate of increase in channel access competition reduces with an increase of λ. In Fig. 7 (b), at λ = 0.6, RI-LD-WuR provides (almost) 58.0% and 40.0% lower D than RI-WuR and RI-CPT-WuR, respectively.

Figure 7 (c) shows the variation in AEC with λ. Here, AEC is affected by two factors: channel access competition and an average number of packets received by S per cycle. AEC increases with an increase in the first factor (channel access competition) and decreases with an increase in the second factor. In all the three protocols, channel access competition increases with an increase of λ. In RI-CPT-WuR, on success in channel contention, a node can send more number of data packets to S than RI-WuR and RI-LD-WuR. Therefore, unlike RI-LD-WuR and RI-WuR, in RI-CPT-WuR, AEC decreases with an increase of λ. However, for each value of λ, AEC in RI-LD-WuR is lesser than RI-WuR and RI-CPT-WuR. The reason is that, in RI-LD-WuR, channel access competition is significantly lesser than RI-WuR and RI-CPT-WuR. AEC increases with an increase in channel access competition due to an increase in idle listening, overhearing, and retransmissions. In Fig. 7 (c), at λ = 0.6, AEC of RI-LD-WuR is (almost) 87.0% and 49.0% lower than RI-WuR and RI-CPT-WuR, respectively.

C. **Performance with varying** Q

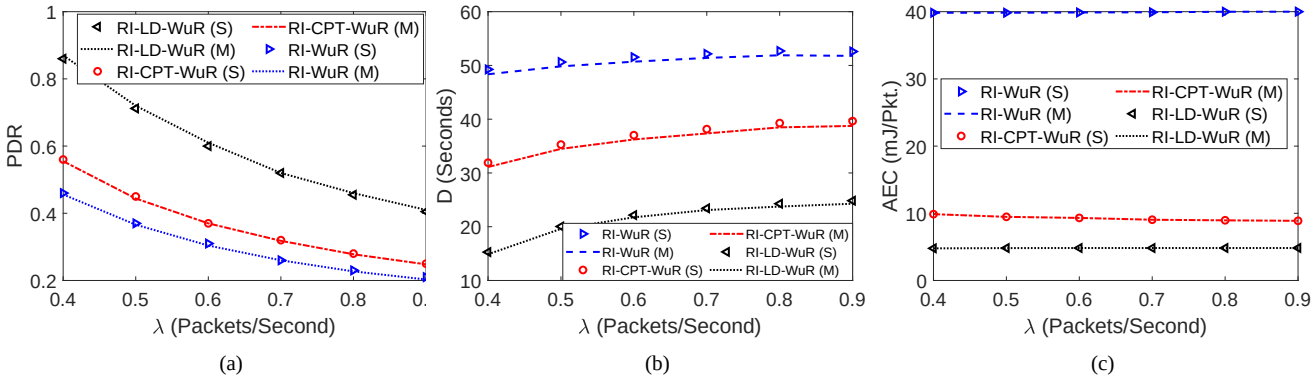
Figure 8 shows the variation in performance with Q for the case of N = 30.0, λ = 1.0, K^* = 6.0, and T_{CYCLE} = 1.0 second. Figure 8 (a) shows the variation in PDR with Q. It can be noted that PDR of RI-LD-WuR and RI-CPT-WuR increases with an increase in Q. The reason is that an increase in Q increases the packets received by S per cycle. For each value of Q, in RI-LD-WuR, channel access competition is lesser than RI-WuR and RI-CPT-WuR. As a result, RI-LD-WuR provides higher PDR than RI-WuR and RI-CPT-WuR. In Fig. 8 (a), for Q = 6.0, RI-LD-WuR provides (almost) 50.0% and 40.0% higher PDR than RI-WuR and RI-CPT-WuR, respectively.

Figure 8 (b) shows the variation in D with Q. As can be noted in Fig. 8 (b), D of each protocol increases with an increase in Q. The reason is that an increase in Q increases the queuing delay (D_Q). Further, in Fig. 8 (b), RI-LD-WuR provides a lower D than the other two protocols. The reason is that contention delay (D_C) increases with an increase in



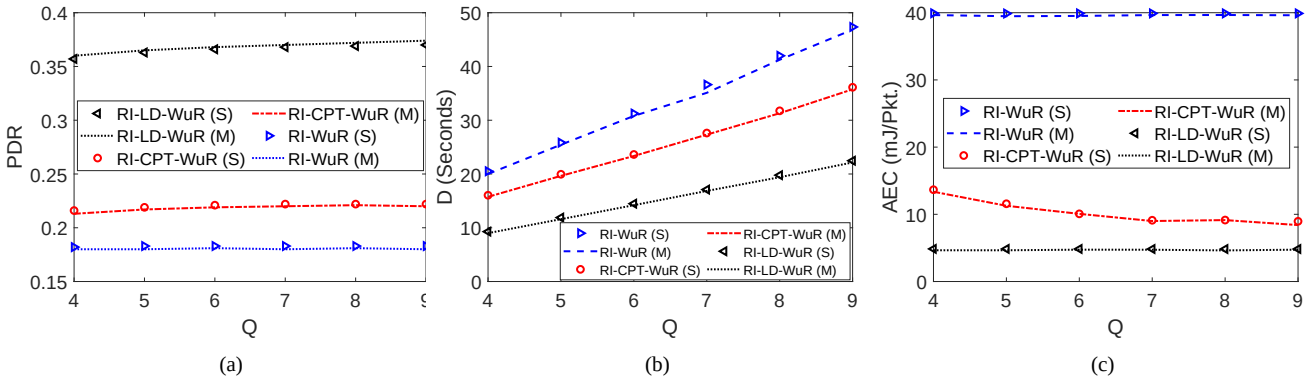
(a) (b) (c)

Fig. 6: Performance with varying N. (a) PDR, (b) D, and (c) AEC.



(a) (b) (c)

Fig. 7: Performance with varying λ. (a) PDR, (b) D, and (c) AEC.



(a) (b) (c)

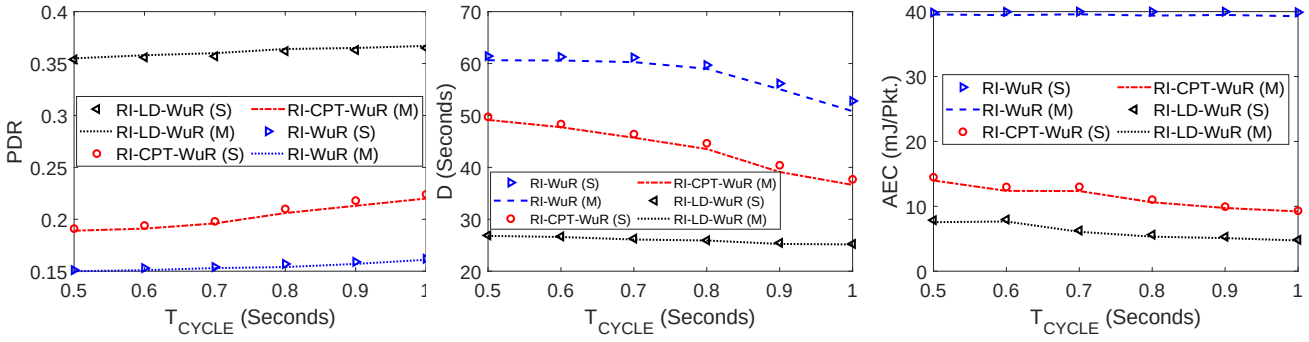
Fig. 8: Performance with varying Q. (a) PDR, (b) D, and (c) AEC.

channel access competition. In Fig. 8 (b), for Q = 6.0, RI-LD-WuR provides (almost) 53.0% and 38.0% lower D than RI-WuR and RI-CPT-WuR, respectively.

Figure 8 (c) shows the variation in AEC with Q. In RI-CPT-WuR, on success in channel contention, a node can send more number of packets to S than RI-WuR and RI-LD-WuR. Therefore, AEC of RI-CPT-WuR decreases with a higher rate than RI-LD-WuR and RI-WuR. However, for each value of Q, AEC of RI-LD-WuR is lesser than RI-WuR and RI-CPT-WuR. The reason is that idle listening, overhearing, and retransmissions increase with an increase in channel access competition. In Fig. 8 (c), for Q = 6.0, AEC of RI-LD-WuR is (almost) 86.0% and 52.0% lower than RI-WuR and RI-CPT-WuR, respectively.

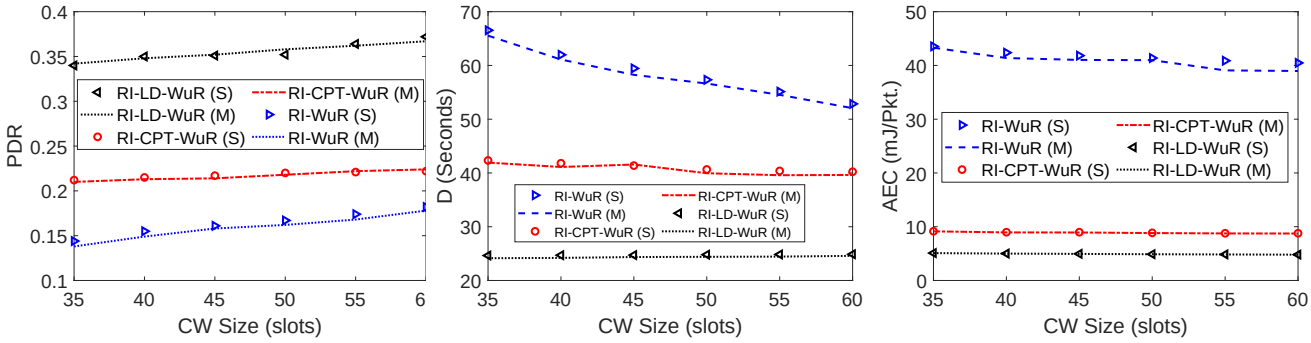
D. Performance with varying T_{CYCLE}

Figure 9 shows the variation in performance with T_{CYCLE} for the case of N = 30.0, λ = 1.0, and Q = 10. Figure 9 (a) shows the variation in PDR with T_{CYCLE}. In RI-LD-WuR, K^* increases with an increase in T_{CYCLE}. It is to be noted that channel access competition reduces with an increase in K^*. Therefore, PDR of RI-LD-WuR increases with an increase in T_{CYCLE}. In addition to this, with an increase in T_{CYCLE}, the average number of packets received by S per second increases



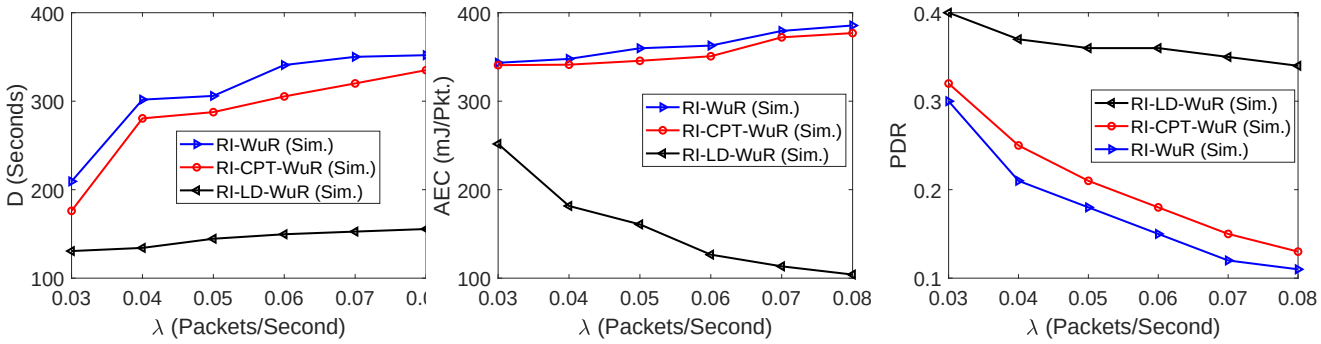
(a) (b) (c)

Fig. 9: Performance with varying T_{CYCLE}. (a) PDR, (b) D, and (c) AEC.



(a) (b) (c)

Fig. 10: Performance with varying CW. (a) PDR, (b) D, and (c) AEC.



(a) (b) (c)

Fig. 11: Performance with varying λ in a multi-hop WSN. (a) PDR, (b) D, and (c) AEC.

due to a reduction in the number of WuCs broadcasted per second. In RI-LD-WuR, channel access competition is lesser than the other two protocols for each given value of T_{CYCLE}. In Fig. 9 (a), for $T_{CYCLE} = 0.7$ seconds, RI-LD-WuR provides (almost) 58.0% and 45.0% higher PDR than RI-WuR and RI-CPT-WuR, respectively.

Figure 9 (b) shows the variation in D with T_{CYCLE}. As can be noted in Fig. 9 (b), D of each protocol reduces with an increase in T_{CYCLE}. The reason is that an increase in T_{CYCLE} increases the number of packets received by S per second. Further, RI-LD-WuR provides lower D than RI-WuR and RI-CPT-WuR due to low channel access competition. Recall that D_Q and D_C increase with an increase in channel access competition. In Fig. 9 (b), for $T_{CYCLE} = 0.7$ seconds, RI-LD-WuR provides (almost) 57.0% and 43.0% lower D than RI-WuR and RI-CPT-WuR, respectively.

Figure 9 (c) shows the variation in AEC with T_{CYCLE}. In Fig. 9 (c), AEC reduces with an increase in T_{CYCLE}. It happens because of energy-consumption in idle listening, overhearing, and retransmissions reduce with a reduction in channel access competition. In addition to this, the number of WuCs broadcasted per second also reduces with an increase in T_{CYCLE}. In Fig. 9 (c), AEC of RI-LD-WuR is lesser than RI-WuR and RI-CPT-WuR due to low channel access

competition. In Fig. 9 (c), for $T_{CYCLE} = 0.7$ seconds, AEC of RI-LD-WuR is (almost) 84.0% and 51.0% lower than RI-WuR and RI-CPT-WuR, respectively.

E. Performance with varying CW

Figure 10 shows the variation in performance with CW size for the case of $N = 30.0,\ \lambda = 1.0,\ K^* = 6.0,\ Q = 10$ and $T_{CYCLE} = 1.0$ second. Figure 10 (a) shows the variation in PDR with CW size. It can be noted that the PDR of each protocol increases with an increase in CW size. The reason for this is that the collision probability reduces with an increase in CW size. For each value of CW size, in RI-LD-WuR MAC, channel access competition is lesser than RI-WuR and RI-CPT-WuR MACs. As a result, RI-LD-WuR MAC provides higher PDR than RI-WuR and RI-CPT-WuR MACs. In Fig. 10 (a), for CW Size = 45, RI-LD-WuR MAC provides (almost) 19.0% and 13.50% higher PDR than RI-WuR and RI-CPT-WuR MACs, respectively.

Figure 10 (b) shows the variation in D with CW size. As can be noted in Fig. 10 (b), D of each protocol decreases with an increase in CW size. The reason for this is that the contention delay (D_C) decreases with an increase in CW size. Further, in Fig. 10 (b), RI-LD-WuR MAC provides a lower D than the other two protocols. This happens because D_C increases with an increase in channel access competition. In Fig. 10 (b), for CW size = 45, RI-LD-WuR MAC provides (almost) 88.26% and 44.82% lower D than RI-WuR and RI-CPT-WuR MACs, respectively.

Figure 10 (c) shows the variation in AEC with CW size. As can be noted in Fig. 10 (c), AEC of each protocol decreases with an increase in CW size. The reason behind this is that the collision probability decreases with an increase in CW size. Further, for each value of CW size, AEC of RI-LD-WuR MAC is lesser than RI-WuR and RI-CPT-WuR MACs. The reason for this is that idle listening, overhearing, and retransmissions increase with an increase in channel access competition. In Fig. 10 (c), for CW size = 45, AEC of RI-LD-WuR MAC is (almost) 58.42% and 40.75% lower than RI-WuR and RI-CPT-WuR MACs, respectively.

F. Performance with varying λ in a Multi-Hop WSN

The proposed protocol is also applicable in a multi-hop WSN. Unlike single-hop WSNs, in a multi-hop WSN, a node follows back-off (BO) and clear channel assessment (CCA) before WuC broadcast to avoid collisions. To analyze the performance in a multi-hop WSN, we construct a network by randomly deploying N sensor nodes in a rectangular region of $150 \times 600\ m^2$ area. We placed a sink (S) at one corner of the network. We set communication and carrier sensing range of each node equal to 250 m and 550 m, respectively. Therefore, most of the sensor nodes are at two or more hops distance from S.

Figure 11 shows variation in performance with λ in case of $K^* = 6.0,\ Q = 10,\ N = 30.0$ and $T_{CYCLE} = 1.0$ second. As can be noted in Fig. 11, RI-LD-WuR MAC provides lower delay and higher PDR than RI-CPT-WuR and RI-WuR MAC protocols. Further, AEC is also lesser than RI-CPT-WuR and RI-WuR MAC protocols. Note that our proposed DTMC model and expressions (of PDR, D, and AEC) are applicable only for single-hop WSNs. Therefore, in Fig. 11, we have shown only simulation results.

VI. Conclusions

In this paper, we have proposed a receiver-initiated low delay WuR (RI-LD-WuR) MAC protocol for WSNs. We improved the delay and energy-efficiency by reducing the channel access competition. We utilized the idea of node grouping to decrease the channel access competition. We proposed an algorithm to determine the optimum number of groups. Further, we developed a DTMC model to evaluate the packet delivery ratio (PDR), delay (D), and average energy consumption per transmitted data packet (AEC). The results showed that RI-LD-WuR MAC significantly improves PDR, D, and AEC in comparison to RI-WuR and RI-CPT-WuR MAC protocols.

In the proposed DTMC model, we have assumed that the channel is error-free and the retransmission limit is infinite. Extension of our proposed model to analyze the performance of RI-LD-WuR MAC in the case of finite retransmission and/or error-prone channel conditions could be an interesting direction for future work.

References

[1] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5G: A tutorial overview of standards, trials, challenges, deployment, and practice", *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1201-1221, Jun. 2017.

[2] L. Guntupalli, H. Farag, A. Mahmood, and M. Gidlund, "Priority-oriented packet transmissions in Internet of Things: Modeling and delay analysis", *in Proc. IEEE ICC*, 2018, pp. 1-6.

[3] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks ?", *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674-687, Apr. 2016.

[4] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: A hardware and networking survey", *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2117-2157, 2017.

[5] F. Z. Djiroun and D. Djenouri, "MAC protocols with wake-up radio for wireless sensor networks: A review", *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 587-618, 2017.

[6] L. Guntupalli, D. Ghose, F. Y. Li, and M. Gidlund, "Energy efficient consecutive packet transmissions in receiver-initiated wake-up radio enabled WSNs", *IEEE Sensors J.*, vol. 18, no. 11, pp. 4733-4745, Jun. 2018.

[7] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems", *in Proc. IEEE INFOCOM*, Apr./May 2015, pp. 522-530.

[8] J. M. Lebreton, S. Kandukuri, N. Murad, and R. Lorion, "An energyefficient duty-cycled wake-up radio protocol for avoiding overhearing in wireless sensor networks", *Wireless Sensor Netw.*, vol. 8, pp. 176-190, Aug. 2016.

[9] F. A. Aoudia, M. Gautier, and O. Berder, "OPWUM: Opportunistic MAC protocol leveraging wake-up receivers in WSNs", *J. Sensors*, vol. 2016, Art. no. 6263719, 2016.

[10] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for longlived wireless sensing systems", *in Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 522-530.

[11] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "RI-MAC: A receiverinitiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks", *in Proc. ACM Sensys*, 2008, pp. 510-527.

[12] D. Ghose, F. Y. Li, and V. Pla, "MAC protocols for wake-up radio: Principles, modeling and performance analysis", *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2294-2306, May 2018.

[13] D. Ghose and F. Y. Li, “Enabling backoff for SCM wake-up radio: Protocol and modeling”, *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1031-1034, May 2017.

[14] D Ghose, A Frytlog, and F. Y. Li, “Enabling early sleeping and early data transmission in wake-up radio-enabled IoT networks”, *Computer Netw.*, vol. 153, pp. 132-144, Apr. 2019.

[15] D. Ghose, L. Tello-Oquendo, F. Y. Li, and V. Pla, “Lightweight relay selection in multi-hop wake-up radio enabled IoT networks”, *in Proc. IEEE GLOBECOM*, 2018, pp. 1-6.

[16] M. Zhang, D. Ghose, and F. Y. Li, “Collision avoidance in wake-up radio enabled WSNs: Protocol and performance evaluation”, *in Proc. IEEE ICC*, 2018, pp. 1-6.

[17] D. Ghose, A. Froytlog, and F. Y. Li, “Reducing overhearing energy in wake-wp radio-enabled WPANs: Scheme and performance”, *in Proc. IEEE ICC*, 2018, pp. 1-6.

[18] M. Zhang, D. Ghose, and F. Y. Li, ““Does wake-up radio always consume lower energy than duty-cycled protocols ?”, *in Proc. IEEE VTC-Fall*, 2017, pp. 1-5.

[19] K. R. Chowdhury, N. Nandiraju, D. Cavalcanti, and D. P. Agrawal, “CMAC: A multi-channel energy efficient MAC for wireless sensor networks”, *in Proc. IEEE WCNC*, 2006, pp. 1172-1177.

[20] S. H. Lee, Y. S. Bae, and L. Choi, “On-demand radio wave sensor for wireless sensor networks: Towards a zero idle listening and zero sleep delay MAC protocol”, *in Proc. IEEE GLOBCOM*, 2012, pp. 560-566.

[21] P. Sthapit and J.-Y. Pyun, “Effects of radio triggered sensor MAC protocol over wireless sensor network”, *in Proc. IEEE CIT*, 2011, pp. 546-551.

[22] A. Pegatoquet, T-N. Le, and M. Magno, “A Wake-Up Radio based MAC Protocol for Autonomous Wireless Sensor Networks”, *IEEE/ACM Trans. Netw.*, Vol. 27, no. 1, pp. 56-70, Feb. 2019.

[23] R. Duan, Q. Zhao, H. Zhang, Y. Zhang, and Z. Li, “Modeling and performance analysis of RI-MAC under a star topology”, *Comput. Commun.*, vol. 14, pp. 133-144, May 2017.

[24] L. Guntupalli and M. Gidlund, “Multiple packet transmissions in duty cycling WSNs: A DTMC based throughput analysis”, *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 480-483, Dec. 2017.

[25] F. A. Aoudia, M. Magno, M. Gautier, O. Berder, and L. Benini, “Analytical and experimental evaluation of wake-up receivers based protocols”, *in Proc. IEEE GLOBECOM*, 2016, pp. 1-7.

[26] N. S. Mazloum and O. Edfors, “Influence of duty-cycled wakeup receiver characteristics on energy consumption in single-hop networks”, *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3870-3884, Jun. 2017.

[27] J. M.- Bauset, L. Guntupalli, and F. Y. Li, “Performance analysis of synchronous duty-cycled MAC protocols”, *IEEE Wireless Commun. Lett.*, vol. 4, no. 5, pp. 469-472, Jun. 2015.

[28] L. Guntupalli, J. M.- Bauset, F. Y. Li, and M. A. Weitnauer, “Aggregated packet transmission in duty-cycled WSNs: Modeling and performance evaluation”, *IEEE Trans. on Veh. Technol.*, vol. 66, no. 1, pp. 563-579, Mar. 2016.

[29] L. Guntupalli, M. Gidlund, and F. Y. Li, “An on-demand energy requesting scheme for wireless energy harvesting powered IoT networks”, *IEEE Int. Things J.*, vol. 5, no. 4, pp. 2868-2879, Aug. 2018.

[30] R. Singh, B. K. Rai, and S. K. Bose, “A novel framework to enhance the performance of contention-based synchronous MAC protocols”, *IEEE Sensors J.*, vol. 16, no. 16, pp. 6447-6457, Aug. 2016.

[31] S. Ganeriwal, R. Kumar, and M. B. Srivastava, “Timing-sync protocol for sensor networks”, *in Proc. ACM Embedded Netw. Sensor Syst.*, 2003, pp. 138-149.

[32] R. Singh, B. K. Rai, and S. K. Bose, “Modeling and performance analysis for pipelined-forwarding MAC protocols for linear wireless sensor networks”, *IEEE Sensors J.*, vol. 19, no. 15, pp. 6539-6552, Aug. 2019.

[33] O. Yang and W. Heinzelman, “Modeling and performance analysis for duty-cycled MAC protocols with applications to S-MAC and X-MAC”, *IEEE Trans. Mobile Comput.*, vol. 11, no. 6, pp. 905-921, Jun. 2012.

[34] F. Tong, L. Zheng, M. Ahmadi, M. Li, and J. Pan, “Modeling and analyzing duty-cycling, pipelined-scheduling MACs for linear sensor networks”, *IEEE Trans. on Veh. Technol.*, vol. 65, no. 4, pp. 2608-2620, Apr. 2016.

[35] M. S. Hefeida, T. Canli, and A. Khokhar, “CL-MAC: A cross-layer MAC protocol for heterogeneous wireless sensor networks”, *Ad Hoc Netw.*, vol. 11, no. 1, pp. 213-225, Jan. 2013.

Ripudaman Singh , received the B.E. (Hons.) degree in Electronics and Communication Engineering from University of Rajasthan, Jaipur, India, in 2009, and the M. Tech. and the Ph.D. degrees in Electronics and Electrical Engineering from Indian Institute of Technology Guwahati, Guwahati, India, in 2012 and 2018, respectively. Currently, he is an Assistant Professor at the Department of Electronics and Communication Engineering, Indian Institute of Information Technology Guwahati, Guwahati, India. From May 2020 to December 2020, he was a Postdoctoral Fellow at Electrical and Computer Engineering Department NUS Singapore, Singapore. His research interests include Wireless Sensor Networks, Distributed Algorithms, and Performance Modeling of communication Systems.

Biplab Sikdar (S’98-M’02-SM’09) received the B.Tech. degree in electronics and communication engineering from North-Eastern Hill University, Shillong, India, in 1996, the M.Tech. degree in electrical engineering from the IIT Kanpur, Kanpur, India, in 1998, and the Ph.D. degree in electrical engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 2001. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. His research interests include wireless MAC protocols, transport protocols, network security, and queuing theory.