Media Access Protocol for Wireless Sensor Network Using Active Reception Scheme-Based Energy-Efficient Technique

Anushree Goud and Bindu Garg

Abstract Reception Media Access Control is a TDMA-based MAC protocol for data transmission in WSNs. The feature is to preserve energy through restrain traffic overhearing, in-active listening, retransmission of data and packet collision. R-MAC uses a distributed scheduling procedure to assign a time slot for reception called Reception Slot (RS) to each sensor, and share message of reception slot to each of its neighbouring device. A sensor who has data to transmit can consequently wake up in reception slot of its intended receiver only. R-MAC guarantees that only one one-hop neighbour can transmit at a time in reception slot of its intended receiver, which assures collision avoidance. The result analysis of R-MAC is analysed with respect to power consumption, traffic and message delay in multi-hop networks through detailed simulation. The result of R-MAC is compared against SMAC. Network simulator 2 (NS-2) is used for simulation.

Keywords Reception Slot (RS) · Δ-slot · Slot request · Slot response · Data request · Data response

1 Introduction

WSNs consist of small individual sensors coordinating among themselves to analyse environmental conditions, such as pressure, sound, temperature and vibrations [1]. Sensors in WSNs have sensing, computation and wireless transfer capabilities. WSNs have emerged as a substitute of large network infrastructures for a large number of application, for example, precision agriculture, habitat monitoring,

---

A. Goud (✉)
Narsee Monjee Institute of Management studies, Mumbai, Maharashtra, India
e-mail: anushree.goud@nmims.edu

B. Garg
Bharati Vidyapeeth College of Engineering, Pune, India
e-mail: brgarg@bvucoep.edu.in

A. Haldorai et al. (eds.), *3rd EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, EAI/Springer Innovations in Communication and Computing, https://doi.org/10.1007/978-3-030-78750-9_12

structure monitoring, environmental and medical diagnostics. Detecting complex scenario interactions is the most dramatic applications of WSNs.

Designing MAC scheme for WSNs involves many challenges. Energy consumption is one of the most vital issue in WSN as sensors have limited battery power, and it is very hard or sometimes impossible to recharge or replace batteries of sensors. Our main aim is to consider the sources of energy waste while designing an efficient protocol. Idle listening is the first cause of unwanted energy wastage. It occurs usually when the node is listening to the environment for expected traffic but there is no traffic in the channel.

Collision is the second source of unwanted energy wastage. When multiple messages are transmitted in the same channel simultaneously, collision may occur and they will be of no use and must be discarded. Retransmission of collided messages consumes energy. Collision is a vital issue to consider in contention-based MAC scheme. Next source is overhearing, which occurs when a sensor listens to the traffic that is destined to separate sensors. Where traffic load is high, overhearing can be a serious issue to deployed network. Sending and receiving control packets and control packet overhearing are other sources of energy waste as it does not directly convey useful data.

In this research work, MAC scheme is proposed for data transmission in WSNs, which is energy efficient. This solution is suited for network scenarios where sensor mobility is low, traffic rate is low-to-moderate. Environmental monitoring, structure monitoring and precision agriculture are some examples of such applications. Our proposed MAC scheme limits the energy waste through limiting overhearing, packet collision, retransmission, idle listening and packet over-emitting. R-MAC is a time division multiple access-based protocol, which uses a distributed procedure that allocates duration to each node for data and spreads the information about allocated time slot among neighbouring sensors.

The chapter is structured as follows: In Sect. 2, literature review is presented, Sect. 3 describes the proposed MAC scheme referred to as R-MAC. Section 4 contains the simulation results followed by conclusions in further section.

2 Literature Review

In literature, many MAC protocols have been proposed, which can be categorised as contention-free and contention-based scheme. It offers low traffic rate scenarios, since they mainly work on the basis of Carrier Sense Multiple Access (CSMA) or Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA), allowing collisions to occur, and try to recover from these collisions by using back-off algorithm. Examples of contention-based protocols are S-MAC (Sensor MAC) [6, 13], D-MAC (Data-gathering MAC) [7, 16], T-MAC (Timeout MAC) [8, 15]. S-MAC implements periodical listening and sleeping mechanism to minimize energy waste. In other word, each node goes to sleep for some time, and then wakes up and listens to see if any other node wants to transmit to it [6].

On the other side, contention-free scheduled MAC protocols use TDMA slotted structure, where time slots are allocated to the nodes in the network. Energy efficiency is achieved by ensuring that each node is asleep whenever it is not scheduled to transmit or receive. Examples of contention-free MAC schemes are SPARE-MAC [9], TDMA-ASAP [10], TRAMA [11], DEE-MAC [12]. A reception schedule is a time slot during which a sensor becomes active to receive data. This conserves energy by limiting overhearing and idle listening at the expense of increased collision probability [9]. Collision happens when multiple nodes transmit to the same node in the same reception schedule [14]. does not use collision avoidance, but it recovers after collisions happen.

3 R-MAC Description

R-MAC is a TDMA-based scheme, where individual sensor is having a time slot for reception called Reception Slot (RS), and sensors know the RS of all its receivers. A RS is a time slot when sensor node becomes active for exchange of data. The rationale behind R-MAC is that if a sensor needs to transmit, the receiving sensor restrains its all other one-hop neighbours from transmitting during ongoing transmission to avoid collision. Furthermore, time synchronization among sensors is the primary requirement of R-MAC, which can be achieved through different proposed Schemes [2–5].

In the next section, frame structure of R-MAC is presented. The reception slot assignment procedure is followed by data transmission.

3.1 *Frame Structure*

R-MAC describes the frame structure shown in Figs. 1 and 2. Each is divided into N Reception Slots (RSs). Each RS is further divided into N + 3 Δ-slots and a data slot. Δ-slots are used for coordination among neighbouring sensors. On the other hand, data slot is reserved for data reception. The parameter N depends on the traffic requirements and topology of network. Δ-slots forming 1 to N are assigned to the neighbouring sensors. Following are Δ-slot assignment requirements:

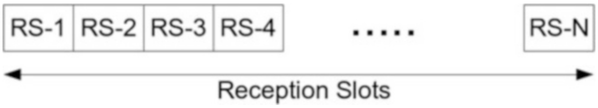

Fig. 1 Frame structure

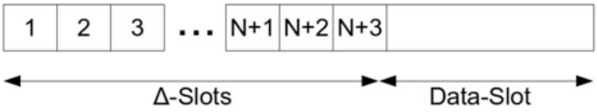

Fig. 2 Reception slot structure

- Each sensor is assigned a Δ-slot in all its potential receivers' RS.
- Each sensor knows its Δ-slot in all its potential receivers' RS.

3.2 Reception Slot Assignment

Next step after activation of a sensor is to gain a reception slot. Δ-slots N + 2 and N + 3 in each RS are reserved from reception slot assignment. Reception slot assignment completes in three steps.

4 Step 1

After activation, the newcomer sensor sends a slot request in N + 2th Δ-slot of every RS. A slot request contains sender's RS number and Δ-slot number of receiver in sender's RS. In step 1, slot request contains NULL for both sender's RS number and Δ-slot number of receiver *in sender's RS.*

Rule 1: *Each sensor remains active during all Δ-slots of its own RS.*

Each receiver of slot request checks Δ-slot number of sender in its own RS, if Δ-slot is not assigned yet, assigns a Δ-slot in its own RS to the sender and sends a slot-response in N + 3th Δ-slot. A slot response contains sender's RS number and receiver's Δ-slot number in sender's RS. If newcomer receives no slot response in N + 3th Δ-slot, marks the current RS as AVAILABLE. If newcomer receives one slot response, marks the current RS RESERVER, stores the RS number of the sender and its own Δ-slot number in sender's RS and assigns a Δ-slot to the sender in its own RS. Collision may happen at newcomer where multiple nodes send slot response on the same RS. If collision happens at newcomer, it marks the corresponding RS COLLIDE.

Rule 2: *An AVAILABLE reception slot must not interfere with any other reception slot of one-hop neighbours.*

Rule 3: *Two one-hop neighbours cannot have same reception slot.*

After step 1, newcomer acquires a RS from the list of *AVAILABLE* reception slots.

5 Step2

In the following frame, newcomer sends *slot requests* in $N + 2th$ Δ-slot of each *RESERVED* and *COLLIDE* reception slot. If RS is coded *COLLIDE,* the value of *Δ-slot number of receiver in sender's RS* in *slot request* is set *NULL.* At the receiver end, receiver stores the sender's RS number. If the value of Δ-*slot number of receiver in sender's RS* is not *NULL, it* stores in its own Δ-slot number in sender's RS and sends a slot response in $N + 3th$ Δ-slot.

The NULL value of Δ-*slot number of receiver in sender's RS* is interpreted as the indication of collision.

6 Step 3

Each collision-occurring node refrains from exchanging *slot request* for a number of frames, interpreted as *i*

$$i = \begin{cases} \text{random}\left[1, 2^k\right] & \text{, If } k \leqslant 10 \\ \text{random}\left[1, 2^{10}\right] & \text{, otherwise} \end{cases}$$

where k is the number of consecutive collisions.

After *i* frames, the receiver sends slot request in $N + 2th$ Δ-slot of newcomer's RS. After receiving *slot request*, newcomer stores RS number of sender and its own Δ-slot number in sender's RS, assigns a Δ-slot to the sender in its own RS and sends a *slot-response.* Receiver of slot response stores its own Δ-slot number in sender's RS.

Figure 3 shows the case of a new sensor entering the system (sensor 3). Sensor 3 waits for the next frame and sends out a *slot request* in sixth Δ-slot of RS-1 which is acquired by sensor 1. Sensor 1 assigns a Δ-slot to sensor 3 in its own RS (RS-1) and sends out *slot response* in seventh Δ-slot. Sensor 3 receives *slot response* from sensor 1 containing RS of sensor 1 and Δ-slot of sensor 3 in RS of sensor 1. Sensor 3 assign a Δ-slot to sensor 1 in its own RS and marks RS-1 *RESERVED.* Same procedure is followed in RS-2 which is owned by sensor 2. Sensor 3 sends *slot request* in sixth Δ-slot of RS-3 and RS-4 but does not get *slot response* in seventh Δ-slot of RS-3 and RS-4. So, sensor 3 marks RS-3 and RS-4 *AVAILABLE.* After frame 1, sensor 3 acquires a RS from the list of *AVAILABLE* reception slots. In the next frame (frame 2), sensor 3 sends out a *slot request* in sixth Δ-slot of RS-1 to sensor 1 containing RS number of sensor 3 and Δ-slot number of sensor 1 in the RS of sensor 3. Sensor 1 sends a *slot response* in seventh Δ-slot of RS-1 to sensor 3. Same procedure is followed in RS-2. Sensor 3 does not send slot requests in RS-3 and RS-4 because they are coded *AVAILABLE.* After frame 2, sensor 3 has acquired a RS and Δ-slot in each reception slot of its neighbours and each neighbour of sensor 3 knows RS number of sensor 3 and has a Δ-slot in RS of sensor 3.

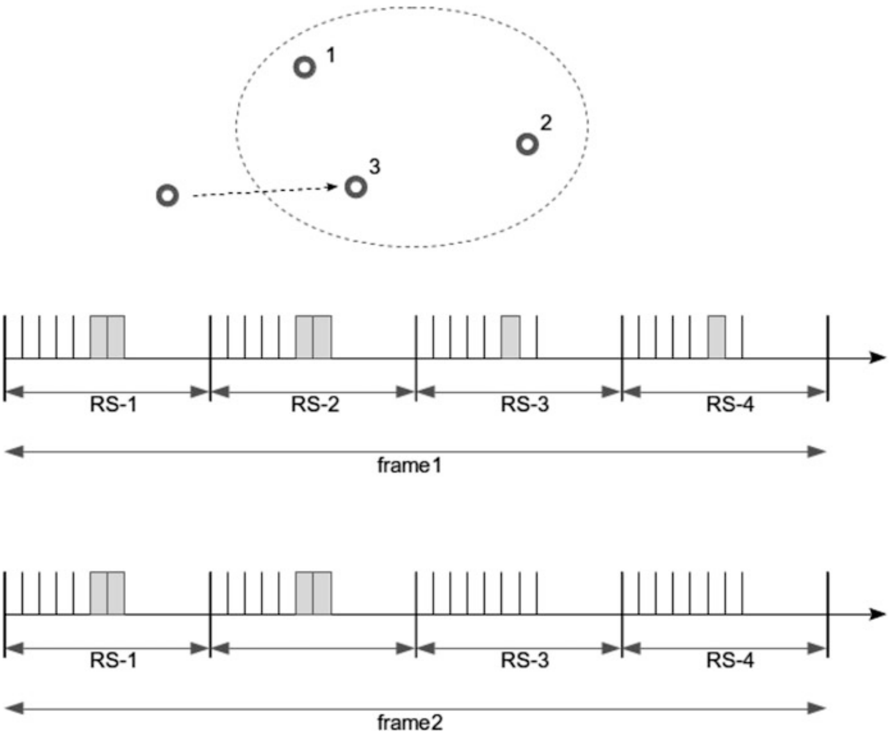

Fig. 3 Reception slot assignment procedure

6.1 Data Transfer

Once a sensor has gained a reception slot mentioned in the procedure explained, it can receive and transmit data. For this purpose, it contains information of one-hop neighbours with the corresponding reception slot and its own Δ-slot in its one-hop neighbour's RS.

Rule 4: *A node can transfer data request only in its own assigned Δ-slot in receiver's RS.*

At the receiver end, every node wakes up in its own reception slot and receives data request. As soon as data request is received by its intended receiver, the receiver switches off its radio and sets timer to become active in N + 1th Δ-slot of current RS. All data requests except first data request intended to the same destination are not received as the radio of the receiver is switched off. In N + 1th Δ-slot of current RS, receiver sends data response which announces the winner of the following data slot of current RS. In other words, data response is destined to the winner of the following data slot of current RS but it is also received by all one-hop neighbours who want to transmit in current RS. The winner of data slot starts sending in following data slot of current RS and all other one-hop neighbours of receiver refrain from transmitting in current RS.

R-MAC does not allow two nodes to transmit to a node in one RS. If a sensor does not get any data request in first N Δ-slots, switch off its radio for the entire data slot.

The idea behind R-MAC is to conserve energy through keeping the node's radio off while it is neither receiving nor transmitting. To achieve this, a sensor turns on its radio only:

- During its Δ-slot, $N + 1th$ Δ-slot and data slot in the reception slot of the receiver when it is ready to transmit.
- During its reception slot.
- **Data-request collision at receiver:** Collision can occur at the receiver if it overhears a data request destined to any other sensor and receives a data request destined to it. After collision, receiver waits for data request in the following Δ-slots. If it receives any data request in following Δ-slots than send a data response in N + 1th Δ-slot, otherwise switch off its radio for the entire data slot.

For illustration, consider the network shown in Fig. 4, where sensor 1 and 2 have same RS (according to **Rule 3,** sensor 1 and 2 can have same RS as they are two-hop neighbours). Δ-slot number of sensor 3 in RS of sensor 1 is same as Δ-slot number of sensor 4 in RS of sensor 2.Assume that sensor 3 has traffic to transmit to sensor 1, and sensor 4 has traffic to transmit to sensor 2. Because sensor 3 is in the range of sensor 2, if sensor 3 and 4 were to transmit in the same frame (node 1 and 2 has same RS) there would be collision at sensor 2. Hence, only sensor 1 transmits data response and sensor 3 subsequently transmits in the following data slot whereas sensor 4 will have to wait for next frame.

Data-Response Collision at Sender Collision can occur at the sender if it overhears a data response destined to any other sensor and receives a data response destined to it. After collision, the sender refrains from transmitting in current RS and retries in the next frame.

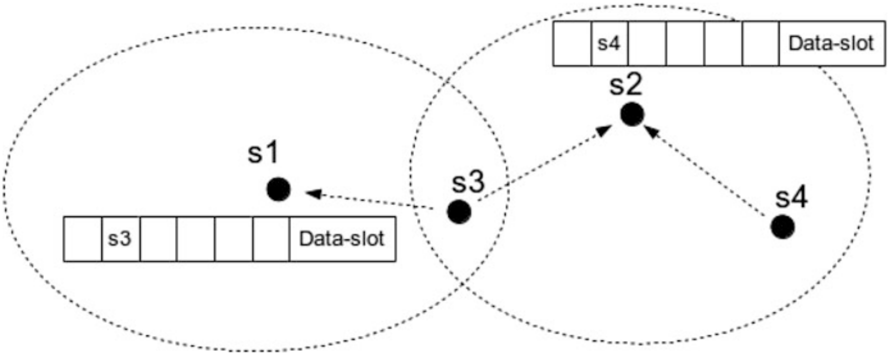

Fig. 4 Data-request collision at receiver

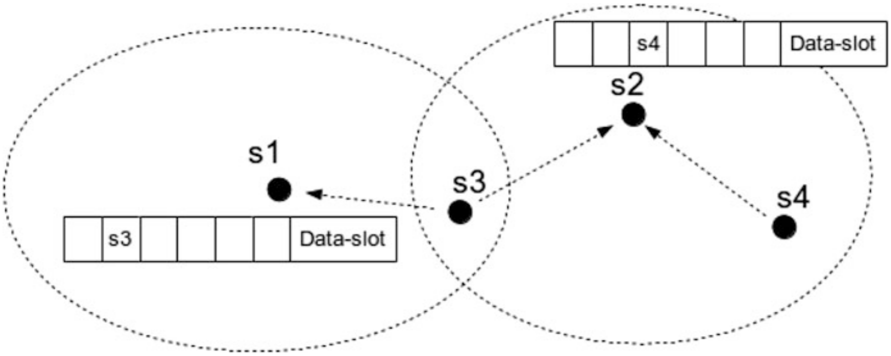

Fig. 5 Data-response collision at sender

For illustration, consider the network shown in Fig. 5, where sensors 1 and 2 have same RS. Δ-slot number of sensor 3 in RS of sensor 1 is different from Δ-slot number of sensor 4 in RS of sensor 2. Assume that sensor 3 has traffic to transmit to sensor 1, and sensor 4 has traffic to transmit to sensor 2. If both sensor 3 and 4 try to transmit in same frame, both sensor 3 and 4 will send data request in their corresponding Δ-slots. Sensor 1 will send data response in N + 1th Δ-slot to sensor 3, and sensor 2 will send slot response in the same Δ-slot to sensor 4 which will be overheard by sensor 3. Hence, there will be collision at sensor 3, and it will refrain from transmitting in current RS and sensor 4 can transmit to sensor 2 in current RS.

7 Evaluation

7.1 Simulation Environment

To test the performance of R-MAC, simulation analysis is conducted in two different scenarios as shown in Fig. 6 [9]. The first one, Fig. 6a is a fully connected cluster where every individual node is in the range of every other node in the network. The second topology (Fig. 6b) is a tree topology where traffic passes from leaf to a data sink node.

The analysis of Reception-based Media Access Control is assessed using three performance metrics: offered traffic, consumed power and end-to-end delivery delay. Table 1 contains the standard simulation parameters.

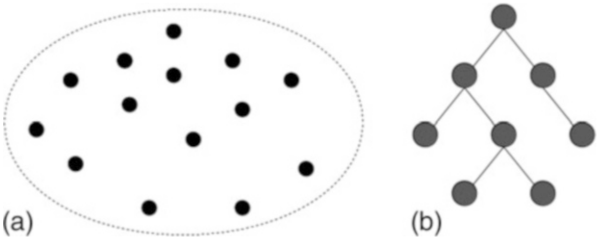

Fig. 6 (**a**) Fully connected cluster topology, (**b**) tree topology

Table 1 Simulation parameter setting

Parameter	Value
Simulation time	10,000 s
Bandwidth	250Kbps
Data slot	565byte
Δ-slot	64byte
Packet length	512byte
Transmission power	24 mW
Reception power	13.5 mW
Idle power	13.5 mW
Sleep power	5 µW

7.2 Fully Connected Cluster Topology

Figure 7 shows the comparison between average delivery delay measured for different values of *N*. Simulation results shows that average delivery delay increases very slowly as offered traffic is increased. Figure 6 also shows that average delivery delay is directly proportional to the value of *N*.

Figure 8 shows the comparison between average power consumption for different values of N in fully connected cluster topology. Average power consumption is inversely proportional to the value of *N*.

Performance of R-MAC is also compared with SPARE-MAC, which is a TDMA-based protocol. Bandwidth and packet length are the same as given in Table 1 for simulation of SPARE-MAC (Fig. 9).

In Fig. 10, comparison of SPARE-MAC and R-MAC is shown for average consumed power versus achieved throughput.

A. *Tree Topology.*

In this section, simulation results are presented for networks where sensors are organized in two-level tree topology, leaf sensors (sensors at level 2) transmit to data sink (sensors at level 0) (Figs. 11 and 12).

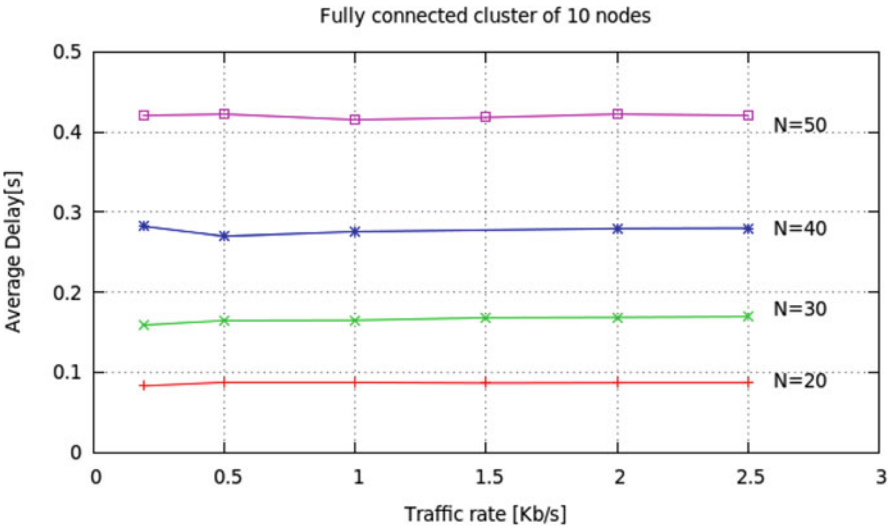

Fig. 7 Average delivery delay versus the offered traffic rate

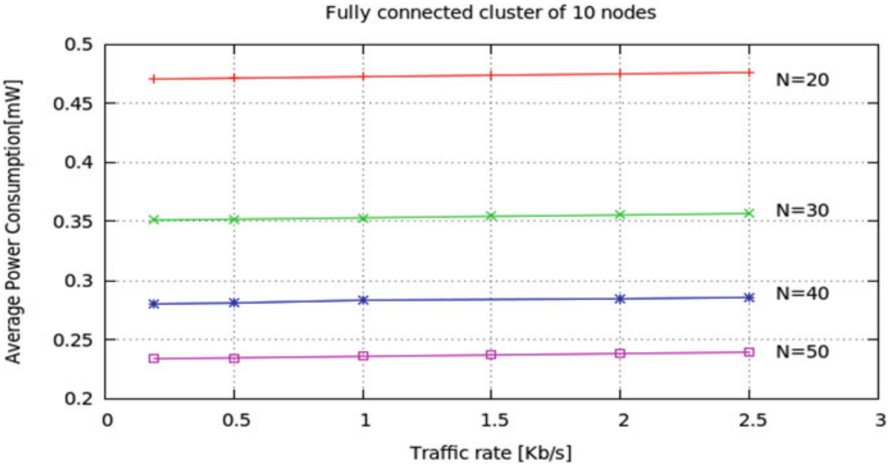

Fig. 8 Average power consumption versus the offered traffic

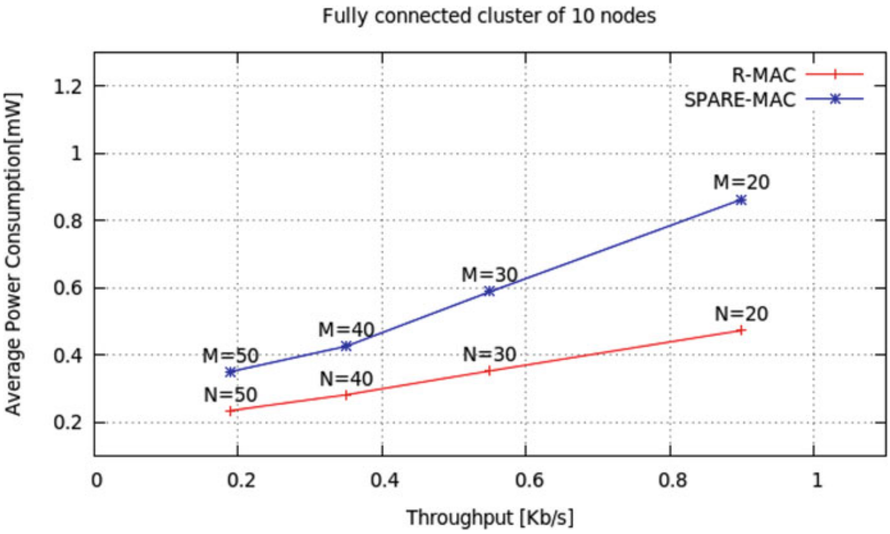

Fig. 9 Comparison between SPARE-MAC and R-MAC. Average delay versus achieved throughput in cluster topology [14]

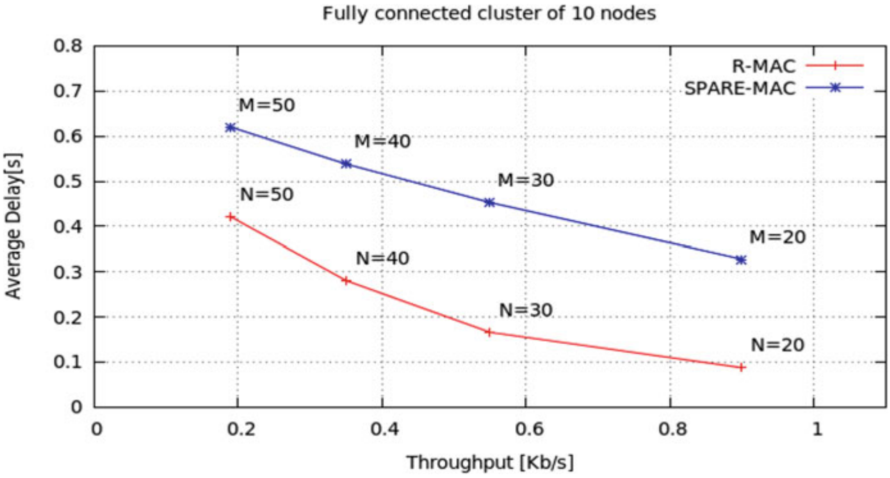

Fig. 10 Comparison between SPARE-MAC and R-MAC. Average consumed power versus achieved throughput

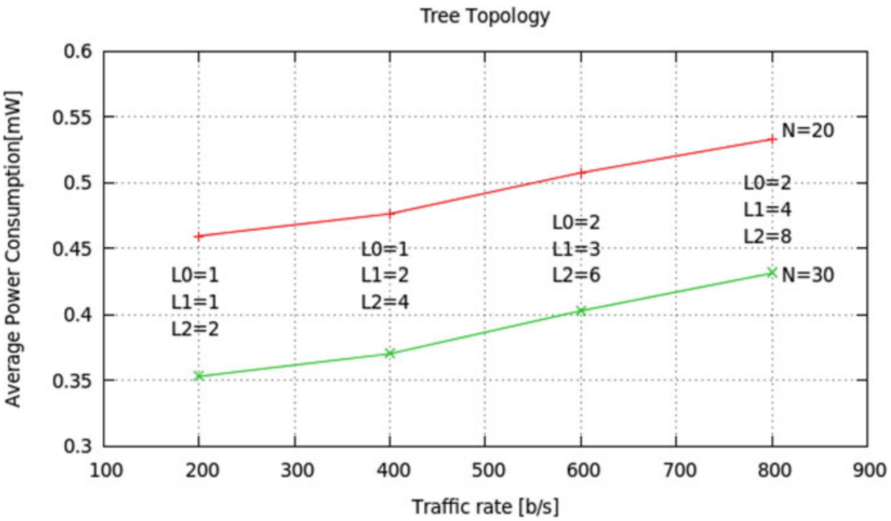

Fig. 11 Average power consumption versus traffic rate at data sink in tree topology (Figure 6b is a tree topology. It is average power consumption versus traffic rate)

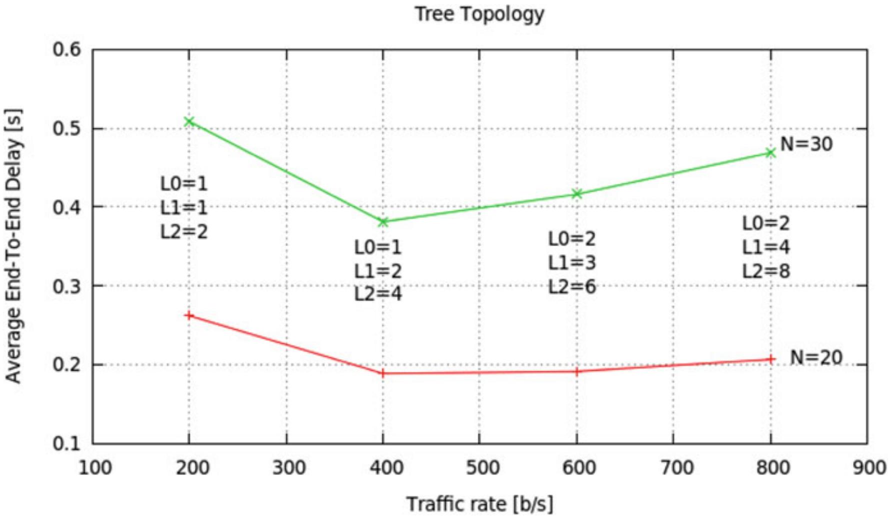

Fig. 12 Average delay versus traffic rate in bits per second at data sink in tree topology (Figure 6b is a tree topology. It is average delay vs traffic rate)

8 Conclusion

The work R-MAC is an energy-efficient Media Access Control approach, which is proposed for message transmission in Wireless Sensor Network and shown improvement in different energy levels.

References

1. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A survey of sensor networks. *Proceedings of IEEE Communication Magazine*.
2. Rowe, A., Mangharam, R., & Rajkumar, R. (2006). RT-link: A time-synchronized link protocol for energy constrained multi-hop wireless networks. *Proceedings of IEEE SECON*, 402–411.
3. Mangharam, R., Rowe, A., & Rajkumar, R. (2006). FireFly: A time synchronized scalable real-time sensor network Plateform. *Proceedings Of IEEE SECON*.
4. Elson, J., & Estrin, D. (2001). Time synchronization for wireless sensor networks. *Proceedings of IEEE IPDPS*, 1965–1970.
5. Ganeriwal, S., Kumar, R., & Srivastava, M. (2003). Timing-sync protocol for sensor networks. *Proceedings Of ACM SenSys*, 138–149.
6. Ye, W., Heidemann, J., & Estrin, D. (2002). An energy-efficient MAC protocol for wireless sensor networks. *Proceedings of IEEE INFOCOM*.
7. Lu, G., Krishnamachari, B., & Raghavendra, C. (2004). An adaptive energy-efficient and low-latency MAC for data-gathering in sensor networks. In *Proceedings of the 4th international IEEE workshop on algorithms for wireless, Mobile, ad hoc and sensor networks (WMAN)*.
8. Dam, T. V., & Langendoen, K. (2003). An adaptive energy-efficient MAC protocol for wireless sensor networks. *Proceedings of ACM SenSys*.
9. Campelli, L., Capone, A., Cesana, M., & Ekici, E. (2007). *A receiver oriented MAC protocol for wireless sensor networks*. IEEE.
10. Gobriel, S., Mosse, D., & Cleric, R. (2009). *Sensor network TDMA scheduling with adaptive slot-stealing and parallelism*. IEEE.
11. Rajendran, V., Obraczka, K., & Garcia-Luna-Aceves, J. J. (2003). Energy-efficient, collision-free medium access control for wireless sensor networks. *Proceedings of ACM Sensys*.
12. Cho, S., Kanuri, K., Cho, J.-W., Lee, J.-Y., & June, S.-D. (2005). Dynamic energy efficient TDMA-based MAC protocol for wireless sensor networks. *Proceedings of ICAS/ICNS/IEEE*.
13. Ye, W., Heidemann, J., & Estrin, D. (2004). Medium access control with coordinated, adaptive sleeping for wireless sensor networks. *ACM/IEEE Transactions on Networking, 12*(3), 493–506.
14. Turati, F., Cesana, M., & Campelli, L. (2009). SPARE MAC enhanced: A dynamic TDMA protocol for wireless sensor networks. *IEEE "GLOBECOM" 2009 proceedings*.
15. Khatarkar, S., & Kamble, R. (2013). Wireless sensor network MAC protocol: SMAC & TMAC. *Indian Journal of Computer Science and Engineering (IJCSE)*., ISSN 0976-5166*, 4*(4).
16. Swapna Kumar, S., Nanda Kumar, M., Sheeba, V. S., & Kashwan, K. R. (2012). Power efficient dynamic MAC protocol (D-MAC) for wireless sensor networks. *Journal of Information & Computational Science, 9*(7), 1795–1805.