www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 7 July 2015, Page No. 13336-13340

A Fast and Efficient RBTRC-MAC Protocol for Event-Driven Wireless Sensor Networks

[1]Hasna P., [2]Annes Philip

[1] M-Tech Student, Computer Science and Engineering, MES college of Engineering
Malappuram, Kerala, India
hasnapulipra@gmail.com

[2] Assistant Professor, Computer Science and Engineering, MES college of Engineering
Malappuram, Kerala, India
annes.philip@gmail.com

Abstract: ***Event-driven wireless sensor networks have been identified as one of the key areas in the field of wireless communication, where most of the time the network will be in light traffic load situation. When an event has detected, large number of packets will be generated. A MAC protocol designed for this kind of WSNs should be able to swiftly adapt to both light and heavy traffic load situations of the network. Here, implemented a receiver-centric MAC protocol called RC-MAC that integrates duty cycling and receiver centric scheduling. To handle heavy traffic load situation triggered by an event, RC-MAC takes advantage of tree based topology and multichannel technique to assist medium access scheduling. Different parent children sets are assigned to different channels. During light traffic load situation, there will be idle listening of sender in existing RC-MAC protocol, when short beacon interval is used. It reduces energy efficiency of existing RC-MAC. Also, during heavy traffic load situation, packet scheduling is not used in existing RC-MAC. So, a randomization of beacon technique enabled RC-MAC is designed to provide energy efficiency in both short beacon interval and large beacon interval cases. Also, packet scheduling is introduced along with receiver-centric scheduling and hence provides better end to end delay compared to existing approach.***

Keywords: Event-driven wireless sensor network, MAC protocol, Duty cycling, Receiver-centric scheduling.

1. Introduction

Wireless sensor networks (WSNs) [1] are wireless networks consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions at different locations. Ambient conditions in the environment surrounding the sensors are measured by them and then these measurements are transformed into signals. These can be processed to reveal some characteristics about the phenomena located in the area around these sensors. Basically, each sensor node comprises sensing, processing, communication and power units. Sensing unit senses data from environment, processing unit performs certain computations on these data, communication unit connects nodes to the network and power unit provides necessary power. Sensor nodes send these collected data towards sink and sink transfers it to corresponding destination.

Main concern in WSN is energy efficiency due to sensor nodes' limited battery power. Reasons for energy wastage during communication in WSN are collision, overhearing, control packet overhead and idle listening. Medium Access Control (MAC) protocols can avoid all these wastages. Task of MAC protocol is to coordinate the time that several sensor nodes access a shared communication medium. In event driven WSN, sensor nodes normally operate under light traffic load situation. When an event has occurred, all sensors in the event area have to collect data about event and should report to the sink. The channel is expected to be very busy at this moment. Event-driven wireless sensor networks have many applications such as surveillance, intrusion detection, target tracking, etc. Mac protocols designed for event driven wireless sensor networks should be able to adapt to both light traffic load and heavy traffic load conditions of network. Energy efficiency is important during light traffic load situation.

1.1 Duty Cycling Technique

Duty cycling is one of the primary mechanisms for achieving low energy consumption in WSNs. In this approach, each node periodically cycles between active state and sleep state. That is whenever communication is not required; the radio is put into sleep mode or low power. Active periods of nodes will be less for low duty cycling based MAC protocols. Nodes operate on low duty cycle consume less energy which helps to increase the lifetime of the sensor nodes. Low duty cycling based MAC protocols are classified as TDMA-based, contention-based and hybrid protocols. In TDMA-based MAC protocols, each node is assigned to a time slot for transmitting or receiving packets to or from other nodes. Contention based protocols achieved duty cycling by tightly integrating channel access functionalities with a sleep/wakeup scheme. Contention-based duty MAC protocol can be classified into two categories: synchronous and asynchronous protocols. In synchronous protocol, the nodes broadcast their next wake up time to their neighbors and here tight time synchronization is required. Asynchronous protocols are not tightly time synchronous, node can operate on their own different duty cycle schedule. Finally, hybrid based protocols combines the strengths of the both protocols while off setting their drawbacks. It is based on the principle of switching the protocol's behavior between TDMA and CSMA depends upon the traffic in the network.

2. Related works

The previous study includes various energy efficient MAC protocols that use low duty cycling technique. Wei Ye et al. [2] proposed S-MAC (Sensor Mac). It is a synchronous duty cycled Mac protocol. For allowing duty cycling in sensor networks, here loose synchronization between nodes is used. To achieve low power duty cycling, the protocol uses three techniques: periodic sleep, virtual clustering, and adaptive listening. The nodes in the network periodically wake up, receive and transmit data, and return to sleep. When the awake period begins, a node exchanges synchronization and schedule information with its neighbors to guarantee that the node and its neighbors wake up concurrently. Here, fixed duty cycling technique is used. Tijis Van dam et al. [3] proposed a protocol named T-MAC (Timeout Mac). It is synchronous based MAC protocol. It shortens the awake period if the channel is idle, thus it improves S MAC. In S-MAC, even if there is no transmission or reception occurs in the network, nodes will remain awake throughout the entire awake period. T-MAC improves S-MAC by listening to the channel for only a short time after the synchronization phase, and if no data is received or send within this time, the node returns to sleep mode. If data is received, the node remains awake until the awake period ends or no further data is received. T-MAC uses adaptive duty cycling. It reduces energy usage for variable workload. But it causes reduced throughput and increased latency.

Michael Buettner et al. [4] proposed a Mac protocol named X-MAC. It is asynchronous duty cycling based protocol. Here, when a node want to send data, a sequence of short preambles are firstly transmitted, after that only data is transmitted. The ID of target node is included in each short preamble packet. When preamble packet is received by a node, it looks at target node ID in packet. Node returns to sleep mode immediately if it is not the intended receiver. Node remains awake for subsequent data packet if it is intended receiver. X-MAC cannot perform well in case of burst traffic. Yanjun Sun et al. [5] proposed RI-MAC. It is a receiver-initiated asynchronous duty cycling based Mac protocol for dynamic traffic loads in WSN. Here, data transmission is initiated by the receiver. In RI-MAC, to check whether there is any incoming data frame for a node, each node periodically wakes up based on a schedule. When node wakes up, if medium is idle it broadcast beacon message into network. Sender remains active and waits for beacon message from receiver. When beacon message is received by sender, it starts data transmission and receiver send acknowledgement using another beacon. After broadcasting beacon, if there is no data is received, node goes to sleep mode. Here, senders have to wait for receiver's beacon to start transmission.

Injong Rhee et al. [6] proposed a hybrid Mac protocol named Z-MAC (Zebra Mac). It combines the strengths of TDMA and CSMA and avoids their weaknesses. Z-MAC achieves high channel utilization and low latency under low contention like CSMA and achieves high channel utilization under high contention like TDMA. It reduces collision among two hop neighbors. It uses slot stealing technique, which may cause collisions among nodes. Z-MAC assigns time slots to all nodes even though; nodes that are not on active path do not have data to send. Gahng-Seop Ahn et al. [7] proposed a hybrid Mac protocol named Funneling-MAC. It is localized and sink oriented MAC protocol. It reduces funneling effect in WSN. Here, CSMA is implemented network-wide, with a localized TDMA algorithm implemented in the funneling region. It assigns slots only to nodes that are in active route. It cannot manage the nodes that are out of communication range of sink.

Recently Pei Huang et al. [8] proposed a receiver-centric MAC protocol called RC-MAC that integrates duty cycling and receiver centric scheduling. During light traffic load situation it uses receiver-initiated low power probing technique. It means when a node want to send data it has to first receive a beacon message from receiver, then only sender can start data transmission. To handle heavy traffic load situation triggered by an event, RC-MAC takes advantage of tree based topology and multichannel technique to assist scheduling of medium access. Even though this method is efficient in many ways, it has two main drawbacks. During light traffic load, it uses receiver centric low power probing technique. When nodes' beacons can be spread out in a beacon interval, it uses offset based method to help a node to estimate the proper time to wake up. If short beacon interval is used, RC-MAC cannot spread out all nodes' beacons. So, offset based method cannot be used to predict receiver's beacon. Another problem is during heavy traffic load situation, in RC-MAC, scheduling is done to child nodes only. That means which node is next sender is selected by parent node. There is no scheduling for the packets that each node has to send. The protocol proposed in this section 3 effectively overcome those problems. Later in section 4 these two protocols are compared on their performance.

3. Proposed work

In this section a new and efficient MAC protocol named RBTRC-MAC protocol is introduced. The framework of the proposed system is as follows. Figure 1 shows the framework of the proposed protocol. After the nodes are deployed in to the network, tree is constructed. Initially all nodes will be in light traffic load situation. Then, a beacon interval is entered into the terminal. If beacon interval is large, receiver initiated LPP technique and offset based method is used. If small beacon interval, receiver initiated LPP technique is used along with randomization of beacon technique and on-demand prediction error correction mechanism. When event has occurred, there will be heavy traffic load situation. Then receiver-centric scheduling is used along with packet scheduling for transmitting data about the event towards sink node. Then each parent children set is assigned to different channel. If vacant channel is not available for all parent children sets, dynamic adjustment of scheduling cycle length is used. Then, channel switching technique is used by nodes for transmitting packets to sink.

3.1 Offset based method

Nodes first enter into set up phase when they are deployed, and all will stay awake during that time. Then, each node randomly selects a time to send beacon and periodically broadcast beacon messages. When a node receives beacon from its neighbor, it will calculate beacon offset, which is difference between node's beacon time and neighbor's beacon time. Suppose, a node i receives a beacon message from node j, it calculates beacon offset $O_{i,j}$ as, $O_{i,j}=B_j-B_i$. So, when a node wants to send data to its neighbor, it can predict neighbor's beacon time by inverting offset calculation. So, sender can wake up earlier than receiver if it wants to send data towards receiver. Details can be found in [8].

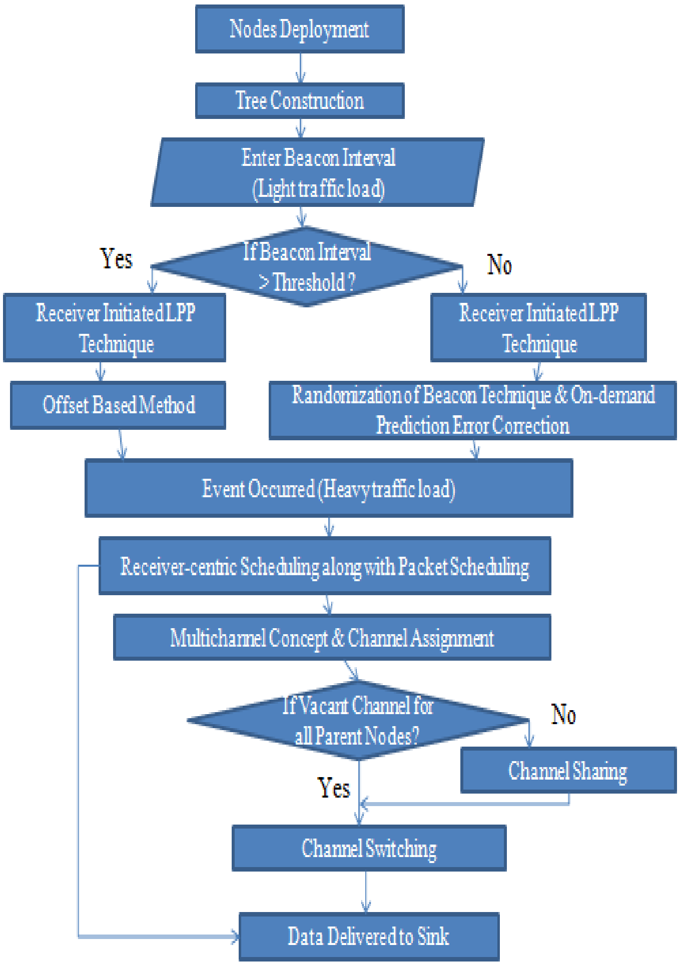

Figure 1: Framework of the proposed system

3.2 Randomization of beacon technique

When short beacon interval is used, during light traffic load situation, then offset based method cannot be used by existing RC-MAC. So, during that situation, in order to avoid idle listening of sender, to predict receiver's beacon time by the sender, randomization of beacon technique [10] is used. In which, all nodes in network compute their wakeup time using a pseudo-random wakeup-schedule generator. Linear congruential generator (LCG) is used for building pseudo-random number generator. It is given by,

$$X_{n+1} = (aX_n + c) \bmod m \quad (1)$$

Where, $m>0$, $0<a<m$, $0<c<m$, $0<X_n<m$. Using this, all nodes calculate their wake up time periodically. Each X_{n+1} generated can be used as a pseudo-random number and becomes the new seed. The parameters a, X_n, c, m will be different for all nodes. When a node S knows the parameters of another node R, then S can predict all future wakeup times of R using that equation. Suppose, when a node S want to send a packet to node R, initially, S does not know the wakeup time of R. So, S waits for a beacon from R. After receiving R's beacon, then S transmits the DATA packet, in which it sets a special flag to request R's prediction state.

Prediction state of R contains the parameters and current seed of the pseudo-random number generator of R as well as the current time of R. When R receives this DATA packet, it sends another beacon that serves both to acknowledge the DATA packet reception (i.e., an ACK beacon) and to allow additional DATA packets to be sent to R. As a response to the prediction state request from S, R also includes its current time and prediction state in the beacon. Then, S compute the time difference between S's and R's clocks using current time of R. Using the prediction information received from the ACK beacon, node S can predict future wakeup times of R. In the future, if S wants to send another DATA packet to R, S wakes up before the predicted wakeup time of R. To prevent senders from missing the wakeup of receivers due to factors like clock drift, on-demand prediction-error correction mechanism is used.

3.3 On-demand prediction error correction mechanism

Prediction error for a wakeup of a node R is defined as the difference between the actual wakeup time and the predicted wakeup time of R. When the sender node notices that prediction error is greater than sender wakeup advance time, then sender will request the receiver, for getting an update of prediction state of receiver. Details can be found in [9].

3.4 Receiver-centric scheduling

Nodes initially operate in duty cycling mode. When event occurs, heavy traffic load will be there in the network. Tree based topology is used to send packets to sink. RC-MAC uses parent children concept. It consists of one parent and multiple children. When event occurs, nodes in event area disable duty cycling and switch to full active mode. Receiver centric scheduling is used in heavy traffic load situation in order to avoid collision within parent children set. RC-MAC operates as pulling data instead of traditional pattern of pushing data. RC-MAC schedules next sender through overhearing.

A node schedules it's children's data transmission by reusing ACK. It means, when receiver (parent) gets data from a sender (child), receiver broadcast an ACK in which ID of next sender is included. It will be overheard by all children. But only scheduled child node send data towards parent node. In order to ensure fairness among source nodes, a scheduling pattern is used by parent nodes. Each time a child is selected as the next sender, the corresponding number of remaining packets of that child is reduced by one. Two variables are used to remember the two children from which it has just received packets. Excluding the two children, parent finds the child with the largest remaining number of packets and selects it as the next sender. RC-MAC does not schedule same node as next sender consecutively, thus fairness is ensured. Detailed process is described in [8].

3.5 Packet scheduling

During heavy traffic load situation, in the existing RC-MAC, receiver-centric scheduling is used. In which, scheduling is done to children nodes by the parent node. There is no scheduling for the packets that each child node has to send towards its parent. In the proposed protocol, packet scheduling is used along with receiver-centric scheduling.

Earliest deadline first packet scheduling is used. Each packet contained in a child node will have a deadline within which it should be delivered to its destination. The packet having earliest deadline is scheduled first by the scheduled child node and it will be forwarded towards its parent node. So, there is no need for the packet having earliest deadline to wait much. Hence, the protocol reduces end to end delay.

3.6 Multichannel concept

Each parent children set need to be placed inside a particular channel to avoid collision between parent children sets. Each node has to keep track of three channels. They are default common channel, data gathering channel and data forwarding channel. Data forwarding channel of a child should be same as data gathering channel of its parent. Parallel data gathering can be achieved since different channels are assigned to different parent children sets.

To assign different channels to parent children set, first step is each node has to obtain an interference list. It contains ID of nodes that are on same level, two levels higher and two levels lower. When a node obtains its complete interference list, for each channel i, it calculates winner of the channel based on priority. If a node cannot find any vacant channel for data gathering after testing all channels, it has to share a channel with other parent children set.

3.7 Channel sharing

When multiple sets share a common channel, it is important to ensure that fairness is guaranteed. For that dynamic adjustment of scheduling cycle length is used. In a scheduling cycle, half of remaining buffer size is allowed to be filled. When a node is received enough packets from its children, to terminate current scheduling cycle, it responds with an ACK that sets PENALIZE as next sender. When such ACK is received, all children of previous set will not participate in data transmission for several rounds. But neighboring nodes contend for sending immediately. So, it reduces contention, since same set cannot win medium access opportunities consecutively. Hence, fairness is ensured.

3.8 Channel switching

Initially all nodes will be in default common channel. Sink uses common channel for data gathering. When sink notices occurrence of event, it performs receiver centric scheduling. Then it checks if required number of packets that it needs to be gathered from its children are got. If yes, then sink send an ACK containing PENALIZE as next sender, in order to indicate termination of scheduling cycle. Then sink waits for some time to give its children a chance to gather data. When children of sink receive this ACK, they set a channel switch timer. It indicates that when timer expires, they should switch back to sink's data gathering channel for data forwarding. Then children of sink perform data gathering. When they receive a packet each of them select a data gathering channel. Then perform scheduling and attach selected data gathering channel to ACK and broadcast to their own children.

When ACK is received, children record their data forwarding channel as obtained data gathering channel from ACK and scheduled child send packet towards parent. If children of sink had completed their one cycle of scheduling, they send ACK containing PENALIZE as next sender and waits for some time to give its children a chance to gather data. When timer interrupt occurs, children of sink need to switch to sink's data gathering channel for forwarding data towards sink. This procedure is repeated for all levels. Detailed process is described in [8].

4. Results and Analysis

The project is implemented using NS2. Nodes are static. Tree is constructed and parents are randomly selected. Receiver initiated low power probing technique is used before event is occurred. When event is detected, nodes in event area send data towards their parents and forwarded to sink. Every parent children set is assigned to different channels for allowing parallel data gathering. Here, 2.4 GHz frequency band is used. It contains total 16 channels. There is 5 MHz spacing between the channels. When a node has multiple children, performance of data collection is improved by using receiver-centric scheduling along with packet scheduling.

Parameter	Value
Number of nodes	31
Simulation area	1000X1000
Node Type	Static
Traffic	CBR

Table 1: Network Simulation Parameters

The proposed randomization of beacon technique enabled RC-MAC (RBTRC-MAC) protocol is compared with the existing RC-MAC. The following graphs (figure 2-3) compare the energy, end to end delay of the two protocols. It is found that the energy consumption due to idle listening is reduced in proposed protocol since; protocol can operate in both short beacon interval and large beacon interval situations during light traffic load. In the existing protocol, if short beacon interval is used during light traffic load situation, then energy is wasted due to idle listening of the sender. But once modification is done it is found that if short beacon interval is used then energy is improved in the proposed protocol since, randomization of beacon technique is used to avoid idle listening of the sender. The graphical results show that end to end delay is considerably reduced in the proposed RC-MAC protocol since; earliest-deadline first packet scheduling is introduced along with receiver-centric scheduling during heavy traffic load situation.

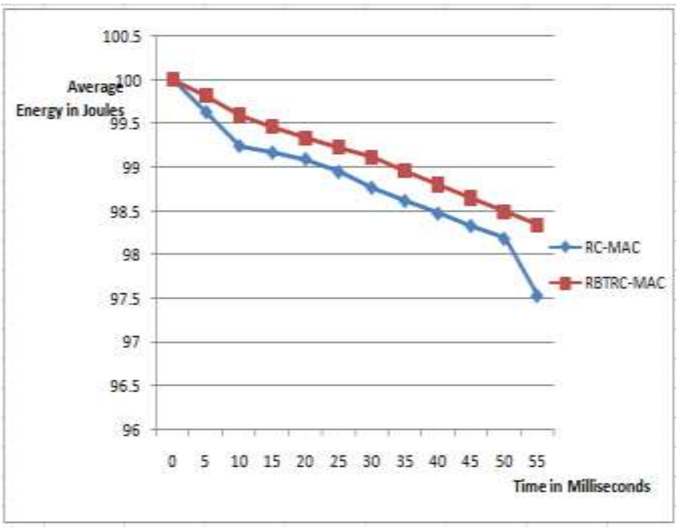

Figure 2: Average Energy Vs Simulation Time

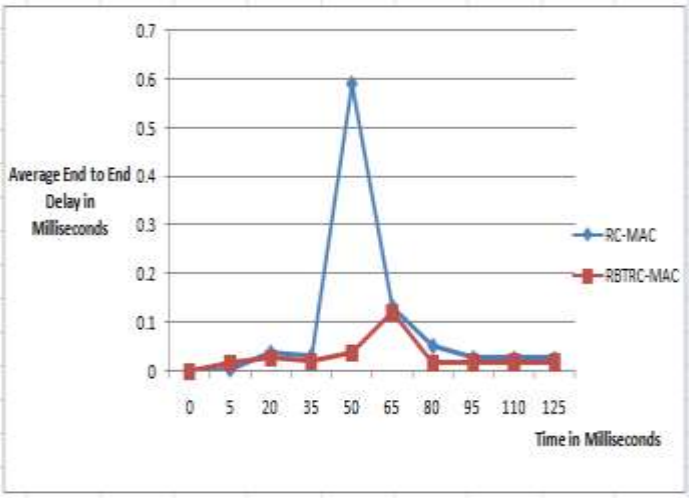

Figure 3: Average End to End Delay Vs Simulation Time

5. Conclusion

Duty cycling is a technique to improve energy efficiency of nodes in wireless sensor network. In this work, several low duty cycling based Mac protocols are studied in detail. Out of them RC-MAC protocol is found more efficient. But it has some drawbacks. During light traffic load situation, if short beacon interval is used, then RC-MAC cannot use offset based method for predicting beacon message of receiver by the sender. Also, during heavy traffic load situation, no scheduling is done for the packets that each node has to send. But the proposed method, which is the modified version of RC-MAC overcome the problems effectively. The experimental results conclude that the proposed method reduces the end to end delay and increases energy efficiency.

References

[1] Holger Karl, Andreas Willig., "Protocols and Architectures for Wireless Sensor Networks", John Wiley Sons, Ltd. ISBN: 0-470-09510-5, 2005.

[2] W. Ye, J. Heidemann, and D. Estrin., "An energy-efficient MAC protocol for wireless sensor networks", in Proc. INFOCOM, vol. 3, pp. 1567-1576, Jun. 2002.

[3] T. van Dam and K. Langendoen., "An adaptive energy-efficient MAC protocol for wireless sensor networks," in Proc. SenSys, pp. 171-180, Nov. 2003.

[4] M. Buettner, G. V. Yee, E. Anderson, and R. Han,"X-MAC: A short preamble MAC protocol for duty-cycled Wireless sensor networks," in Proc. Sensys, pp. 307-320, 2006.

[5] Y. Sun, O. Gurewitz, and D. B. Johnson,"RI-MAC: A receiver initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in Proc. Sensys, pp. 1-14, 2008.

[6] I. Rhee, A. Warrier, M. Aia, and J. Min,"Z-MAC: A hybrid MAC for wireless sensor networks," in Proc. Sensys, pp. 90-101, 2005.

[7] G. S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo,"Funneling-MAC: a localized, sink oriented MAC for boosting fidelity in sensor networks," in Proc. Sensys, pp. 293-306, 2006.

[8] Pei Huang, Chen Wang, Li Xiao,"RC-MAC: A Receiver-Centric MAC protocol for event-driven wireless sensor networks," in IEEE Transactions on Computers, 2013.

[9] Lei Tang, Yanjun Sun, Omer Gurewitz, David B. Johnson "PW-MAC: An Energy Efficient Predictive-Wakeup MAC Protocol for Wireless Sensor Networks," in IEEE INFOCOM, 2011.