Chapter 5

A SURVEY OF MAC PROTOCOLS FOR SENSOR NETWORKS

Piyush Naik and Krishna M. Sivalingam
Dept. of CSEE, University of Maryland, Baltimore County, Baltimore, MD 21250
piyush.naik@umbc.edu,krishna@umbc.edu

Abstract In this chapter, we continue the discussion on medium access control protocols designed for wireless sensor networks. This builds upon the background material and protocols presented in Chapter 4. We first present protocols that are based on random access techniques such as Carrier Sense Multiple Access. These include the Sift protocol, the T-MAC protocol and other protocols. The second set of protocols are based on static access and scheduling mechanisms. These include the UNPF framework, T-RAMA protocols and other related protocols.

Keywords: Medium access control, sensor networks, wireless ad-hoc networks.

5.1 INTRODUCTION

Wireless sensor networks have attracted a considerable attention from the researchers in the recent past as described in previous chapters in this book. Though the initial impetus came from military applications, the advancements in the field of pervasive computing have led to possibilities of a wide range of civilian, environmental, bio-medical, industrial and other applications. In order to practically realize such networks, Medium access control (MAC) is one of the basic protocol functionality that has to be appropriately defined.

The previous chapter presented some of the fundamental issues underlying the design of MAC protocols for sensor networks. In this chapter, we continue this discussion and present a comprehensive survey of other MAC protocols studied for sensor networks. We first present protocols that are based on random access techniques such as Carrier Sense Multiple Access. These include the Sift protocol [1], the T-MAC protocol [2] and other protocols presented in [3, 4]. The second set of protocols are based on static access and scheduling mechanisms. These include the UNPF framework [5], T-RAMA protocols [6] and the work presented in [7].

5.2 RANDOM ACCESS BASED PROTOCOLS

This section presents MAC protocols based on a random access mechanism.

5.2.1 CSMA-BASED EXPERIMENTS

One of the first experimental results for sensor networks based on the Berkeley motes was presented in [3]. The protocol is based on CSMA and its variants based on tuning many system parameters such as: (i) whether random delay is used before transmission, (ii) whether the listening time is constant or random and (iii) whether fixed or exponential window backoff mechanisms are used. An experimental testbed consisting of 10 sensor nodes and a base station was used for the analysis.

A detailed analysis of each of these CSMA schemes is performed through simulations and actual experiments. The 802.11 CSMA with ACK scheme was used as the baseline for comparison. A simple single-hop star topology and a more complex multi-hop tree topology were used in the analysis. The performance metrics considered include the average energy consumed per packet and the fraction of packets delivered to the base station. It was observed that a combination of random delay, constant listening and backoff with radio powered off provided the best results, for the metrics of interest. Interestingly the performance was found to be almost insensitive to the backoff mechanism.

The paper also presents the Adaptive Rate Control (ARC) mechanism. This mechanism tries to balance the originating traffic with the route-through traffic. It is similar to the congestion control scheme of TCP and works as follows. The transmission rate of the either traffic is given by Sp, where S is the original transmission rate and p is the probability of transmission. The factor p is governed by linear increase and multiplicative decrease: p is incremented by adding a constant α $(0 < \alpha < 1)$ on a successful transmission and decremented by multiplying with β $(0 < beta < 1)$ in the case of a failure. In short *alpha* is a reward while *beta* is a penalty. Naturally, a large α makes the scheme aggressive in acquiring the channel while a small β makes it very conservative.

Given that the network has invested more resources in the route-through traffic, it is given more consideration. The penalty (in case of a failure) for the route-through traffic is set to 50% less than that for the originating traffic. Also in order to provide a fair proportion of bandwidth to each node routing through it, the α_{route} is given by $\alpha_{route} = \frac{\alpha_{originate}}{n+1}$, where n is the number of nodes routing through that node.

Experiments and Results. Studies through simulation and actual experiment support the analytical claims and expectations. Delivered bandwidth per node is nearly constant for all the nodes with ARC mechanism as compared to the IEEE 802.11 mechanisms or simple RTS/CTS mechanism where the variance is quite high. This observation clearly proves that the protocol is fair to all the network nodes. Average energy cost per packet is lower than IEEE 802.11 for smaller values of α. This is to be expected because larger values of α tends to be aggressive and injects more originating traffic.

To summarize, Adaptive Rate Control provides a good balance between energy efficiency and fairness of the network.

5.2.2 SIFT: AN EVENT-DRIVEN MAC PROTOCOL

The *Sift* protocol [1] exploits the event driven nature of the sensor networks for MAC protocol design. This work points out that the contention among the sensor nodes is often spatially correlated. This means that at a given time, only a set of adjacent $(R|R \leq N)$ sensors have data to transmit and this is most likely to be after detection of some specific event. Thus, contention resolution may be limited to these *R* sensors rather than the entire set of *N* sensors. The protocol adopts a typical random access protocol such as CSMA or CSMA/CA and uses a fixed size contention window with a non-uniform probability distribution for choosing the contention slot for a node.

Protocol Details. At system initialization, every node uses a large estimate of a node population and hence correspondingly small transmission probability, p_r. Each node also continuously monitors the contention slots and reduces the node count estimate after every contention slot that goes free. That is, a free slot is taken as an indication of fewer number of sensors than assumed. Likewise, the node increases its transmission probability multiplicatively after each free slot. Thus the contention is reduced to geometrically decreasing number of nodes for the same number of contention slots. This is the core idea of Sift wherein the protocol *sifts* the slot-winner node from the others.

The geometrical distribution is chosen to be:

$$p_r = \frac{(1-\alpha)\alpha^{CW}}{1-\alpha^{CW}}\alpha^{-r}, r = 1, ..., CW$$

where α is the distribution parameter and *CW* is the fixed size of the contention window.

Note that p_r increases exponentially with increasing r. This means that later slots have higher probability of transmission than the earlier ones. When a node successfully transmits in a slot or when there is a collision, the other nodes select new random contention slots and repeat this backoff procedure. An example slot probability allocation is shown in Table 5.1.

Table 5.1. Example slot probabilities with the Sift protocol.

Slot	1	2	3	4	5	6	7	8
p_r	0.02	0.03	0.04	0.06	0.1	0.15	0.24	0.37
N_r	24	15	10	6	4	3	2	1

This protocol is designed on the premise that often only a subset of all the nodes need to send their reports. Thus the goal is to minimize the latency for the first R sensors sending their reports and suppress the rest. The scheme indeed works well for this purpose. The detailed probability analysis presented in [1] determines the optimal value of alpha to be 0.82 for a system with 512 sensors. The merit of the protocol lies in the fact that the performance degrades gracefully for more than 512 sensors. The probability of success in both the cases is almost the same until the performance of Sift starts degrading marginally for $N > 512$.

Experiments and Results. A set of experiments is conducted to observe the protocol performance with respect to different metrics viz. latency, throughput and fairness. To capture the burstiness of the sensor data traffic realistically, the work considers a motion-sensor video camera focused on a street and logging every motion event. The log contains the time and x, y coordinates of the motion event. This trace is mapped to an imaginary sensor field with randomly placed sensors. Sensors near the given x, y position at a given time from the log send the reports. The experiments are run to compare Sift with IEEE 802.11 [8]. Latency experiments show a seven-fold latency reduction compared to 802.11. Furthermore, Sift is found to be the least susceptible to the changes in latency with changes in number of reporting sensors and the variation in report times. Throughput analyses show that Sift shows promise under both event-driven and non-event-driven workloads.

Under constant bit rate workload, Sift lags behind 802.11 for a small number of flows (≤ 2). This is due to the higher delay per slot as compared to 802.11 which wins slot early. However as the number of flows increases, Sift performs better and surpasses 802.11 for 8 or more number of flows.

However, the work does acknowledge that Sift does not focus on the energy consumption issue since it constantly listens during the backoff period like 802.11. However, it is possible to integrate the Sift mechanisms with other wireless MAC protocols that focus on minimizing the energy consumption [9].

5.2.3 THE T-MAC PROTOCOL

The T-MAC protocol presented in [2] attempts to improve upon the performance of the S-MAC protocol [10]. It proposes using a dynamic duty cycle as against the fixed one in S-MAC to further reduce the idle listening periods. It also introduces some additional features described below.

Protocol details. Since idle listening is a major source of overhead, T-MAC, similar to S-MAC, maintains a sleep-sense cycle. However instead of having a fixed duty cycle like in S-MAC (say 10% sense and 90% sleep), it has a variable duty cycle. The idea is similar to that of a screen-saver. Just as the screen-saver starts after a certain period of inactivity, the node switches itself to a sleep mode when no activation event has occurred for a predetermined time period. The activation event can be a reception of some data, expiration of some timer, sensing of the communication, knowledge of an impending data reception through neighbors' RTS/CTS and so on. Synchronization of the schedules is achieved in an exactly similar manner as S-MAC through the scheme dubbed as virtual clustering.

T-MAC uses a fixed contention interval to send an RTS. A special case arises in the RTS transmission due to the dynamic duty cycle. When a node sends an RTS, it may not get a CTS back if that RTS was lost or if the receiving node could not transmit because one or more of its neighbors were communicating. In this case, the sender node might go to sleep if it does not hear a CTS for the predetermined time, resulting in a reduced throughput. To correct this problem, T-MAC specifies that the RTS be sent twice before the sender gives up.

The paper also describes another type of problem, called the *early sleeping problem.* Consider a scenario where a node X may constantly lose the contention to transmit an RTS to its neighbor (say N). This can happen if another neighbor of X (say Z), which is not a neighbor of N, is communicating with its own neighbor (say A). As a result the node X has to remain silent either because of an RTS transmitted to it by Z or because of an overheard CTS of the neighbor of Z. The situation is illustrated in the Figure 5.1. Ultimately, the active period of node N ends and it goes to sleep. Now node X can only transmit to N in its next active period. This plight of node X affects the throughput badly. This is termed as the *Early Sleeping problem* because a node (N in this case) goes to sleep even when another node (X here) has data to send to it.

T-MAC offers two solutions to this problem. In the first solution, the blocking node (X) sends a Future-RTS packet to its intended receiver (N), with the information about the blocking duration. The receiving node now knows that it is the future receiver at a particular time and must be awake by then. However this solution is found to increase the throughput and the energy requirements considerably. In the other solution, the node gives itself a higher priority if its

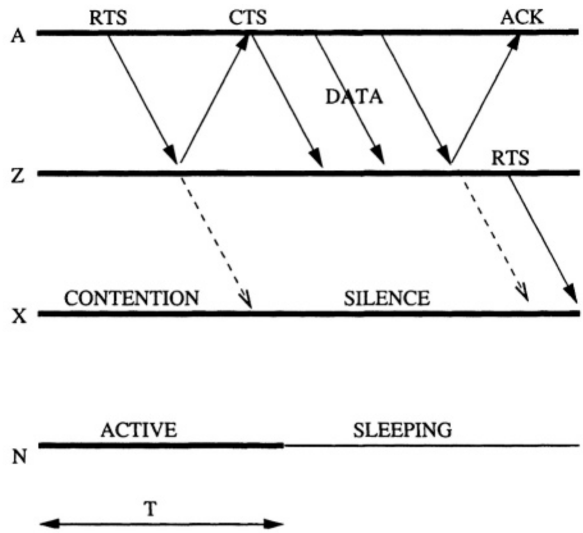

Figure 5.1. The Early Sleeping Problem.

transmission buffers are full. In other words, it will not send a CTS reply to an RTS but would rather send its own RTS for its own contention for the medium. This solution is termed as *full-buffer-priority.* Although this reduces the probability of the early sleeping problem, it is not advisable under high traffic load conditions because it will increase the number of collisions dramatically due to increased aggression. To avoid this, T-MAC puts a constraint that a node may only use the priority if it has lost the contention at least twice.

Experiments and results. Simulations are carried out with energy consumption as the primary metric. In a comparison of T-MAC with CSMA and S-MAC with different duty cycles for a homogeneous local unicast traffic, T-MAC is shown to perform far better than CSMA and at least as well as S-MAC. In a Nodes-To-Sink type of communication, T-MAC again outperforms S-MAC and CSMA especially at higher traffic loads. However it results in lower throughput as compared to S-MAC because of the early sleeping problem even with the FRTS and full-buffer-priority solutions. In a more realistic scenario with event based local unicast where nodes send unicast messages to their neighbors upon the occurrence of certain events, T-MAC is shown to perform the best. Once again, the early sleeping problem limits the overall throughput of T-MAC.

In a separate comparison of the solutions to the early sleeping problem, FRTS provides higher throughput at a higher energy cost while the full-buffer-priority scheme provides a slightly lesser throughput than FRTS but with no additional energy costs. Lastly in a combined simulation of the event based unicast and Nodes-To-Sink reporting, T-MAC has the least energy consumption.

T-MAC has been experimentally implemented on the EYES nodes [11]. Through this implementation, extensive experiments are carried out which provide important power usage characteristics of the sensor nodes.

5.2.4 MEDIUM ACCESS CONTROL WITH CHANNEL STATE INFORMATION

In [4], the authors develop a protocol based on the hypothesis that a sophisticated physical layer model can help improve the MAC protocol. A MAC protocol that uses the channel state information (CSI) in presented in this work. The authors propose a variation of the slotted ALOHA protocol in which the nodes transmit with a probability that is a function of the observed channel state in a particular slot. At the end of the time slot, base station transmits the indices of the nodes from which it received the packets successfully. The channel state is assumed to be identically and independently distributed from slot to slot and from node to node.

A metric termed Asymptotic Stable Throughput (AST) is introduced, where AST is defined as the maximum stable throughput achieved as the number of users goes to infinity while keeping the total packet rate constant [4]. It is evident that given a scheduler which uses the channel state information, AST can be significantly improved. The channel state is chosen to be proportional to the transmit power and propagation channel gain.

This concludes the discussion on the CSMA-based MAC protocols for sensor networks.

5.3 STATIC ACCESS BASED PROTOCOLS

This section present the different static access schemes such as those based on Time Division Multiplexed Access (TDMA).

5.3.1 UNPF PROTOCOLS

In [5], a unified protocol framework denoted *UNPF,* that comprises a network organization protocol, a MAC protocol and a routing protocol, are presented for a large-scale wireless sensor network architecture. In this network architecture, the network nodes are organized into layers where the layers are formed based on a node's hop-count to the base station [12]. For example, a node two hops away from the base station belongs to layer 2 and so on. Fig. 5.2 illustrates how the 10 sensor nodes are organized into three layers. When the network is first created, the nodes are organized into layers by using periodic *beacon* packets. The details of the network organization phase are explained in

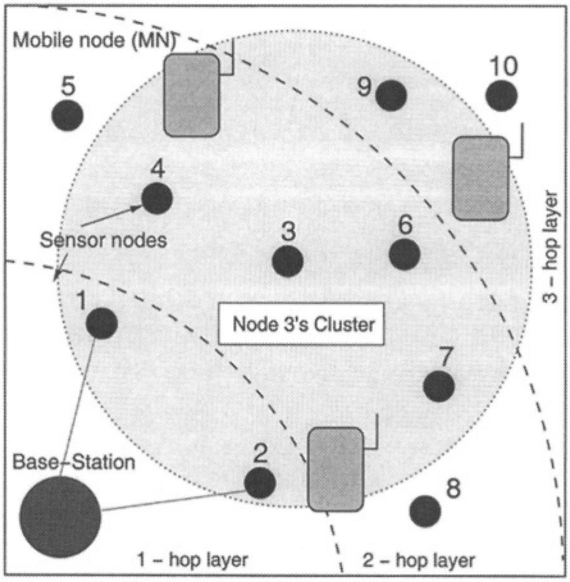

Figure 5.2. An example of MINA with 10 nodes organized into 3 layers. The shaded region represents the broadcast region of node 3.

[5]. The layered organization leads to a simple multi-hop routing mechanism well aligned with the limited memory and computation capabilities of the sensor node. A source node in layer i selects one of its inward neighbors in layer $(i - 1))$ as its forwarding node, and sends the data packet to that node. This node then forwards the packet to *its* forwarding node in layer $(i - 2)$, and so on until the packet reaches the BS. Thus, a node k in layer $i - 1$ acts as a forwarding node for a set of nodes in layer i and transmissions to node k need to be coordinated with a MAC protocol as explained below.

Protocol details. A Time Division protocol is proposed as the MAC protocol. The MAC protocol assumes the availability of a number of channels either in the form of a code or a frequency. The receiver of a forwarding node receiver (in layer i) is assigned a unique channel (with spatial reuse possible) and the MAC protocol is designed to share this channel among the transmitters of the forwarding node's client nodes in layer $i - 1$. A simple scheduling scheme is used for this purpose. This protocol is termed as Distributed TDMA Receiver Oriented Channeling (DTROC). Thus, each forwarding node can define its own schedule for its clients. Also, a node may be able to reach several nodes in its inward layer and chooses one among those as its forwarding node based on criteria such as available buffer space and remaining energy.

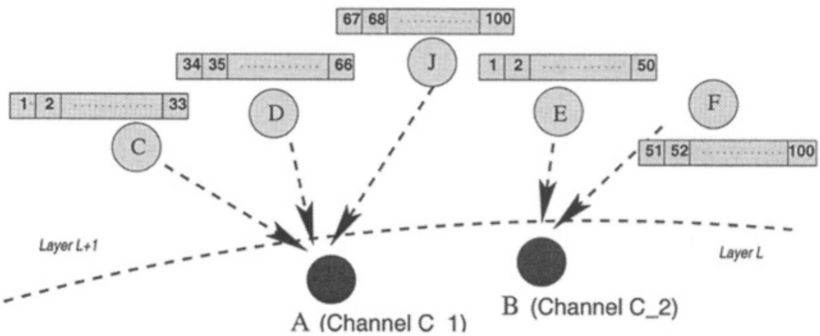

Figure 5.3. Slot allocation for forwarding nodes A and B, in a specific frame.

Fig. 5.3 shows a sample slot allocation and channel assignment for a sensor network with $\beta = 100$ slots in each data frame. In the figure, node A uses channel C_1 and $\mathcal{S}_A = \{C, D, J\}$; similarly, node B uses channel C_2 and $\mathcal{S}_B = \{J, E, F\}$. Note that in the figure, node **J** can choose between node **A** and node **B** as a forwarding node. In this frame, it has selected node A.

Experiments and results. With the layered architecture, the number of layers and the transmission range play an important role. In fact the latter influences the former. The simulations are therefore geared towards studying the effect of variation in the transmission range on different parameters. These parameters include packet latency, average energy consumed per packet, energy-delay product, time to first node death and time to network partition. All the simulations are carried out for varying node density viz. 200, 400, 600, and 800 nodes for a given field size.

With the increase in the transmission range, the number of layers decreases and so does the hop-count. This results in a lesser average delay. However for the transmission range greater than 40-60m, delay actually increases because there are lesser intermediate layers available which causes more queuing delays.

Similarly the energy per packet decreases for the transmission range of up to 60m. This again can be attributed to the lesser hop-count. However for range greater than 60m, the transmit power required is higher resulting in higher energy per packet. Energy-delay product too predictably follows the same trend. However the optimal value of the range varies with the number of nodes in this case.

Network lifetime is quantified by two metrics namely time to first node death and time to network partition. Both the metrics increase with the increase in transmission range. However for the range greater than 120m, the network lifetime drops significantly due to the added cost of transmission.

Thus from these simulations, the optimal range of the transmission is identified to be between 40m and 60m. Thus the MAC protocol for MINA provides a clean way to perform medium access control and complements the other protocols of the suite perfectly.

5.3.2 TRAFFIC ADAPTIVE MEDIUM ACCESS PROTOCOL (TRAMA)

The goal of the TRAMA protocol [6] is to provide a completely collision free medium access and thus achieve significant energy savings. It is primarily a scheduled based MAC protocol with a random access component for establishing the schedules. TRAMA relies on switching the nodes to a low power mode to realize the energy savings. The Protocol has different phases or components namely: Neighbor Protocol (NP), Schedule Exchange Protocol (SEP) and Adaptive Election Algorithm (AEA). NP uses the random access period to gather the one-hop and two-hop neighbor information. SEP helps establishing the schedules for a given interval among the one-hop and two-hop neighbors. Finally, AEA decides the winner of a given time slot and also facilitates the reuse of unused slots.

Protocol details. TRAMA derives from the idea proposed in the Neighbor-Aware Contention Resolution (NCR) [13] to select the winner of the given time slot in a two-hop neighborhood. For every one-hop and two-hop neighbor, a node calculates a MD5 hash of the concatenation of the node-id and the time slot t. This gives the priority of a node for a given time slot. The node with the highest priority is chosen to be the slot winner. After the Neighbor Protocol has gathered the neighbor information using the signaling packets in a random access mode, the node computes a certain SCHEDULE_INTERVAL. This is the duration in which a node may transmit data and is based on the rate at which packets are generated from the application layer. The node further pre-computes the priorities to identify its own winning slots for the duration of SCHEDULE_INTERVAL. These schedules are announced in a schedule packet. Instead of including the receiver addresses in the schedule packet, a bitmap is included for its every winning slot. Each bit in the bitmap corresponding to its every one-hop neighbor; 1 if it is the intended receiver and 0 otherwise. This also simplifies the broadcast and multicast mechanisms.

Broadcast involves a bitmap with all 1's while for the multicast, specific bits corresponding to the intended receivers are set. Looking at the schedule packet, a node may go into sleep mode if it is not the intended receiver of any of its neighbors. This is helpful from the energy efficiency point of view. Also a node may not have enough data to transmit during all of its winner slots. In order not to waste these vacant slots and allow their re-use, the node announces these slots with a zero bitmap. However the last of the winning slots is reserved for the node to announce its schedule for the next SCHEDULE_INTERVAL. Due to this provision for the re-use of slots, schedules for the nodes may no longer remain synchronized. This is because a node may use some other node's unused slot to transmit. To alleviate the problem, schedules are timed out after a certain time period. Furthermore, schedule summaries are sent piggybacked along with the data packets to help maintain the synchronization.

The Adaptive Election Algorithm (AEA) determines the state of the node at a given time and facilitates the slot re-use. For every node, the protocol keeps track of the nodes in its neighborhood that need extra slots to transmit. The set of such nodes is called as *Need Contender Set*. Every slot that is owned but unused by a node X is given to the node with the highest priority in the Need Contender Set of node X. However, inconsistencies may arise as shown in Figure 5.4. For node B, node D is the winner since it has the highest priority in its two-hop neighborhood. But for node A, node D is not visible and hence it assumes itself as the winner. Thus both nodes A and D may transmit. Suppose node B is not the intended receiver for node D and goes to sleep. However the node A may transmit to B in which case the transmission will be lost. In order to deal with such a problem, a node also keeps track of an Alternate winner along with the Absolute winner. The node has to account for the Alternate winner as well if the Absolute winner does not have any data to send. In this case, D is the Absolute winner and A is the Alternate winner for B.

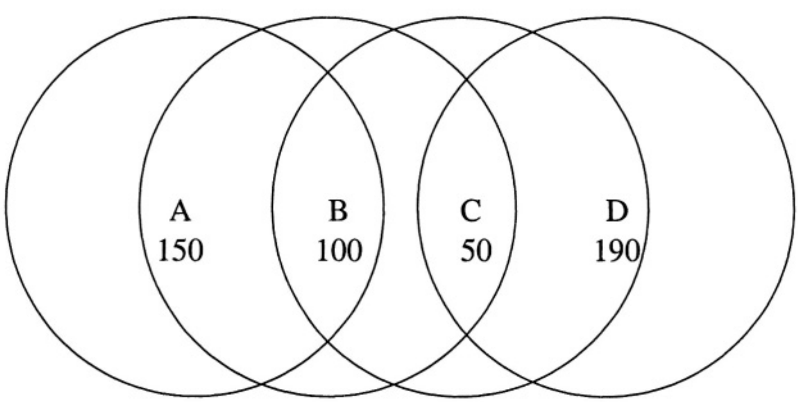

Figure 5.4. The slot inconsistency problem.

Experiments and Results. The performance of TRAMA has been studied using detailed discrete-event simulation based experiments. The performance of TRAMA is compared with both contention based protocols (IEEE 802.11 and S-MAC) as well as a scheduled based protocol (NAMA [13]). One set of experiments is conducted using with an exponential inter-arrival time for the data. In this case, a neighbor is randomly selected to perform either the unicast or the broadcast. TRAMA is shown to achieve higher throughput than the contention based protocols. This is to be expected given the collision-free nature of TRAMA and the fact that contention based protocols perform poorly due to the collisions. As to NAMA, the other scheduled access based protocol, the throughput achieved is comparable to that of TRAMA. Broadcasts are also found to be more feasible in case of scheduled based protocols, in particular TRAMA. This again can be attributed to the collision freedom guaranteed by TRAMA.

For the study of the energy efficiency, detailed investigations are performed by comparing the performance of TRAMA with that of S-MAC [10] and IEEE 802.11. The metrics considered are *sleep time percentage,* defined as the ratio of number of sleep slots to the total slots and the *average sleep interval,* used to measure the number of radio mode switches. In case of average sleep time percentage, S-MAC with 10% duty cycle (10% sense, 90% sleep) fares better than TRAMA. However TRAMA has better average sleep interval than S-MAC. This means that the switching between radio modes is more frequent in the case of S-MAC. The price to pay for the scheduled access based protocols is the higher latency. TRAMA incurs higher average queuing delays than the IEEE 802.11 and S-MAC. However it performs better in this respect than its other counterpart, NAMA.

The simulations are also performed under different sensor scenarios by varying the position of the sink node in the field (edge, corner or center). The results are almost similar to the ones observed before with the notable exception in case of the percentage sleep time. TRAMA exhibits higher energy savings than S-MAC in all the scenarios.

5.3.3 ENERGY AWARE TDMA BASED MAC

Another approach based on TDMA is considered in [7]. It assumes the presence of "gateway" nodes which act as the cluster heads for clusters of sensors. The gateway assigns the time slots to the sensor nodes in its cluster. Naturally this TDMA scheme eliminates majority of the potential collisions. Marginal possibility of collisions still exists in the case that a node does not hear the slot assignment. However this is highly limited.

Protocol details. The protocol consists of four phases namely, data transfer, refresh, event-triggered rerouting and refresh-based rerouting. Data transfer phase, understandably is the longest of all. Refresh phase is used by nodes to update the gateway about their current state (energy level, position etc.) This information is used by the gateway to perform rerouting if necessary. This is done during the event-triggered rerouting phase. Another form of rerouting occurs during refresh-based rerouting, which is scheduled periodically after the refresh phase. In both the rerouting phases, the gateway runs the routing algorithms and sends the new routes to the sensors.

The paper presents two algorithms for the slot assignment based on Breadth First Search (BFS) and Depth First Search (DFS). These graph-parsing strategies specify the order in which slot numbers are assigned to the nodes starting from the outermost active sensors. In BFS, the numbering starts from the outermost nodes giving them contiguous slots. On the other hand, the DFS strategy assigns contiguous time slots for the nodes on the routes from outermost sensors to the gateway. Figure 5.5 illustrates the two ideas.

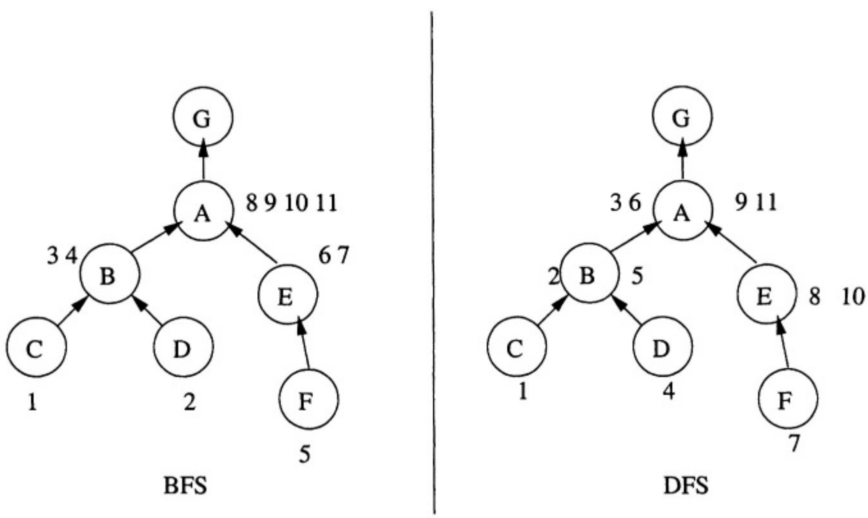

Figure 5.5. **Slot Assignments for BFS and DFS Strategies. Note that Node E serves both as a sensor as well as a relay while nodes A and B only serve as relays.**

With the BFS strategy, the relay nodes need only turn on once to route their children's packets. If the cost of turning the sensor nodes ON and OFF is high, this scheme offers a good economical option. On the other hand, the relay nodes need to have sufficient buffer capacity to store the data packets until it is time to forward them. This makes it susceptible to the buffer overflows and associated packet drops. The DFS strategy, on the other hand, does not demand any buffer capacity for the relays. However the relays have to switch on and off multiple times, which makes it a less attractive choice from the energy savings point of view.

Experiments and Results. The simulations are performed for a 1000 × 1000 square meter field with 100 randomly deployed nodes. The effects of buffer size on various parameters (e.g. end-to-end delay, throughput, energy consumed per packet, node lifetime, packet drop count etc.) are considered for both BFS and DFS mechanisms. Because there are no sensor state changes, BFS consumes less energy per packet and hence offers a higher node lifetime. However, DFS offers lesser end-to-end delay and lesser packet drop count and thus higher throughput. This is because there are no overheads associated with the buffers.

5.4 SUMMARY

This section presented a survey of the some of the recent medium access control protocols specifically designed for wireless sensor networks. The protocols were categorized based on the random access or static access nature. Further research is necessary in this topic to address highly scalable MAC protocols for networks involving a very large number (say 10,000) of nodes. Also, a comprehensive qualitative evaluation of the various protocols for different scenarios and traffic patterns is necessary. In addition, more experimental MAC-level protocol results especially for scheduling-based protocols will be useful.

ACKNOWLEDGMENTS

We would like to thank DAWN lab members Aniruddha Rangnekar and Uttara Korad for their valuable suggestions and help in preparing this document.

REFERENCES

[1] K. Jamieson, H. Balakrishnan, and Y. C. Tay, "Sift: A MAC protocol for event-driven wireless sensor networks," Tech. Rep. MIT-LCS-TR-894, Massachusetts Institute of Technology, May 2003.

[2] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *ACM Intl. Conference on Embedded Networked Sensor Systems (SenSys),* (Los Angeles, CA), Nov. 2003.

[3] A. Woo and D. Culler, "A transmission control scheme for media access in sensor networks," in *Proc. of ACM Mobicom,* (Rome, Italy), Aug 2001.

[4] S. Adireddy and L. Tong, "Medium access control with channel state information for large sensor networks," in *Proc. of IEEE Intl. Workshop on Multimedia Signal Processing,* (St. Thomas, US Virgin Islands), Dec. 2002.

[5] J. Ding, K. M. Sivalingam, R. Kashyapa, and L. J. Chuan, "A multi-layered architecture and protocols for large-scale wireless sensor networks," in *Proc. IEEE Semiannual Vehicular Technology Conference – Fall,* (Orlando, FL), Oct. 2003.

[6] V. Rajendran, K. Obraczka, and J. Garcia-Luna-Aceves, "Energy-efficient collision-free medium access control for wireless sensor networks," in *ACM Intl. Conference on Embedded Networked Sensor Systems (SenSys),* (Los Angeles, CA), Nov. 2003.

[7] K. Arisha, M. Youssef, and M. Younis, "Energy-aware TDMA based MAC for sensor networks," in *IEEE Workshop on Integrated Management of Power Aware Communications Computing and Networking (IMPACCT 2002),* 2002.

[8] IEEE, "Wireless LAN medium access control (MAC) and physical layer (PHY) Spec." IEEE 802.11 standard, 1998.

[9] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J.-C. Chen, "A survey of energy efficient network protocols for wireless networks," *ACM/Baltzer Wireless Networks,* vol. 7, no. 4, pp. 343–358, 2001.

[10] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proc. IEEE INFOCOM,* (New York, NY), June 2002.

[11] P. Havinga, "Energy efficient sensor networks: European research project (ist-2001-34734)." http://eyes.eu.org, 2003.

[12] R. Kashyapa, "Medium access control and routing protocols for data gathering using wireless sensor networks: Design and analysis," Master's thesis, Washington State University, Pullman, Aug. 2001.

[13] L. Bao and J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," in *Proc. ACM MobiCom,* (Rome, Italy), July 2001.