

Addressing Mobility in Wireless Sensor Media Access Protocol

Huan Pham, Sanjay Jha
School of Computer Science and Engineering
The University of New South Wales
Email: {huanp, sjha} @cse.unsw.edu.au

Abstract

Handling mobility in wireless sensor networks presents several new challenges. Techniques developed for other mobile networks, such as mobile phone or mobile adhoc networks can not be applicable, as in these networks energy is not a very critical resource. This paper presents a new adaptive Mobility-aware Sensor MAC protocol (MS-MAC) for mobile sensor applications. In MS-MAC protocol, a node detects its neighbor's mobility based on a change in its received signal level from the neighbor, or a loss of connection with this neighbor after a timeout period. By propagating mobility presence information, and distance from nearest border node, each node learns its relative distance from the nearest mobile node and from nearest border node. Depending on the mobile node movement direction, the distances from mobile and border nodes, a node may trigger its neighbor search mechanism to quicken the connection setup time. The simulation results show that, the new mobility-aware MAC protocol can work very energy-efficiently when the network is stationary, whereas it performs much better in terms of throughput than the existing sensor MAC (SMAC) protocol in scenarios involving mobile sensors.

1. INTRODUCTION

In wireless sensor networks, maximizing battery life is a very important design criterion, because in many applications changing or recharging battery after deployment is normally not economical or not feasible. To maximize the battery lifetime, protocols at the data link layer usually put sensors in a sleep mode for most of the time, and only let them wake up periodically for data communication. This mode of operation, which trades network Quality of Service (QoS) such as delay, throughput for energy saving, proves to be very effective in stationary network, where connection formations and break-ups are rare events. However, these protocols, which are customized for stationary networks, may not work well in mobile scenarios, such as a patient assistance system which monitors patients' health conditions via wearable bio-sensors, workers equipped with sensor device in disaster recovery situation, soldiers in battle field, and provides them with appropriate care when needed.

In such mobile sensor applications, each communication node could be very mobile and the level of mobility may

vary depending on the time of the day. Existing protocols, which heavily rely on periodic sleep and infrequent neighbor search to save energy, may not create a new connection for mobile sensors fast enough and the network performance may degrade so badly that it may become unworkable. The trade-off between QoS for the energy saving in this case need to be reconsidered. To be effective in both stationary and mobile scenarios, we need protocols that can work efficiently in terms of saving energy for sensors when they are stationary, and at the same time these protocols need to provide an acceptable performance level when sensors are mobile. Such protocols need to be mobility-aware. None of the Medium Access Control (MAC) protocols proposed so far has such a capability.

Following are the main contributions of this paper:

- Detailed design of a new mobility-aware MAC protocol to save energy consumption and to provide reasonable QoS in both stationary and mobile scenarios.
- Proposal of a novel mechanism to handle mobility adaptively according to node mobility status.
- To our knowledge, this is the first MAC protocol to address the sensor mobility aspect.
- Validation of hypothesis through discrete event simulation.

The remaining of this paper is organized as follows. Section 2 presents the background and related work on MAC layer design for sensor networks, and mobility handling issues. Section 3 presents detailed design of the new MS-MAC protocol. Section 4 discusses the simulation setup and results. Section 5 concludes the paper with its main contribution and the future work.

2. BACKGROUND AND RELATED WORK

A. Different sleep mode implementations

In sensor network applications, where sensors are densely deployed (a few meters apart), the power required for the idle listening (or receiving) mode is quite significant compared to the power required for the transmit mode. If the nodes are in the transmit mode for one percent of the time, while in the idle listening mode for rest of the time (please refer to data for Crossbow motes in Table 1 [1]), the total energy spent in idle listening could reach as much as 97% of the total energy consumption. Therefore, to maximize the

TABLE 1: POWER BREAKDOWN OF CROSSBOW MICA2DOT MOTES

Consumed by	Mode	Current drawn
CPU	Active (5MHz)	8mA
	Sleep	< 15 μA
Radio	Transmit	27 mA
	Receive	10 mA
	Sleep	< 1 μA

battery life, MAC for sensor networks have to minimize the time that nodes spend in the idle listening state.

For this reason, many techniques have been introduced to shut the nodes' radio down, i.e., to put nodes in the sleep mode, as long as possible. Related MAC protocols for sensor networks are reviewed in the rest of this section.

A Carrier Sense Multiple Access (CSMA) based MAC protocol proposed in [2], [3] lets nodes sleep for most of the time, and only wake up periodically to listen to the channel. Nodes use the same sampling rate, at which they wake up to listen for the channel, but they do not have to be synchronized in time. Before transmitting any packet, a node has to send a long preamble signal (at least of the sampling period). The receiver hears the preamble and stays awake until after the transmission of data finishes. The energy wasted on idle listening is reduced. However, the transmission energy as well as the chance of collision increases.

Another way to reduce idle listening while not requiring long preambles is to have nodes time-synchronized and they only wake up for very short time at the beginning of each cycle to communicate in a contention based mode. Nodes need to be time synchronized only in each virtual cluster rather than network-wide. Nodes on the border of two clusters can communicate directly with each other, as long as they know the schedules of the both clusters. This idea is proposed in S-MAC [4]. However, under current S-MAC implementation, there is no effective support for mobility. To create a new connection with a new node running a different schedule, a node has to wait for next synchronization period of 10 seconds which appears in a semi-regular time interval of 5 minutes. A new mobility aware technique for forming new connections should result in a more energy-efficient protocol and should improve successful transmission rate in scenarios with mobile sensors.

Time Division Multiple Access (TDMA) is a contention-less MAC scheme, in which different pairs of nodes get access to the medium in different time slots. The biggest natural advantage of TDMA over all other multiple access schemes is its ability to turn a node's radio off for most of the time, and only turn it on for the assigned time slots. The idle listening issue is, therefore, reduced very significantly. The biggest challenge associated with the TDMA scheme is the requirement for stringent time synchronization. Such a network-wide time synchronization is hard to be implemented in adhoc scenarios, and is expensive in terms of communication overhead.

Self-Organizing Medium Access Control for Sensor Networks (SMACS) [5] is a TDMA based MAC protocol that

overcomes the requirement for all nodes to be synchronized to each other by dividing nodes into subnets and having them time synchronized in each subnet only. To enable mobile nodes or newly introduced nodes to create new connections, stationary nodes broadcast BI (Broadcast Invite) messages at semi-regular time, and listen for any response from mobile nodes or newly added nodes. Mobile nodes, that regularly monitor the channel, simply eavesdrop these messages and respond only if a new connection is required.

However, this protocol is only designed to save energy for stationary nodes. Mobile nodes are assumed to have plenty of energy. When the mobile nodes are stationary, they can not work like other stationary nodes to conserve energy because mobile nodes have different time slot structure from that of stationary nodes. The protocol, therefore, can not be used in scenarios where mobile nodes are also energy-limited.

B. Mobility handling

Mobility handling has been dealt with in mobile telephone networks and mobile ad-hoc networks for many years. However, in these two former technologies, energy is not a critical consideration. Specifically in mobile phone networks, it is most important to minimize the hand-off drop rate, call block rate and the hand-off delay. In mobile adhoc networks, the primary objective is to maximize the throughput, and to minimize the packet loss and delay. Energy consumption is only a secondary design consideration.

In mobile telephone networks, since hand-offs can be controlled centrally, many sophisticated strategies to make mobile handoff seamless have been developed using macro/micro cell handoff scheme, different channel assignment approaches which may take into account user tracking information. All those techniques are not applicable for adhoc networks such as wireless sensor networks, where there is no central control body.

In mobile adhoc networks, where energy consumption is only secondary issue, nodes could be awake all the time to monitor the wireless channel for new data and for new neighbors. The IEEE 802.11 standard suite, for instance, specifies a contention-based MAC protocol where all the nodes within the coverage range continually monitor and share the channel. Mobile nodes can quickly see new neighbors and set up new connections. This property makes mobility handling at the link layer not an issue and mobility is mainly dealt with at the network (routing) layer. It should be noted that according to IEEE 802.11 specifications, nodes which do not have data to transmit or receive may be turned off while the nearby nodes are in transmission. However this sleep feature is mainly to deal with the unnecessary interference to surrounding nodes, and not primarily to save energy.

In wireless sensor networks, energy is the primary design constraint. Much work has been proposed to address mobility at the network layer, taking into account the energy constraint, such as [6], [7]. In our opinion, since majority of energy consumption is for idle-listening at the data link layer, significant energy saving could be achieved if mobility handling

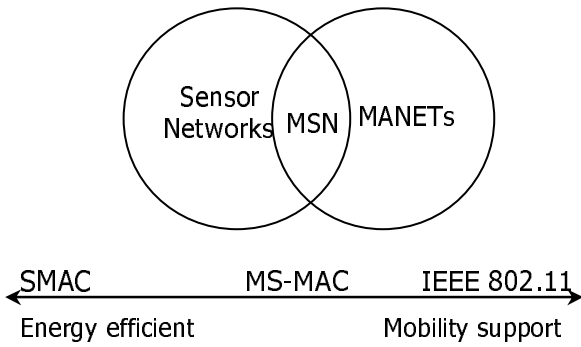


Fig. 1: MS-MAC duty cycle varies with level of mobility

is addressed right at this layer. Handling mobility in wireless sensor networks at the link layer centers around how to create new connections quickly for mobile sensors (how quickly is an application specific consideration) while maintaining a low energy consumption level. Mobility awareness is crucial in addressing this challenge.

None of the existing protocols proposed for sensor networks in the literature are designed to be adaptive to a mobile's presence. In the existing protocols, nodes are not intelligent enough to sleep more often to conserve energy when they are stationary. Similarly they do not wake up more often when there are mobile nodes around. It is clear that mobility is a big sink of energy and handling mobility at the MAC layer needs to be aware of the presence of mobile nodes.

3. MS-MAC PROTOCOL

A. An extension to SMAC

The objective of design for the new MAC protocol for sensor networks is a that it should work energy efficiently in both stationary and mobile scenarios. To achieve this objective, we take S-MAC as a starting point and extend the protocol to support mobile sensors. The mobility-aware MAC protocol for sensor networks (MS-MAC) would work similar to S-MAC to conserve energy when nodes are stationary. At the other extreme, nodes may stay awake similar to IEEE802.11 [8] for a very mobile scenario. Figure 1 shows the relationship between our MS-MAC protocol and SMAC as well as IEEE802.11.

S-MAC is basically a CSMA/CA MAC protocol, based on IEEE 802.11. SMAC introduces periodic coordinated sleep/wakeup duty cycles, thus extending the battery lifetime for sensor nodes. Nodes do not have to use the same sleep/wakeup schedules for the entire network, but they are divided into *virtual clusters*, in which nodes follow same schedules. In each virtual cluster nodes wake up at the same time to transmit/receive. If a node does not have any packet to transmit and there are not any packets addressed to it, it goes back to sleep again. Data transmission follows the carrier sense and RTS/CTS procedure proposed in IEEE 802.11 to overcome the hidden node problem. The awake/sleep cycle is shown in Figure 2.

Virtual cluster formation is a random process, depending on the time when nodes first turn on. Basically, the one that

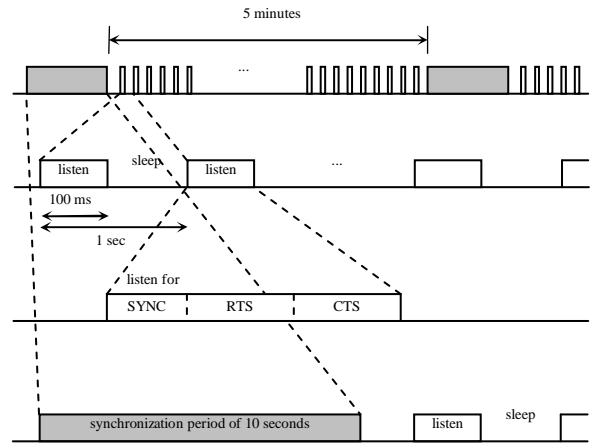


Fig. 2: SMAC listening, sleeping, synchronization cycle

wakes up later will adopt the schedule of its first neighbor as the primary schedule, and the size of virtual clusters keep on increasing as new nodes turn on. When a node that already has some neighbors sees new neighbors with different schedules, it should work as a border node and it will keep the schedules of new neighbors as well. The border node uses these secondary schedules to communicate with nodes belonging to other virtual clusters.

To avoid time drift, nodes need to be resynchronized periodically. After a predefined number (ten) of cycles, each node broadcasts its schedule, so that its neighbors can update that information in their schedule tables. To avoid two neighbor nodes being unable to see each other (for example, due to SYNC packet corruption, interference, or because the medium keep busy and SYNC packets can not be sent in time), each node periodically follows the neighbor discovery scheme. For this scheme, the *synchronization period* (10 seconds) is repeated every 5 minutes if the node has at least one neighbor. It is repeated more often when the node does not have any neighbors, for instance, every 30 seconds instead of every 5 minutes. The synchronization period is shown as dark blocks in Figure 2.

The SMAC protocol described above works well when the network is mainly stationary, in which case the connection formations and brake-ups are not frequent. If a mobile node wants to set up a new connection with a new node in a different cluster, it has to wait for a new synchronization period (which is 10 seconds every 5 minutes), to be able to detect the SYNC message from the new node.

The S-MAC proposal to increase the frequency of synchronization period of a mobile node to twice a minute, when the mobile node does not have any connection, does quicken the time it takes for a mobile to create new connections with new neighbors. However, this is a *"reactive"* scheme that gets activated only when all connections with the mobile have been lost. The time it takes for a node to create a new connection is large. This can be as long as 30 seconds (waiting for the next synchronization period) plus any timeout period (of several minutes) before a node discovers that it does not

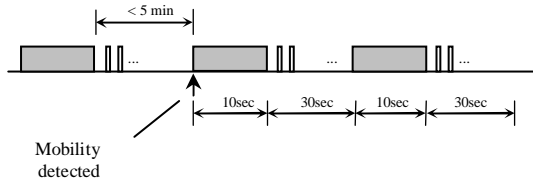


Fig. 3: MS-MAC goes directly into synchronization period when mobility detected

have any neighbors left. During this time, the mobile node is disconnected from the network, resulting in a high packet loss rate, or even an unworkable situation.

If this proposal is changed to become a *“proactive”* one, i.e., to have the resynchronization period frequency increased as the number of neighbors decreases, it does shorten the time it takes for a mobile to find new neighbors. However, this also means that in sparse part of the network, nodes have to stay active unnecessarily even though that proactive neighbor search mechanism does not lead to better network connection.

In addition, S-MAC proposal of increasing the frequency of synchronization period if a node does not have any connection does not work efficiently within a single virtual cluster either. This is because nodes in a virtual cluster all follow the same schedule. Increasing the frequency of synchronization period does increase the energy consumption, yet, in this case it does not increase the chance of successfully finding a new neighbor.

The proposed Mobility aware Sensor MAC protocol is designed as an extension to SMAC and is more suitable for mobile sensor network setting. Instead of passively waiting for connections to be lost before actively searching for new neighbors, MS-MAC proactively activates the neighbor search mechanism based on the mobility information (Figure 3). This extension is discussed in more detail in the next sections.

B. Mobility aware protocol

To be able to work energy-efficiently as well as to guarantee some level of performance under both stationary and mobile scenarios, we need a new mobility-aware protocol, which can *“make before break”* connections for mobile nodes. In other words, nodes need to be able to detect the presence of nearby mobile nodes and search for new neighbors more intensively compared to the stationary case.

There are several ways of detecting mobility, e.g., based on radio, IR, acoustic, ultrasound signals or some combination. The mobility information can also come from the application layer such as by using a motion sensor or a speed meter. In the proposed MS-MAC protocol, a change in the radio signal level of received SYNC messages is used as an indication of mobile’s presence. Although this method does not provide a high level of accuracy, it works with the proposed MS-MAC protocol, which will be described in more details in the rest of this section. A wrong detection of mobile node may trigger neighbor search unnecessarily, but it does not degrade the network performance. Unlike other methods using IR, acoustic or ultrasound signals, the radio signal of SYNC messages is available at no extra cost in terms of energy consumption.

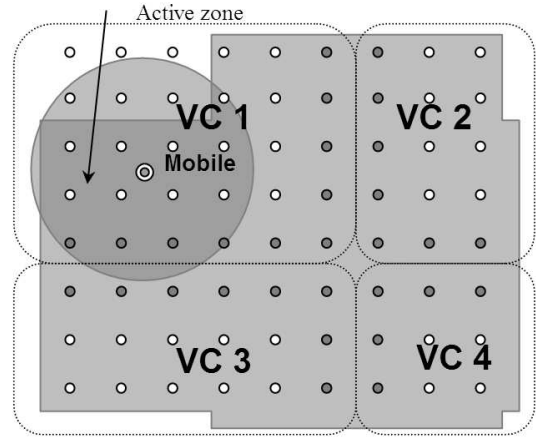


Fig. 4: MS-MAC active zone is created to facilitate connection setup for mobile sensors

In a single virtual cluster, where all nodes follow same wakeup/sleep schedules, two nodes that come within the radio coverage of each other can quickly find each other upon receiving the next SYNC message (that carries the schedule information) from the other. This connection setup process normally takes less than 10 seconds. In contrast, getting into a new virtual cluster without knowing the new schedule is a undesirable situation for the mobile sensor as it has to wait for long time for the next synchronization period (by default it is 5 minutes) during which the mobile is disconnected from the entire network.

To reduce the chance of a mobile node crossing the border region without detecting the new schedule of the new cluster, we would like the nodes that are within R hops from mobile node and R hops from border nodes to form an active zone. In other words, the active zone is an overlap of the area around the mobile node and the border region (Figure 4). R is a *“hand-off”* parameter and by default it is chosen as 2. A higher value of R will increase the size of the active zone, supporting better node mobility, but on the other hand, it is more likely to increase the chance of a node going to the active state unnecessarily, i.e., consuming more energy.

In the active zone, nodes immediately go into synchronization period without waiting for the next scheduled synchronization period. In addition, nodes in the active zone also repeat this synchronization period more often than in normal operation at an interval of 30 seconds instead of 5 minutes.

These nodes in the active zone aim at expediting new connection setups with new neighbors for mobile nodes transparently while moving from between clusters.

To be able to create the active zone in this way, instead of carrying only information on the schedule of the sender nodes (as is the case in SMAC), each MS-MAC SYNC message also carries the distances (hop counts) from the sender node to its nearest mobile and to the nearest border nodes. Upon receiving SYNC messages from neighbors, each node works out the distance from itself to its nearest mobile and nearest border node and whether it belongs to the active zone.

In addition, if the mobile is approaching (i.e., the hop count from this node to the mobile reduces) then the node renews its active status by immediately starting over the synchronization period and resetting the active status timer. Otherwise, the node goes back to normal operation after a timeout period of 5 minutes.

This mobility aware mechanism of MS-MAC protocol allows nodes to work efficiently in both stationary and mobile scenarios. Under a stationary scenario or when the mobile node moves only within a single virtual cluster, all nodes work in a very energy efficient mode. Similar to SMAC, apart from waking up for a very short time at the beginning of each cycle, each node only stays awake for the synchronization period of 10 seconds every 5 minutes.

When there is a mobile node crossing cluster borders, the mobile node and surrounding nodes form an "active zone" R hops away around the mobile node. In the active zone, nodes work with a higher duty cycle. The nodes in the active zone go into synchronization period sooner than in normal operation and the synchronization periods are repeated more often. This full duty cycle synchronization period allows nodes to set up connections with new neighbors in a timely basis.

4. SIMULATIONS RESULTS AND DISCUSSIONS

To evaluate the performance of the new MS-MAC protocol, we have implemented it in NS2 [9] and have set up identical network scenarios to compare the performance of MS-MAC and SMAC. It should be noted that the current implementation of SMAC in NS2 that comes with the NS2 distribution does not have any support for node mobility. Also, there is no implementation of synchronization periods, neighbor updates and neighbor timeouts. We have introduced these functionalities into the NS2 implementation of SMAC so that we could compare MS-MAC and SMAC performance under the same mobile sensor network scenarios.

A network of 10×10 nodes, which has the grid size of 20 meters, is set up for simulations. The radio coverage of each node is set to be 35 meters so that each node can reach a maximum of 8 other neighbors. Nodes wake up at different times and form 3 virtual clusters as shown in Figure 5. A special node (mobile node) is located near the bottom left corner of the network and works as a sink to collect data that are broadcast periodically every 50 second from the node in the bottom left corner. Simulations are run for 4000 seconds, and at 800, 1500, 2000, 2500, 3000, 3500 seconds the mobile node travels at a speed v for the next destination (D1, D2, D3, D4, D5, D6) along the complete route of the mobile shown in Figure 5. A series of simulations are run with MS-MAC, and SMAC as the MAC layer protocol, and with mobile speed v ranging from 0 (stationary case) to 20 meters per second. For simplicity, flooding is used as the routing protocol. The simulation results are shown in Figures 6, 7, 8.

Figure 6 shows the average packet loss measured at the sink for MS-MAC and SMAC at different mobile speed. It is indicated that at low speed of less than or equal to 3 m/s, SMAC and MS-MAC have very similar drop rates of about

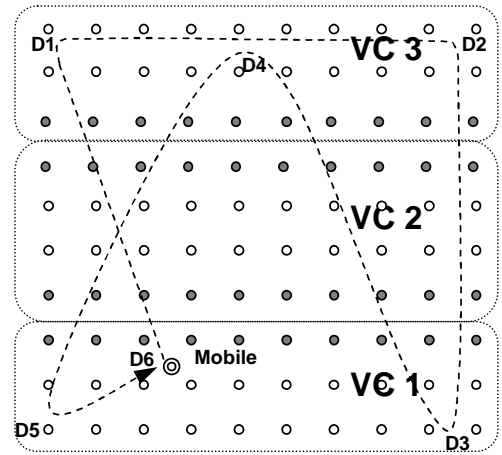


Fig. 5: Simulation scenario

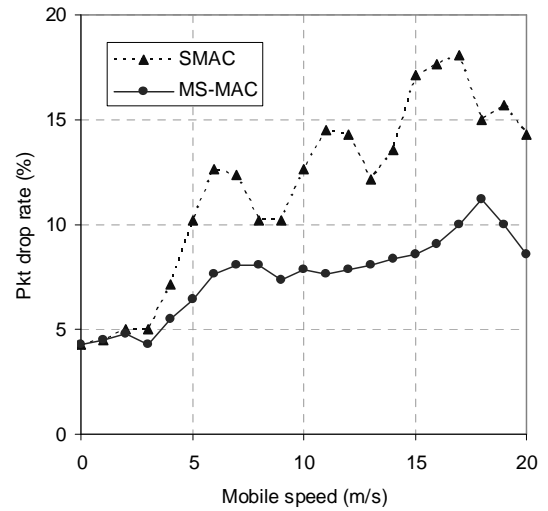


Fig. 6: Packet drop rate

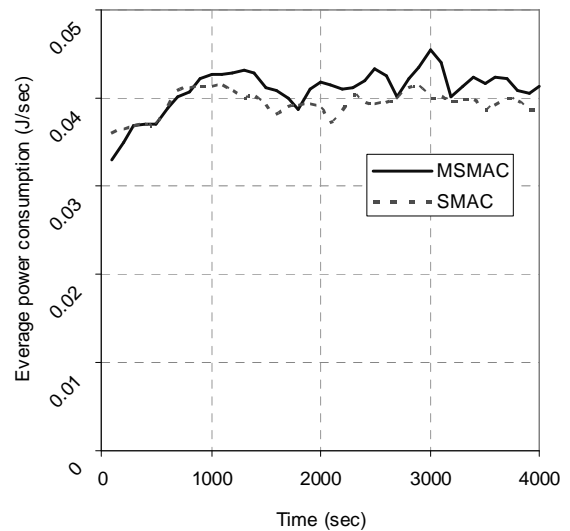


Fig. 7: Average power consumption when $v = 5\text{m/s}$

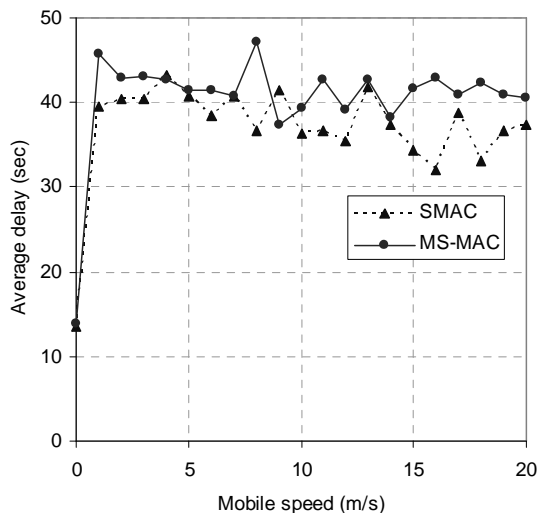


Fig. 8: Average end-to-end delay

5%. In this case, both SMAC and MS-MAC can maintain connection for the mobile node with neighbors while traveling and losses are mainly due to packet collision.

However, as the mobile travels at higher speed, the losses due to the mobile's disconnection from the network start to occur. The difference in the performance of SMAC and MS-MAC becomes evident. At the speed of 5m/s or higher, the packet losses due to mobility if SMAC is used are about 2 to 3 times higher than that of MS-MAC.

This sharp difference in performance results from the fact that in MS-MAC, the active zone is created around the mobile node and border region, while SMAC does not have such support. The active zone lowers the chance of a mobile being invisible in the border region and reduces the time it takes for mobile to see neighbors in the new cluster if the mobile does not yet have the new schedule. In other words, the chance the mobile in MS-MAC case is disconnected from the remaining of the network is lower than in SMAC case.

Figure 7 shows the average energy consumption per second when the speed of the mobile v is 5m/s. As we can see, MS-MAC consumes just a bit more energy than SMAC does. The main source of energy consumption in two cases is the idle listening mode that each node spends in 10% of their duty cycle. The energy overhead used for mobility handling in MS-MAC is negligible because in the given simulation scenario, the time the mobile node spends in the active state is much smaller than the time it is in the "stationary" mode. This simulation result proves that MS-MAC is very energy efficient when the mobile node only travels occasionally.

As far as the end-to-end delay is concerned, figure 9 shows that the two protocols give similar performance. Since the link layer and transport layer are both best-effort where there is no guarantee on the delivery of the transmitted data, the end-to-end delay is mainly influenced by the position of the source and the sink. However, considering the difference in the loss rate of MS-MAC and SMAC, if a reliable transmission mechanism is implemented at the link layer (hop by hop) or

transport layer (end to end), then MS-MAC would result in much lower end-to-end packet delay than SMAC.

In summary, in scenarios where nodes could be mobile for sometime and stationary for the other time, MS-MAC consumes relatively just a fraction more energy than SMAC. Yet, MS-MAC performs much better than SMAC in terms of packet drop rate and potentially end-to-end delay if a reliable transmission is implemented at the link layer or transport layer.

5. CONCLUSION AND FURTHER RESEARCH

The new mobility and speed aware MS-MAC protocol aims to work energy-efficiently in both stationary and mobile scenarios, while not compromising the performance requirements. To achieve this objective, we introduce a novel mobility handling mechanism at MAC layer. This mechanism is activated and controlled based on the presence of mobile nodes. The simulation results show that the MS-MAC protocol performs better than the S-MAC protocol for sensor networks in scenarios involving mobile sensors. More specifically, it would provide much lower packet drop rate than S-MAC when the network has mobile nodes, whereas it consumes almost the same energy compared to S-MAC. Our future work aims at prototyping the protocol using Xbow nodes to test the protocol performance in the real hardware.

ACKNOWLEDGMENT

This work is a part of Networking Infrastructure for Intelligent Environment Applications project, funded by the Smart Internet Technology CRC, Australia.

REFERENCES

- [1] Crossbow, "MPR/MIB Mote Hardware Users Manual", Crossbow website at <http://www.xbow.com/Support/manuals.htm>.
- [2] A. El-Hoiydi, "Aloha with Preamble Sampling for Sporadic Traffic in Ad Hoc Wireless Sensor Networks", Proc. of ICC 2002.
- [3] A. El-Hoiydi, J. D. Decotignie, C. Enz and E. Le Roux, "Poster Abstract: WiseMAC, and Ultra Low Power MAC protocol for the WiseNET Wireless Sensor Network", Proc. of First International Conference on Embedded Networked Sensor Systems (SenSys 2003), November, 2003.
- [4] W. Ye, J. Heidemann, D. Estrin, "Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks", Technical Report ISI-TR-567, USC/Information Sciences Institute, January, 2003.
- [5] K. Sohrabi, J. Gao, V. Alawadhri, and G. J. Pottie, "Protocols for Self-Organization of a Wireless Sensor Network", IEEE Personal Communications Magazine, October, 2000.
- [6] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks", Proc. of The ACM Annual International Conference on Mobile Computing and Networking (Mobicom), August 2000.
- [7] H. S. Kim, T. F. Abdelzaher, W. H. Kwon, "Minimum-Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks", Proc. of First International Conference on Embedded Networked Sensor Systems (SenSys 2003), November, 2003.
- [8] LAN MAN Standards Committee of the IEEE Computer Society. "IEEE Std 802.11-1999", Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE, 1999.
- [9] NS Network Simulator, <http://www.isi.edu/nsnam/ns>
- [10] S. Singh and C. Raghavendra, "PAMAS: Power Aware Multi-Access Protocol with Signalling for Ad-hoc Networks", ACM SIGCOMM Computer Communication Review, 1998.
- [11] A. Woo and D.E. Culler, "A Transmission Control Scheme for Media Access in Sensor Networks", Proc. of The ACM Annual International Conference on Mobile Computing and Networking (Mobicom) 2001.