

# MoX-MAC: A Low Power and Efficient Access Delay For Mobile Wireless Sensor Networks

Papa Dame BA

Université Cheikh Anta DIOP de Dakar, Senegal  
papadam83.ba@ucad.edu.sn

Bamba GUEYE

Université Cheikh Anta DIOP de Dakar, Senegal  
bamba.gueye@ucad.edu.sn

Ibrahima NIANG

Université Cheikh Anta DIOP de Dakar, Senegal  
iniang@ucad.sn

Thomas NOEL

Université de Strasbourg, France  
noel@unistra.fr

**Abstract**—Mobility management in Wireless Sensor Networks (WSNs) is a complex problem that must be taken into account in all layers of the protocol stack. But this mobility becomes very challenging at the MAC level in order to do not degrade the energy efficiency between sensor nodes that are in communication. However, among medium access protocols, sampling protocols reflect better the dynamics of such scenarios. Nevertheless, the main problem, of such protocols, remains the management of collisions and idle listening between nodes. Previous approaches like B-MAC and X-MAC, based on sampling protocols present some shortcomings. Therefore, we address the mobility issue of WSNs that use as medium access sampling protocols. It is worth noticing that sampling protocols remain the most adaptive protocols for dynamic scenarios.

In this paper, we propose a mobile access solution based on the X-MAC protocol which remains a reference protocol. This proposal, called MoX-MAC, incorporates different mechanisms in order to mitigate the energy consumption of mobile sensor nodes. Through an extensive experimental validation, we show that MoX-MAC enables to reduce the energy consumption of sensor nodes, as well the medium access delay of node with respect to the channel, and the average packets loss.

**Index Terms**—Wireless Sensor Networks, Mobility, MAC layer, Performance.

## I. INTRODUCTION

Recent advances in miniaturization of electronic systems have given rise to low power, low cost and multifunctional devices: wireless sensors. The organization of such sensors, in a network in order to cooperatively accomplish a task, takes the name of Wireless Sensor Network (WSN) [1], [2], [3]. However, WSNs are often deployed in hostile environments where optimization of the energy consumption of sensor nodes becomes crucial for the network lifetime [4].

Nowadays, the distributed robotics and low embedded systems have led to a new class of Mobile Sensor Networks (MSNs) that can be used for a wide range of other applications (eg. ocean exploration, monitoring, automobile applications). MSNs have a same architecture with respect to WSN where sensors are fixed.

Therefore, MSNs are constrained by the same energy and processing limitations, but they are supplemented with implicit or explicit mechanisms that enable these devices to move in space (e.g. motor or sea/air current) over time. Among

numerous advantages that they have over the static WSNs (sensors are fix), MSNs need an efficient handling of mobility in all layers with respect to the sensor network protocol stack. The requirement to handle mobility adds another dimension to sensor network protocols, in addition to conservation of energy and computation resources [5].

In this paper, we address the problem of mobility in WSNs, and we focus on the MAC (Medium Access Control) layer. We explore in particular the main existing categories of MAC protocols in WSNs, and afterwards we identify the problems caused by mobility, and expose the most significant existing solutions.

Through a comparative study, we note that the sampling protocols (protocols that send a preamble before sending the data) [6], [7], [8] remain the most adaptable category for dynamic scenarios. Nevertheless, in this category, the solution proposed in [8], which is based on the *B-MAC* protocol [7], does not provide an effective mechanism against nodes' over-hearing [9] caused by the use of a long preamble.

In the literature, *X-MAC* protocol [6] is a reference protocol in terms of energy efficiency according to sampling protocols. In this paper, we highlight the problems that it faces in dense and dynamic networks and then propose a new protocol called *MoX-MAC*. Note that, this protocol is based on the well known protocol X-MAC which is designed for static WSNs. The MoX-MAC protocol is able to deal with static sensors as well those that are in movement. Therefore, MoX-MAC allows to reduce collisions during communication between mobile and static nodes while maintaining the performance of X-MAC.

The remaining of this paper is organized as follows. Section II presents the background and related work on MAC protocols for WSNs and mobility challenges. In Section III, we introduce our hybrid medium access protocol and point out the contributions of MoX-MAC in contrast to previous approaches. Following that we present results obtained through extensive simulation. Finally, we conclude and present some research perspectives in Section IV.

## II. BACKGROUND

Medium access control is critical for enabling successful network operation in order to avoid collision meanwhile to fairly and efficiently share the bandwidth resources among multiple nodes. According to the underlying control mechanisms such as collision avoidance, medium access, MAC protocols can be typically classified into three main categories: sampling protocols, slotted protocols and hybrid protocols [1].

In sampling protocols [6], [7], [8], nodes send a preamble before sending data. Each node in the network periodically switches its radio and listens the medium. If no signal is detected, the node turns off his radio. In contrast, if a preamble is detected, the node stays awake to receive the subsequent data. The preamble thus serves to synchronize a set of nodes to ensure they are ready to receive data sent by the transmitter of the preamble.

B-MAC [7] is the most famous protocol in this category. However, B-MAC suffers from node's "Overhearing" caused by the sending of a long preamble. The phenomena of Overhearing is due to the fact that a node receives packets that are destined to other nodes [9]. In contrast, X-MAC [6], overcomes this problem by splitting the long preamble used in B-MAC into small ones.

In slotted protocols [10], [11], nodes are organized around a common timetable. Time is divided into slotted intervals, which are used by nodes to send or receive data, or to turn off their radios [8]. S-MAC [10] is the most famous protocol based on this principle.

Hybrid protocols [12], [1] combine the strength of the slotted access (*TDMA*) and random access (*CSMA*) while offsetting the weaknesses of both access methods. For instance, the *Z-MAC* protocol [12] can be characterized as a hybrid protocol.

### A. Mobility challenges in MAC solutions

In addition to the five main sources of energy consumption that are: overhearing, collisions, overemitting, idle listening, the control-packet overhead [9], mobility in WSNs brings some new challenges in the design of MAC protocols, including managing the scheduling, transmission and packet resolution. MAC protocols such as sampling, slotted, and hybrid described in Section II may present several shortcomings when they are used in dynamic networks.

Slotted protocols can hardly integrate mobile sensors in their communication scheduling algorithms. Sampling protocols may however face synchronization problems between mobile and fixed nodes. In fact, when a node receives a preamble either with B-MAC or X-MAC, and afterwards it becomes mobile, it can happen that when the data arrives the node is far away to signal radio. Therefore, the data will be lost.

Such issue may happen when a mobile node transmits a preamble while moving. Furthermore, sending preambles reduces the channel availability and therefore increases the competition among the nodes. According to the frequency of the data collection in the network, the performance of mobile sensors can rapidly decrease [8].

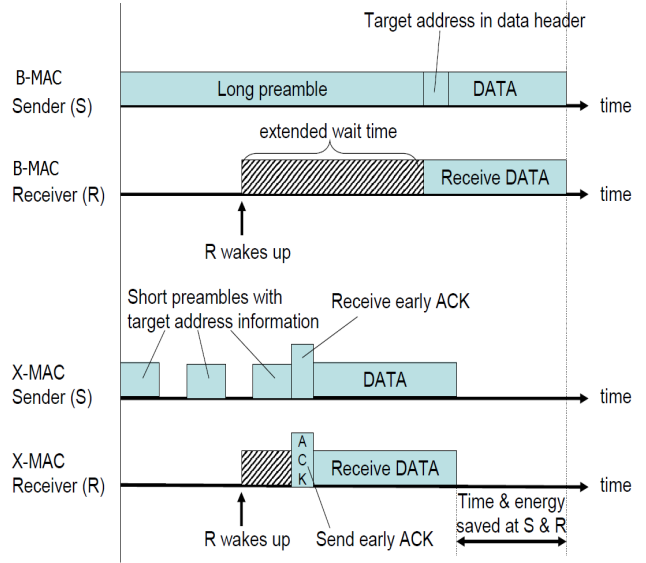


Fig. 1. Comparison of the timelines between B-MAC and X-MAC protocols.

Hybrid protocols seem to have a good adaptability to traffic conditions, but suffer in contrast to the problem of complexity of the control-packet overhead that leads to a high energy consumption.

Due to different issues illustrated above, our MoX-MAC protocol proposed here is based on a sampling approach.

### B. Brief overview on sampling protocols

Following B-MAC [7] approach, the sender should send the entire preamble even though the receiver was woken up during the starting of the preamble's transmission. It is worth noticing that the goal of the preamble is to synchronize the sender and the receiver, as well to freeze the transmission of other nodes that hear this preamble. In so doing, the preamble mitigates the risk of collision.

The entire preamble needs to be sent before every data transmission because there is no way for the sender to know that the receiver has woken up. The Figure 1 illustrates a comparison of the use of a preamble with respect to X-MAC [6] and B-MAC [7] protocols.

According to B-MAC Sender (Figure 1), the data is transmitted after the entire preamble was sent. In contrast, for X-MAC, the preamble is splitted in short preambles with target information [6]. Therefore, as soon as that a short preamble is received by the receiver, it sends an "ACK", and thus, the sender stops the transmission of the remaining short preambles (Figure 1). Afterwards, the transmission of data can start. Put simply, one can have a higher probability to transmit firstly its data according to X-MAC compared to B-MAC (see Figure 1).

We have other shortcomings with respect to the transmission of along preamble. After the first sender begins its preamble's transmission, subsequent transmitters stay awake and wait until the channel is free. Therefore, a sender can send a

preamble in order to wake up a node that is already woken up by preamble sent previously.

In addition to shortening the preamble by the use of acknowledgement (ACK), X-MAC also addresses the problem of multiple transmitters sending the entire preamble even though the receiver is already awake. In X-MAC, when a transmitter is attempting to send, but detects a preamble, it waits until the channel is free. However, if during its listening, the node hears an acknowledgement (frame) that comes from the node that it wishes to communicate later, the transmitter starts a backoff (*ie.*, waits a random amount of time). After, its backoff, the transmitter sends its data to this given node without a preamble [6].

### C. Related work on mobility-based MAC solutions

*MS-MAC* is a slotted protocol proposed by Huan et al. in [13]. It is an improved version of the *S-MAC* [14] protocol in order to handle mobility. *MS-MAC* uses a simple mobility estimation algorithm to estimate the mobility in a neighborhood.

As discussed in [5], one disadvantage of running the synchronization algorithm very often leads to higher energy consumption. Therefore, *MS-MAC* leads to high energy consumption in order to maintain connections between nodes during their mobility.

*MMAC* is a slotted protocol proposed by Ali et al. in [15]. *MMAC* is an improvement of the *TRAMA* [16] protocol by adding a mobility adaptive algorithm to overcome the problems encountered by *TRAMA* under mobile scenarios. *TRAMA* is a scheduled based protocol, however under mobility, the two-hop topology information becomes inconsistent. Furthermore, *TRAMA* uses a fixed time frame, which makes the mobile node to wait longer to join the network.

*MMAC* has an adaptive mobility algorithm which addresses these problems by adjusting the frame size according to the mobility status in the network. As discussed in [5], the disadvantages of *MMAC* are the highly complex scheduling algorithm to calculate the transmitter of each slot in a frame time. Note that the control overhead is high due to the explicit transmission of scheduling packet. *MMAC* consider also a duty cycle which is also high due to the use of a random access period and huge amount of collisions following the mobility of nodes.

*MACHIAVEL*, a sampling protocol proposed by Kuntz et al. in [8], reiterates the sampling protocols principles. Therefore, the preamble is followed by a short "SYNC" message, sent by a fixed node, that enables the neighborhood to prepare the reception of the trailing data. *MACHIAVEL* makes the mobile nodes benefit from this synchronization work.

When a mobile node wishes to emit data, it first samples the medium. If it does not detect any signal, it follows the standard procedure: it sends a preamble, a SYNC and then the data. If it detects a preamble, it is allowed to take possession of the medium at the end of the current preamble and SYNC being sent by a fixed node. For that purpose, *MACHIAVEL* specifies a delay (MIFS, *MACHIAVEL* Inter-Frame Space) that fixed

nodes have to observe between the SYNC and their data. The value of the MIFS delay may vary according to the time that a node should take to sample the channel.

*MEMAC* (Mobility aware and Energy efficient Medium Access Control), proposed by Yahya et al. in [5], is a hybrid protocol. *MEMAC* uses a hybrid approach of both scheduled (TDMA) and contention based (CSMA) medium access schemes. *MEMAC* differentiates between data and control messages; long data messages are assigned scheduled TDMA slots (only those nodes, which have data to send are assigned slots), while short control messages are assigned random access slots. This technique limits message collisions and reduces the total energy consumed by the radio transceiver. Furthermore, *MEMAC* uses a dynamic frame size to enable the protocol to effectively adapt itself to changes in mobility conditions.

Mobility prediction through the use of the first order auto-aggressive moving average model is used to dynamically adjust the frame size and control the channel access in an efficient way according to the mobility conditions.

## III. MOX-MAC: A MOBILE ACCESS SCHEME FOR X-MAC

### A. Architecture and description

As specified in [8], sending a preamble, in sampling protocols, reduces channel availability and thus increases the competition between the nodes. This problem is especially highlighted when a node sends preambles while moving, it might put in overhearing situation all nodes within range of its radio. This is a shortcoming according to X-MAC.

Therefore, we also take as hypothesis that the mobile nodes do not send preamble. Moreover, in *MACHIAVEL*, a mobile node sends its data to the static receiver node (preamble receiver). Nevertheless, in X-MAC, this receiver has the possibility to communicate with other static nodes after its first communication. In so doing, its lifetime will be reduced if it communicated in addition with mobile nodes.

1) *Communication architecture in presence of mobile node:* The main assumption in *MoX-MAC* is that mobile nodes should send their data to static nodes that should route these data towards the sink. Put simply, any communication initiated by a mobile node should go towards a fixed one.

As illustrated in Figure 2, we use a scenario similar to X-MAC: a series of short preambles sent by a static node in order to make ready node's neighborhood to receive data; the node (recipient of data) that receives one of these short preambles, automatically sends an acknowledgment frame (ACK) to the issuer of the preambles to say that it is ready to receive data. *MoX-MAC* uses this ACK frame for the benefit of mobile nodes.

When a mobile node wants to send data, it samples the medium in the hope of receiving an ACK. If it detects no signal, it follows the standard procedure of X-MAC (sending short preambles and data). If it detects an ACK, it waits until the end of the originally scheduled transmission of this ACK. Afterwards, it can send its data to the static node that has

transmitted the preambles previously (receiver of the ACK). Figure 2 depicts this algorithm.

A mobile node waits a random time (backoff) before sending its data toward a static node (see Figure 2). The main reason is due to the fact that more than one mobile node can become potential transmitters. By doing this we mitigate the risk of contention. In other words, this backoff prevents collisions between competitive nodes.

Therefore, the static transmitter node remains awake after any initially scheduled transmission to eventually receive data (this period equals to the maximum of backoff period of the mobile node).

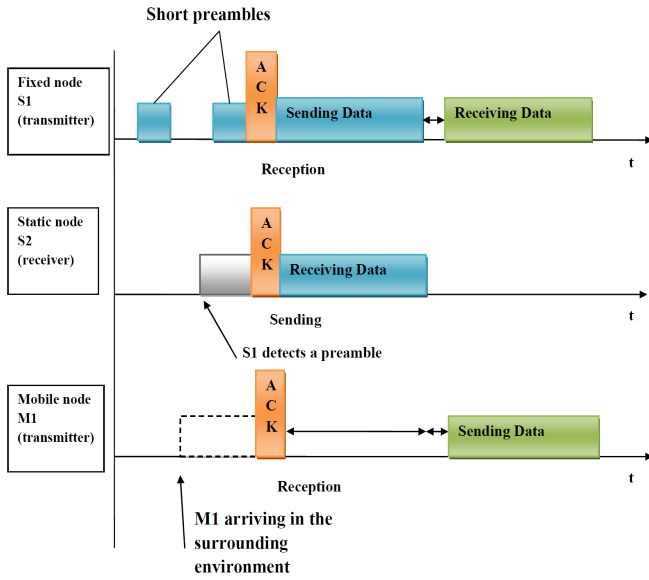


Fig. 2. Communication between mobile and static nodes following MoX-MAC protocol.

2) *Communication architecture without mobility*: In the case where no mobile node occupies the medium, the behavior of static nodes is very similarly to nodes in X-MAC, except the backoff performed by the static node's transmitter before returning to sleep as illustrated by the node *S1* in Figure 3.

### B. Performance analysis and validation

Using *COOJA*, a simulator integrated to *Contiki OS* [17] which implements the X-MAC protocol, we performed extensive simulations to check the performance of a given mobile node that uses our protocol. Simulations parameters are presented in Table I.

To compare the performance between MoX-MAC and X-MAC, it is convenient to measure some metrics for the mobile node: the average energy consumption, the average packet loss, the average medium access delay.

1) *Average energy consumption*: The Figure 4 depicts the average energy consumption of a given mobile node with respect to other static nodes in the overall network. Note that the number of static nodes varies from 10 to 350. The x-axis

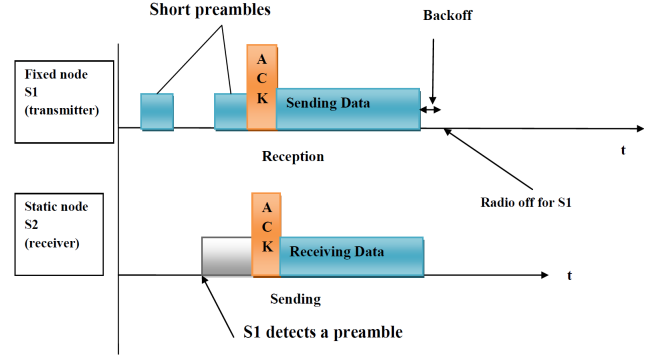


Fig. 3. Communication between static nodes without mobility.

Simulation parameters	Values
Topology	Square (150mx150m), mobile and fixed sensors are distributed randomly for each simulation
Number of sensors considered during each simulation	10, 20, 30, 50, 75, 100, 150, 200, 250, 300, 350
Mobility model	Random Way Point
Min speed of a mobile node	1.0 m/s
Max speed of a mobile node	4.0 m/s
Radio model	Chipcon CC2420 IEEE 802.15.4
Data size	16 Bytes
Duration	100 seconds

TABLE I  
SIMULATION PARAMETERS

show the number of given static nodes during each simulation and the y-axis the average energy consumption, in millijoule, of a given mobile node. It should be noted that during our extensive simulations, one node is mobile with respect to the remaining nodes.

As illustrated in Figure 4, up to 50 static nodes MoX-MAC and X-MAC have the same trend. Nevertheless, X-MAC outperforms a little bit MoX-MAX. The main reason is due that we have a low density nodes. Following MoX-MAC, the mobile node needs to hear ACK messages before transmitting its message. Since there is a limited number of nodes in the network we should wait a long time in order to hear an ACK. In so doing, it consumes its energy.

In contrast, when the mobile node uses X-MAC, since there is few nodes in the network, we have less probability that nodes experience collisions. Hence, the mobile node sends its data rapidly, and thus still awakes during less time.

Afterwards, with a number of static nodes upper than 50 the energy consumption of the mobile node when it uses MoX-MAC decreases compared to X-MAC (Figure 4). With X-MAC protocol, the number of collisions increases when the number of nodes augment. Therefore, nodes use more frequently this battery. Following MoX-MAC, when the number of static nodes increase (high density), the probability that the

mobile node hears an ACK message is high. Therefore, it will be not necessary that the mobile node sends a preamble before the transmission of its data (as illustrated in Figure 2). Since collisions and idle listening are the main sources of energy consumption according to sampling protocols, it means that our approach gives better results compared to X-MAC. Indeed, following MoX-MAC, the mobile node hears more frequently ACK messages and sends faster its data.

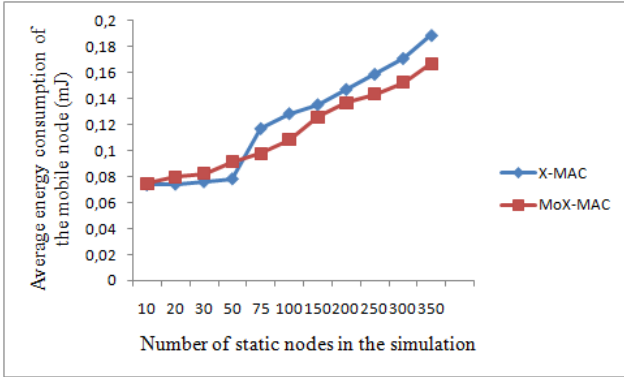


Fig. 4. Average energy consumption of a given mobile node.

2) *Average packet loss*: The Figure 5 depicts the average packet loss of a given mobile node with respect to other static nodes in the overall network. Note that the number of static nodes varies from 10 to 350. The x-axis show the number of given static nodes during each simulation and the y-axis the average packet loss of a given mobile node.

MoX-MAC and X-MAC present the same trend when the number of static nodes is less than 50 nodes. In this case, the mobile node has the same percentage of packet loss. The percentage of packet loss is roughly equals to 0.05. In contrast, when the number of nodes is upper than 50 nodes, the percentage of packet loss for the mobile nodes is less when it uses the MoX-MAC protocol (Figure 5).

Furthermore, we note a gap between MoX-MAC and X-MAC from 50 to 350 nodes. In such interval, MoX-MAC outperforms the X-MAC protocol. For instance, for 350 static nodes the percentage of packet loss experienced by a given mobile node is roughly 0.25 and 0.20 respectively for X-MAC and MoX-MAC. It is worth noticing that when the node is moving the probability to lose data is high. It's the reason why the gap between both curves (Figure 5) is reduced. Nevertheless, in high density, Figure 5 exhibits clearly that MoX-MAC outperforms X-MAC.

3) *Average medium access delay*: The Figure 6 illustrates the average medium access delay of a given mobile node with respect to other static nodes in the overall network. It is worth noticing that the number of static nodes varies from 10 to 350. The x-axis show the number of given static nodes during each simulation and the y-axis the average medium access, in millisecond, of a given mobile node.

A high medium access delay can saturate the queue of packets of a mobile node. As illustrated in Figure 6, below 50

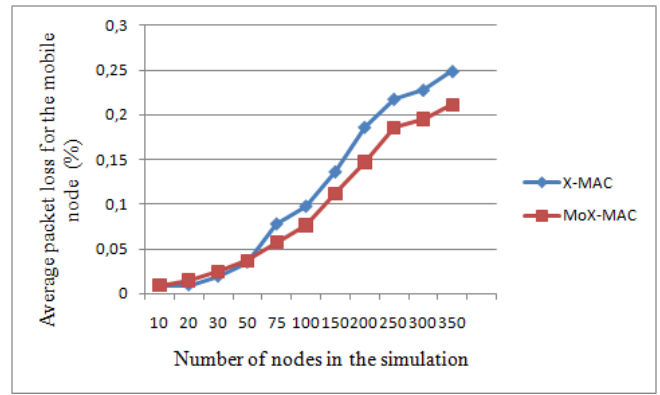


Fig. 5. Average packet loss of a given mobile node.

nodes, X-MAC protocol has a higher efficiency compared to MoX-MAC but beyond 50 nodes our approach outperforms the X-MAC. The reason why X-MAC outperforms MoX-MAC in low density is due to the fact that a mobile node using X-MAC protocol, has the advantage to automatically send a preamble after a listening of the channel. Indeed, in low-density there is less communication in the network and thus the channel is less busy.

In contrast to MoX-MAC, before sending its data the mobile node should hear an ACK. Therefore, since the number of nodes is reduced, the probability to hear an ACK is reduced, and thus, the node should listen the channel more time. Nevertheless, when the number of nodes is upper than 50 nodes, *ie.* high density, the time that the mobile node should wait according to X-MAC is very important. Indeed, we have more competitive transmissions.

Following MoX-MAC, the mobile node has a high probability to hear an ACK, and thus send its transmission to the potential receiver when it finishes its early transmission. The gap noticed in Figure 6 is more important compared to gap that Figures 4 and 5 illustrated. It should be noted that when the access delay is low it means that when a given wants to send its data it spends less time in the network, and thus consumes less power.

In summary, according to Figures 4, 5, and 6 we argue that MoX-MAC is able to reduce considerably the energy consumption of a given mobile node as well the average packet loss and the average medium access.

#### IV. CONCLUSION

Wireless Sensor Networks exhibit undoubtedly a major breakthrough for the future of human being in several application areas: medical, military, agricultural, domestic, etc. In this paper we were interested in handling mobility at the MAC layer of WSNs.

We proposed a mobile access scheme to overcome the limitations of X-MAC protocol. Our proposed solution, called MoX-MAC, allows specific channel access to mobile nodes while maximizing energy efficiency. The simulation results

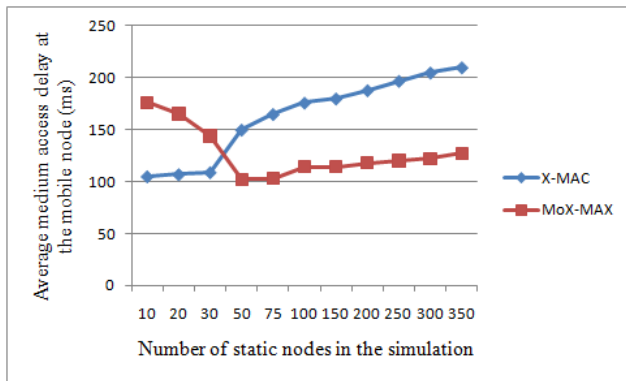


Fig. 6. Average medium access delay for a given mobile node.

were satisfying with respect to the mobility of nodes. According to our MoX-MAC protocol, we are able to improve the energy consumption, to reduce the average packet loss and to mitigate the medium access delay of a given mobile node. The evaluation of MoX-MAC has shown its benefits, especially in dense networks where packet loss rate is significantly reduced for the mobile node.

As perspectives, we investigate to evaluate MoX-MAC on other aspects. Firstly, we plan to study the protocol behavior when the ratio of mobile nodes increases in the network. Moreover, the impact of mobile node's speed also seems to be an important consideration. Secondly, we plan to compare MoX-MAC with MACHIAVEL protocol.

## REFERENCES

- [1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Comput. Netw.*, vol. 52, pp. 2292–2330, August 2008.
- [2] F. Lewis, "Wireless sensor networks," in *Smart Environments: Technologies, Protocols, Applications*, J. Wiley, Ed.
- [3] G. Pottie, "Wireless sensor networks," in *Information Theory Workshop*, June 1998.
- [4] M. Ba, I. Niang, B. Gueye, and T. Noel, "Deterministic key management scheme for securing cluster-based sensors networks," in *IEEE International Workshop on Wireless Network Algorithm and Theory*, dec 2010.
- [5] B. Yahya and J. Ben-Othman, "An adaptive mobility aware and energy efficient mac protocol for wireless sensor networks," in *Computers and Communications, 2009. ISCC 2009. IEEE Symposium on*, 5-8 2009, pp. 15–21.
- [6] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-mac: A short preamble mac protocol for duty-cycled wireless sensor networks," in *SenSys*, 2006, pp. 307–320.
- [7] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 95–107.
- [8] R. Kuntz and T. Noel, "Machiavel: Accessing the medium in mobile and dense wsn," in *Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on*, 13-16 2009, pp. 1088–1092.
- [9] I. Demirkol, C. Ersoy, and F. Alagöz, "MAC Protocols for Wireless Sensor Networks: a Survey," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 115–121, Apr. 2006.
- [10] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 3, pp. 493–506, 2004.
- [11] N. Riga, I. Matta, and A. Bestavros, "A geometric approach to slot alignment in wireless sensor networks," in *GLOBECOM, 2007*, pp. 817–822.
- [12] I. Rhee, A. Warriier, M. Aia, J. Min, and M. L. Sichitiu, "Z-mac: a hybrid mac for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 511–524, 2008.
- [13] H. Pham and S. Jha, "Addressing mobility in wireless sensor media access protocol," in *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004. Proceedings of the 2004*, 14-17 2004, pp. 113–118.
- [14] J. Zheng and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective*, W.-I. Press, Ed. Wiley-IEEE, October 2009.
- [15] M. Ali, T. Suleman, and Z. Uzmi, "Mmac: a mobility-adaptive, collision-free mac protocol for wireless sensor networks," in *Performance, Computing, and Communications Conference, 2005. IPCCC 2005. 24th IEEE International*, 7-9 2005, pp. 401–407.
- [16] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-efficient collision-free medium access control for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, ser. SenSys '03, 2003, pp. 181–192.
- [17] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki - a lightweight and flexible operating system for tiny networked sensors," in *LCN '04: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 455–462.