Energy Evaluation of Preamble Sampling MAC Protocols for Wireless Sensor Networks

Giorgio Corbellini*†, Cedric Abgrall*, Emilio Calvanese Strinati*, and Andrzej Duda†
*CEA-LETI, MINATEC, Grenoble, France
†Grenoble Institute of Technology, CNRS Grenoble Informatics Laboratory UMR 5217, France
Email: [Giorgio.Corbellini, Cedric.Abgrall, Emilio.Calvanese-Strinati]@cea.fr, Andrzej.Duda@imag.fr

***Abstract*—The paper presents a simple probabilistic analysis of the energy consumption in preamble sampling MAC protocols. We validate the analytic results with simulations. We compare the classical MAC protocols (B-MAC and X-MAC) with LA-MAC, a method proposed in a companion paper. Our analysis highlights the energy savings achievable with LA-MAC with respect to B-MAC and X-MAC. It also shows that LA-MAC provides the best performance in the considered case of high density networks under traffic congestion.**

I. Introduction

Wireless Sensor Networks (WSN) have recently evolved to support diverse applications in various and ubiquitous scenarios, especially in the context of Machine-to-Machine (M2M) networks [1]. Energy consumption is still the main design goal along with providing sufficient performance support for target applications. Medium Access Control (MAC) methods play the key role in saving energy [2] because of the part taken by the radio in the overall energy budget. Thus, the main goal in designing an access method consists of reducing the effects of both *idle listening* during which a device consumes energy while waiting for an eventual transmission and *overhearing* when it receives a frame sent to another device [2].

To save energy, devices aim at achieving low duty cycles: they alternate long sleeping periods (radio switched off) and short active ones (radio switched on). As a result, the challenge of MAC design is to synchronize the instants of the receiver wake-up with possible transmissions of some devices so that the network achieves a very low duty cycle. The existing MAC methods basically use two approaches. The first one synchronizes devices on a common sleep/wake-up schedule by exchanging synchronization messages (SMAC [3], TMAC [4]) or defines a synchronized network wide TDMA structure (LMAC [5], D-MAC [6], TRAMA [7]). With the second approach, each device transmits before each data frame a *preamble* long enough to ensure that intended receivers wake up to catch its frame (Aloha with Preamble Sampling [8], Cycled Receiver [9], LPL (Low Power Listening) in B-MAC [10], B-MAC+ [11], CSMA-MPS [12] aka X-MAC [13], BOX-MAC [14], and DA-MAC [15]). Both approaches converge to the same scheme, called *synchronous preamble sampling*, that uses very short preambles and requires tight synchronization between devices (WiseMAC [16], Scheduled Channel Polling (SCP) [17]).

Thanks to its lack of explicit synchronization, the second approach based on *preamble sampling* appears to be more easily applicable and more scalable than the first synchronous approach. Even if methods based on *preamble sampling* are collision prone, they have attracted great research interest, so that during last years many protocols have been published. In a companion paper, we have proposed LA-MAC, a Low-Latency Asynchronous MAC protocol [18] based on preamble sampling and designed for efficient adaptation of device behavior to varying network conditions.

In this paper, we analytically and numerically compare B-MAC [10], X-MAC [13], and LA-MAC in terms of energy consumption. The novelty of our analysis lies in how we relate the energy consumption to traffic load. In prior energy analyses, authors based the energy consumption on the average Traffic Generation Rate (TGR) of devices [17] as well as on the probability of receiving a packet in a given interval [13]. In contrast to these approaches, which focus on the consumption of a "transmitter-receiver" couple, we rather consider the global energy cost of a group of neighbor contending devices. Our analysis includes the cost of all radio operations involved in the transmission of data messages, namely the cost of transmitting, receiving, idle listening, and overhearing.

The motivation for our approach comes from the fact that in large, dense, and multi-hop networks, traffic distribution is not uniformly spread over the network. Thus, the energy consumption depends on traffic pattern, *e.g. convergecast*, *broadcast*, or *multicast*, because instantaneous traffic load may differ over the network. In our approach, we estimate the energy consumption that depends on the instantaneous traffic load in a given localized area. As a result, our analysis estimates the energy consumption independently of the traffic pattern.

II. Background

We propose to evaluate the energy consumption of a group of sensor nodes under three different preamble sampling MAC protocols: B-MAC, X-MAC, and LA-MAC. In large, dense, and multi-hop networks, the instantaneous traffic distribution over the network is not uniform. For example, in the case of networks with the *convergecast* traffic pattern (all messages go to one sink), the traffic load is higher at nodes that are closer to the sink in terms of number of hops. Due to this *funneling effect* [19], devices close to the sink exhaust their energy much faster than the others.

The evaluation of the energy consumption in this case is difficult and the energy analyses published in the literature often base the energy consumption of a given protocol on the traffic generation rate of the network [17]. In our opinion, this approach does not fully reflect the complexity of the problem, so we propose to analyze the energy consumption with respect to the number of messages that are buffered in a given geographical area. This approach can represent different congestion situations by varying the instantaneous size of the buffer.

In our analysis, we consider a "star" network composed of a single receiving device (*sink*) and a group of N devices that may have data to send. All devices are within 1-hop radio coverage of each other. We assume that all transmitting devices share a global message buffer for which B sets the number of queued messages, B is then related to network congestion. Among all N devices, N_s of them have at least one packet to send and are called *active* devices. Remaining devices have empty buffers and do not participate in the contention, nevertheless, they are prone to the *overhearing effect*. Thus, there are $N_o = N - N_s$ *over-hearers*. According to the global buffer state B, there are several combinations of how to distribute B packets among N sending devices: depending on the number of packets inside the local buffers of active devices, N_s and N_o may vary for each combination. For instance, there can be B active devices with each one packet to send or less than B active devices with some of them having more than one buffered packet.

In the remainder, we explicitly separate the energy cost due to transmission E_t, reception E_r, polling (listening for any radio activity in the channel) E_l, and sleeping E_s. E_o is the overall energy consumption of all over-hearers. The overall energy consumption E is the sum of all these energies. The power consumption of respective radio states is P_t, P_r, P_l, and P_s for transmission, reception, channel polling, and sleeping. The power depends on a specific radio device. We distinguish the polling state from the reception state. When a node is performing channel polling, it listens to any channel for activity—to be detected, a radio transmission must start after the beginning of channel polling. Once a radio activity is detected, the device immediately switches its radio state from polling to receiving. Otherwise, the device that is polling the channel cannot change its radio state. The duration of a message over the air is t_d. The time between two wake-up instants is $t_f = t_l + t_s$, where t_l and t_s are respectively the channel polling duration and the sleep period. These values are related to the duty cycle.

III. Preamble Sampling MAC Protocols

In this section, we provide the details of the analyzed preamble sampling protocols. Figure 1 presents the operation of all protocols.

In B-MAC [10], all nodes periodically repeat the same cycle during their lifetime: wake up, listen to the channel, and then go back to sleep. When a node wants to transmit a data frame, it first transmits a preamble long enough to cover the entire sleep period of a potential receiver. Then, it transmits the data frame. When the receiver wakes up and detects the preamble, it remains in receiving mode until the complete reception of the data frame. Even if the lack of synchronization results in low overhead, the method presents several drawbacks due to the length of the preamble: high energy consumption of transmitters, high latency, and limited throughput. We denote by t_p^B the duration of the B-MAC preamble.

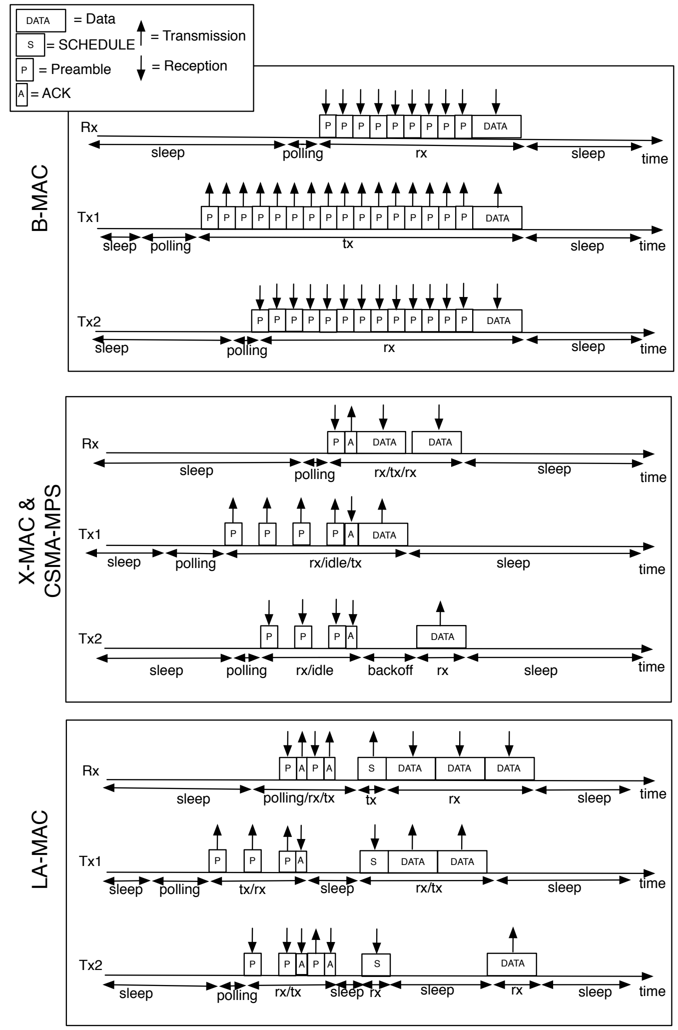

Figure 1. Comparison of analyzed MAC methods.

In CSMA-MPS [12] and X-MAC [13], nodes periodically alternate sleep and polling periods. After the end of a polling period, each active node transmits a series of short preambles spaced with gaps. During a gap, the transmitter switches to the idle mode and expects to receive an ACK from the receiver. When a receiver wakes up and receives a preamble, it sends an ACK back to the transmitter to stop the series of preambles. After the reception of the ACK, the transmitter sends a data. After data reception, the receiver remains awake for a possible transmission of a single additional data frame. If another active node receives a preamble destined to the same receiver it wishes to send to, it stops transmitting to listen to the channel for an incoming ACK. When it overhears the ACK, it sets a random back-off timer at which it will send its data frame. The transmission of a data frame after the back-off is not preceded by any preamble. The duration of each short preamble is t_p^X and the ACK duration is t_a^X.

LA-MAC [**?**] is a scalable protocol that aims at achieving low latency and limited energy consumption by building on three main ideas: efficient forwarding based on proper scheduling of children nodes that want to transmit, transmissions of frame bursts, and traffic differentiation. It assumes that the network is organized according to some complex structure (tree, DAG, partial mesh) and takes advantage of the network structure to support efficient multi-hop forwarding—a parent of some nodes becomes a coordinator that schedules transmissions in a localized region.

The method periodically adapts local organization of channel access depending on network dynamics such as the number of active users and the instantaneous traffic load. In LA-MAC, nodes periodically alternate long sleep periods and short polling phases. During polling phases each receiver can collect several requests for transmissions included inside short preambles. After the end of its polling period, the node that has collected some preambles processes the requests, compares the priority of requests with the locally backlogged messages and broadcasts a SCHEDULE message. The goal of the SCHEDULE message is to temporarily organize the transmission of neighbor nodes to avoid collisions. If the node that ends its polling has not detected any channel activity and has some backlogged data to send, it starts sending a sequence of short unicast preambles containing the information about the burst to send. As in B-MAC and X-MAC, the strobed sequence is long enough to wake-up the receiver. When a receiver wakes up and detects a preamble, it clears it with an ACK frame containing the instant of a *rendezvous* at which it will broadcast the SCHEDULE frame. If a second active node overhears a preamble destined to the same destination it wants to send to, it waits for an incoming ACK. After ACK reception, a sender goes to sleep and wakes up at the instant of the rendezvous. In Figure 1, we see that after the transmission of an ACK to Tx1, Rx device is again ready for receiving preambles from other devices. So, Tx2 transmits a preamble and receives an ACK during the same *rendezvous*. Preamble clearing continues until the end of the channel polling interval of the receiver.

IV. Energy Analysis

In this section, we provide an analytic evaluation of the energy consumption for B-MAC, X-MAC, and LA-MAC based on the instantaneous *global message buffer* B, the number of buffered messages that must be sent by a group of contending devices. Due to the limited space, we only explicit the analytic expression for the case of $B = 1$, nevertheless we provide later the analytic and numerical results for a wide range of values of B (cf. Figures 2–5). The complete analysis for all buffer sizes is available in [20]. If $B = 0$, all protocols behave in the same way: nodes periodically wake-up, poll the channel, then go back to sleep because of the absence of channel activity. Thus, the consumption only depends on the time spent in the polling and sleeping states. If there is one message to send ($B = 1$), there is only one *active* device that has a message to send (N_s = 1). Other devices are *over-hearers*, that is, they waste energy by overhearing useless transmissions. The number of overhears is $N_o = N - 1$.

B-MAC. When the sender wakes up, it polls the channel and then starts sending one long preamble that anticipates data transmission. Even if data is unicast, the destination field is not included in the preamble. Therefore, all neighbors need to hear both the preamble and the header of the following data to know the identity of the intended receiver. Provided that devices are not synchronized, we assume that each device will hear the half of the preamble on the average. The cost of a transmission is the cost of an entire preamble plus the cost of transmitting data:

$$E_t^B(1) = (t_p^B + t_d) \cdot P_t \tag{1}$$

The energy consumption due to the reception of the packet depends on the probability of *quasi-synchronization* p of the "transmitter-receiver" couple. Due to the lack of explicit synchronization, it may happen that at the instant when the receiver wakes up, the sender is already polling the channel. The probability of this event is $p = t_l/t_f$, so if the receiver wakes up during this period, it will perform a half of the polling period and then it will listen to the entire preamble. Otherwise, if the receiver wakes up after the end of the polling of the sender, it will listen during a half of the preamble (with probability $1 - p$). In the remainder of this paper, we say that with probability p the transmitter and the receiver are *quasi-synchronized*. In this paper, we assume that sensors are equipped with packetized radios, thus large preambles are obtained by transmitting a sequence of short preambles one after the other. For this reason, if device A_2 wakes up and polls the channel while device A_1 is sending a *long* preamble, the radio of device A_2 will remain in the polling state for a very short time until the beginning of the next short preamble is receive. Afterwards radio of A_2 will switch to the receiving state (consuming more energy) (cf. Figure 1). The cost of reception is:

$$E_r^B(1) = (p \cdot t_p^B + (1-p) \cdot \frac{t_p^B}{2} + t_d) \cdot P_r \tag{2}$$

The cost of polling activity is the cost of sender polling plus the cost of receiver polling weighted by the probability of *quasi-synchronization*:

$$E_l^B(1) = (1 + \frac{p}{2}) \cdot t_l \cdot P_l \tag{3}$$

Following the same reasoning, the cost of sleeping is:

$$E_s^B(1) = (2 \cdot t_f - (\frac{t_p^B}{2} \cdot (p+3) + 2 \cdot t_d + t_l \cdot (1 + \frac{p}{2}))) \cdot P_s \tag{4}$$

Under B-MAC, there is no difference in terms of energy consumption between overhearing and receiving a message. Therefore, the cost of overhearing is:

$$E_o^B(1) = N_o \cdot (E_r^B(1) + p \cdot \frac{t_l}{2} \cdot P_l + \quad (5)$$

$$(t_f - (p \cdot (\frac{t_l}{2} + t_p^B) + (1-p) \cdot \frac{t_p^B}{2} + t_d)) \cdot P_s)$$

X-MAC. When the sender wakes up, it polls the channel and starts sending a sequence of unicast preambles separated by a time interval for *early* ACK reception. When the intended receiver wakes up and polls the channel, it receives the preamble and clears it. Afterwards the sender can transmit its message. After data reception, the receiver remains in the polling state for an extra back-off time t_b. All devices that have no message to send, overhear channel activity and go to sleep as soon as they receive any unicast message (preamble, ACK, or data). Due to the lack of explicit synchronization, the expected number of preambles that are needed to *wake-up* the receiver is γ^X. The average number of preambles depends on the duration of the polling period, the preamble and ACK messages as well as the time between two wake-up instants [13]. γ^X is the inverse of the *collision* probability of one preamble during the polling period of the receiver. Each preamble sent is a trial of a geometric distribution, thus we say that before a collision between preamble and the polling period, there are $(\gamma^X - 1)$ preambles whose energy is wasted.

$$\gamma^X = (\frac{t_l - t_a^X - t_p^X}{t_f})^{-1} \quad (6)$$

In the case of *quasi-synchronization* (same probability p previously defined), the receiver will be active on the average during a half of the polling periods and afterwards it will be able to clear the first preamble of the strobe. Otherwise, the sender will waste some energy for the transmission of γ^X preambles and for waiting for an ACK. The transmission cost is thus:

$$E_t^X(1) = ((1-p) \cdot \gamma^X + p) \cdot t_p^X \cdot P_t + t_a^X \cdot P_r + t_d \cdot P_t \quad (7)$$

The cost of the receiving activity is represented by the transmission of one ACK and the reception of both data and preamble:

$$E_r^X(1) = (t_d + t_p^X) \cdot P_r + t_a^X \cdot P_t \quad (8)$$

The expected cost of polling depends on *quasi-synchronization*:

$$E_l^X(1) = ((1-p) \cdot (\frac{t_p^X + t_a^X}{2} + (\gamma^X - 1) \cdot t_a^X) + (\frac{p}{2} + 1) \cdot t_l + t_b) \cdot P_l \quad (9)$$

The sleep activity of the active couple is twice t_f minus the time during which both devices are active.

$$E_s^X(1) = (2 \cdot t_f - 2 \cdot t_d - p \cdot \frac{t_l}{2} - t_p^X - t_a^X - (1-p) \cdot \frac{t_p^X + t_a^X}{2} \quad (10)$$

$$-t_l - ((1-p) \cdot \gamma^X + p) \cdot (t_p^X + t_a^X) - t_b) \cdot P_s$$

As other devices, the over-hearers can wake-up at a random instant. However, differently from active devices, as soon as they overhear some activity they go back to sleep. Therefore, their energy consumption depends on the probability that such nodes wake up while the channel is busy or not. The probability that at the wake-up instant the channel is free depends on the polling duration, buffer states, the number of senders, *quasi-synchronization* of active couple etc. In total, we have identified 9 different combinations with specific probability to happen. Due to the limited space, we omitted the final expression for the cost of overhearing messages that is the sum of the costs of each combination weighted by the probability of the combination to happen.

LA-MAC. When sender wakes up, it polls the channel, and sends preambles as in X-MAC. However, differently from X-MAC, after *early* ACK reception, the sender goes back to sleep and waits for the SCHEDULE message to be sent. When the intended receiver receives one preamble, it clears it and completes its polling period to detect additional possible preambles to clear. Immediately after the end of the polling period, the receiver processes requests and broadcasts the SCHEDULE message. In LA-MAC, over-hearers go back to sleep as soon as they receive any unicast message (preamble, ACK, or data) as well as the SCHEDULE. The expected number of preambles necessary to wake-up the receiver follows the X-MAC modeling with a different size of preambles t_p^L and ACK t_a^L. When the sender wakes up, it polls the channel and starts sending preambles to *wake-up* the receiver. With probability p, the first preamble sent will wake up the receiver, so the sender will immediately receive an *early* ACK. Otherwise, if nodes are not synchronized, the sender will wake up its destination on the average after γ^L preambles. Similarly to X-MAC, we have:

$$E_t^L(1) = ((1-p) \cdot \gamma^L \cdot t_p^L + p \cdot t_p^L + t_d) \cdot P_t + t_a^L \cdot P_r + t_g \cdot P_r \quad (11)$$

The cost of reception depends on the duration of the preamble, ACK, data, and SCHEDULE messages.

$$E_r^L(1) = (t_p^L + t_d) \cdot P_r + (t_a^L + t_g) \cdot P_t \quad (12)$$

The cost of polling activity is similar to X-MAC, however in this case, the receiver will complete its polling period even if it clears one preamble, so its radio will remain in the polling state for the duration of a full polling period less the time for preamble reception and ACK transmission:

$$E_l^L(1) = ((t_l + (1-p) \cdot (\gamma^L - 1) \cdot t_a^L) + (t_l - t_p^L - t_a^L)) \cdot P_l \quad (13)$$

When the active nodes are not transmitting, receiving, or polling the channel, they can go to sleep.

$$E_s^L(1) = (2 \cdot t_f - (t_l + (1-p) \cdot \gamma^L \cdot t_p^L + \quad (14)$$

$$p \cdot t_p^L + t_a^L + (1-p) \cdot (\gamma^L - 1) \cdot t_a^L + t_d + t_g) - (t_l + t_d + t_g)) \cdot P_s$$

Similarly to X-MAC, as soon as over-hearers receive some messages, they go back to sleep. In LA-MAC, we have identified 11 main combinations. When the number of messages in

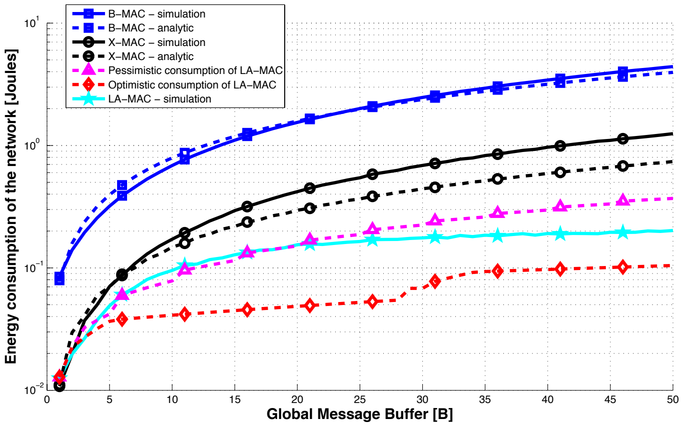

Figure 2. Energy analysis and OMNeT++ simulations versus the global message buffer.

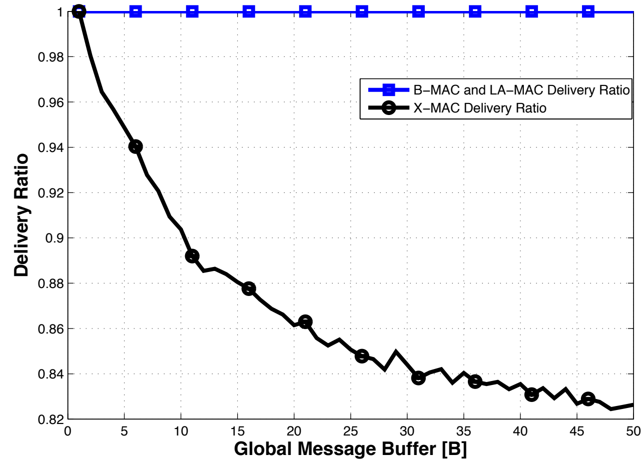

Figure 3. Delivery ratio versus the global message buffer.. In X-MAC, most collisions happen when messages are sent after the back-off time.

the global buffer is high, the evaluation of energy consumption for LA-MAC becomes difficult. Therefore, we propose two analytic expressions, the first one is *optimistic* and it assumes local transmissions without collision of preambles; the second one is *pessimistic* because it assumes that preambles may collide resulting in higher latency because all messages in the global buffer need more time to be sent [20].

V. Numerical Validation

We have implemented the analyzed MAC protocols in the OMNeT++ simulator [21] for numerical evaluation. Each numerical value is the average of 1000 runs and we show the corresponding confidence intervals at 95% confidence level. We assume that devices use the CC1100 [22] radio stack with bitrate of 20 Kbps. The values of power consumption for different radio states are specific to the CC1100 transceiver considering a 3 V battery. In the following, we assume $N = 9$ senders. The periodical wake-up period is the same for all protocols: $t_f = t_l + t_s = 250\ ms$. Also the polling duration is the same for all protocols: $t_l = 25\ ms$, thus the duty cycle with no messages to send is 10%. We provide numerical and analytic results for buffer size $B \in [1, 50]$. We compare the protocol performance with respect to several criteria:

Latency [s]: the delay between the beginning of the simulation and the instant of packet reception at the sink (we present the latency averaged over all nodes).

Energy Consumption [Joules]: the averaged energy consumed by all nods due to radio activity.

Delivery Ratio: the ratio of the number of received packet by the sink to the total number of packets sent.

In Figure 2, we show the comparison between the proposed energy consumption analysis and numerical simulations for different values of the global buffer size. We assume that at the beginning of each simulation all messages to send are already buffered. Each simulation stops when the last message in the buffer is received by the sink. Figure 2 highlights the validity of the analytic expressions for energy consumption—we can see that the curves reflect the main trends. The simulation results exceed the analytic data because the simulation reflects the detailed behavior for the protocols, which cannot be captured in simple expressions. As expected, B-MAC is the most energy consuming protocol: as the buffer size increases, the transmission of a long preamble locally saturates the network resulting in high energy consumption and latency (cf. Figure 4). In X-MAC, short preambles mitigate the effect of the increasing local traffic load, thus both latency and energy consumption are reduced with respect to B-MAC. Even if X-MAC is more energy efficient than B-MAC, Figure 3 shows that even for small buffer sizes, the delivery ratio for this protocol is lower than 100 % most likely because packets that are sent after the back-off collide at the receiver. LA-MAC is the most energy saving protocol and it also outperforms other protocols in terms of latency and the delivery ratio. Energy consumption of LA-MAC lies in between the *pessimistic* and the *optimistic* curves. We observe that when the instantaneous buffer size is lower than 2 messages, the cost of the SCHEDULE message is paid in terms of a higher latency with respect to X-MAC (cf. Figure 4); however, for larger buffer sizes the cost of the SCHEDULE transmission is compensated by a high number of delivered messages. In Figure 5, we show the percentage of the time during which devices spend in each radio state versus the global buffer size. Thanks to efficient message scheduling of LA-MAC, devices sleep most of the time independently of the buffer size and all messages are delivered.

VI. Conclusions

In the present paper, we have analyzed the energy consumption of preamble sampling MAC protocols by means of a simple probabilistic modeling. The analytic results are then validated by simulations. We compare the classical MAC protocols (B-MAC and X-MAC) with LA-MAC, a method proposed in a companion paper. Our analysis highlights the energy savings achievable with LA-MAC with respect to B-

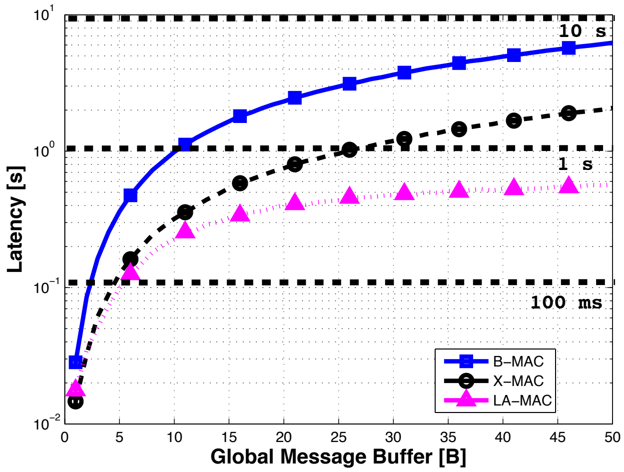

Figure 4. Average latency versus the global message buffer.

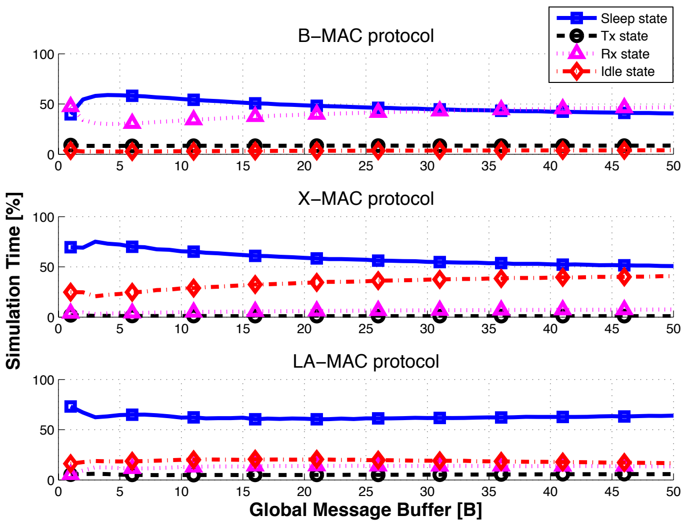

Figure 5. Percentage of the time spent in each radio state versus the global message buffer.

MAC and X-MAC. It also shows that LA-MAC provides the best performance in the considered case of high density networks under traffic congestion.

ACKNOWLEDGMENTS

A part of this work has been performed in the framework of the ICT project ICT-5-258512 EXALTED partly funded by the European Commission. The author Calvanese Strinati would like to acknowledge the contributions of their colleagues from the EXALTED project, although the expressed views are those of the authors and do not necessarily represent the project.

REFERENCES

[1] "ICT-258512 EXALTED project." [Online]. Available: http://www.ict-exalted.eu/.

[2] K. Langendoen, "Energy-Efficient Medium Access Control," *Book chapter : Medium Access Control in Wireless Networks, H. Wu and Y. Pan (editors), Nova Science Publishers*, 2008.

[3] Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *in Proc. of Annual IEEE International Conference on Computer Communications, INFOCOM*, vol. 3, pp. 1567–1576, 2002.

[4] T. van Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," *in Proc of. ACM International Conference on Embedded Networked Sensor Systems, SenSys*, pp. 171–180, 2003. [Online]. Available: http://portal.acm.org/citation.cfm?id=958512

[5] L. van Hoesel, P. Havinga, "A Lightweight Medium Access Protocol (LMAC) for Wireless Sensor Networks: Reducing Preamble Transmissions and Transceiver State Switches," *in Proc. of IEEE INSS*, Tokyo, Japan 2004.

[6] G. Lu, B. Krishnamachari, and C. Raghavendra, "An Adaptive Energy-efficient and Low-latency MAC for Data Gathering in Wireless Sensor Networks," *in Proc. of 18th IEEE IPDPS*, p. 224, 2004.

[7] V. Rajendran, J. J. Garcia-Luna-Aceves, and K. Obraczka, "Energy-Efficient, Application-Aware Medium Access for Sensor Networks," *in Proc. of IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS*, pp. 8–pp, 2005.

[8] A. El-Hoiydi, "Aloha with Preamble Sampling for Sporadic Traffic in Ad Hoc Wireless Sensor Networks," *in Proc. of the Int. Conf. on Communications*, vol. 5, pp. 3418–3423, New York, NY, USA, April 2002.

[9] E-Y. Lin, J. Rabaey, A. Wolisz, "Power-Efficient Rendez-vous Schemes for Dense Wireless Sensor Networks," *in Proc. of the Int. Conf. on Communications*, Paris, France, June 2004.

[10] J. Polastre, J. Hill and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," *in Proc of. ACM International Conference on Embedded Networked Sensor Systems, SenSys*, 2004.

[11] M. Avvenuti, P. Corsini, P. Masci, and A. Vecchio, "Increasing the Efficiency of Preamble Sampling Protocols for Wireless Sensor Networks," *in Proc. of the First Mobile Computing and Wireless Communication International Conference, MCWC*, pp. 117–122, 2006.

[12] S. Mahlknecht and M. Boeck, "CSMA-MPS: A Minimum Preamble Sampling MAC Protocol for Low Power Wireless Sensor Networks," *in Proc. of IEEE Workshop on Factory Communication Systems*, pp. 73–80, Vienna, Austria, September 2004.

[13] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks," *in Proc of. ACM International Conference on Embedded Networked Sensor Systems, SenSys*, pp. 307–320, 2006. [Online]. Available: http://portal.acm.org/citation.cfm?id=1182807.1182838

[14] R. Kuntz, A. Gallais, and T. Noel, "Auto-adaptive MAC for energy-efficient burst transmissions in wireless sensor networks," *in Proc. of IEEE Wireless Communications and Networking Conference, WCNC*, pp. 233 –238, march 2011.

[15] G. Corbellini, E. Calvanese Strinati, E. Ben Hamida, and A. Duda, "DA-MAC: Density Aware MAC for Dynamic Wireless Sensor Networks," *in Proc. of 22nd IEEE Personal Indoor Mobile Radio Communications (PIMRC'11 - LPAN)*, pp. 920–924, September 2011.

[16] C. Enz, A. El-Hoiydi, J. Decotignie, V. Peiris., "WiseNET: An Ultra-Low-Power Wireless Sensor Network Solution," *IEEE Computer Society Press*, vol. 37, no. 8, pp. 62–70, August 2004.

[17] W. Ye, F. Silva and J. Heidemann, "Ultra-Low Duty Cycle MAC with Scheduled Channel Polling," *in Proc of. ACM International Conference on Embedded Networked Sensor Systems, SenSys*, pp. 321–334, Boulder, CO, USA, November 2006.

[18] G. Corbellini, E. Calvanese Strinati, and A. Duda, "LA-MAC: Low-Latency Asynchronous MAC for Wireless Sensor Networks," *IEEE PIRMC'12*.

[19] C. Wan, S. Eisenman, A. Campbell, and J. Crowcroft, "Siphon: Overload Traffic Management Using Multi-radio Virtual Sinks in Sensor Networks," *in Proc of. ACM International Conference on Embedded Networked Sensor Systems, SenSys*, pp. 116–129, 2005.

[20] G. Corbellini, "Adaptive Medium Access Control for Heterogeneous Wireless Sensor Networks," http://tel.archives-ouvertes.fr/tel-00690964, 2012.

[21] "OMNeT++ Discrete Event Simulator," http://www.omnetpp.org.

[22] "Texas Instruments, CC1100 datasheet," http://focus.ti.com/docs/prod/folders/print/cc1100.html.