Continuous-Monitoring Using Event Driven Reporting for Cluster-Based Wireless Sensor Networks

Nizar Bouabdallah, Mario E. Rivero-Angeles, *Member, IEEE,* and Bruno Sericola

***Abstract*— Continuous-monitoring applications are an important class of wireless sensor networks (WSNs) applications. These applications require periodic refreshed data information at the sink nodes. To date, this entails the need of the sensor nodes to transmit continuously in a periodic fashion to the sink nodes, which may lead to excessive energy consumption.**

In this paper, we show that continuous-monitoring does not imply necessarily continuous reporting. Instead, we demonstrate that we can achieve continuous-monitoring using an event-driven reporting approach. Building on this, we propose two new mechanisms that enable energy conservation in continuous-monitoring WSNs. The first mechanism can augment any existing protocol, whereas the second is conceived for cluster-based WSNs. With both mechanisms, sensor nodes only transmit information whenever they sense relevant data. To evaluate the efficiency of our proposals, the basic unscheduled transmission model and three well-known cluster-based protocols are used as baseline examples. Specifically, new analytical models for conventional cluster-based systems and for our approach-enabled systems are complemented by simulations in order to present a quantified perspective of the potential benefits of the proposed reporting technique. We prove that significant energy conservation can be achieved using our reporting approach.

***Index Terms*— Wireless sensor networks, clustering, energy-efficiency, continuous-monitoring, event-driven reporting.**

I. INTRODUCTION

Wireless sensor networks (WSNs) rely on the cooperative effort of the densely deployed sensor nodes to gather data information from the supervised area [1] [2], typically either to achieve environmental monitoring or target tracking and sensing. Both applications produce light traffic compared to traditional wireless networks, but with different characteristics. While environmental monitoring often requires continuous-monitoring of the supervised area (i.e., each sensor node transmits periodically its sensed data to the sinks node), target tracking requires instead Event-Detection Driven (EDD) WSNs, where communications are triggered by the occurrence of a pre-specified type of events. In this case, once an event occurs, it is reported to the sink node by the sensors within the event area.

Manuscript received February 20, 2008; revised October 28, 2008.

Nizar Bouabdallah is with INRIA Rennes - Bretagne Atlantique (Institut National de Recherche en Informatique et en Automatique), Campus de Beaulieu, 35042 Rennes cedex, France (phone:+33299847471; fax:+33299847171; e-mail: nizar.bouabdallah@inria.fr).

M. E. Rivero-Angeles is with INRIA Rennes - Bretagne Atlantique (Institut National de Recherche en Informatique et en Automatique), Campus de Beaulieu, 35042 Rennes Cedex, France, on leave from the Telematic Section, Advanced Technologies and Interdisciplinary Engineering Professional Unit (UPIITA-IPN), Mxico City, 07340, Mxico (e-mail: Mario.Rivero@inria.fr).

Bruno Sericola is with INRIA Rennes - Bretagne Atlantique (Institut National de Recherche en Informatique et en Automatique), Campus de Beaulieu, 35042 Rennes cedex, France (e-mail: Bruno.Sericola@inria.fr).

In this work, we focus on the class of continuous monitoring applications where the final user requires the most recent values sensed by the sensor nodes. Hence, each sensor node periodically produces data information and reports to one or several sink nodes. This typically implies that sensor nodes continuously transmit their information regardless of whether they have relevant data or not [14]. By relevant data we refer to data that contains different information from the previous data information transmitted by the same sensor.

In view of this, we propose two schemes that perform intelligent reporting of the data information to the sink nodes by avoiding the transmission of extra and non relevant data information. For example, consider a continuous-monitoring temperature application, where each sensor node transmits periodically the sensed temperature to the sink node [2]. In such application, it may happen that sensors have very similar reading during long periods of time and it would not be energy-efficient for sensors to continuously send the same value to the sink node. The network lifetime would be greatly increased by programming the sensors to transmit only when they have sensed a change in the temperature compared to the last transmitted information. In doing so, the end user would have a refresh value of the temperature in the supervised area even if the sensors are not transmitting continuously in a periodic fashion. The final user would have exactly the same information gathered by the WSN as with the classical continuous-monitoring applications, but while the sensors only transmit when there is relevant data.

We refer to this technique as Continuous-Monitoring based on an Event Driven Reporting (CM-EDR) philosophy. Specifically, our proposed CM-EDR mechanism can be viewed as a particular type of EDD applications, where an event is defined as an important change in the supervised phenomenon compared to the last reading sent to the sink node. However, the main difference with typical EDD applications is that with CM-EDR, the end user would have a continuous reading of the phenomenon of interest, which is not the case with EDD applications

We emphasize the difference in terms of goals and produced traffic between CM-EDR applications and classical event-detection driven (EDD) applications. For instance, considering EDD temperature application, typically used for fire detection, the sensor nodes advertise the sink node only if the sensed temperature exceeds a pre-specified threshold. Instead, with CM-EDR-enabled continuous-monitoring temperature application, the sensor node advertises the sink node with the new information, each time the sensed temperature changes with respect to the last transmitted data.

Specifically, assume that the nodes sense periodically, each δ period of time, the local temperature. Consider the scenario where the temperature sensed by a given sensor node evolves as follows: at t_0, $Temp(t_0) = 68°F$; $Temp(t_0 + \delta) = 68°F$;

$Temp(t_0 + 2\delta) = 70°F$. Considering an EDD application with a threshold equal to $100°F$, no report is transmitted to the sink node. With a classical continuous-monitoring application, the sensor node transmits periodically its sensing data, i.e., three reports are transmitted to the sink node. Enabling the CM-EDR option, the sensor node transmits only two reports at t_0 and at $t_0 + 2\delta$.

Using the CM-EDR philosophy, we propose a first mechanism to improve the energy-efficiency in continuous-monitoring WSNs. This mechanism is protocol-independent in the sense that it can be applied to any reference protocol. This mechanism is designed therefore as an extension layer that can augment any of the existing WSN protocols (i.e., MAC and routing protocols).

In this study, the basic case where sensors communicate directly with the sink node (i.e., unscheduled architecture [2]) in addition to three cluster-based reference protocols [15] – [17] are used as baselines to which the CM-EDR improvements could be compared. In addition to the unscheduled architecture, using cluster-based architectures is motivated by the results in [15] – [17], which highlight the significant energy conservation that could be achieved when clustering sensors into groups, so that sensors communicate information only to cluster heads (CHs), which communicate the aggregated information to the sink node. In view of this, the second mechanism based on the CM-EDR philosophy, which is more appropriate to the class of cluster-based WSNs, is proposed in this work.

In both WSNs architectures, unscheduled or cluster-based, contention-based MAC protocols (i.e., random access protocols) need to be used. In particular, for the cluster-based architectures, random access protocols are used at the set-up phase of the clusters, i.e., when forming the different clusters. The energy consumed in this phase is far from being negligible and need to be considered in order to conduct a fair comparison between cluster-based and unscheduled architectures. Even though, a common assumption in previous works is the negligence of the energy consumed in the cluster formation phase. However, as it is shown in this paper, the random access protocol and the backoff policy implemented to resolve access conflicts in the cluster formation phase have an important impact on the WSN performance. In this paper, we provide an in-depth and fair comparison between cluster-based and unscheduled architectures by considering various random access protocols at the cluster formation phase.

The main contributions of our work are summarized as follows:

- First, an in-depth comparison between cluster-based and unscheduled architectures is realized in order to explore the main interest of WSN clustering. To achieve this, we consider various contention-based MAC protocols. Specifically, different variations of the carrier sense multiple access (CSMA) are considered such as non-persistent CSMA (NP-CSMA), one-persistent CSMA (1P-CSMA) and CSMA with collision avoidance (CSMA/CA). Moreover, different backoff policies are investigated such as geometric backoff (GB), uniform backoff (UB), binary exponential backoff (BEB) and negative exponential backoff (NEB).
- As a second main contribution of our work, two mechanisms based on the CM-EDR philosophy are proposed to conserve energy in continuous-monitoring WSNs. The first mechanism can augment any existing protocol, whereas the second is conceived for cluster-based WSNs. With both mechanisms, sensors only transmit their data whenever an event occurs. Moreover, with the second mechanism, if a CH has not received data from any of the cluster members (CMs) during a certain number of consecutive slots, then the CH goes to sleep for a predefined period of time which is advertised to all CMs. During this period, the sensor nodes transmit directly to the sink node if needed. Once the CH wakes up again, CMs resume their transmissions through the CH using their previously designated time slots. In doing so, we aim at achieving further energy conservation in cluster-based WSNs. This benefit, however, comes at the cost of increasing the transmission power at the sensor nodes during the CH sleeping period. This tradeoff between the energy consumption at the CHs and their associated CMs is also investigated in this work.
- Finally, new analytical models for conventional cluster-based systems and for our approach-enabled systems are complemented by simulations in order to quantitatively evaluate the benefits of the proposed reporting techniques. The mathematical models provide explicit expressions of both the energy consumption and the reporting latency.

The organization of the paper can be divided in two parts: In the first part, sections II, III and IV are dedicated to present, compare and analyze the reference protocols used to appreciate the gains introduced by the CM-EDR mechanism. In the second part, sections V and VI analyze and compare the CM-EDR enabled protocols with different system parameter values. Specifically, section II reviews the reference cluster-based protocols used as baselines to which the CM-EDR improvements are compared. Following this, section III specifies the system model and compares the basic LEACH cluster-based architecture [15] to the unscheduled architecture using different random access protocols in order to investigate the main interest of WSN clustering. In section IV, a theoretical framework is developed to evaluate the energy consumption and the reporting latency with the basic LEACH protocol. Then, section V describes our proposed CM-EDR mechanisms and extends the analytical model to evaluate the WSN performance when using our CM-EDR philosophy. Simulations have been performed to validate the analytic results; using these results, a study of the characteristics of the CM-EDR system is presented in section VI. The performance of our proposal is evaluated, using the basic unscheduled architecture and three well-known clustering protocols as baseline examples. The article concludes with a summary of our conclusions and contributions.

II. Reference Protocols

As stated before, in this work we focus mainly in cluster-based reference protocols for the introduction of the CM-EDR mechanism. The reason for this is that, as show in section III, clustering sensor nodes provides several advantages compared to the unscheduled case. It allows reducing the energy consumption due to collisions, idle listening and overhearing by coordinating sensor nodes belonging to each cluster with a common schedule. The CH assigns resources by clarifying which sensor nodes should utilize the channel at any time ensuring thus a collision-free access to the shared data channel. Specifically, the following WSNs will be considered in the analysis as baselines to which the CM-EDR improvements are compared.

- Unscheduled MAC protocol-based WSNs [2]: In this case, the sensor nodes transmit directly their sensing data to the sink node without any coordination between them.
- Cluster-based WSNs (i.e., scheduled MAC protocol-based WSNs): The WSN is divided into clusters. Each sensor communicates information only to the CH, which communicates the aggregated information to the sink node. In our study, we considered three well-known clustering protocols: Low Energy Adaptive Clustering Hierarchy (LEACH) [15], Hybrid Energy-Efficient Distributed clustering (HEED) [16] and the clustering protocol of [17] that allows multi-hop communications inside the clusters. This latter, called henceforth as MH clustering, can be seen as an extension of the LEACH protocol to the multi-hop case.

It is worth noting that we use both scheduled and unscheduled MAC protocol-based WSNs to show the capability of our proposed CM-EDR mechanism to augment any WSN design.

Four main sources of energy wastage can be identified in WSNs: collisions, overhearing (when a node receives an unintended packet), idle listening (lost energy while listening to the medium to receive possible traffic that is not sent) and overhead (due to exchange of signaling messages required for the protocol execution) [2]. Compared to the scheduled protocols, the unscheduled MAC protocols experience, in general, a higher rate of collisions, overhearing and idle listening. A scheduled MAC protocol, such as in the cluster-based architectures, addresses all of these issues inherently since it coordinates transmission among sensor nodes. Once the clusters are formed, each node will be affected an exclusive time slot, preventing thus collisions. Moreover, since each node knows when to transmit, it does not need to be awake during the complete TDMA frame but only at its specific time slot. As such, there is neither overhearing nor idle listening. But, again, these benefits compared to the basic unscheduled model come at the cost of coordination message overhead during the cluster formation phase. In this work, we investigate this tradeoff considering different random access protocols. As opposed to the previous works, we do not neglect the energy consumption at the cluster formation phase. In the following, we review the clustering protocols used in our analysis.

A. LEACH

The LEACH protocol [15] groups sensors into clusters in order to conserve energy. To balance the energy consumption inside the network, the CH role is rotated among all sensor nodes. CHs are selected in a fully distributed manner, without needing the exchange of signaling messages, which are required, however, for the CH announcement. The local decision to become a CH takes into account when the node served as a CH for the last time. As such, a sensor node that has not been a CH for a long period is more likely to become a CH in the next round.

Each sensor node selected as a CH, transmits an accepting message to the remaining sensor nodes. Cluster members (CMs) that receive multiple CH announcements select the CH that requires the lowest energy for communication by sending a cluster join message. Once a CH received all the CM announcements, it computes its schedule and assigns time slots to the different CMs. Hence, a TDMA frame shared among CMs is formed. Each CM can enter the sleep mode during the TDMA frame and wakes up only at its associated slots. Instead, the CH never enters the sleep mode and at the end of each TDMA frame it transmits the aggregate data to the sink node.

The LEACH operation is composed therefore of two phases: set-up and steady state phases. While the set-up phase refers to cluster formation, the steady phase corresponds to the TDMA operation. The duration of the steady phase is fixed by the network administrator. It is generally preferred that the steady phase lasts much longer than the set-up phase in order to limit the energy consumption due to coordination message overhead. However, rotation of the CH role among the sensor nodes is needed to balance the energy consumption inside the WSN.

B. HEED

Likewise LEACH, the HEED protocol [16] operates in two phases: the set-up phase where clusters are formed and the steady phase where the sensor nodes transmit their data using the TDMA frames. HEED differs from LEACH in the way CHs are selected.

The choice of the CHs with HEED is done in an iterative way. The aim is to achieve a better distribution of the CHs in the WSN at the cost of more complexity and increased overhead compared to LEACH, which does not guarantee a good distribution of the CHs inside the WSN. To elect CHs, HEED considers a new metric that reflects the residual energy at the sensor nodes. An elected CH advertises only its neighbors as opposed to the LEACH protocol where all the WSN nodes are advertised. In doing so, HEED ensures a better distribution of the CHs inside the WSN. CMs that receive multiple CH announcements select the CH that requires the lowest energy for communication.

C. MH Clustering

The two previous clustering protocols consider single hop architecture inside each cluster, i.e., all CMs communicate directly to the CH, which in turns transmits directly to the sink node. [17] can be seen from this perspective as an extension of the LEACH protocol to the multi-hop case.

Likewise LEACH, each CH advertises itself to the neighboring sensor nodes, which relay the advertisement in a multi-hop fashion. The advertisement is forwarded to sensors that are at most h hops away the CH. CMs that receive multiple CH announcements select the closest CH in terms of hop count. On the other hand, a sensor node that is neither a CH nor has received any CH announcement becomes a forced CH.

Operating the WSN in a multi-hop fashion enables further energy conservation in communications compared to single hop transmissions, mainly in large WSNs. This gain comes at the cost of additional complexity, for example the one-hop CMs need to perform data gathering from the two-hop CMs, so on and so forth. In addition, the overhead in the set-up phase may increase considerably since CH messages have to be forwarded through multiple hops. This MH clustering example allows us to investigate the impact the CM-EDR mechanisms when multi-hop communication is enabled inside the cluster.

III. Comparison between Cluster-Based and Unscheduled WSNs

In this section, we focus on the analysis of the LEACH protocol as it represents the basic clustering protocol in WSNs.

Results regarding the remaining reference protocols are provided in subsequent sections. Specifically, we explore the main interest of WSN clustering by comparing the LEACH cluster-based model to the basic unscheduled model, where communications are performed directly between the sensor nodes and the sink node.

As a distinguishing future from previous works, we consider in our study the energy consumption due to overhead in the cluster formation phase. We show that the energy consumed in this phase is far from being negligible. Recall that the main philosophy behind clustering is to reduce the energy consumption compared to the unscheduled systems by reducing collisions, idle listening and overhearing at the cost of coordination message overhead during the cluster formation phase.

A. Network Model

In our analysis, we consider different variations of CSMA protocol to arbitrate the access to the medium among the sensor nodes at the cluster formation phase. Specifically, the NP-CSMA, 1P-CSMA and CSMA/CA variations are considered along with different backoff policies are investigated (i.e., GB, UB, BEB and NEB).

According to the CSMA technique, a sensor node listens to the medium before transmission. If the medium is sensed idle, the node starts transmission. Otherwise, in NP-CSMA, the node draws a random waiting time (backoff period) before attempting to transmit again. During this time, the sensor does not care about the state of the medium. In 1P-CSMA, after detecting activity on the medium, the node continues to sense the channel until the end of the ongoing transmission and then immediately transmits. Since in a wireless environment, nodes can not hear collisions, another variant of CSMA called CSMA/CA is used, such as the one used in the Distributed Coordination Function (DCF) of the IEEE 802.11 protocol [18]. Accordingly, the node first senses the medium and if it is idle it does not immediately transmits but rather waits for a certain period of time called Distributed Inter Frame Space (DIFS). If the channel remains idle, the node transmits, otherwise, it continues listening to the channel until it becomes idle for a DIFS period and then enters to the backoff procedure to avoid collisions.

Whenever a collision occurs, sensor nodes must retransmit their packet according to the different backoff policies. For instance, considering the CSMA/CA case, the sending node attempts to send its frame again when the channel is free for a DIFS period augmented by the new backoff value, which is sampled according to the backoff policy. Let W_i (expressed in terms of time slots) be a random variable representing the backoff delay at a node experiencing i consecutive collisions. W_i is calculated as follows according to the different backoff policies:

- UB: W_i is uniformly chosen from the range $[1, w]$.
- BEB: W_i is uniformly chosen from the range $[1, 2^{i-1}w]$, where w is the initial backoff window size. This means that the range of the backoff delay is incremented in a binary exponential manner according to the number of collisions suffered by the packet. Following to each unsuccessful transmission, the backoff window size is doubled until a maximum backoff window size value equal to $2^m w$ is reached, where m is the number of backoff stages.
- GB: W_i is geometrically distributed with a probability q.
- NEB: W_i follows a negative exponential distribution with mean $1/R$.

Based on these random access protocols, a comparison between the LEACH cluster-based WSN and the basic unscheduled WSN is performed using the following assumptions and system parameters:

- The total number of sensor nodes in the system is $N = 100$.
- Sensor nodes are uniformly distributed in an area between $(0, 0)$ and $(100, 100)$ meters (i.e., square 100×100 area).
- The sink node is situated outside of the supervised area at the coordinate $(50, 175)$ as in [15].
- All sensor nodes have the same amount of initial energy (2 J).
- Each sensor node senses its area periodically, each $T_{sensing} = 1s$, and transmits the produced data information to the sink node.
- All nodes can transmit with enough power to reach directly the sink node. Additionally, nodes can use power control to vary the amount of transmit power.
- The energy consumed to transmit a packet depends on both the length of the packet l and the distance between the transmitter and receiver nodes d. We use the same model as in [15] where:

$$E_{tx}(l, d) = \begin{cases} l \times E_{elec} + l \times \epsilon_{fs} \times d^2, & \text{if } d < d_0 \\ l \times E_{elec} + l \times \epsilon_{mp} \times d^4, & \text{if } d \geq d_0 \end{cases} \quad (1)$$

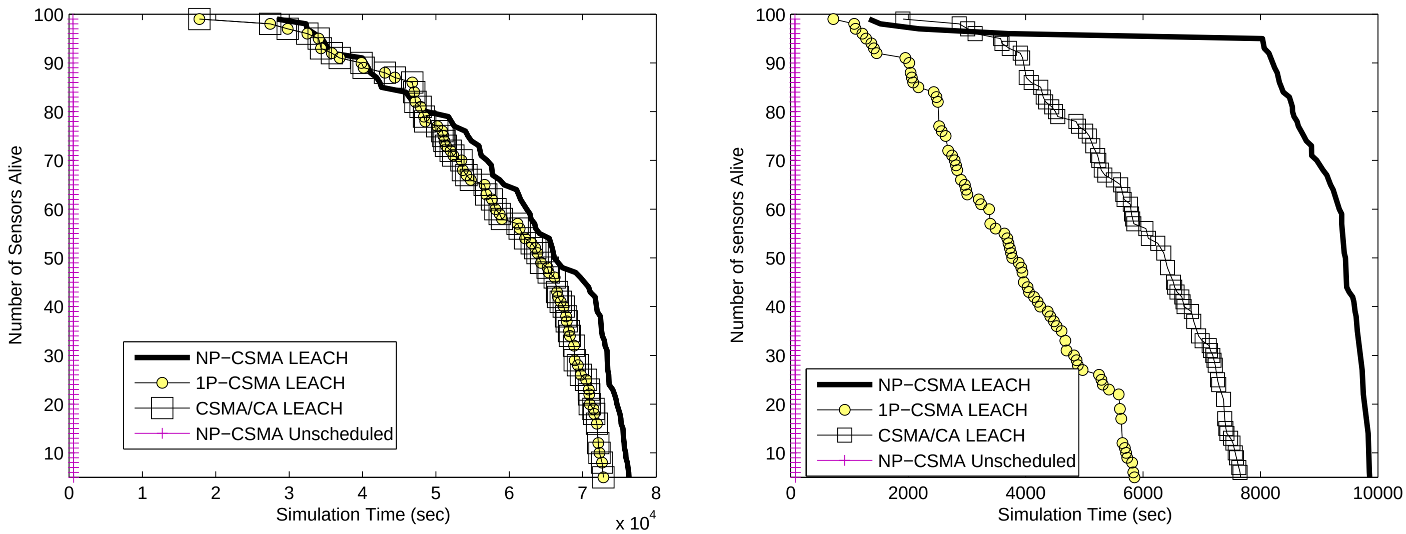
  where E_{elec} is the electronics energy, $\epsilon_{fs} \times d^2$ or $\epsilon_{mp} \times d^4$ are the amplifier energies that depends on the distance to the receiver, and d_0 is a distance threshold between the transmitter and the receiver over which the multipath fading channel model is used (i.e., d^4 power loss), otherwise the free space model (i.e., d^2 power loss) is considered.
- The energy to receive a packet depends only on the packet size, then:

$$E_{rx}(l) = l \times E_{elec} \quad (2)$$

- Considering LEACH, each CH dissipates energy in reception, transmission and in aggregating the signals received from the CMs. The energy for data aggregation is set as $E_{DA} = 5$ nJ/bit/signal.
- CHs perform ideal data aggregation.
- The expected number N_{CH} of CHs following the cluster formation phase is set equal to 5. In this section, we used the same network topology as in [15], where it was demonstrated that LEACH is most efficient when the number of CHs, N_{CH}, is equal to 5 in a 100-node network. Hence, the results shown here for LEACH are obtained by choosing the best parameter value for N_{CH}.
- The rest of the parameters are listed in Table I.

B. Impact of the Random Access Protocol

Figure 1 shows the evolution in time of the number of sensors still alive in the WSN in the LEACH and the unscheduled cases. In the unscheduled case, access is arbitrated using NP-CSMA with GB policy. In the LEACH case, three random access strategies are considered: NP-CSMA, 1P-CSMA and the CSMA/CA, all with the GB policy. We use the same backoff policy (i.e., GB) in order to perceive the impact of the random



(a) $q = 0.01$ (b) $q = 0.3$

Fig. 1. Evolution in time of the number of sensors still alive in the WSN

Parameter	Value
ϵ_{fs}	10 pJ/bit/m^2
ϵ_{mp}	0.0013 pJ/bit/m^4
E_{elec}	50 nJ/bit
E_{DA}	5 nJ/bit/signal
Idle power	13.5 mW
Sleep power	15 μW
Initial energy per node	2 J
Transmission bit rate	40 kbs^{-1}
Round time	20 sec.

TABLE I
PARAMETERS SETTING

access strategy on the WSN performance. Typically, we fix the backoff policy and we vary the random access strategy. Note that similar results can be obtained with the other backoff policies.

Let us first focus on the LEACH performance. Figure 1 shows that for low values of q, the different access protocols provide comparable results, whereas for moderate values of q the NP-CSMA is the best (see Fig. 1(b)). Indeed, with low values of the probability q, all the access protocols enable practically collision-free transmission and achieve thus similar energy consumption. It is worth noting that in this range of q, achieving practically collision-free transmission comes at the cost of excessive access delay to the medium. In this context, the energy wasted due to idle listening while waiting to transmit or to receive a packet is dominant compared to the energy wasted due to collisions.

In contrast, for moderate values of q, the energy wasted due to collisions is dominant since collisions are more likely to happen. In this case, NP-CSMA allows the lowest energy consumption. On the other hand, 1P-CSMA presents the highest collision probability leading thus to the highest energy consumption per unit of time when LEACH is enabled as can be seen in Fig. 2. In view of this, the WSN experiences the fastest sensor node energy drain with 1P-CSMA (see Fig. 1(b)).

Let us now compare LEACH to the basic unscheduled case from energy consumption perspective. We can see in Figs. 1 and 2 that LEACH achieves always significant gain compared to the basic unscheduled transmission case. This is because LEACH

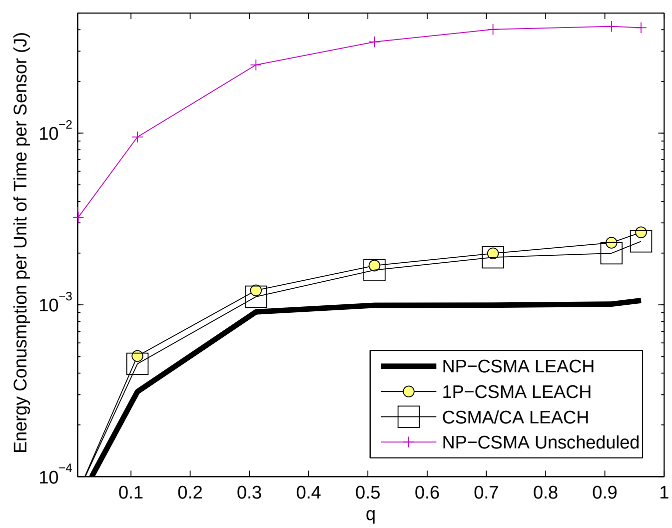


Fig. 2. Average energy consumption per unit of time per sensor node

coordinates the sensor nodes' transmissions with a common schedule in the steady phase, which eliminates collisions, idle listening and overhearing. This gain depends on the access protocol choice. For example, Fig. 1(b) shows that using the 1P-CSMA access protocol with LEACH provides the smallest gain. This is because 1P-CSMA causes excessive collisions among the signaling messages at the cluster formation phase. This harmful wastage of energy at the cluster formation phase slows down the gain that achieves LEACH in the steady phase due to its scheduled transmission compared to the unscheduled case.

Let us now focus on the latency performance. Figure 3 depicts the reporting and the cluster formation latencies. The reporting latency is defined as the time between the report generation and its reception by the sink node. The cluster formation latency is the time needed to form the clusters, i.e., to elect the cluster heads and to construct the TDMA frames. Again, NP-CSMA allows the best results when LEACH is enabled. In this case, the reporting latency

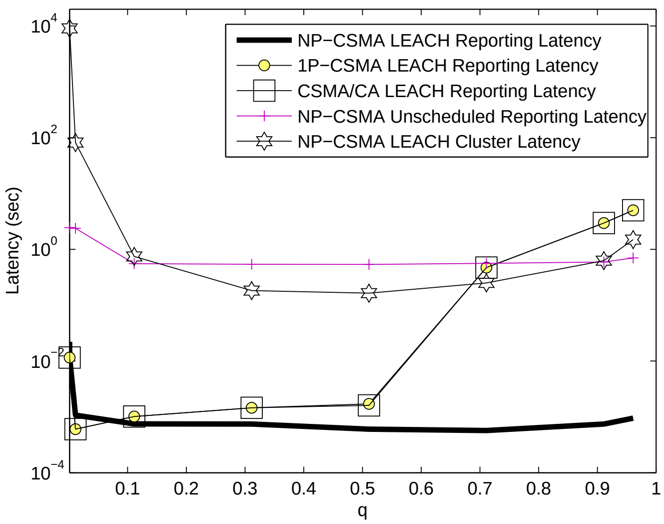


Fig. 3. Average reporting and cluster formation latencies

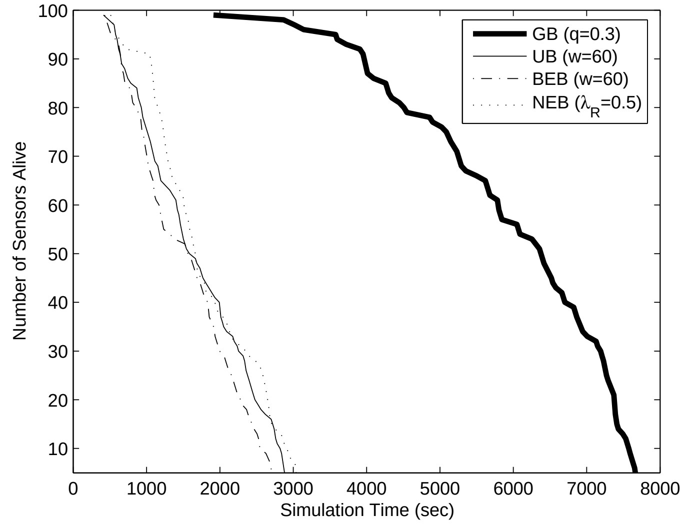


Fig. 4. Evolution in time of the number of sensors still alive in the WSN when varying the backoff policy

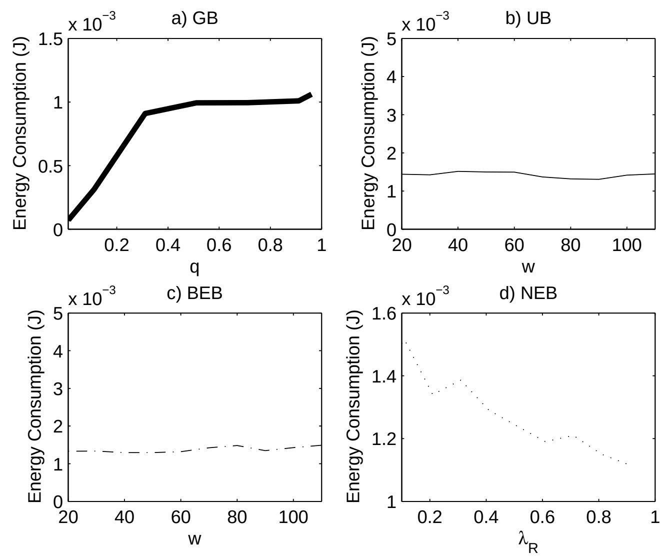


Fig. 5. Average energy consumption per unit of time per sensor node when varying the backoff policy

curve follows the same pace as that of the cluster formation latency curve, which is a convex function of the probability q. The rationale behind this can be explained as follows. For small values of q, the access delay to the medium during the set-up phase is very large, which induces large cluster formation latency. On the other hand, large values of q cause excessive collisions, increasing thus the time needed to transmit correctly a signaling message. Hence, the optimal cluster formation latency is a tradeoff between the above opposite requirements. In our scenario, the minimal cluster formation time is obtained when q ranges between 0.3 and 0.5. It is worth noting that the reporting latency is always lower than the cluster formation latency, since after the set-up phase, packets are transmitted in a contention-free way and sensor nodes only have to wait for their assigned time slots inside the TDMA frame.

Finally, compared to unscheduled case, the NP-CSMA-based LEACH achieves lower latencies thanks to its collision-free transmission during the steady phase.

According to the above results regarding both the energy consumption and the reporting latency, we can draw two important conclusions: i) the cluster-based LEACH architecture performs always better than an unscheduled one and ii) the NP-CSMA behaves better than the 1P-CSMA or CSMA/CA protocols for the different parameters of the backoff policy. Therefore, for the rest of the paper, we use the NP-CSMA as access strategy. In the next subsection, different backoff policies are used with the NP-CSMA in order to analyze their performances.

C. Impact of the Backoff Policies

In this subsection, we analyze the NP-CSMA-based LEACH protocol using different backoff policies. Recall that in the previous subsection, we proved that, using the same access protocol, the cluster-based systems outperform always the unscheduled systems. Moreover, we showed that NP-CSMA stands out as the best access strategy for cluster-based systems. In this subsection, we rather look for the best backoff policy that enables further energy conservation as well as reduced reporting delay.

Figures 4 and 5 compare the energy efficiency among the four backoff policies: GB, UB, BEB and NEB. The main observation is that GB provides the lowest energy consumption compared to the remaining policies, which on the other hand exhibit similar results. Specifically, Fig. 5 shows that the energy consumption with the GB policy is always below 1 mJ per unit of time, whereas it is around 1.5 mJ with the other backoff policies.

Figure 6 shows the reporting and the cluster formation latencies for the four backoff policies. Again, we can observe in Fig. 6(a) that using the GB policy the reporting and cluster latencies are convex functions of q, where minimum delays are obtained for q in the range of $[0.3, 0.5]$. Moreover, the GB policy achieves similar results (although sometimes slightly higher) as the remaining backoff policies.

Since the GB policy achieves better results in terms of energy

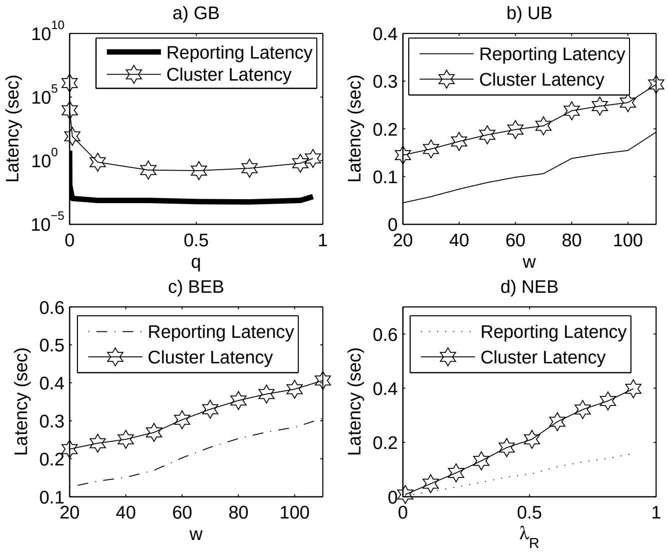


Fig. 6. Average reporting and cluster formation latencies when varying the backoff policy

consumption, even at the cost sometimes of slightly higher latencies compared to the other backoff policies, then the NP-CSMA with GB policy will be used as the access strategy for the rest of the manuscript.

IV. Mathematical Model for LEACH

In this section, we present a mathematical model for the LEACH-enabled WSNs. Compared to [15], we consider the energy consumption and the delay introduced by the cluster formation phase. We present explicit expressions for the average energy consumed per unit of time by a sensor node, the average reporting latency and the average cluster formation time. We consider the LEACH protocol with the NP-CSMA access strategy and the GB policy, where a packet transmission is done with probability q. It is important to note that the results provided by this model will be used as baselines to which the CM-EDR improvements are compared. In the next section, we present the analytical model when the CM-EDR strategy is enabled. In Table II we present the mathematical symbols used in this section for both the energy consumption and latency analysis.

A. Energy Consumption Analysis

At the beginning of each new cycle or round, a new set of N_{CH} CHs is elected. The CH role is rotated among all sensor nodes in order to balance the energy consumption inside the WSN. The cluster formation phase can be divided into three steps: CH announcement, CM join and CH schedules. In the first step, each elected CH advertises all the sensor nodes in the WSN. Once the CH announcement step is completed, each sensor node transmits a CM join message to its associated CH. Based on this information, each CH transmits a message indicating the schedule to its associated CMs. In what follows, each step will be analyzed separately.

1) CH announcement step: At the beginning of the set-up phase, all the elected CHs try to advertise the remaining sensor nodes at the same time, leading thus to a collision occurrence. All the CH nodes undergo hence the backoff procedure. Accordingly, the channel is divided into time slots that can be used by the CHs to transmit their announcement messages. The duration of a time slot t_{sig} is by definition the time that takes a sensor to transmit a control packet.

In order to calculate the energy consumption in the CH announcement step, we consider that at any time slot, the system can be defined according to the number of potential nodes that can initiate transmission, n, and the number of actual transmissions made, m, at the beginning of the time slot. Hence, the system can be described by the duple (n, m). In the Appendix, we present a methodology based on a transitory Markov chain to derive the average number of time slots that the system can be found at state (n, m), namely:

$$E[N_B] = E\left[N_{\{(n,m)\}}\right] = \frac{p_a(n,m)}{p_a(n,1)} \tag{3}$$

where $p_a(N_{CH}, m) = \binom{N_{CH}}{m} q^m (1-q)^{N_{CH}-m}$ for $m = 0, ..., N_{CH}$.

Accordingly, the total energy consumption in the WSN during the CH announcement step can be calculated as follows:

$$\begin{aligned} E_{CH\_Announ} &= f(N_{CH}, l_{sig}) \\ &= N_{CH} E_{tx}(l_{sig}, d_{\max}) + (N - N_{CH})\, E_{rx}(l_{sig}) \\ &\quad + \sum_{n=1}^{N_{CH}} \sum_{m=1}^{n} E\left[N_{\{(n,m)\}}\right] \Big(m E_{tx}(l_{sig}, d_{\max}) \\ &\qquad + (N-m)\, E_{rx}(l_{sig}) \Big) \\ &\quad + \sum_{n=1}^{N_{CH}} E\left[N_{\{(n,0)\}}\right] N E_{idle} t_{sig} \end{aligned} \tag{4}$$

where l_{sig} denotes the size of a control packet, $d_{\max} = \sqrt{2}M$ the diameter of the $M \times M$ square supervised area and E_{idle} the average amount of energy consumed per unit of time by a sensor node in the idle state. We highlight that the first element of (4) corresponds to the energy dissipated in the WSN due to the first collision among all the CHs when attempting to send for the first time all together their announcement messages at the beginning of the set-up phase. The remaining elements of (4) correspond to the energy consumption during the backoff procedure that undergo the N_{CH} CHs.

2) CM join step: As explained before, once the CH announcement step is completed, each sensor node transmits a CM join message to its associated CH. Similarly to the CH announcement step, the $N - N_{CH}$ sensor nodes try to join their CHs at the same time, leading thus to a collision occurrence. Then, the sensor nodes enter in backoff procedure to transmit their CM join messages.

Following the same reasoning as in the CH announcement step (i.e., using (4)), we obtain the average energy dissipated during the CM join step as:

$$E_{CM\_Join} = f(N - N_{CH}, l_{sig}) \tag{5}$$

3) CH schedules step: In this step, each CH transmits a message indicating the schedule to its associated CMs. Using the same reasoning as before, the average energy consumed during the CH schedules step is given by:

$$E_{CH\_Sched} = f(N_{CH}, l_{sig}) \tag{6}$$

Notation	Description of the Notation
N	Number of sensor nodes in the network
N_{CH}	Average number of cluster heads in the network
N_{CM}	Average number of cluster members in the network
M	Side length of the square supervised area
l_{sig}	Size of the control packet
l_{data}	Size of the data packet
d_{max}	Diameter of the MxM square supervised area
$d_{CM(i)\_CH}$	Average distance between the cluster member node *i* and its associated cluster head
d_{CH_SN}	Average distance from the cluster head to the sink node
t_{sig}	Duration of the control packet
t_{data}	Duration of the data packet
$T_{sensing}$	Sensing period
T_{frame}	Average duration of a TDMA frame
T_{round}	Round time after which the cluster head nodes are elected anew
T_{set-up}	Average time spent in the cluster formation phase
T_{CH_Announ}	Average cluster head announcement time
T_{CH_Join}	Average cluster member join time
T_{CH_Sched}	Average cluster head schedule time
$T_{set-up(LEACH)}$	Average time needed to form the clusters
$T_{reporting(LEACH)}$	Average reporting latency
E_{tx}	Energy consumed at transmission
E_{rx}	Energy consumed at reception
E_{idle}	Average energy consumed per unit of time per sensor in the idle state
E_{sleep}	Average energy consumed per unit of time per sensor in the sleep state
E_{CH_Announ}	Average energy consumption at the cluster head announcement phase
E_{CM_Join}	Average energy consumption at the cluster member join step
E_{CH_Sched}	Average energy consumed at the cluster head schedules step
E_{Set-up}	Average energy consumed at the set-up phase
$E_{CM(LEACH)}$	Average energy consumed per cluster member node during the sensing period
$E_{CH_frame(LEACH)}$	Average energy consumed by a cluster head node during a TDMA frame
$E_{CH(LEACH)}$	Average energy consumed per cluster head node during the sensing period
$E_{WSN(LEACH)}$	Average energy consumed in the network during the sensing period
$E_{steady(LEACH)}$	Average energy consumed in the network during the steady phase
$E_{sensor(LEACH)}$	Average energy consumed in the network per unit of time in LEACH

TABLE II
MATHEMATICAL NOTATION FOR THE LEACH SYSTEM ANALYSIS

Finally, the average amount of energy dissipated to form clusters is:

$$E_{Set\text{-}up}(LEACH)=E_{CH\_Announ}+E_{CM\_Join}+E_{CH\_Sched} \quad (7)$$

4) Energy consumption in the steady phase: Let us now calculate the average amount of energy consumed during the steady phase, where each CH receives periodically a TDMA frame from its CMs. In our study, we assume that the N sensor nodes are uniformly distributed in the supervised area. Hence, there are on average N/N_{CH} nodes, including the CH, in each cluster.

In continuous-monitoring WSNs, each sensor node senses its area periodically, each $T_{sensing}$ period of time, where $T_{sensing} \geq T_{frame}$. We note that $T_{frame} = \frac{N}{N_{CH}} t_{data}$ is the duration of a TDMA frame, where t_{data} is the duration of a time slot needed by a sensor to transmit a data packet of size l_{data}. In the particular case where $T_{sensing} = T_{frame}$, the WSN operates in the saturation regime, i.e., a sensor node always has data to send to the sink node. Since each sensor node wakes up only during its attributed time slot, then the energy consumed by a CM i node during a sensing period $T_{sensing}$ is:

$$E_{CM}(i) = (T_{sensing} - t_{data})\, E_{sleep} + E_{tx}(l_{data}, d_{CM(i)\_CH})$$

where E_{sleep} is the average amount of energy consumed by a sensor node per unit of time in the sleep state and $d_{CM(i)\_CH}$ is the distance between the CM node i and its associated CH. In [15], it was demonstrated that if the density of nodes is uniform throughout the cluster area, then the expected square distance from the CM nodes to the CH is given by:

$$E\left[(d_{CM\_CH})^2\right] = \frac{M^2}{2\pi N_{CH}}$$

where M is the side length of the square supervised area. Hence the average amount of energy consumed by a CM node during a sensing period is:

$$E_{CM}(LEACH) = (T_{sensing} - t_{data})\, E_{sleep} + E_{tx}\left(l_{data}, \frac{M}{\sqrt{2\pi N_{CH}}}\right) \quad (8)$$

In turn, each CH consumes energy in receiving and aggregating the data sent by its CMs as well as in the transmission of that aggregated data to the sink node. The energy consumed by a CH node during a TDMA frame is therefore:

$$E_{CH\_frame}(LEACH) = \left(\frac{N}{N_{CH}} - 1\right) E_{rx}(l_{data}) + \frac{N}{N_{CH}} l_{data} E_{DA} + E_{tx}(l_{data}, d_{CH\_SN}) \quad (9)$$

where d_{CH_SN} is the average distance from the CH to the sink node.

Thus, the energy consumed by a CH node during a sensing period is:

$$E_{CH}(LEACH) = E_{CH\_frame}(LEACH) + (T_{sensing} - T_{frame})\,E_{sleep} \quad (10)$$

The energy consumed in the network during a sensing period is therefore:

$$E_{WSN}(LEACH) = N_{CH}\Bigg(\left(\frac{N}{N_{CH}} - 1\right)E_{CM}(LEACH) + E_{CH}(LEACH)\Bigg) \quad (11)$$

and the total energy consumed in the network during the steady phase is:

$$E_{Steady}(LEACH) = E_{WSN}(LEACH) \times \frac{T_{round} - T_{set-up}(LEACH)}{T_{sensing}} \quad (12)$$

where T_{round} is the round time after which the CH nodes are elected anew and $T_{set-up}(LEACH)$ is the average time spent in the cluster formation phase, which will be derived in the next subsection.

Finally, we obtain the average amount of energy consumed by each sensor node in the WSN per unit of time when the basic LEACH clustering is adopted:

$$E_{sensor}(LEACH) = \frac{E_{Steady}(LEACH) + E_{Set-up}(LEACH)}{NT_{round}} \quad (13)$$

B. Latency Analysis

In this subsection we derive both the average cluster formation time and the average reporting latency.

1) The average cluster formation time: It is the time needed to form the clusters, i.e., to perform the CH announcement, the CM join and the CH schedules steps. Using the same model introduced in the Appendix, the CH announcement time is simply the time elapsed from the beginning of the cluster formation procedure to the instant where all the CHs successfully transmit their announcement message. As such, the CH announcement time can be expressed as follows:

$$T_{CH\_Announ} = g(N_{CH}, t_{sig}) = \left(1 + \sum_{n=1}^{N_{CH}} \sum_{m=0}^{n} E\left[N_{\{(n,m)\}}\right]\right) t_{sig} \quad (14)$$

We highlight that (14) is the sum of the time lost due to the first collision among all the CHs when attempting to send for the first time all together their announcement messages (i.e., t_{sig}) and the average duration of the backoff procedure that undergo the N_{CH} CHs.

Following the same reasoning, we obtain the average time spent in the CM join and the CH schedules steps as follows:

$$T_{CM\_Join} = g(N - N_{CH}, t_{sig}) \quad (15)$$

$$T_{CH\_Sched} = g(N_{CH}, t_{sig}) \quad (16)$$

Finally, the average time needed to form clusters is:

$$T_{Set-up}(LEACH) = T_{CH\_Announ} + T_{CM\_Join} + T_{CH\_Sched} \quad (17)$$

2) The average reporting latency: It is the time needed by a generated report to be received by the sink node. In continuous-monitoring WSNs, the sensor nodes produce data information at the beginning of each sensing period. In the steady phase, the average reporting time is simply the transmission time of a TDMA frame. Considering the extra delay spent in the construction of the clusters, the reporting latency increases slightly as follows:

$$T_{reporting}(LEACH) = T_{frame} + \frac{T_{set-up}(LEACH)T_{sensing}}{T_{round}} \quad (18)$$

V. Continuous-Monitoring through Event Driven Reporting Philosophy

This section introduces our CM-EDR scheme. In the previous section, we presented a mathematical analysis for the classical continuous-monitoring LEACH WSNs. In this section, we analyze the corresponding CM-EDR-aware extension. Comparing the new results, i.e., the average energy consumption, the average reporting latency and the average cluster formation time, to that obtained with the classical approach, we can gauge the benefits introduced by the proposed CM-EDR technique.

A. The CM-EDR Scheme

The main idea behind the CM-EDR introduction is avoiding the extra transmission of non relevant data information, typical in classical continuous-monitoring WSNs. With CM-EDR, continuous-monitoring does not imply indeed continuous reporting. By reporting only relevant data, the sink node would gather exactly the same information as with classical continuous-monitoring applications while receiving less reports and thus dissipating less energy.

Enabling the CM-EDR technique, each sensor node continues to produce periodically data information. However, the sensed information is reported to the sink node only if it differs from the last transmitted data information. In doing so, the sensor node dissipates also less energy in communications, achieving thus significant energy conservation. Clearly, the energy consumption will greatly depend on the rate of variation of the phenomenon that the sensors are monitoring.

With CM-EDR, each sensor node needs to storage the last transmitted data (i.e., only a single packet). Evidently, this does not entail the need to increase the memory capacity of sensor nodes. Following to each periodic observation, the sensor node compares the new reading to the stored one. If both readings are similar, the new generated data packet is discarded. Otherwise, the new information is reported to the sink node and the stored information is updated. In this case, we deal with relevant data, referred to us also as an event.

It is worth noting that our approach can be seen as a new alternative to reduce the transmission of redundant information, by profiting from the natural temporal correlation among the sensed data information. Our technique complement the data fusion or aggregation techniques [20] – [23] and the spatial-correlation based schemes [24] – [26]. With aggregation techniques, paths from different sources to the sink form an aggregate tree, where the redundant data at the branching nodes are replaced by a single message. As a result, the number of packets traversing the network is considerably reduced, which leads to significant energy conservation [20] – [23]. Another way to reduce the transmission of redundant information is to profit from the spatial correlation among the densely deployed sensor nodes [24] – [26]. Typically, close nodes would most likely produce similar information. Letting therefore only one representative node to report data information inside the correlated area would help reducing the energy consumption. From this perspective, the CM-EDR technique can be viewed as a third and complementary alternative to limit the transmission of redundant information.

B. Illustrative Example: CM-EDR-enabled LEACH WSNs

This subsection describes a typical example of CM-EDR-enabled WSNs. The CM-EDR mechanism is introduced to a basic LEACH WSN. As already indicated, the LEACH operation comprises two phases: the set-up and the steady state phases. The CM-EDR mechanism deals only with the way sensing data is reported to the sink node. In this regard, its inclusion does not affect the set-up phase in the sense that the clusters are formed exactly in the way as with classical LEACH.

In contrast, during the steady phase, a sensor node reports only relevant data to its CH. Specifically, if the sensor node has not sensed a relevant data, it keeps unused its reserved slot on the TDMA frame. In turn, the CH transmits to the sink node only if it senses or receives relevant data from its CMs. We refer to this frame as relevant frame as opposed to the empty or free frame.

C. Analytical Model for the CM-EDR-enabled LEACH WSNs

This subsection extends the analysis done in section IV to the case where the CM-EDR technique is enabled. Since the CM-EDR technique does not affect the set-up phase, the analysis for this phase remains unchanged. Hereafter, we focus on the analysis of the steady phase. Again, we present in Table III the complete list of the mathematical symbols used in the mathematical analysis of this section.

Assume that the variations on the sensed information, for example the temperature around a sensor node, happen following a Poisson process of rate λ. In other words, the time between two variations of the temperature is exponentially distributed. In our case, each sensor node senses its area periodically, each $T_{sensing}$ period of time. $T_{sensing}$ is chosen by the administrator such that the probability that two or more changes on the sensed information occurs during $T_{sensing}$ be negligible, i.e., be below a certain threshold ε as follows:

$$\Pr\{N_{event} \geq 2\} = 1 - e^{-\lambda T_{sensing}} - \lambda T_{sensing} e^{-\lambda T_{sensing}} \leq \varepsilon \tag{19}$$

where N_{event} is the number of changes that occurs on the sensed information during $T_{sensing}$. As such, $T_{sensing}$ must verify:

$$T_{sensing} \leq \sup\{t \mid 1 - e^{-\lambda t} - \lambda t e^{-\lambda t} \leq \varepsilon\} \tag{20}$$

Hence, the probability that the sensed information be relevant, for example the temperature changes between two observations, i.e., during the last $T_{sensing}$ period, is given by:

$$P_{event} \simeq \Pr\{N_{event} = 1\} = \lambda T_{sensing} e^{-\lambda T_{sensing}} \tag{21}$$

Based on this model, during the steady phase each CM-EDR-enabled sensor node transmits on its reserved slot (i.e., uses the current frame) according to a geometric process of probability P_{event}. Assuming that a CM node enters the sleep mode during the sensing period and wakes up only on its associated slot if it has relevant data to transmit, the average amount of energy consumed by a CM node during a sensing period is:

$$\begin{aligned} E_{CM}(CM{-}EDR) &= P_{event} E_{CM}(LEACH) \\ &\quad + (1 - P_{event})\, T_{sensing} E_{sleep} \end{aligned} \tag{22}$$

On the other hand, each CH consumes energy in receiving and aggregating the data sent by its CMs as well as in the transmission of that aggregated data to the sink node. The average amount of energy dissipated by a CH node in the reception of a frame can be given by:

$$\begin{aligned} E_{CH\_rec} = &\sum_{k=0}^{\left\lceil \frac{N}{N_{CH}} \right\rceil -1} \binom{\left\lceil \frac{N}{N_{CH}} \right\rceil -1}{k} (P_{event})^k (1-P_{event})^{\left\lceil \frac{N}{N_{CH}} \right\rceil -1-k} \\ &\times \left(k E_{rx}(l_{data}) + t_{data} E_{idle} \left(\left\lceil \frac{N}{N_{CH}} \right\rceil -1-k \right) \right) \end{aligned}$$

Assuming perfect data aggregation, the average amount of energy dissipated by a CH node due to aggregation is:

$$\begin{aligned} E_{CH\_agg} = &\sum_{k=0}^{\left\lceil \frac{N}{N_{CH}} \right\rceil} \binom{\left\lceil \frac{N}{N_{CH}} \right\rceil}{k} (P_{event})^k \left(1 - P_{event}\right)^{\left\lceil \frac{N}{N_{CH}} \right\rceil -k} \\ &\times (k l_{data} E_{DA}) \end{aligned}$$

The average amount of energy dissipated by a CH for a possible transmission of the aggregated data to the sink node is:

$$E_{CH\_tr} = \left(1 - (1 - P_{event})^{\frac{N}{N_{CH}}}\right) E_{tx}(l_{data}, d_{CH\_SN})$$

Hence, the total energy consumed by a CH node during a TDMA frame when CM-EDR is enabled is:

$$E_{CH\_frame}(CM{-}EDR) = E_{CH\_rec} + E_{CH\_agg} + E_{CH\_tr} \tag{23}$$

and the energy consumed by a CH node during a sensing period is:

$$\begin{aligned} E_{CH}(CM{-}EDR) &= E_{CH\_frame}(CM{-}EDR) \\ &\quad + (T_{sensing} - T_{frame})\, E_{sleep} \end{aligned} \tag{24}$$

The energy consumed in the network during a sensing period is therefore:

$$\begin{aligned} E_{WSN}(CM{-}EDR) &= N_{CH} \Bigg(E_{CH}(CM{-}EDR) \\ &\quad + \left(\frac{N}{N_{CH}} - 1 \right) E_{CM}(CM{-}EDR) \Bigg) \end{aligned} \tag{25}$$

and the total energy consumed in the network during the steady phase is:

$$\begin{aligned} E_{Steady}(CM{-}EDR) &= E_{WSN}(CM{-}EDR) \\ &\quad \times \frac{T_{round} - T_{set-up}(LEACH)}{T_{sensing}} \end{aligned} \tag{26}$$

Finally, we obtain the average amount of energy consumed by each sensor node in the WSN per unit of time when the CM-EDR option is enabled:

$$\begin{aligned} E_{sensor}(CM{-}EDR) = &\Big(E_{Steady}(CM{-}EDR) \\ &+ E_{Set-up}(LEACH) \Big) \frac{1}{N T_{round}} \end{aligned} \tag{27}$$

Notation	Description of the Notation
λ	Variation rate of the supervised phenomenon
N_{event}	Number of changes on the sensed information per sensing period
P_{event}	Probability of having relevant information in a sensing period
$E_{CM}(CM-EDR)$	Average energy consumption per cluster member in a sensing period for the CM-EDR scheme
$E_{CM}(OCM-EDR)$	Average energy consumption per cluster member in a sensing period for the OCM-EDR scheme
E_{CH_rec}	Average energy consumption per frame reception per cluster head
E_{CH_agg}	Average energy consumption per cluster head due to aggregation
E_{CH_tr}	Average energy consumption per frame transmission to the sink node per cluster head
$E_{CH\_frame}(CM-EDR)$	Average energy consumption per TDMA frame per cluster head
$E_{CH}(CM-EDR)$	Average energy consumption per cluster head in a sensing period for the CM-EDR scheme
$E_{CH}(OCM-EDR)$	Average energy consumption per cluster head in a sensing period for the OCM-EDR scheme
$E_{WSN}(CM-EDR)$	Average energy consumption in the network in a sensing period
$E_{steady}(CM-EDR)$	Average energy consumption in the network during the steady phase
$E_{sensor}(CM-EDR)$	Average energy consumption per sensor node per unit of time
N_{sleep}	Number of sensing periods that the cluster head remains sleep in the OCM-EDR mechanism
N_{idle}	Number of idle frames before a cluster head goes to sleep mode in the OCM-EDR mechanism
$Y(k)$	State of the cluster head at the sensing period k
P_{free}	Probability that the cluster head does not report to the sink node
K	Number of sensing periods during a round
P_{CH_sleep}	Percentage of sensing periods in a round

TABLE III
MATHEMATICAL NOTATION FOR THE CM-EDR AND OCM-EDR LEACH SYSTEM ANALYSIS

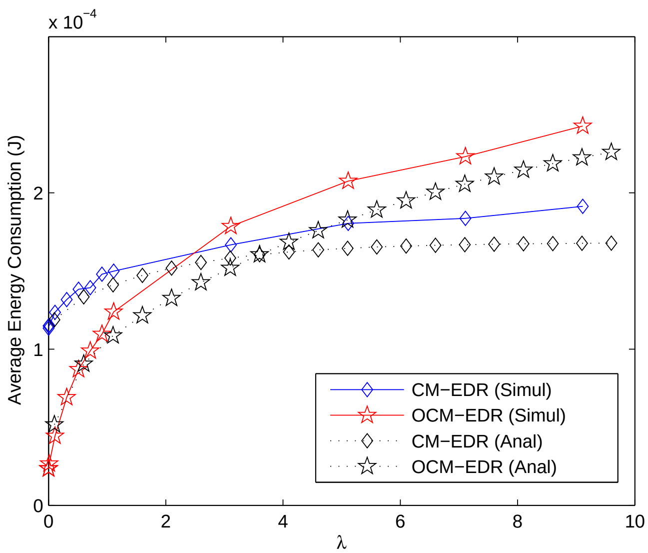


Fig. 7. Average energy consumption per unit of time per sensor node

With regard to the latency performance, it is worth noting that the CM-EDR scheme does not impact the latency compared to the classical LEACH case. Indeed, a relevant data packet is received by the sink node at the same time whether the CM-EDR mechanism is enabled or not. The CM-EDR mechanism avoids only the transmission of non relevant data.

A discrete event simulation model has been developed in order to validate the analytic results. Figure 7 compares the simulation results of the energy consumption with CM-EDR to that given by equation (27) as a function of the rate λ. In this case, $T_{sensing}$ is chosen such that it verifies the constraint given by (20) with $\varepsilon = 10^{-4}$. Figure 7 shows that there is a good fit between the simulation and analytical results, which exhibits the accuracy of our analysis.

D. Optional Mechanism for CM-EDR-enabled Cluster-Based WSNs

Using CM-EDR, a CH node transmits to the sink node only if it senses or receives relevant data from its CMs. As such, the CH may not transmit to the sink during a long period if it does not receive any relevant information. Even though, it dissipates energy due to idle listening. The energy wasted due to idle listening is far from being negligible and can account for a significant portion of the energy a sensor dissipates in some cases [27].

To achieve further energy conservation, the CH will be allowed with the optional CM-EDR (OCM-EDR) to enter sleep mode during N_{sleep} sensing periods if it does not receive any relevant data during N_{idle} consecutive frames. The CH assumes indeed that the supervised environment is *"calm"* and it is improbable that an event occurs in the next sensing periods. In this case, the CH advertises its CMs that it will undergo the sleep state during N_{sleep} sensing periods. However, during this period, a CM node may sense a relevant data that needs to be reported immediately (i.e., in the current frame) to the sink node, otherwise continuous-monitoring property is lost. To do so, the sensor node is allowed to transmit directly this information to the sink node during its reserved slot.

Let us now calculate the average energy consumption by a sensor node when this optional mechanism is enabled. Let $Y(k)$ be the CH state at the sensing period k of the steady phase defined by the tuple (i, j), where $i = 0$ if the CH is in the sleep state and $i = 1$ otherwise. Moreover, if $i = 0$, $j = 1, ..., N_{sleep}$ signifies that the CH has been for j sensing periods in the sleep state (including the current sensing period); otherwise (i.e., if $i = 1$) $j = 1, ..., N_{idle}$ indicates the number of consecutive empty (non relevant) frames that has received the CH. The process $Y = \{Y(k), k \geq 1\}$ is a discrete time Markov chain with the state space $S = \{(i, j) \mid 0 \leq i \leq 1, 1 \leq j \leq N_{sleep} 1_{\{i=0\}} + N_{idle} 1_{\{i=1\}}\}$. For every $s \in S$, we denote by

$$\Pi_s = \lim_{k \to +\infty} \Pr\{Y(k) = s\}$$

where $\Pi = [\Pi_s]$ is the steady state distribution of the Markov chain Y, which satisfies

$$\Pi P = \Pi \text{ and } \sum_{s \in S} \Pi_s = 1, \qquad (28)$$

and $P = (P(s, s'))$, $s = (i, j)$, $s' = (i', j') \in S$, is the transition probability matrix of Y given by:

$$P(s,s') = \begin{cases} P_{free} & \text{if } \left(i = i' = 1 \text{ and } j' = j+1\right); \\ 1 - P_{free} & \text{if } \left(s' = (1,1) \text{ and } s = (1,j) \text{ with } j < N_{idle}\right); \\ 1 & \text{if } \begin{cases} \left(i = i' = 0 \text{ and } j' = j+1\right) \\ \text{or } \left(s = (1, N_{idle}) \text{ and } s' = (0,1)\right) \\ \text{or } \left(s = (0, N_{sleep}) \text{ and } s' = (1,1)\right); \end{cases} \\ 0 & \text{otherwise.} \end{cases} \quad (29)$$

where P_{free} is the probability that the CH node does not transmit to the sink node since it has not any relevant data to forward. P_{free} is given by:

$$P_{free} = (1 - P_{event})^{\frac{N}{N_{CH}}} \quad (30)$$

Let $K = \left\lceil \frac{T_{round}}{T_{sensing}} \right\rceil$ denote the number of sensing periods during a round. We denote by P_{CH_sleep} the percentage of sensing periods in a round, during which a CH is in the sleep state. P_{CH_sleep} can be expressed as follows:

$$P_{CH\_sleep} = \frac{1}{K} \sum_{j=1}^{N_{sleep}} V_{(0,j)}(K) \quad (31)$$

where $V_{(0,j)}(K)$ is the number of visits to the state $(0,j)$ during a round, i.e., during the K first transitions of process Y. Then, P_{CH_sleep} is given by:

$$\begin{aligned} P_{CH\_sleep} &= \frac{1}{K} \sum_{j=1}^{N_{sleep}} \sum_{k=1}^{K} \Pr\{Y(k) = (0,j)\} \\ &= \frac{1}{K} \sum_{j=1}^{N_{sleep}} \sum_{k=1}^{K} \left(\alpha P^k\right)_{(0,j)} \end{aligned} \quad (32)$$

where α is the initial probability distribution of Y and $\left(\alpha P^k\right)_{(0,j)}$ is the $(0,j)$ element of the vector αP^k. Note that when K goes to the infinity, P_{CH_sleep} denotes the probability that a CH is in the sleep state during a sensing period, i.e.,

$$\lim_{K \to +\infty} P_{CH\_sleep} = \sum_{s \in S} \Pi_s 1_{\{i=0\}} = \sum_{j=1}^{N_{sleep}} \Pi_{(0,j)} \quad (33)$$

Deriving the steady state distribution of the Markov chain Y, we get

$$\begin{aligned} \lim_{K \to +\infty} P_{CH\_sleep} &= \sum_{j=1}^{N_{sleep}} N_{sleep} \left(P_{free}\right)^{N_{idle}-1} \Pi_{(1,1)} \\ &= \frac{N_{sleep}\left(1 - P_{free}\right)\left(P_{free}\right)^{N_{idle}-1}}{1 - (P_{free})^{N_{idle}} + N_{sleep}(1 - P_{free})(P_{free})^{N_{idle}-1}} \end{aligned} \quad (34)$$

Now, we can derive the average amount of energy consumed by a CM node during a sensing period as follows:

$$\begin{aligned} E_{CM}(OCM{-}EDR) = & \left(1 - P_{event}\right) T_{sensing} E_{sleep} \\ & + P_{event}\left(1 - P_{CH\_sleep}\right) E_{CM}(LEACH) \\ & + P_{event} P_{CH\_sleep} \Big(E_{tx}\left(l_{data}, d_{CM\_SN}\right) \\ & + \left(T_{sensing} - t_{data}\right) E_{sleep} \Big) \end{aligned} \quad (35)$$

where d_{CM_SN} is the average distance between a CM node and the sink node.

On the other hand, the average energy consumed by a CH node during a sensing period with OCM-EDR is:

$$\begin{aligned} E_{CH}(OCM{-}EDR) = & \left(1 - P_{CH\_sleep}\right) E_{CH}(CM{-}EDR) \\ & + P_{CH\_sleep} T_{sensing} E_{sleep} \end{aligned} \quad (36)$$

Using the expressions of $E_{CM}(OCM - EDR)$ and $E_{CH}(OCM - EDR)$ given by (35) and (36), respectively, we derive in the way as in (25), (26) and (27) the average energy consumed by a sensor node with OCM-EDR.

It is worth noting that with regard to the latency performance, the OCM-EDR scheme achieves slightly better results compared to the basic CM-EDR scheme, since some relatively long indirect transmissions to the sink through the CH are replaced by fast direct transmissions. As for the energy consumption performance, it is interesting to notice that the OCM-EDR mechanism performs better at low values of λ ($\lambda < 2.5$) while the CM-EDR mechanism should be preferred for higher values of λ. In section VI we analyze in further detail this behavior.

Figure 7 compares the simulation results of the energy consumption with OCM-EDR to that given by the analytical model as a function of λ. In this case, we consider $N_{idle} = 1$, $N_{sleep} = 10$ and $\varepsilon = 10^{-4}$. Figure 7 shows again that there is a good fit between the simulation and analytical results, which exhibits the accuracy of our analysis.

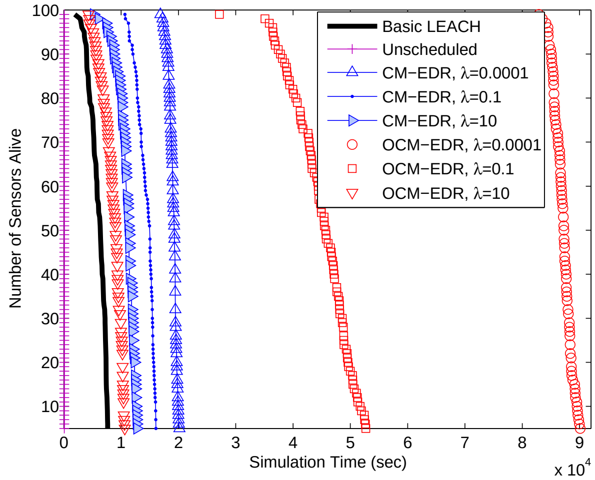
VI. Numerical & Simulation Results

In this section, we evaluate the efficiency of our proposed mechanisms (i.e., CM-EDR and OCM-EDR). We first study the gain that they introduced using four baseline examples: the case of unscheduled WSNs and three variants of cluster-based WSNs. Then, we compare between the CM-EDR and OCM-EDR mechanisms.

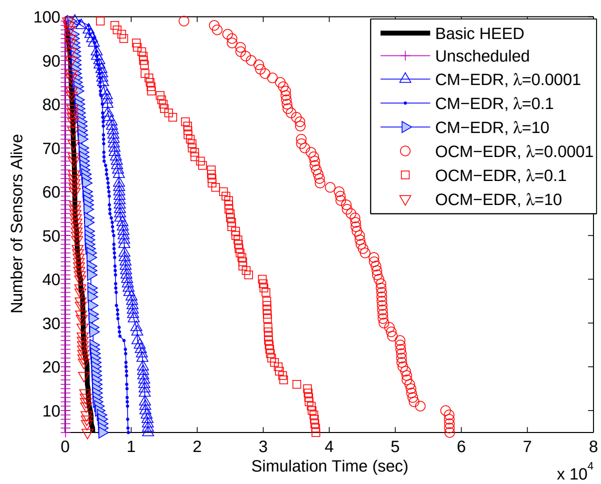
A simulation model has been developed in order to validate the analytic results. The system of WSNs was implemented as a discrete event simulation. Numerous evaluations were performed in order to confirm the analytic results. In all cases, the results matched very closely as shown in section V. For the remainder of the results, it has been confirmed that there is a good fit between the simulation and analytical results. Therefore, for presentation purposes, all remaining figures show only the simulation results. We assume the same network topology used in the previous sections, i.e., 100 sensor node-network. We assume also that $\varepsilon = 10^{-4}$, i.e., $T_{sensing} = \sup\{t — 1 - e^{-\lambda t} - \lambda t e^{-\lambda t} \leq 10^{-4}\}$. Moreover, unless explicitly notified, we consider $q = 0.3$, $N_{idle} = 1$ and $N_{sleep} = 10$. The parameters setting in our experiments are listed in table I.

Figure 8 shows the evolution in time of the number of still alive sensor nodes when LEACH, HEED and the MH clustering protocols are adopted. In each sub-figure, we plot the corresponding results when the CM-EDR and OCM-EDR extensions are enabled. The basic case of unscheduled transmission is also presented in each sub-figure and serves as a baseline that shows the gain that can be achieved by using clustering techniques.

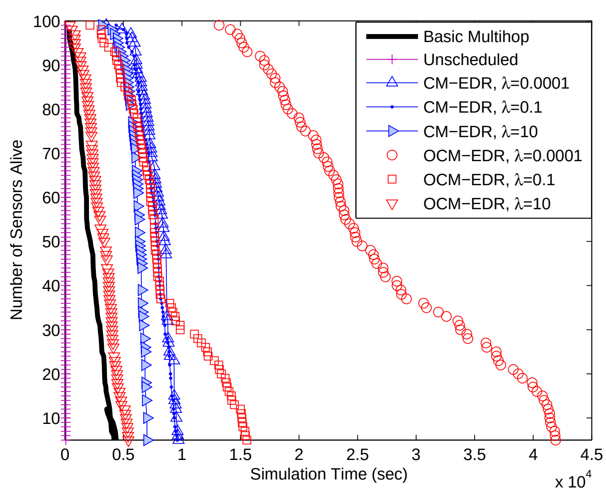
When our CM-EDR mechanisms are disabled, the sensor nodes always transmit their sensed data regardless of whether they have or not relevant information. Enabling our CM-EDR mechanisms,



(a) LEACH



(b) HEED

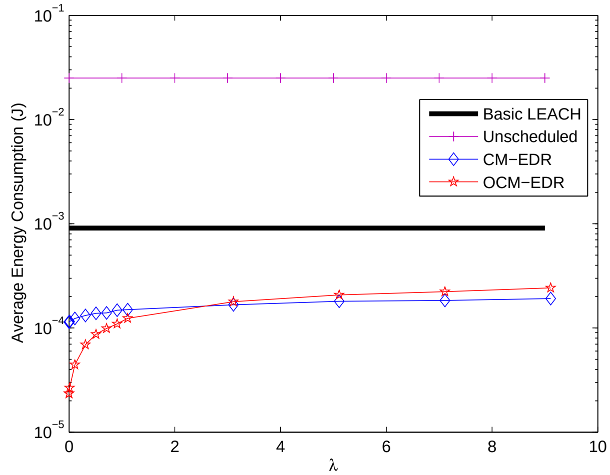


(c) MH clustering

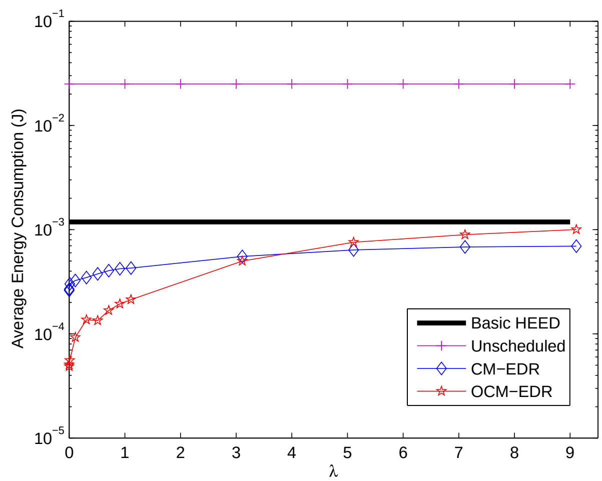
Fig. 8. Evolution in time of the number of sensors still alive in the WSN

a sensor node transmits on its assigned time slot only when it senses relevant data. Figure 8 shows the results for different values of the rate λ that specifies the speed of variations occurring on the sensed information. Small values of λ refer to relatively calm and static supervised environment, whereas large values of λ indicate agitated and variable environment.

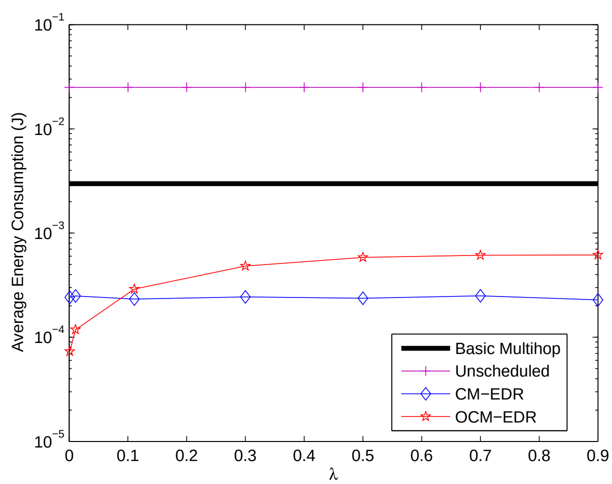
According to these results we can draw three main observations:



(a) LEACH



(b) HEED



(c) MH clustering

Fig. 9. Average energy consumption per unit of time per sensor node

- Clustering achieves always significant gain in terms of energy consumption compared to the basic unscheduled transmission case. This is because using clustering coordinates the sensor nodes' transmissions with a common schedule in the steady phase, which eliminates collisions, idle listening and overhearing. Further energy conservation can be achieved when the CM-EDR mechanisms are enabled, which brings us to the second observation.
- The sensor node lifetime is increased considerably when enabling our CM-EDR mechanisms. Clearly, the CM-EDR

abilities provide an advantage over the classical WSNs, by preventing the transmission of redundant data. For reference, Fig. 9 shows the relative decrease in the energy consumption by a sensor node per unit of time of the CM-EDR networks compared to the classic networks. The magnitude of the increase regarding the sensor node lifetime decreases as the rate λ grows. In other words, the relative improvement decreases when the supervised area becomes agitated since less non relevant data are transmitted by the classical WSNs.
- The OCM-EDR mechanism outperforms the CM-EDR one, when we deal with calm WSNs, whereas in agitated WSNs, it is better to use the basic CM-EDR mechanism. The rationale behind this can be explained as follows. Allowing the CHs to go to sleep with OCM-EDR results in the occurrence of expensive direct transmissions from the CMs to the sink node. In agitated environment, the energy conservation achieved at the CHs due to their asleep abilities is dominated by the additional energy consumed at the CM nodes due to frequent direct communications to the sink node. These direct communications become rare in calm WSNs.

Clearly, the CM-EDR systems are a major improvement over the classic networks. Figure 9 shows the average amount of energy consumed by a sensor node per unit of time as a function of the rate λ. Again, we can observe that the CM-EDR abilities provide significant energy conservation, notably in calm WSNs. This improvement decreases with λ. Moreover, enabling the optional version OCM-EDR is helpful only for small to moderate values of λ; otherwise, the basic version of CM-EDR performs better.

Figure 10 provides more insight into the effectiveness of using the OCM-EDR extension instead of the basic CM-EDR mechanism in the context of cluster-based WSNs. In this case, the two variants of the CM-EDR technique are introduced over a classical LEACH WSN. Note that similar results can be obtained when using the remaining clustering protocols. Figure 10 shows the performance of OCM-EDR as a function of the setting parameters N_{idle} and N_{sleep} for various values of the rate λ. Recall that with the optional OCM-EDR, the CH enters the sleep mode during N_{sleep} sensing periods if it does not receive any relevant data during N_{idle} consecutive frames. Four main conclusions may be drawn based on these results:

- The energy consumption with OCM-EDR is a convex function of N_{idle} (see Fig. 10(a)). For low values of N_{idle}, the CHs enter frequently to the sleep mode. Hence, the sensor nodes are most likely transmitting directly to the sink node instead of passing through the CHs. As a result, the energy consumption increases since the energy conservation achieved at the CHs due to their asleep abilities is dominated by the additional energy consumed at the CM nodes due to frequent direct communications to the sink node. On the other hand, when N_{idle} gets large values, the CHs almost never enter the sleep mode and can not profit from the calm periods of the supervised environment. Hence, the energy consumption increases. For moderate values of N_{idle}, the CHs enter the sleep mode without really penalizing the sensor nodes. In our scenario, setting $N_{idle} = 25$ enables the minimal energy consumption in the network (see Fig. 10(a)).
- In the same way, the energy consumption with OCM-EDR is a convex function of N_{sleep} (see Fig. 10(b)). Decreasing N_{sleep}, the CHs enter into the sleep state for very short periods of time and hence can not really profit from the calm periods of the supervised environment. As such, the CHs waste significant amounts of energy due to idle listening. This can be alleviated by increasing N_{sleep}. However, for large values of N_{sleep}, the gain achieved by the CHs is lost at the CM nodes due to frequent and high-consuming direct communications to the sink node. In our example, the energy consumption is minimal when $N_{sleep} = 36$ (see Fig. 10(b)).
- Comparing OCM-EDR to CM-EDR from energy consumption perspective, we can see again that OCM-EDR fits better calm environment, whereas CM-EDR is more convenient for agitated environment.
- With regard to reporting latency, we can see that OCM-EDR achieves always better results than the basic CM-EDR. This is because the OCM-EDR mechanism replaces some relatively long multi-hop transmissions (i.e., through the CH) by short direct transmissions. This gain increases with the increase of N_{sleep} (see Fig. 10(d)) and the decrease N_{idle} (see Fig. 10(c)) since in doing so more multi-hop transmissions are replaced by fast direct transmissions. This gain, however, may come at the cost of increasing energy consumption.

To conclude this paper, we can state that the CM-EDR philosophy enables significant energy conservation while ensuring continuous-monitoring applications. The decision to use the optional OCM-EDR instead of the basic CM-EDR mechanism depends on the supervised environment, whether it is calm or agitated. When OCM-EDR is preferred, the optimal parameter values of N_{idle} and N_{sleep} should be used to configure the sensor nodes.

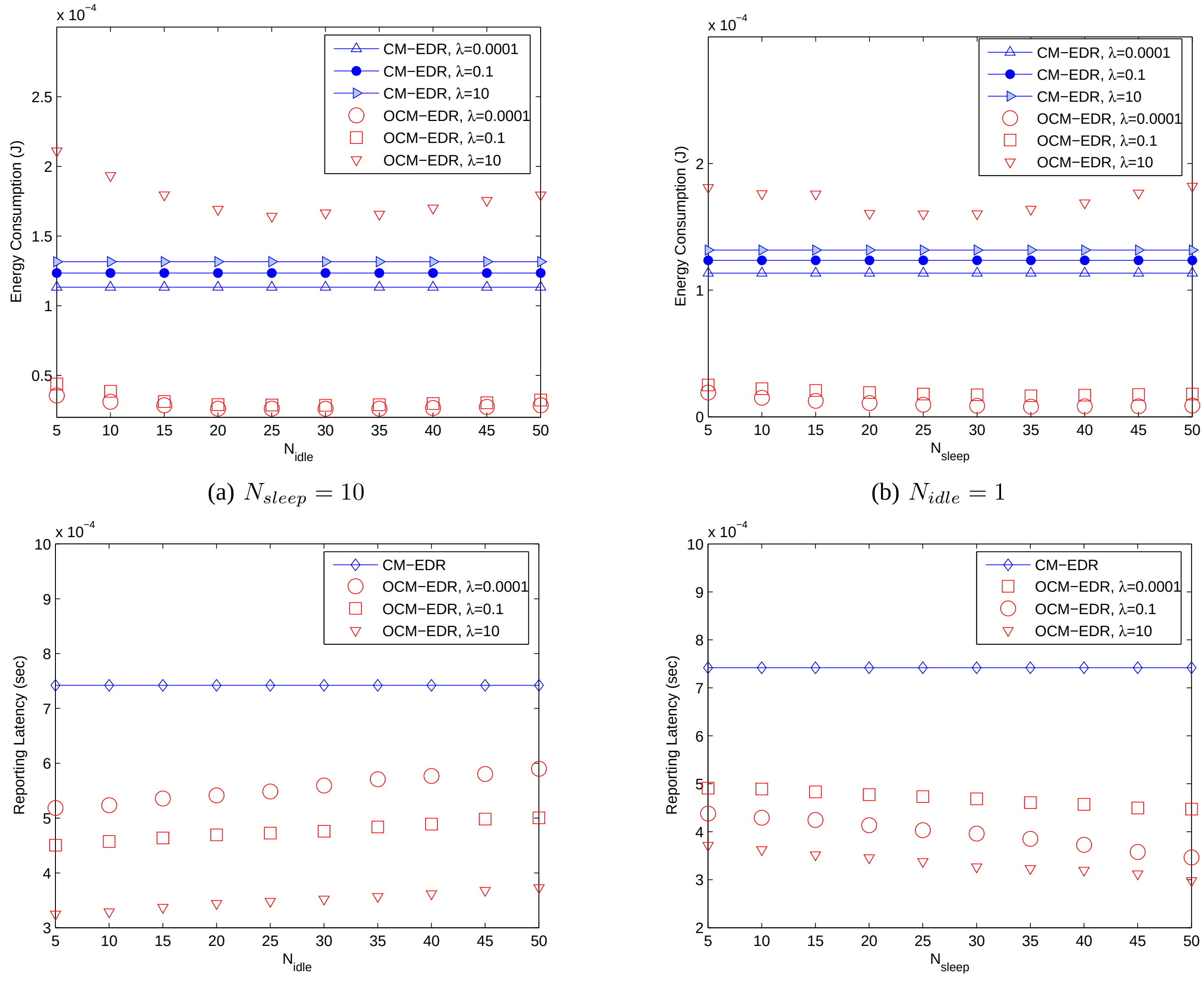
VII. Conclusion

This work has focused on studying the benefits to the energy consumption that can be gained by adding CM-EDR capabilities to systems of classical, unscheduled and cluster-based WSNs. The resulting continuous-monitoring WSN has been modeled, analyzed, simulated and studied.

The model developed describes a LEACH WSN. It calculates the energy consumption in both the set-up and steady phases. It then adds the CM-EDR capabilities by describing how only relevant data are reported to the sink node. The elaborated model is then extended to handle the case of optional OCM-EDR, where CHs can enter the sleep mode. The system can be described by a Markov process, with discrete time and finite state space.

Using this Markov process, the system was analyzed to find expressions for the energy consumption and reporting latency metrics. Simulations were also prepared and shown to closely agree with the analytical model. Using these two tools, a number of experiments were performed, in order to evaluate the characteristics of such systems.

Through these experiments, the potential performance gains of applying CM-EDR have been quantified. It has been verified that CM-EDR can allow for an improvement in the network lifetime while ensuring the continuous-monitoring task. More significantly however, it has been shown that for calm supervised environment,



(a) $N_{sleep} = 10$ (b) $N_{idle} = 1$

(c) $N_{sleep} = 10$ (d) $N_{idle} = 1$

Fig. 10. Comparison between OCM-EDR and CM-EDR

it is more convenient to use the optional OCM-EDR, whereas in agitated environment, it is better to use the basic CM-EDR mechanism. These results are summarized in Tables IV, V and VI.

Table IV presents a comparison in the energy consumption between the basic and the CM-EDR and OCM-EDR enhanced protocols considering different values of λ. With LEACH and HEED protocols, low λ signify values in the interval of $0 \leq \lambda \leq 3$ while with the Multihop protocol, low λ are values in the interval of $0 \leq \lambda \leq 0.1$.

It is worth noting that enabling the CM-EDR and OCM-EDR mechanisms reduces always the energy consumption. On the other hand, the OCM-EDR mechanism has superior performance in terms of energy consumption for low values of λ while for higher values of λ the CM-EDR mechanism provides lower energy consumption.

Table V summarizes our conclusions regarding the energy consumption when OCM-EDR is enabled in a LEACH-based WSN considering different values of N_{idle} and N_{sleep}. For both low and moderate values of λ, with low or high values of N_{idle}, OCM-EDR provides good performance. Considering moderate values of N_{idle} the performance is superior since the energy consumption decreases. A similar effect can be seen when N_{sleep} is varied. In this case, OCM-EDR has the best performance when λ is low. On the other hand, when the value of λ is high, OCM-EDR presents relatively bad performance for any values of N_{idle} and N_{sleep} since the energy consumption is higher than the basic CM-EDR mechanism.

To summarize the results regarding the latency performance, Table VI shows that for high values of λ, OCM-EDR has always the best performance since it allows the lowest latency. When λ is small, the system has higher latency. Even though, it still enables better latency than the basic CM-EDR mechanism.

Future research directions will be to consider our CM-EDR technique, which takes its advantage from the natural temporal correlation among the sensed data information, in conjunction with spatial-correlation based schemes, which reduce the transmission of redundant information by profiting from the spatial correlation among the densely deployed sensor nodes. From this perspective, the CM-EDR technique can be viewed as a complementary alternative to limit the transmission of redundant information. Energy savings would be much higher if sensors not only do not transmit non relevant data, but also considers the data form neighbor nodes in order to decide to transmit or enter the

Protocol	Low λ	High λ
Basic LEACH	Bad	Bad
CM-EDR LEACH	Good	Superior
OCM-EDR LEACH	Superior	Good
Basic HEED	Bad	Bad
CM-EDR HEED	Good	Superior
OCM-EDR HEED	Superior	Good
Basic Multihop	Bad	Bad
CM-EDR Multihop	Good	Superior
OCM-EDR Multihop	Superior	Good

TABLE IV

ENERGY CONSUMPTION COMPARISON BETWEEN THE BASIC PROTOCOLS AND THE CM-EDR AND OCM-EDR ENABLED PROTOCOLS

Protocol	$N_{idle} < 20$, $N_{idle} > 40$	$20 \leq N_{idle} \leq 40$	$N_{sleep} < 30$, $N_{sleep} > 45$	$30 \leq N_{sleep} \leq 45$
OCM-EDR LEACH Low λ	Good	Superior	Superior	Superior
OCM-EDR LEACH Medium λ	Good	Superior	Good	Good
OCM-EDR LEACH High λ	Bad	Bad	Bad	Bad

TABLE V

ENERGY CONSUMPTION COMPARISON FOR DIFFERENT PARAMETERS OF THE OCM-EDR LEACH ENABLED PROTOCOL

Protocol	Low N_{idle}	High $Nidle$	Low N_{sleep}	High N_{sleep}
OCM-EDR LEACH Low λ	Inferior	Inferior	Inferior	Inferior
OCM-EDR LEACH Medium λ	Good	Good	Good	Good
OCM-EDR LEACH High λ	Superior	Superior	Superior	Superior

TABLE VI

LATENCY COMPARISON FOR DIFFERENT PARAMETERS OF THE OCM-EDR LEACH ENABLED PROTOCOL

sleep mode.

APPENDIX

In this Appendix we make use of a transitory Markov chain in order to derive the average number of time slots that the LEACH system remains in the state (n, m) at the cluster formation phase, where n represents the number of CHs with a backlog packet (i.e., CHs that have not yet transmitted correctly their announcement messages) at the beginning of the slot k and $m \in \{0, ..., n\}$ represents the number of nodes that transmit on the slot k.

Let $X(k)$ be the system state at the slot k defined by the tuple (n, m) Then, the event $\{X(k) = (n, 0)\}$ means that no node transmits on the slot k and hence the slot remains free. $\{X(k) = (n, m)\}$ with $m > 1$ means that a collision occurs on the slot k. Finally, $\{X(k) = (n, 1)\}$ means that a successful transmission of a CH announcement message is achieved on the slot k. In this case, the next slot system state will be $X(k+1) = (n-1, m')$ with $m' \in \{0, ..., n-1\}$.

The transmission of each backlog node on a slot is achieved according to a geometric process with a probability q. Hence, the process $\{X(k), k \geq 1\}$ is a discrete time Markov chain with the state space $S = \{(n, m) \mid 0 \leq n \leq N_{CH}, 0 \leq m \leq n\}$ as depicted in Fig. 11. The space state S can be also expressed as follows:

$$S = \bigcup_{n=0}^{N_{CH}} S_n, \text{ with } S_n = \{(n, m) \mid 0 \leq m \leq n\}$$

To calculate the average energy consumption during the CH announcement step, we need to calculate the average number of visits of each state $s \in S$ before entering the $(0, 0)$ absorbing state.

The initial number of backlog CHs is N_{CH}. Hence, the system evolution starts at a state $s \in S_{N_{CH}}$. Specifically, $X(1) = (N_{CH}, m)$, with a probability

$$p_a(N_{CH}, m) = \binom{N_{CH}}{m} q^m (1-q)^{N_{CH}-m}, \ \forall\ m = 0, ..., N_{CH}. \quad (37)$$

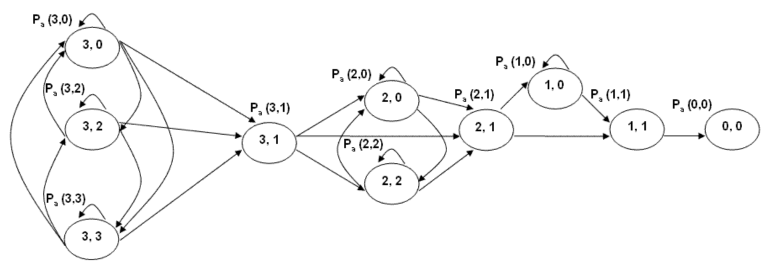


Fig. 11. State transition diagram of the Markov chain X: case $N_{CH} = 3$

Note that $\sum_{m=0}^{N_{CH}} p_a(N_{CH}, m) = 1$.

Any state $s \in S_{N_{CH}}$, i.e., $s \in \{(N_{CH}, m),\ m = 0, ..., N_{CH}\}$, could be visited several times until the system visits the state $(N_{CH}, 1)$, let say at slot k. This signifies that a successful CH transmission occurs at slot k and hence the remaining number of backlog CHs becomes $N_{CH} - 1$. The system evolves thus to the state $X(k+1) \in S_{N_{CH}-1}$ with a probability $p_a(N_{CH} - 1, m)$, $m = 0, ..., N_{CH} - 1$. Again this set of states $S_{N_{CH}-1}$ continues to be visited until the system visits the state $(N_{CH} - 1, 1)$, and so on and so forth.

Building on these observations, we can see that the number of visits to a state $(n, 1)$, $1 \leq n \leq N_{CH}$, before entering the absorbing state $(0, 0)$ is equal to 1. Moreover, calculating the number of visits of the process X to a generic state (n, m), with $1 \leq n \leq N_{CH}$ and $0 \leq m \neq 1 \leq n$, before entering the absorbing state $(0, 0)$ turns out at calculating the number of visits of the state (n, m) before entering the state $(n, 1)$, given that the system starts its evolution at the set of states S_n with an initial probability distribution $(p_a(n, 0), \ldots, p_a(n, n))$.

Hence, instead of studying the general process $\{X(k), k \geq 1\}$ to compute the average number of visits of a state (n, m), we can limit our study to the process $Z_n = \{(Z_n(r), r \geq 1\}$ shown in Fig. 12. Z_n is a Markov chain on the finite space

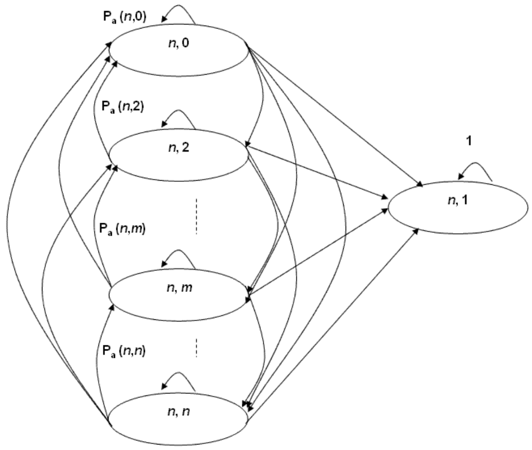


Fig. 12. State transition diagram of the Markov chain Z_n

$S_n = \{(n,0), \ldots, (n,n)\}$, where $S_n \backslash (n,1)$ is the set of the transient states and $(n,1)$ is the absorbing state.

Let denote by $B = \{(n,m)\}$ and by D the remaining transient states $D = \{(n,0), (n,2), \ldots, (n,m-1), (n,m+1), \ldots, (n,n)\}$. With these notations, we have $S_n = B \cup D \cup \{(n,1)\}$. $\alpha = (p_a(n,m), p_a(n,0), p_a(n,2), \ldots, p_a(n,m-1), p_a(n,m+1), ..., p_a(n,n), p_a(n,1))$ denotes the initial probability distribution of Z_n. According to the decomposition of S_n, we define also $\alpha_B = p_a(n,m)$ and $\alpha_D = (p_a(n,0), p_a(n,2), \ldots, p_a(n,m-1), p_a(n,m+1), ..., p_a(n,n))$. We denote by $P = (P(i,j))$, $i,j \in S_n$ the transition probability matrix of Z_n. P is given as follows:

$$P(i,j) = \begin{cases} 1 & \text{if } i = j = (n,1); \\ 0 & \text{if } i = (n,1) \text{ and } j \neq (n,1); \\ p_a(j) & \text{otherwise.} \end{cases} \quad (38)$$

To derive the number of visits of the state (n,m), the matrix P will be decomposed with respect to the partition $\{B, D, (n,1)\}$ as follows:

$$P = \begin{pmatrix} P_B & P_{BD} & P_{B\{(n,1)\}} \\ P_{DB} & P_D & P_{D\{(n,1)\}} \\ 0 & 0_D & 1 \end{pmatrix} \quad (39)$$

where 0_D is a row vector of dimension $n-2$ with all entries equal to 0.

Let N_B be the random variable representing the total number of visits of Z_n to the state (n,m). That is,

$$N_B = \sum_{r=1}^{+\infty} 1_{\{Z(r)=(n,m)\}}$$

Based on [19], we have

$$\Pr\{N_B = i\} = \begin{cases} 1-\beta & \text{if } i = 0; \\ \beta h^{i-1}(1-h) & \text{if } i \geq 1. \end{cases} \quad (40)$$

where β and h are given by:

$$\begin{cases} \beta = \alpha_B + \alpha_D (I_D - P_D)^{-1} P_{DB} \\ h = P_B + P_{BD} (I_D - P_D)^{-1} P_{DB} \end{cases} \quad (41)$$

where I_D is identity matrix with dimension $(n-2)$.

To derive β and h, we need to calculate $(I_D - P_D)^{-1} P_{DB}$ as follows:

$$(I_D - P_D)^{-1} P_{DB} = p_a(n,m) (I_D - P_D)^{-1} \mathbb{1}_D$$

where $\mathbb{1}_D$ is a column vector of dimension $n-2$ with all entries equal to 1.

Since P_D is a stochastic matrix up to a constant, i.e., $P_D \mathbb{1}_D = (1 - p_a(n,m) - p_a(n,1)) \mathbb{1}_D$, we have,

$$(I_D - P_D)^{-1} = \sum_{k=0}^{+\infty} (P_D)^k$$

This yields to

$$\begin{aligned} & (I_D - P_D)^{-1} P_{DB} \\ = \; & p_a(n,m) \sum_{k=0}^{+\infty} (P_D)^k \mathbb{1}_D \\ = \; & p_a(n,m) \sum_{k=0}^{+\infty} (1 - (p_a(n,m) + p_a(n,1)))^k \mathbb{1}_D \\ = \; & \frac{p_a(n,m)}{p_a(n,m) + p_a(n,1)} \mathbb{1}_D \end{aligned} \quad (42)$$

Replacing (42) in (41), we get

$$\beta = h = \frac{p_a(n,m)}{p_a(n,m) + p_a(n,1)} \quad (43)$$

As a result, we get

$$\Pr\{N_B = i\} = \begin{cases} 1-\beta & \text{if } i = 0; \\ \beta^i (1-\beta) & \text{if } i \geq 1; \end{cases}$$

and the average number of visits of Z_n to the state (n,m) is therefore:

$$E[N_B] = E\left[N_{\{(n,m)\}}\right] = \frac{\beta}{1-\beta} = \frac{p_a(n,m)}{p_a(n,1)} \quad (44)$$

REFERENCES

[1] I. Akiyldiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, *A survey on sensor networks*, IEEE Communications Magazine, Vol. 40, Issue 8, pp. 102–114, August 2002.
[2] K. Kredo II, P. Mohapatra, *Medium access control in wireless sensor networks*, Computer Networks, Volume 51, Issue 4, 14, pp. 961–994, March 2007.
[3] L. Wang, Y. Xiao, *A suervey of energy-efficient Scheduling Mechanisms in sensor networks*, Mobile Networks and Applications, Vol. 11, no. 5, pp. 723–740, oct. 2006.
[4] M. Miller and N. Vaidya, *A MAC protocol to reduce sensor network energy consumption using a wake-up radio*, IEEE Transactions on Mobile Computing, Vol. 4, Issue 3, pp. 228–242, May-June 2005.
[5] W. Ye, J. Heidemann, and D. Estrin, *Medium access control with coordinated adaptive sleeping for wireless sensor networks*, IEEE/ACM Transactions on Networking, vol. 12, no. 3, pp. 493–506, Jun. 2004.
[6] T. van Dam and K. Langendoen, *An adaptive energy-efficient MAC protocol for wireless sensor networks*, in Proc. of ACM SenSys 2003, pp. 171–180, Los Angeles, CA, Nov. 2003.
[7] J. Polastre, J. Hill, D. Culler, *Versatile low power media access for wireless sensor networks*, in Proc. of ACM SenSys 2004, pp. 95–107, 2004.
[8] I. Rhee, A. Warrier, M. Aia, J. Min, *Z-MAC: a hybrid MAC for wireless sensor networks*, in Proc. of ACM SenSys 2005, pp. 90–101, 2005.
[9] A. El-Hoiydi, J.-D. Decotignie, *WiseMAC: an ultra low power MAC protocol for multi-hop wireless sensor networks*, in Proc. of the International Workshop on Algorithmic Aspects of Wireless Sensor Networks (Algosensors), pp. 18–31, 2004.
[10] W. Ye, F. Silva, and J. Heidemann, *Ultra-Low Duty Cycle MAC with Scheduled Channel Polling*, in Proc. of SenSys'06, Boulder, Colorado, USA, November 2006.
[11] X. Shi, G. Stromberg, *SyncWUF: An Ultra Low-Power MAC Protocol for Wireless Sensor Networks*, IEEE Transactions on Mobile Computing, vol. 6, no. 1, pp. 115–125, Jan. 2007.
[12] S. Singh and C.S. Raghavendra, *PAMAS: Power Aware Multi-Access Protocol with Signaling for Ad Hoc Networks*, ACM Computer Communication. Review., pp. 5–26, July 1998.

[13] C. F. Chiasserini, and M. Garetto, *An Analytical Model for Wireless Sensor Networks with Sleeping Nodes*, IEEE Transactions on Mobile Computing, Volume 5, Issue 12, pp. 1706–1718, December 2006 .
[14] Y. Wu, S. Fahmy, N. B. Shroff, *Energy Efficient Sleep/Wake Scheduling for Multi-hop Sensor Networks: Non-convexity and Approximation Algorithm*, in Proc. IEEE INFOCOM 2007, pp. 1568–1576, May 2007.
[15] W. B. Heinzelman, A. P. Chandrakasan, H. Balakrishnan, *An application-specific protocol architecture for wireless microsensor networks*, IEEE Transactions on Wireless Communication, vol. 1, no. 4, pp. 660–670, Oct. 2002.
[16] O. Younis, S. Fahamy, *Distributed clustering in Ad-Hoc sensor networks: A hybrid, energy-efficient approach*, in Proc. IEEE INFOCOM 2004, vol.1, pp. 629–640, March 2004.
[17] S. Bandyopadhyay, E. J. Coyle, *An energy efficient hierarchical clustering algorithm for wireless sensor networks*, in Proc. IEEE INFOCOM 2003, vol. 3, pp. 1713–1723, April 2003.
[18] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ISO/IEC IEEE 802.11 Standard, 1999.
[19] B. Sericola, *Closed form solution for the distribution of the total time spent in a subset of states of a Markov process during a finite observation period*, Journal of Applied Probability, 27, pp. 713–719, 1990.
[20] C. Intanagonwiwat, R. Govindan, and D. Estrin, *Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks,* (2000), in Proc. Sixth Annual International Conference on Mobile Computing and Networks (MobiCom 2000), Boston, Massachusetts, August 2000.
[21] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, *Impact of network density on data aggregation in wireless sensor networks,* in Proceedings 22nd International Conference on Distributed Computing Systems 2002, pp 457 – 458, July 2002.
[22] T. Pham, E. J. Kim, and M. Moh, *On data aggregation quality and energy efficiency of wireless sensor network protocols - extended summary*, in Proc. of the IEEE First Conference on Broadband Networks (BROADNETS '04), pp. 730–732, Oct. 2004.
[23] M. Larrea, C. Martin, J. J. Astrain, *Hierarchical and fault-tolerant data aggregation in wireless sensor networks,* in Proc. IEEE Second International Symposium on Wireless Pervasive Computing (ISWPC '07), pp. 531–536, Feb. 2007.
[24] F. Bouabdallah, N. Bouabdallah, and R. Boutaba, *Towards Reliable and Efficient Reporting in Wireless Sensor Networks*, IEEE Transaction on Mobile Computing, to appear.
[25] M. C. Vuran, Ö. B. Akan, and I. F. Akyildiz, *Spatio-temporal correlation: theory and applications for wireless sensor networks*, Computer Networks Journal (Elsevier), vol. 45, no. 3, pp. 245–259, Jun. 2004.
[26] M. C. Vuran, and I. F. Akyildiz, *Spatial correlation-based collaborative medium access control in wireless sensor networks*, IEEE/ACM Transactions On Networking, vol 14, Issue 2, pp. 316–329, April 2006.
[27] A. Woo, D.E. Culler, *A transmission control scheme for media access in sensor networks*, in Proc. of the International Conference on Mobile Computing and Networking (MobiCom 2001), pp. 221–235, 2001.

Nizar Bouabdallah received the B.S. degree in telecommunications engineering from Ecole Superieur des Communications (Sup'Com), Tunis, Tunisia, in 2001, and the M.S. and Ph.D. degrees in network and computer science from the University of Paris VI, Paris, France, in 2002 and 2004, respectively. He joined Alcatel Research Laboratories, Marcoussis, France, in 2002, while working on his Ph.D. degree. In 2005, he was with the North Carolina State University, Raleigh, NC, USA, as a Postdoctoral Fellow. He is currently a researcher at INRIA (Institut National de Recherche en Informatique et en Automatique). In 2007, he spent six months as a Visiting Researcher at the School of Computer Science, University of Waterloo, Waterloo, ON, Canada. His research interests include optical networking, wireless and sensor networks, performance evaluation, network planning and modeling, as well as control and management architectures.

Mario E. Rivero Angeles (S'00-M'04) was born in Mexico D.F., Mexico, in 1976. He received the BSc degree from Metropolitan Autonomous University (UAM), Mexico, in 1998, the M.Sc. and Ph.D. degrees form CINVESTAV-IPN in 2000 and 2006 respectively in electrical engineering. He is a professor at the Advanced Technologies and Interdisciplinary Engineering Professional Unit (UPIITA-IPN), Mexico since 2002. He is currently a Postdoctoral Fellow at Dyonisos research project in INRIA (Institut National de Recherche en Informatique et en Automatique), Rennes, France. His research interest include random access protocols and data transmission in cellular networks and wireless sensor networks.

Bruno Sericola received the Ph.D. degree in computer science from the University of Rennes I in 1988. He has been with INRIA (Institut National de Recherche en Informatique et Automatique, a public research French laboratory) since 1989. His main research activity is in computer and communication systems performance evaluation, dependability and performability analysis of fault-tolerant architectures and applied stochastic processes.