5

Energy Efficient Transmission Techniques in Continuous-Monitoring and Event-Detection Wireless Sensor Networks

Nizar Bouabdallah, Bruno Sericola, Sofiane Moad
INRIA Rennes-Bretagne Atlantique
France

Mario E. Rivero-Angeles
INRIA Rennes-Bretagne Atlantique / UPIITA-IPN
France / Mexico

1. Introduction

Wireless Sensor Networks (WSNs) can be typically used to achieve Continuous Monitoring (CM) or Event-Detection Driven (EDD) inside the supervised area. For both applications, sensors consume energy for three main reasons: sensing, processing and wireless communicating. The wireless communication refers to data transmission and reception. Among these three operations, it is known that the most power consuming task is data transmission. Approximatively 80% of power consumed in each sensor node is used for data transmission. Hence, unnecessary transmissions and/or unnecessary large data packets reduce the system's lifetime. In this work, we are interested in studying different data transmission schemes that reduce the energy consumption by means of compression, in order to reduce the data packet's length, or by means of avoiding transmission of redundant information.
Continuous-monitoring 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 work, 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. For example, consider a continuous-monitoring temperature application, where each sensor node transmits periodically the sensed temperature to the sink node. 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 refreshed 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.

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. Specifically, we refer to these techniques as Continuous-Monitoring based on an Event Driven Reporting (CM-EDR) philosophy. Our proposed CM-EDR mechanisms 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.

In Event-Detection Driven applications, on the other hand, once an event occurs, it is reported to the sink node by the sensors within the event area. As such, the reporting nodes are expected to be closer to each other compared to the continuous-monitoring case where all nodes in the system are active simultaneously. Therefore, it is possible to take advantage of the spatial correlation inherit in these conditions. In view of this, we propose a compression technique for clustered-based event driven applications in wireless sensor networks. The main idea behind our proposal is to exploit the spatial correlation of such networks in order to reduce the size of the data packets by means of data compression. Specifically, the proposed scheme is composed of two major operations: Cluster Head (CH) selection and data compression.

Data compression is based on the following reasoning: Since the active nodes are inside the event area, they are usually very close to each other and the data correlation is expected to be high. As such, the data values sensed by the different nodes are most likely very similar. The proposed scheme exploits this correlation since nodes transmit only the difference of their sensed data and a reference value which is transmitted constantly by the node selected as CH. As it is shown, fewer bits are required to encode this difference compared to the case where the complete data value is transmitted. The other important procedure of the proposed scheme is the CH selection. This selection is carried out at the sink node (which is assumed to be outside the system's area and therefore is not energy constrained). The sink node receives a sample value of all active nodes at the beginning of the event and then selects the node that minimizes the aggregated data packet's size. Numerical results show that the proposed scheme achieves significant energy conservation compared to a classical clustering scheme [1].

2. 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.

In unscheduled MAC protocol-based WSNs (Kredo et al., 2007), the sensor nodes transmit directly their sensing data to the sink node without any coordination between them.

On the other hand, in 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

[1] This is footnote

communicates the aggregated information to the sink node. In our study, we consider the well known Low Energy Adaptive Clustering Hierarchy (LEACH) (Heinzelman et al., 2002) which is a simple and efficient clustering protocol.

3. 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.

3.1 Network Model

In our analysis, we consider different variations of the 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 (IEEE Specification, 1999). 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 distributed 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 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 parameter 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:

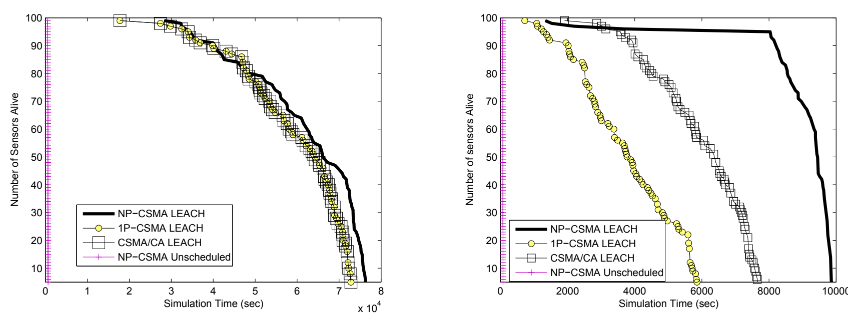
- The total number of sensor nodes in the system is $N = 100$.
- Sensor nodes are uniformly distributed in an area between $(0,0)$ and $(100,100)$ meters (i.e., square 100×100 area).
- The sink node is situated outside of the supervised area at the coordinate $(50,175)$ as in (Heinzelman et al., 2002).
- 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 (Heinzelman et al., 2002) where:

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

  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, $E_{rx}(l) = l \times E_{elec}$
- 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 (Heinzelman et al., 2002), 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.

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 1. Parameters setting



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

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

3.2 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 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

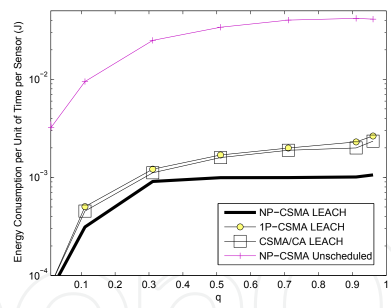


Fig. 2. Average energy consumption per unit of time per sensor node

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 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 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 document, 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.

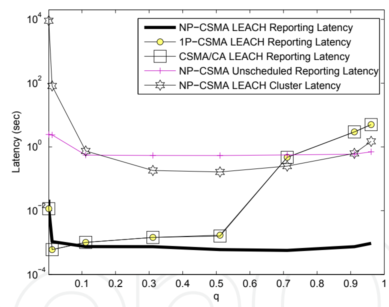
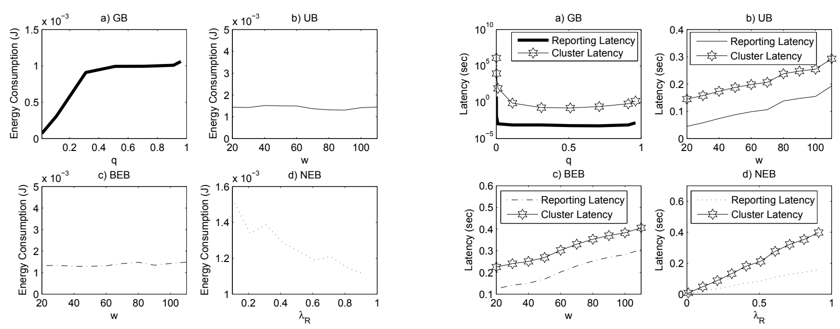


Fig. 3. Average reporting and cluster formation latencies

3.3 Impact of the Backoff Policies



(a) Average energy consumption per unit of time per sensor node when varying the backoff policy

(b) Average reporting and cluster formation latencies when varying the backoff policy

Fig. 4. Impact of the backoff policy on the performance of the system

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.

Figure 4 (a) 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. 4 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 4 (b) shows the reporting and the cluster formation latencies for the four backoff policies. Again, 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 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.

4. Mathematical Model for LEACH

In this section, we present a mathematical model for the LEACH-enabled WSNs. Compared to (Heinzelman et al., 2002), 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.

4.1 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.

4.1.1 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). 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. 5. The space state S can be also expressed as follows:

$$S = \bigcup_{n=0}^{N_{CH}} S_n, \text{ with } S_n = \{(n, m) \mid 0 \leq m \leq n\} \tag{2}$$

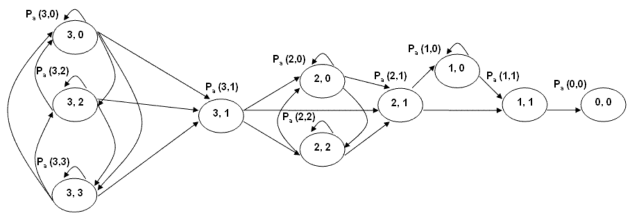


Fig. 5. State transition diagram of the Markov chain X: case $N_{CH} = 3$

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

$$p_a(N_{CH}, m) = \binom{N_{CH}}{m} q^m (1-q)^{N_{CH}-m}, \ \forall\, m = 0, ..., N_{CH}. \tag{3}$$

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 $Z_n = \{(Z_n(r), r \geq 1\}$. Z_n is a Markov chain on the finite space $S_n = \{(n, 0), \ldots, (n, n)\}$, where $S_n \backslash (n, 1)$ is the set of the transient states and $(n, 1)$ is the absorbing state. This Markov chain can be solved as in (Sericola, 1990), (Bouabdallah, 2009) and the average number of visits of Z_n to the state (n, m) is given by:

$$E\left[N_{\{(n,m)\}}\right] = \frac{p_a(n, m)}{p_a(n, 1)} \tag{4}$$

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

$$
\begin{aligned}
E_{CH\_Announ} = f(N_{CH}, l_{sig}) = N_{CH} E_{tx}(l_{sig}, d_{\max}) + (N - N_{CH})\, E_{rx}(l_{sig}) \\
+ \sum_{n=1}^{N_{CH}} \sum_{m=1}^{n} E\left[N_{\{(n,m)\}}\right] \left(m E_{tx}(l_{sig}, d_{\max}) + (N - m)\, E_{rx}(l_{sig}) \right) \\
+ N E_{idle} t_{sig} \sum_{n=1}^{N_{CH}} E\left[N_{\{(n,0)\}}\right]
\end{aligned}
\tag{5}
$$

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 (5) 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 (5) correspond to the energy consumption during the backoff procedure that undergo the N_{CH} CHs.

4.1.2 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 (5)), we obtain the average energy dissipated during the CM join step as $E_{CM\_Join} = f(N - N_{CH}, l_{sig})$.

4.1.3 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 $E_{CH\_Sched} = f(N_{CH}, l_{sig})$.
Finally, the average amount of energy dissipated to form clusters is:

$$E_{Set\text{-}up}(LEACH) = E_{CH\_Announ} + E_{CM\_Join} + E_{CH\_Sched} \tag{6}$$

4.1.4 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:

$$E_{CM}(i) = \left(T_{sensing} - t_{data} \right) E_{sleep} + E_{tx}(l_{data}, d_{CM(i)\_CH}) \tag{7}$$

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 (Heinzelman et al., 2002), 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:

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

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:

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

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:

$$E_{CH}(LEACH) = E_{CH\_frame}(LEACH) + \left(T_{sensing} - T_{frame}\right) E_{sleep}$$

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

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

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

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

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:

$$E_{sensor}(LEACH) = \frac{E_{Steady}(LEACH) + E_{Set-up}(LEACH)}{NT_{round}} \tag{8}$$

4.2 Latency Analysis

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

4.2.1 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 previous section, 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:

$$T_{CH\_Announ} \quad = \quad g(N_{CH}, t_{sig}) = \left(1 + \sum_{n=1}^{N_{CH}} \sum_{m=0}^{n} E\left[N_{\{(n,m)\}}\right]\right) t_{sig} \tag{9}$$

We highlight that (9) 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:

$$T_{CM\_Join} = g(N - N_{CH}, t_{sig}) \tag{10}$$

$$T_{CH\_Sched} = g(N_{CH}, t_{sig}) \tag{11}$$

Finally, the average time needed to form clusters is:

$$T_{Set-up}(LEACH) = T_{CH\_Announ} + T_{CM\_Join} + T_{CH\_Sched} \tag{12}$$

4.2.2 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:

$$T_{reporting}(LEACH) = T_{frame} + \frac{T_{set-up}(LEACH)T_{sensing}}{T_{round}} \tag{13}$$

5. Energy Efficient Protocols for Continuous-Monitoring Applications

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.

5.1 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 (Intanagonwiwat et al., 2000) – (Larrea er al., 2007) and the spatial-correlation based schemes (Bouabdallah et al., 2009) – (Vuran et al., 2006).

5.2 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.
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:

$$\Pr\{N_{event} \geq 2\} = 1 - e^{-\lambda T_{sensing}} - \lambda T_{sensing} e^{-\lambda T_{sensing}} \leq \varepsilon \tag{14}$$

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

$$T_{sensing} \leq \sup\{t \mid 1 - e^{-\lambda t} - \lambda t e^{-\lambda t} \leq \varepsilon\} \tag{15}$$

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:

$$P_{event} \simeq \Pr\{N_{event} = 1\} = \lambda T_{sensing} e^{-\lambda T_{sensing}} \tag{16}$$

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:

$$E_{CM}(CM\text{–}EDR) = P_{event} E_{CM}(LEACH) + (1 - P_{event})\, T_{sensing} E_{sleep}$$

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:

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

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

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

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:

$$E_{CH\_frame}(CM\text{–}EDR) = E_{CH\_rec} + E_{CH\_agg} + E_{CH\_tr} \tag{17}$$

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

$$E_{CH}(CM\text{–}EDR) = E_{CH\_frame}(CM\text{–}EDR) + \left(T_{sensing} - T_{frame} \right) E_{sleep}$$

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

$$E_{WSN}(CM\text{–}EDR) = N_{CH} \left(E_{CH}(CM\text{–}EDR) + \left(\frac{N}{N_{CH}} - 1 \right) E_{CM}(CM\text{–}EDR) \right)$$

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

$$E_{Steady}(CM\text{–}EDR) = E_{WSN}(CM\text{–}EDR) \times \frac{T_{round} - T_{set-up}(LEACH)}{T_{sensing}}$$

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:

$$E_{sensor}(CM\text{–}EDR) = \left(E_{Steady}(CM\text{–}EDR) + E_{Set-up}(LEACH) \right) \frac{1}{NT_{round}}$$

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.

5.3 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 (Woo et al., 2001).

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

$$\Pi P = \Pi \text{ and } \sum_{s \in S} \Pi_s = 1, \tag{18}$$

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

$$P(s,s') = \begin{cases} P_{free} & \text{if } \left(i = i' = 1 \text{ and } j' = j+1\right); \\ 1 - P_{free} & \text{if } \left(s' = (1,1) \text{ and } s = (1,j) \right. \\ & \quad \left. \text{with } j < N_{idle}\right); \\ 1 & \text{if } \begin{cases} \left(i = i' = 0 \text{ and } j' = j+1\right) \\ \text{or } \left(s = (1, N_{idle}) \text{ and } s' = (0,1)\right) \\ \text{or } \left(s = (0, N_{sleep}) \text{ and } s' = (1,1)\right); \end{cases} \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

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:

$$P_{free} = (1 - P_{event})^{\frac{N}{N_{CH}}} \tag{20}$$

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:

$$P_{CH\_sleep} = \frac{1}{K} \sum_{j=1}^{N_{sleep}} V_{(0,j)}(K) \tag{21}$$

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:

$$\begin{aligned} P_{CH\_sleep} &= \frac{1}{K} \sum_{j=1}^{N_{sleep}} \sum_{k=1}^{K} \Pr\{Y(k) = (0, j)\} \\ &= \frac{1}{K} \sum_{j=1}^{N_{sleep}} \sum_{k=1}^{K} \left(\alpha P^k\right)_{(0,j)} \end{aligned} \tag{22}$$

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.,

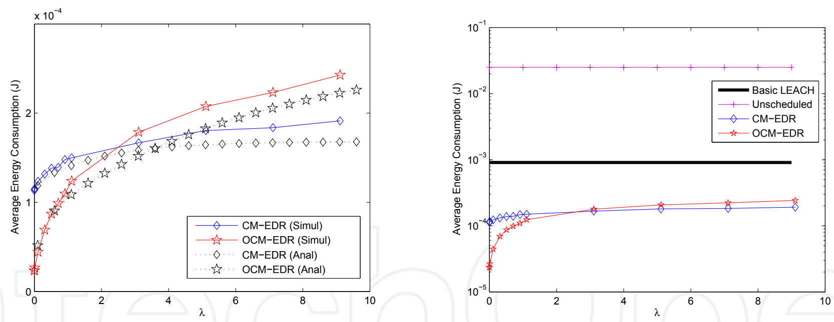
$$\lim_{K\to+\infty} P_{CH\_sleep} = \sum_{s\in S} \Pi_s 1_{\{i=0\}} = \sum_{j=1}^{N_{sleep}} \Pi_{(0,j)} \tag{23}$$

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

$$\begin{aligned} \lim_{K\to+\infty} P_{CH\_sleep} &= \sum_{j=1}^{N_{sleep}} N_{sleep} \left(P_{free}\right)^{N_{idle}-1} \Pi_{(1,1)} \\ &= \frac{N_{sleep}\left(1 - P_{free}\right)\left(P_{free}\right)^{N_{idle}-1}}{1 - \left(P_{free}\right)^{N_{idle}} + N_{sleep}\left(1 - P_{free}\right)\left(P_{free}\right)^{N_{idle}-1}} \end{aligned} \tag{24}$$

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

$$\begin{aligned} E_{CM}(OCM\text{–}EDR) = {} & (1 - P_{event})\, T_{sensing} E_{sleep} \\ & + P_{event} \left(1 - P_{CH\_sleep}\right) E_{CM}(LEACH) \\ & + P_{event} P_{CH\_sleep} \Big(E_{tx}\left(l_{data}, d_{CM\_SN}\right) \\ & + \left(T_{sensing} - t_{data}\right) E_{sleep} \Big) \end{aligned} \tag{25}$$



(a) Proposed protocols (b) Comparison with LEACH

Fig. 6. Average energy consumption per unit of time per sensor node

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:

$$E_{CH}(OCM\text{–}EDR) \;=\; \left(1 - P_{CH\_sleep}\right) E_{CH}(CM\text{–}EDR) + P_{CH\_sleep} T_{sensing} E_{sleep}$$

Using the expressions of $E_{CM}(OCM\text{–}EDR)$ and $E_{CH}(OCM\text{–}EDR)$ given by (we derive in the way as in (18) the average energy consumed by a sensor node with OCM-EDR.

5.4 Numerical Results

We now evaluate the efficiency of our proposed mechanisms 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. Figure 6 (a) compares the simulation results of the energy consumption with CM-EDR to that given by equation (18) as a function of the rate λ. In this case, $T_{sensing}$ is chosen such that it verifies the constraint given by (15) with $\varepsilon = 10^{-4}$. For the OCM-EDR mechanism, Figure 6 compares the simulation results of the energy consumption as a function of λ. In this case, we consider $N_{idle} = 1$, $N_{sleep} = 10$ and $\varepsilon = 10^{-4}$. Figure 6 (a) shows that there is a good fit between the simulation and analytical results, which exhibits the accuracy of our analysis.
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., $T_{sensing} = \sup\{t \mid 1 - e^{-\lambda t} - \lambda t e^{-\lambda t} \leq 10^{-4}\}$. 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.
According to the results presented in Fig. 6 we can draw three main observations:

- Clustering achieves always significant gain in terms of energy 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. 6 (b) 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 6(b) 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.

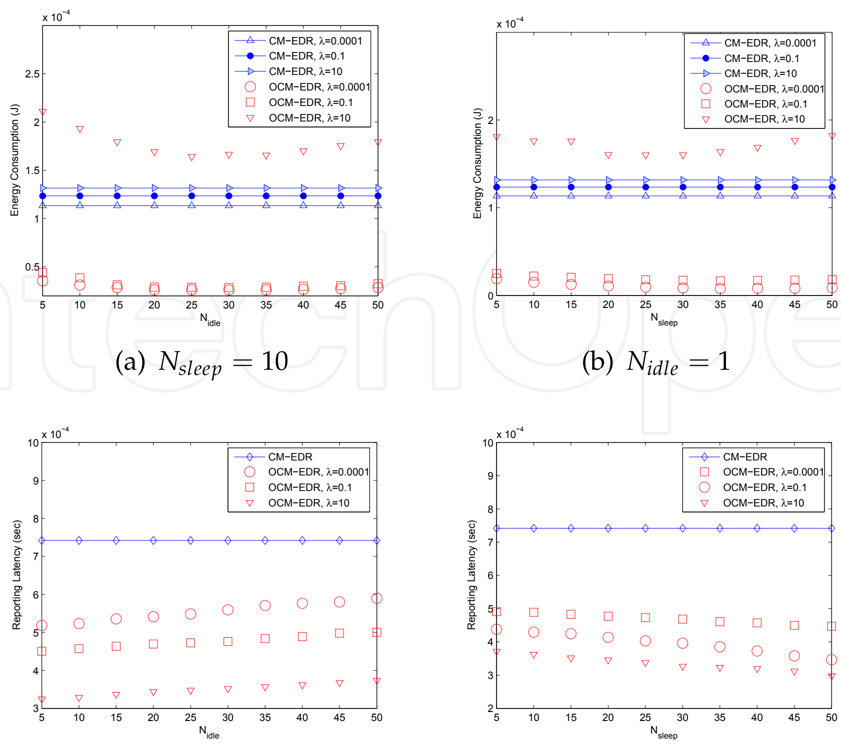
Figure 7 provides more insight into the effectiveness of using the OCM-EDR extension instead of the basic CM-EDR mechanism in the context of cluster-based WSNs. In this case, the two variants of the CM-EDR technique are introduced over a classical LEACH WSN. Note that similar results can be obtained when using the remaining clustering protocols. Figure 7 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.

The energy consumption with OCM-EDR is a convex function of N_{idle} (see Fig. 7(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. 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. 7(a)).

In the same way, the energy consumption with OCM-EDR is a convex function of N_{sleep} (see Fig. 7(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. In our example, the energy consumption is minimal when $N_{sleep} = 36$ (see Fig. 7(b)).

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.,

To conclude this study, we can state that the CM-EDR philosophy enables significant energy conservation while ensuring continuous-monitoring applications. The decision to use the op-



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

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

Fig. 7. Comparison between OCM-EDR and CM-EDR

tional 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.

6. Energy Efficient Protocols for Event-Driven Applications

For the event-driven applications, a new compression technique is proposed. Since the benefits of the clustering technique have been studied, the proposed scheme presented in this section is also clustered-based. The main idea behind our proposal is to exploit the spatial correlation of such networks in order to reduce the size of the data packets that will be sent by means of data compression. The proposed clustering scheme is based on selecting a data value as reference while the rest of the active nodes transmit only the difference between their sensed value and this reference value. Hence, one major issue in the proposed mechanism is to appropriate select the reference node that achieves the highest reduction of the packet size among all active nodes. The proposed scheme is evaluated analytically and by simulations. The results show that the proposed scheme may reduce as much as 11 times the energy consumption compared to a classical clustering scheme.

6.1 Network model

We consider an event-driven WSN consisting of N sensors deployed over a vast field We denote the *i-th* sensor node by n_i and corresponding sensor node set $S = \{n_1, n_2,...,n_N\}$ where $|S| = N$. Some assumptions about the sensor nodes and the underlaying network model are now presented:

- Nodes are uniformly distributed in an $A \times A$ area with (x, y) coordinates. Nodes are homogenous and have the same capabilities. Each node is assigned a unique identifier ID.
- Sensor nodes and the Base Station (BS) are all stationary after deployment. The BS can be reached by sensor nodes under a single high transmission range R_t meters.
- Nodes have two power controls to vary the transmission power which depends on the distance to the receiver. Each node n_i can reach any other node with a transmission range R_c. The BS can be reached with transmission range $R_t > R_c$.
- CHs use the average operation as the aggregation to eliminate the data redundancy. Other aggregation techniques, such as those proposed in (Azim et al., 2010) can also be implemented.

The center of the event is located in a random uniformly distributed point with coordinates (x_{event}, y_{event}) within the network's area. The range of the event area, i.e., the area where sensors can detect the event is R_event meters, where $R\_event \in [1, A] meters$. We also suppose that an event has a duration of T_event seconds. Additionally, in our model, only one event can be active inside the systemńs area and the data value C at the center of the event is constant, i.e, a stationary model in which the measured data do not change during the T_event seconds that the event is active is used.
A clustered based WSN is considered, where only one CH is elected for each event. The clustering process is triggered whenever an event is sensed by the nodes inside the event area.
The spatial correlation of the data from the different active nodes (the nodes that sense the event) can be considered according to the following models:

1. Diffusion propriety (Faruque, et al.).
2. Data is jointly gaussian with the correlation being a function of the distance (Vuran et al., 2006).
3. Data is a function for their joint entropy (Pattem et al., 2004).
4. Correlation is calculated from realistic environmental monitoring and testbeds.

We use the diffusion propriety to model the spatially correlated data (Jindal et al., 2004). The model considered in this paper is the same as in (Faruque, et al.) in which the data reading at a distance d from the center of the event is $D \propto f(1/d)$. Specifically, the data reading in any point at distance d from the center of the event is $D = C/(d+1)^{\alpha}$, where C is a constant representing the value at the center of the event, and α is the diffusion parameter which depends on the particular environment and phenomenon under surveillance, e.g., for light $\alpha = 2$, heat $= \alpha \simeq 1$.
Fig. 8 shows the data reading using the aforementioned model, with different values of α and $C = 250$. On one hand, when $\alpha \geq 1$, we observe a relatively big difference between the value at the event center of the event and the values observed at distance d far away from the center of the event. On the other hand, when $\alpha < 1 (\alpha = 0.1, 0.01$ and $0.001)$, the data readings away from the center of the event are very similar. In our study, we are interested on the type of events where data values are highly correlated.
We use Henizelman's energy consumption model (Heinzelman et al., 2002).

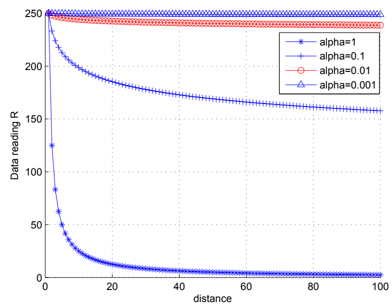


Fig. 8. Variation of data reading with distance d from the center of the event.

6.2 Classical clustering protocol

A classical clustering process is composed by two phases: set up phase and steady state phase. When an event occurs in a random (uniformly distributed) point of the network, nodes inside the event area weak up and start the clustering process. At the beginning of this phase, active nodes compete among each other to become CH. Specifically, active nodes transmit their control packet to the BS according to the specified random medium access protocol. For this protocol the NP-CSMA control protocol is used since it has been proven to be the most energy efficient protocol. The control packet only comprises the node's ID and no data are transmitted at this point. The first node that successfully transmits this packet becomes the CH. All nodes involved in the event reporting immediately send their signaling message to the BS. Therefore, the BS selects the first node that transmitted successfully the signaling message and broadcasts a signaling message over the network for a CH notification. Thus the rest of the nodes inside the event area become CMs. In the steady phase, CMs send their data in a scheduled fashion using a Time Division Multiple Access (TDMA) protocol. The CH aggregates the data values received from its CMs with its own data and sends the resulted data to the BS.

6.3 Proposed clustering protocol

The proposed clustering process is also composed of the same two phases, namely: set up phase and steady state phase. As in the classical protocol, the set up phase is triggered whenever an event occurs in a region of the network. However, in the proposed scheme, the active nodes send their first measured data value to the BS, i.e., they no longer send just their control packet. Instead, active nodes send a data packet. The reason for this is that, this sensed data is used for the CH selection procedure. Indeed, this entails an extra energy consumption at the set up phase compared to the classical protocol. However, this first data transmission allows important energy saving in the steady state phase.
It is important to notice that, our proposed scheme is best suited for environments where the event conditions are fairly stable during the event duration. This is due to the fact that the CH is chosen according to the first sensed data. Hence, if the event conditions suffer a high variation, the originally selected CH may no longer render acceptable energy savings. Example of such applications includes fire surveillance forest, in which when a fire occurs in a region, the temperature remains stationary for the duration of that fire in this region. Another example of application includes target tracking. In this kind of application, the target is the source of the measured data at sensor nodes, such as light or temperature. Here, the measured data remains the same whenever the target stays in the same place and hence the sensor nodes

sense the same measured data during the presence of the target. Next, we describe the set up and the steady phase of the proposed algorithm.

- In the set up phase, after reception of the first data packets of all active nodes, the BS calculates the difference between the data of node n_i and the data of node n_j $(i \neq j$, and $i \leq N, j \leq N)$. Next, these differences are summed over. We call this sum of the data value differences S_i. Then, the BS selects as CH the node which minimizes the total difference calculated value S_i between each node n_i and node n_j $(i \neq j$, and $i \leq N, j \leq N)$. Finally, the BS broadcasts a control message to the active nodes to notify the node selected as CH. Therefore, the rest of the nodes consider themselves as CMs. Note that there is no need for the CMs to send any extra packets since the BS already knows the active nodes.
- In the steady state phase, CMs send the difference between their sensed data and the CH's data value, which corresponds to a compressed value, called Δ_i, rather than the complete data packet, $value\_CM_i$. Therefore, $\Delta_i = |value\_CM_i - value\_CH|$ represents the difference between the *i-th* CM's data value $value\_CM_i$, and the corresponding CH data value $value\_CH$. In order to perform this compression, the CH sends its complete sample data value to the CMs at the beginning of each round. Therefore, the CMs send only the Δ_i to the CH. The main advantage of the proposal scheme is that the S_i calculation is centralized at the BS, which is not energy constrained.

6.4 Mathematical Model

In this section, the mathematical model for the clustering protocol is described. For reasons of clarity, the random access protocol is not considered in this analysis. First, because it has been studied in detail in the previous sections. Second, because we are interested on studying the effect of the compression scheme without the extra energy consumption of the collisions and idle listening and third, because the effect of the energy consumption is considered in the simulation results.

The total energy consumed in the network E_{total} for a duration of the event can be calculated as follows:

$$E_{total} = E_{competing} + E_{reporting} \tag{26}$$

where $E_{competing}$ is the energy consumed during the cluster formation phase and $E_{reporting}$ is the energy consumed during the steady state phase. We calculate hereafter $E[E_{total}]$, the average energy consumed through the network for both the classical and the proposed protocol as $E[E_{total}] = E[E_{competing}] + E[E_{reporting}]$.

6.4.1 Classical protocol

We first calculate $E[E_{competing}]$. The energy consumed at the cluster formation phase is due to the signaling packet transmission of the active nodes in the event area directly to the BS plus the reception of the signaling packet from the BS to the active nodes, then:

$$E[E_{competing}] = m \times [E_{tx}(S, R_t) + E_{rx}(S)] \tag{27}$$

where $m = N\pi(R\_event)^2/A^2$, is the average number of active nodes in the dish of radius R_event and N is the total number of nodes in the network. $S = 24bit$ is the size of signaling message, $m \times E_{tx}(S, R_t)$ is the energy consumed to sent by the m compete messages to the BS, and $m \times E_{rx}(S)$ is the energy consumed by the resulting compete message sent from the BS thought the network. On the other hand, the average energy consumption in the steady phase per event can be calculated as:

$$E[E_{reporting}] = Number\_report \times [E_{tx}(S, R_c) + (m-1) \times E_{rx}(S) + (m-1) \times E_{tx}(fixe, R_c) + (m-1) * E_{rx}(fixe) + E_{DA} \times fixe + E_{tx}(fixe, R_t)]$$

where *fixe* is the size of the full data packets of 32bits, $Number\_report = 29$ is the number of packet sent during the steady phase, $E_{tx}(S, R_c)$ is the energy consumed due to the signaling messages sent by the CH to its CMs to begin the event reporting, $(m-1) \times E_{rx}(S)$ is the energy consumed by CMs to receive this message, $(m-1) \times E_{tx}(fixe, R_c)$ is the energy consumed by the CMs to send the data to the CH, $(m-1) \times E_{rx}(fixe)$ is the energy consumed by the CH to receive the data sent by the CMs, $E_{DA} \times fixe$ is the energy consumed by the CH due to the data aggregation, and $E_{tx}(fixe, R_t)$ is the energy consumed by the CH to send the aggregated data to the BS

6.4.2 Proposed protocol

Note that, at the cluster formation phase, the proposed scheme behaves in the same manner as the classical protocol with the important difference that the nodes transmit the data packet instead of the signaling packet, then:

$$E[E_{competing}] = m \times [E_{tx}(fixe, R_t) + E_{rx}(S)] \tag{28}$$

where $m \times E_{tx}(fixe, R_t)$ is the energy consumed to send the m data packets to the BS and $m \times E_{rx}(S)$ is the energy consumed by the transmission of the compete packets from the BS to the active nodes in the network. The energy consumption in the steady state can be found as follows:

$$E[E_{reporting}] = E_{tx}(fixe, R_c) + (m-1)E_{rx}(fixe) + Number\_report[E_{tx}(S, R_c) + (m-1)E_{rx}(S) + (m-1)E_{tx}(S + log_2(E[\Delta_i]), R_c + (m-1)E_{rx}(S + log_2(E[\Delta_i])) + fixeE_{DA} + E_{tx}(fixe, R_t)])$$

where, $E_{tx}(fixe, R_c)$ is the energy consumed by the data packet transmission that is used as the reference value from the CH to the CMs, $(m-1) \times E_{rx}(fixe)$ is the energy consumed by the CMs to receive the aforementioned reference value, $E_{tx}(S, R_c)$ is the energy consumed from a signaling message sent by the CH to its CMs in order to send their data, $(m-1) \times E_{rx}(S)$ is the energy consumed by CMs to receive this signaling message, $(m-1) \times E_{tx}(S + log_2(E[\Delta_i]), R_c)$ is the energy consumed by the CMs to send the compressed data to the CH, $(m-1) \times E_{rx}(S + log_2(E[\Delta_i]))$ is the energy consumed by the CH to receive the compressed data from the CMs, $E_{DA} \times fixe$ is the energy consumed by the CH due to the data aggregation procedure, $E_{tx}(fixe, R_t)$ is the energy consumed by the CH to send the aggregated data to the BS, $E[\Delta_i]$ is the average data packet size which corresponds to the difference between the CMs' data and the CH's data. It is worth noting that considering a uniform node distribution with a large N, the node that minimizes the distance in the R_event region will be located in the center of R_event. Therefore, to calculate $E[\Delta_i]$ let us first calculate the average distance between active nodes and the CH, $E[d_{toCH}]$.

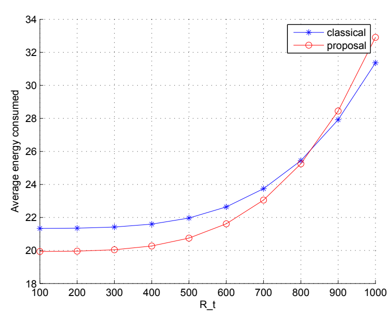
$$\int_0^{R\_event} r2\pi r dr / \int_0^{R\_event} 2\pi r dr = 2R\_event/3$$

If the density of the nodes is uniform through the R_event area. We now calculate $E[\Delta_i]$. The average data difference between the data at the CM and the reference value at the CH C is described by:

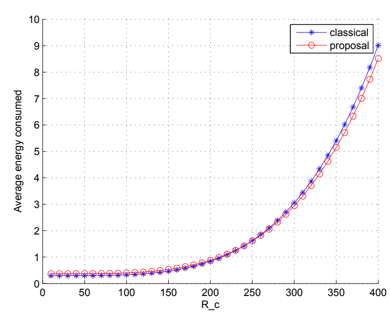
$$E[\Delta_i] = C\left|1 - \frac{1}{(1 + 2R\_event/3)^{\alpha}}\right| \tag{29}$$

6.5 Numerical Results

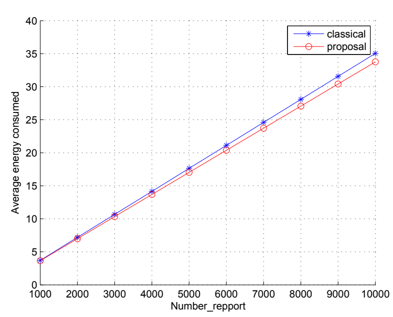
We first present the some important results derived from the analytical model. According to the previous analysis, Fig. 9(a) and 9(b) shows the average energy consumed in the network when $N = 1000$ for different values of R_t and R_c respectively. The results show that our proposal is suitable when the R_t is lower than 830$meters$ and R_c is higher than 230 meters. Exceeding these thresholds makes the competing process very costly due to the complete data packet sent to the BS during the set up phase. Remember that the classical protocol only transmits a control packet in this phase. Therefore the proposed protocol has a higher energy consumption when the distance from the cluster to the BS is high.



(a) Varying R_t.



(b) Varying R_c.



(c) Varying Number_report.

Fig. 9. Analytical results of the energy consumption.

Fig. 9(c) show the average energy consumed in the network varying the $Number\_report$ parameter. Here R_t and R_c are set to 400m and 100m respectively. The result shows that significative energy saving can be achieved by increasing the number of reports sent from the CMs to the CH. For the simulation results, we use TinyOS (Levis et al., 2003) as a simulation tool. The parameters used for this set of results are as follows: Signaling packet length (S) = 24 bits, Data value at the center of the event (C) = 250°, Initial energy per node = 10J, T_event = 200 sec, R_event = 60m, R_c = 100m, and R_t = 400m.

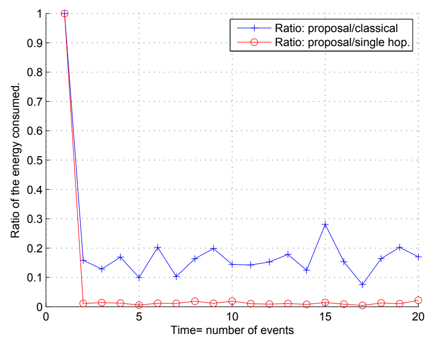
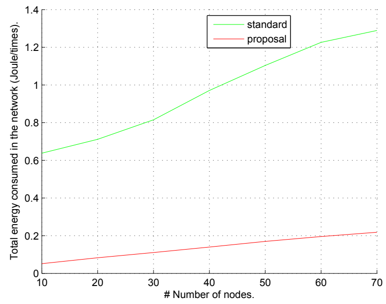


Fig. 10. Average energy consumed in time for direct transmission.

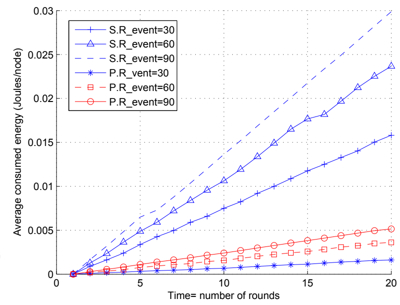
As in the Continuous Monitoring applications, we are interested on investigating the system performance under a cluster based architecture compared to the case where sensor nodes directly transmit to the BS. In order to explore the benefits of the clustering architecture, a scenario where all the nodes transmit directly to the BS is presented with the following modifications to the proposed scheme. All active nodes transmit their initial packet to the BS in order to choose the reference node (note that in this case there is no CH). Then, the BS selects the node that minimizes the data difference as explained in the previous section and then transmits a control packet indicating the ID of the reference node. Following this, in the steady state phase, the active nodes only transmit their difference Δ_i directly to the BS. The results presented in Fig. 10 clearly demonstrate that the proposed scheme conserves more energy compared to the classical scheme. Also, it is clear that the choice of the clustering scheme offers more energy savings than the single hop scheme. The gain ratio may reach up to 11 times more energy conservation than the classical scheme and 119 times more energy conservation than the single hop scheme, which are high benefit ratios.

Fig. 11 (a) shows the average energy consumed in the network per unit of time for different number of nodes. In this case the number of simulated events is 20. The results clearly demonstrates that our proposal outperforms the classical scheme. It can be seen that as the number of nodes in the system increases, also the energy consumption increases. Indeed, when the number of nodes in the network is high, the number of nodes that sense the event is also high. Hence, the number of packet transmissions (both control and data packets) is much higher than for the case where just a few nodes are active per event. The main reason for the better performance of the proposed protocol is that while all active nodes transmit the complete data packet in the steady phase in the classical protocol, for the proposed protocol, only the difference Δ_i is transmitted. Also note that this difference between the classical and proposed protocols increases for higher network densities. The rationale behind this is that for high network densities, the nodes are closer to each other, which in turns entails a higher correlation degree among their sensed data. This in turns renders smaller packet size. Conversely, for the classical scheme, since the packet size is fixed, a higher network density only increases the number of packets transmitted, consuming a lot of energy.

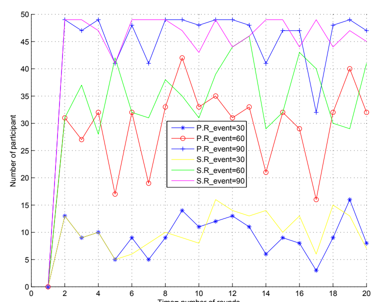
Fig. 11 (b) shows the average energy consumed for different values of R_event. When R_event is varied, also the number of active nodes per event is modified accordingly. Fig. 11 (c) shows the number of active nodes per event. It can be seen that the average number of active nodes for both the classical and the proposed scheme is approximately the same. Indeed, the proposed mechanism has no impact on the number of active nodes. Note that by increasing the



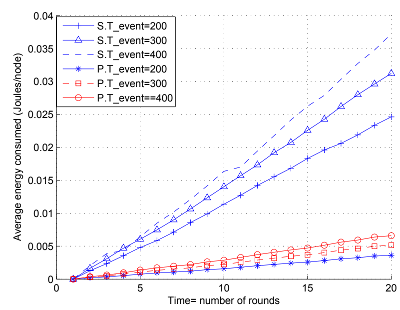
(a) Average energy consumption for unit of time vs number of nodes



(b) Average energy consumed with varying the *R_event* region



(c) Number of active nodes for each round



(d) Average energy consumption vs *T_event*

Fig. 11. Average energy consumption for unit of time for different parameters.

number of active nodes the energy consumption also increases. Observe for instance that the energy consumption when $R\_event = 30$ is less than the consumption when $R\_event = 60$ and 90. In each scenario, we observe that enabling our compression scheme reduces the energy consumption over the network and therefore extends the network lifetime.

Fig. 11 (c) shows the average energy consumed for different values of the T_event period. Increasing T_event also increases the period of the steady state phase and the number of data reported, therefore it can be seen an increase on the energy consumption. That explains why the energy consumed for $T\_event = 200$ sec is less than the energy consumed for $T\_event =$ 300 and 400 sec. In each scenario, we observe that enabling our compression scheme reduces the energy consumption over the network and therefore extends the network lifetime. It is important to note that the proposed mechanism is particularly energy efficient for high event duration times. This is due to the fact that as the event duration increases, the CMs in the classical scheme transmit many full length packets while for the proposed mechanism, the CMs also transmit many packets but with a much smaller length. This renders a slight increase of energy consumption for the proposed mechanism while for the classical scheme there is an important augmentation in the energy consumption when the event duration increases.

7. 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.

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, it is more convenient to use the optional OCM-EDR, whereas in agitated environment, it is better to use the basic CM-EDR mechanism.

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.

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.

For the event detection applications, a new cluster-based compression technique has been proposed. The clustering scheme is based on selecting the node that reduces the packet size among all active nodes in the system. The BS selects the node which minimizes the total amount of data as a CH, therefore it increases the efficiency of the compression technique by sending only the difference, rather than the complete data value to the CH. By varying different parameters of the system, simulation and analytical results conclude that considering the spatial correlation in the communication of WSNs achieves significant energy conservation compared to a classical clustering scheme. The ratio benefit may reach up to 11 times the classical scheme. The proposed scheme extends the network lifetime. In addition, an approximate mathematical model is developed which validate the results.

8. References

K. Kredo II, P. Mohapatra, *Medium access control in wireless sensor networks*, Computer Networks, Volume 51, Issue 4, 14, pp. 961–994, March 2007.

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.

Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC IEEE 802.11 Standard, 1999.

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.

N. Bouabdallah, M. Rivero-Angeles, B. Sericola, *Continuous monitoring using event-driven reporting for cluster-based wireless sensor networks*, IEEE Transactions on Vehicular Technology, vol. 58, No. 7, pp. 3360-3479, Sep. 2007.

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.

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.

F. Bouabdallah, N. Bouabdallah, and R. Boutaba, *Towards Reliable and Efficient Reporting in Wireless Sensor Networks*, IEEE Transaction on Mobile Computing, 2009.

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.

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.

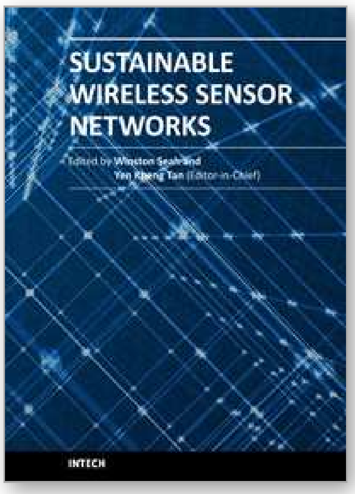
M. A.l Azim, S. Moad and N. Bouabdallah, *SAG: Smart Aggregation Technique for Continuous-Monitoring in Wireless Sensor Networks,* In Proc. of the IEEE International Comunication Conference (ICC), Cap town, May 2010.

J. Faruque and A. Helmy, *Rugged: Routing on fingerprint gradient in sensor network,* in IEEE international confrenece on pervasive Service (ICPS'2004), July 2004.

S. Pattem, B. Krishnmachari, R. Govindan and J. Heidemann, *The impact of spatial correlation on routing with coprerssion in wireless sensor networks,* in Synposium on information Porcessin in Sensor Networks," (IPSN), April 2004.

A. Jindal and K. Psounis,'*Modeling spatially-correlated sensor network data,* In the Proceedings of IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON'04), Santa Clara, CA, October 2004.

P. Levis, N. Lee, M. Welsh and D. Culler, *TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications,* In Proc. of the First ACM Conference on Embedded Networked Sensor Systems (SenSys), 2003.



Sustainable Wireless Sensor Networks
Edited by Yen Kheng Tan

ISBN 978-953-307-297-5
Hard cover, 574 pages
Publisher InTech
Published online 14, December, 2010
Published in print edition December, 2010

Wireless Sensor Networks came into prominence around the start of this millennium motivated by the omnipresent scenario of small-sized sensors with limited power deployed in large numbers over an area to monitor different phenomenon. The sole motivation of a large portion of research efforts has been to maximize the lifetime of the network, where network lifetime is typically measured from the instant of deployment to the point when one of the nodes has expended its limited power source and becomes in-operational â€“ commonly referred as first node failure. Over the years, research has increasingly adopted ideas from wireless communications as well as embedded systems development in order to move this technology closer to realistic deployment scenarios. In such a rich research area as wireless sensor networks, it is difficult if not impossible to provide a comprehensive coverage of all relevant aspects. In this book, we hope to give the reader with a snapshot of some aspects of wireless sensor networks research that provides both a high level overview as well as detailed discussion on specific areas.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821