Leyva Mayorga, I.; Rivero-Angeles, ME.; Carreto-Arellano, C.; Pla, V. (2015). QoS Analysis for a Non-Preemptive Continuous Monitoring and Event Driven WSN Protocol in Mobile Environments. International Journal of Distributed Sensor Networks. 2015:1-16. doi:10.1155/2015/471307.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 471307, 16 pages
http://dx.doi.org/10.1155/2015/471307

Research Article

QoS Analysis for a Nonpreemptive Continuous Monitoring and Event-Driven WSN Protocol in Mobile Environments

Israel Leyva-Mayorga,[1] Mario E. Rivero-Angeles,[2]
Chadwick Carreto-Arellano,[3] and Vicent Pla[1]

[1] *ITACA, Universitat Politècnica de València, 46022 Valencia, Spain*
[2] *Communication Networks Laboratory, CIC-IPN, 07738 Mexico City, DF, Mexico*
[3] *SEPI ESCOM-IPN, 07738 Mexico City, DF, Mexico*

Correspondence should be addressed to Israel Leyva-Mayorga; israle@icloud.com

Received 4 September 2014; Revised 24 December 2014; Accepted 25 December 2014

Academic Editor: Bulent Tavli

Evolution in wireless sensor networks (WSNs) has allowed the introduction of new applications with increased complexity regarding communication protocols, which have to ensure that certain QoS parameters are met. Specifically, mobile applications require the system to respond in a certain manner in order to adequately track the target object. Hybrid algorithms that perform Continuous Monitoring (CntM) and Event-Driven (ED) duties have proven their ability to enhance performance in different environments, where emergency alarms are required. In this paper, several types of environments are studied using mathematical models and simulations, for evaluating the performance of WALTER, a priority-based nonpreemptive hybrid WSN protocol that aims to reduce delay and packet loss probability in time-critical packets. First, randomly distributed events are considered. This environment can be used to model a wide variety of physical phenomena, for which report delay and energy consumption are analyzed by means of Markov models. Then, mobile-only environments are studied for object tracking purposes. Here, some of the parameters that determine the performance of the system are identified. Finally, an environment containing mobile objects and randomly distributed events is considered. It is shown that by assigning high priority to time-critical packets, report delay is reduced and network performance is enhanced.

1. Introduction

Performance in wireless sensor networks (WSNs) protocol is typically determined by its energy consumption under desired conditions, which in turn determines the network life-time (period of time in which the network is considered to be functional). This is due to the energy restriction presented in the nodes (small devices with sensing and wireless transmission capabilities, which are battery supplied). However, as power electronics evolve, other quality of service (QoS) parameters are gaining relevance, such as report delay and packet loss probability. These parameters determine the overall performance of the network. When monitoring time-critical events, such as disaster management or object tracking/detection applications, these parameters become much more relevant. If the network fails to deliver an alarm message in time, the entire environment may be put at risk, resulting in important information or monetary losses for the user. Furthermore, nodes are now capable of allocating several sensors, hence granting the ability to monitor a wide variety of physical factors, such as temperature, soil moisture, air composition, and/or movement simultaneously [1–3]. This increases network versatility but also jeopardizes performance, due to the fact that different data packets require different QoS parameters. This is specially true when both continuous monitoring and event detection applications must coexist in the same network.

In the literature it is common to find wireless sensor networks designed for either continuous monitoring (CntM) [4, 5] or event-driven detection (EDD) applications [1, 6, 7]. EDD WSNs are designed in order to send data in a sporadic fashion whenever a group of nodes detect an event of interest. Conversely, in CntM networks, all nodes in the system are set to transmit data periodically to the sink node; as such, the end user always has updated data from the surveyed region. Applications where both CntM and EDD are required have been largely overlooked. In EDD WSNs [6], communications are only triggered by the occurrence of a prespecified type of event, which typically comprises a sensor measurement that exceeds a determined threshold, which occurs due to unusual conditions inside the network area. In these environments, it is advisable that data packets containing specific events are transmitted faster than the rest of the packets. These event reports may contain, for example, alarm messages related to explosions, chemical leaks, floods, or earthquakes and therefore represent high priority events. On the other hand, other types of events, such as a small increase in some parameters, do not require immediate transmission since they may represent slightly unusual or long-term conditions such as rain occurrence, environmental temperature increase/decrease, or pressure changes. The studied protocol in our previous work [8] presents a cluster-based architecture with EDD and CntM capabilities. As such, while no event is detected, the nodes transmit data in a periodical fashion to the sink node in order to examine the evolution of certain parameters and obtain tendencies. This is performed using a TDMA-like protocol in a clustered architecture. Once an event is detected, the nodes inside the event area shift to contention mode in order to send the event-related information as soon as possible using a random access protocol. The main goal of this protocol is to transmit event data efficiently while sending data periodically, so the network administrators are aware of the adequate operation of the network. By limiting the CntM transmissions, network lifetime and energy consumption are enhanced. Furthermore, when a multievent environment is considered, setting a higher transmission probability to time-critical packets significantly, improves QoS. A previous study is now extended by proposing a simple Markovian model in order to study energy consumption and packet latency in a hybrid WSN (where both CntM and EDD are considered). Also, a priority scheme that allows the transmission of different event-related packets by means of a random access (RA) protocol is proposed and studied. As such, the priority scheme uses different transmission probabilities in order to guarantee a lower report latency in packets from a more important event. Finally, the case where the events are triggered by a mobile target is studied in the context of the hybrid priority-based protocol.

The rest of the paper is organized as follows: Section 2 reviews the previous work related to priority-based and mobility-aware protocols in the context of WSNs, followed by the network model and assumptions described in Section 3. In Section 4, the proposed protocol is described, and in Section 5 it is mathematically analyzed using a discrete-time Markov chain. Section 6 presents some relevant results derived from simulations and the analytical model for several scenarios, including randomly distributed events, mobile environments, and a multievent environment, where object tracking and environmental monitoring are performed. The paper concludes with a summary of results and conclusions.

2. Related Work

Hybrid protocols, such as APTEEN [7], have been proposed for the transmission of both CntM and EDD applications. However, only randomly generated events have been considered for performance analysis purposes. In [9], a transport layer protocol is developed, in which priority is based on the information content of each data packet. Whenever a node receives several packets from its neighbors, it rearranges those packets according to their priority level, thus sending the high priority packet first. Note that the packets, in this case, are generated periodically and the rate is controlled by the sink node, so no collisions can occur. Furthermore, each node generates a specific type of data, which means that all data packets transmitted by a specific source have the same priority whether or not an important event is detected.

In [10], a priority handling scheme has been presented, where CHs (cluster heads) define time intervals in which only packets that contain a high priority label are able to perform channel reservation duties. In case no high priority transmission request is sent, low priority packets are able to contend for channel utilization. While this approach theoretically reduces report delay for important packets, it requires more energy per event transmission due to request-to-send (RTS) and clear-to-send (CTS) messages. Furthermore, if several nodes require using the channel during a single high priority phase, collisions are likely to occur. In PSED [11], several types of events can occur within the area of interest and priority is assigned based on the potential damage that the event represents. Whenever an event is detected, data packets are sent to the CH; then, the CH sends request messages to the mobile nodes within its transmission range, which are guided (based on each event priority) to the detection zone for aiding the event sensing nodes to transmit their packets reliably. In this particular protocol, transmissions occurring between the cluster members and the CH do not consider event priority, and hence event reporting delay is not reduced during this phase. Also, since packet transmission is carried out by the mobile nodes, report delay is highly dependent on their distance from the event and their speed. In this case, mobility in the WSN is achieved by granting mobile capabilities to a determined number of nodes (such as adding a vehicle controlled by the network administrator or the network itself). However, mobility in WSNs may be introduced by means of several other methods, such as granting the nodes the capacity of detecting mobile entities [3] (by including proximity sensors on the nodes or by attaching a radio frequency (RF) transmitter to the mobile entity of interest). In the former, mobility may be handled by defining synchronization intervals in the time frame such as in [12, 13]. This approach reduces synchronization time with the nearest CH. Conversely, little research has been conducted for QoS in static networks in charge of monitoring mobile entities with transmission-only capabilities. This type of networks may be used to report several parameters ranging from vital signs [14, 15] to alarm messages and, by representing independent mobile objects (MOs), their behavior is not controlled neither by the network nor by the user.

The main differences between the aforementioned work and the one presented in this paper are as follows: priority is assigned with a similar purpose as in [10], which is meant to improve report delay on relevant event packets that may contain time-critical data. However, priority is handled by assigning different transmission probabilities to event packets in a RA protocol operating on a slotted channel. This approach is intended to increase the frequency of high priority transmission attempts (considering that collisions occur between high and low priority packets) rather than performing channel reservation duties (which require a higher amount of transmissions per reported packet). Furthermore, an environment where independent mobile objects are distributed within the network and its presence generates high priority packets in the sensor nodes is studied. By doing this, results related to operational parameters that enhance QoS under these conditions are presented, results that have been scarcely studied in previous work.

3. System Model

In this section, the main network parameters and assumptions considered throughout the paper are described.

This work is centered on a cluster-based WSN protocol named WALTER (WSN ALTernating CntM/ED block protocol for nonpreemptive event reporting) with continuous monitoring (CntM) and event detection (EDD) capabilities. This protocol was previously presented in [8] and additional features have been included for the present study. In particular, we consider a clustering protocol similar to LEACH [4], where a certain number of clusters are formed. Each of those clusters contains a node called cluster head (CH). This node gathers the information of all cluster members (CMs) and sends it directly to the sink node. The role of nodes acting as either CHs or CMs shifts constantly throughout the operation of the network. This is done in order to avoid fast battery depletion of nodes acting as CHs, since their transmissions require high power in order to reach the sink node. Once the cluster is formed, nodes send the collected information based on either random access or continuous monitoring, depending on the generated time schedule. During CntM, data is reported periodically to the sink node using a contention-free time division multiple access protocol (TDMA). Since the nodes transmit data continuously, resources are not wasted as each time slot is used by a particular sensor node inside the cluster. Conversely, in the proposed protocol, for both the cluster formation and the event reporting phases a random access protocol based on the NP/CSMA protocol is used. Since these transmissions only occur at certain moments in the operation of the system, it is not practical to preassign resources to specific nodes for these purposes. Furthermore, it is not possible to predict which sensors would be active at any given time, due to the random nature of both, joint packet transmission and event reporting. Hence, the active nodes contend among each other in order to gain access to the medium. It is then essential to carefully select the parameters that trigger the detection and transmission for the random access protocol in order to maintain an acceptable operation of the network. This is usually determined by QoS parameters such as energy consumption and report latency. Specifically, the transmission probability assignment for high priority and low priority events is of particular interest and will be further analyzed in this work.

The following assumptions and system parameters are considered. The total number of sensor nodes in the system is $N_{\text{tot}} = 100$. 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 located outside the supervised area at the coordinate $(200, 0)$. Hence, transmissions to the sink node represent, at minimum, a 100-meter transmission; thus they are highly energy consuming. Each CH uses a distinct CDMA code to transmit the gathered data to the sink node. As such, no collisions among CHs are possible. Average report delay is an important QoS parameter and studies [9, 16] have found that multihop networks may suffer from increased report latency. Then, the basis for this study is a single-hop network. However, the mean increase in report latency due to multihop transmissions is studied to validate this statement. All sensor nodes have the same amount of initial energy. The network operates on a slotted channel, with each slot representing the required time for a data packet to be transmitted (as the transmission bit rate is 40 kbps and data packet length is 2 kb, the duration of each time slot is 0.05 s). Randomly distributed events are generated with probability $\varepsilon = 0.02$ when no node has packets awaiting for transmission and probability $\varepsilon = 0$ when nodes are still attempting any scheduled or event transmission; that is, nodes have a data buffer that can only allocate one event report. As such, whenever there is an overlap of events (multiple events simultaneously), the sensor nodes only transmit information relative to the first sensed event. This is a reasonable assumption since the occurrence of multiple events is unlikely to happen. Additionally, the use of high capacity data buffers would increment the cost of the network. Several methods for event generation are considered in order to study their impact on network performance, and each will be further explained in the following sections. Since the proposed protocol is nonpreemptive, no event-related transmission is performed during CntM phase and no CntM information is sent during EDD phase. The size of the data packet l (2 kbits) comprises the data payload, the identification field, Id, and a *type* field to specify the type of packet: event packet and CntM data packet. These packets are sent during the Steady Phase, where data collecting and transmitting are performed according to a predetermined schedule. The size of the control packet is considered to be 1 kbits, which comprises the same fields but with shorter payload. Control packets are sent during the Setup Phase, which consists in the Cluster Formation and Schedule Broadcast Phases. The energy consumed to transmit a packet depends on the length of the packet l, the path loss exponent Pl (its value is set depending on the selected channel model: $\text{Pl} = 2$ for the free space model, and $\text{Pl} = 4$ for the multipath model [4]), and the distance between the transmitter and receiver nodes, d, as in [4]. Specifically,

$$E_{tx}(l, d) = l \times E_{\text{elec}} + l \times \epsilon_{fs} \times d^{\text{Pl}}, \tag{1}$$

Table 1: Parameter setting.

Parameter	Value
ϵ_{fs}	10 pJ/bit/m^2
E_{elec}	50 nJ/bit
Idle power	13.5 mW
Sleep power	15 μW
Initial energy per node	10 J
Transmission bit rate	40 kbps

where E_{elec} is the electronics energy and $\epsilon_{fs} \times d^{\text{Pl}}$ is the amplifier energy that depends on the required transmission distance. In order to reduce the processing load in the processor of the nodes and ensure data delivery, two power levels are defined for data transmission. A high energy transmission is performed every time a CH sends data to the sink node. For calculating the required power for transmission, the CH is assumed to be located at $(0, 100)$, which is the farther coordinate inside the network from the sink, so the energy used for CH transmission is the maximum energy required to reach the sink from any coordinate within the area of interest, and hence the required energy for a CH transmission is

$$E_{txch}(l, d) = l \times E_{\text{elec}} + l \times \epsilon_{fs} \times \left(\sqrt{200^2 + 100^2}\right)^{\text{Pl}}. \qquad (2)$$

On the other hand, a low-energy transmission is performed whenever a CM sends data to its CH. In this case, cluster nodes are set to transmit their data to up to $d = 35$ m. Therefore, the energy required for every CM packet is

$$E_{tx}(l, d) = l \times E_{\text{elec}} + l \times \epsilon_{fs} \times (35)^{\text{Pl}}. \qquad (3)$$

The energy to receive a packet depends on the time the communication circuits must be enabled; hence, as the transmission rate is set to be 40 kbps, the total time required for transmission depends only on packet size, which gives

$$E_{rx}(l) = l \times E_{\text{elec}}. \qquad (4)$$

It is worth noting that communication circuits in CHs must be on at all times in order to be able to receive packets from their CMs in every phase, so the consumed energy of these nodes depends on the number of time slots in each round. Each CH dissipates energy in receiving and transmitting the signals received from the cluster nodes. The steady state phase is considered to be of 20 seconds. A GB (geometric backoff) policy is employed for collision handling, with parameters τ_h for high priority and τ_l for low priority events. When considering mobility, Random Direction Mobility Model (RDM) [17] is considered due to the fact that the mobile entities tend to get a better distribution compared to Random Waypoint Mobility Model. The rest of the parameters are listed in Table 1.

Simulations in this work are conducted until one node in the network has completely depleted its energy (except when network lifetime is studied) and delay is computed as the average time needed for the transmission of every data packet generated by an event for every cluster. Figure 1 presents the basic system considered in this work. Note that a single-hop system is assumed. The rationale behind this is to avoid the use of a routing algorithm that affects the energy consumption on the system. This is because a multihop system must decide the appropriate routes to relay the packets through the multiple available choices. As such, nodes in a preferred route would consume more energy than nodes further away form the sink node. Since the objective of this work is to study the impact of the random access protocols through the use of priorities, we avoid the aforementioned effects of the selected routing protocol. Specifically, it would be very difficult to know if a certain node consumed its energy due to the use of the priority scheme or due to the fact that it was in a high traffic route in the network. Finally, the use of multihop schemes entails a higher packet delay compared to single-hop schemes. This issue will be explained in detail in a future section. Additionally, many routing protocols have been proposed in the literature specifically aimed at WSNs applications and the selection of one of these is not a trivial task. As such, we leave this issue for future work since it is outside the scope of this paper.

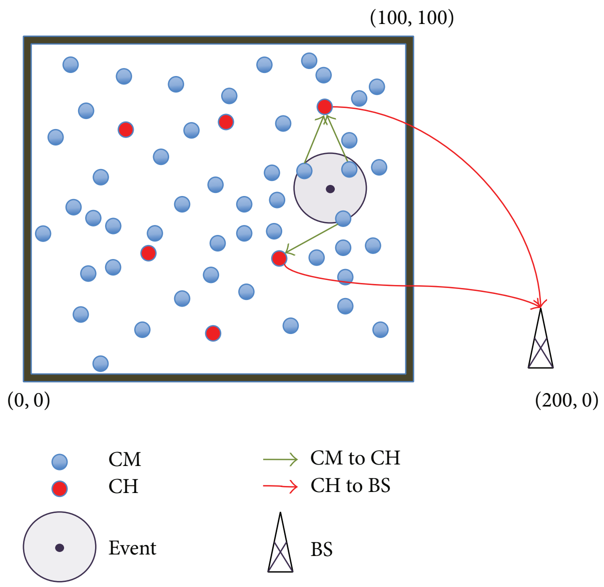


Figure 1: Basic system model for performance analysis.

4. WALTER (WSN Alternating CntM/ED Block Protocol for Nonpreemptive Event Reporting)

Figure 2 presents the basic operation of WALTER, a hybrid protocol with CntM and EDD capabilities.

First, clusters are formed in the CF (cluster formation) interval according to [4]. Once the CF is over (when all nodes are either CMs or CHs and all nodes belong to a particular cluster), event detection is performed using a random access (RA) protocol, and it is denoted by the ED period. Whenever an event packet is received by the CH, it is transmitted to the sink node using a particular CDMA code during the next

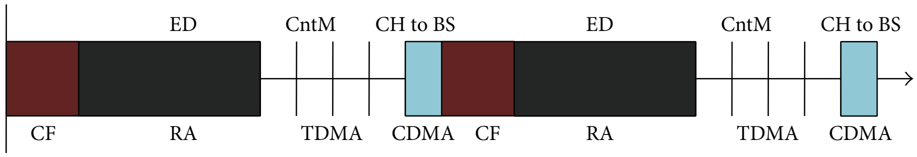
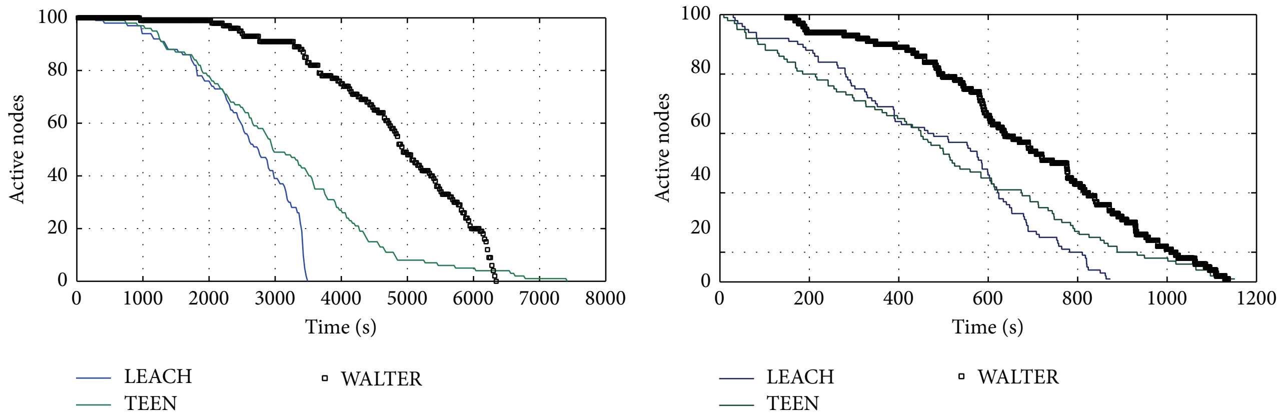


Figure 2: Time frame for WALTER.



(a) Pl = 2

(b) Pl = 4

Figure 3: Network lifetime comparison between two characteristic WSN protocols and WALTER.

time slot. This is in order to avoid collisions between clusters. When the scheduled time for random access is over and no more cluster members have event data for transmission, TDMA is performed. It is worth noting that TDMA starts only if every node within a cluster has no event packets left for transmission; in case that some event packets are awaiting transmission, TDMA is delayed. This approach is used because we consider that, in a hybrid protocol, event reporting has higher priority than CntM data. At the end of the TDMA interval, the CHs transmit the gathered information to the sink node. This process is repeated for 20 seconds, when the scheduled time for the next CF phase is reached.

The benefits of using a hybrid protocol capable of the transmission of event-related and CntM data, as opposed to using either an EDD or CntM specific protocol, have been studied in [8]. Particularly, a hybrid protocol presents a larger network lifetime than LEACH and TEEN when operating in a low-rate event detection environment and when a CntM-like behavior is needed. Network lifetime for an environment with randomly generated events for LEACH, TEEN, and WALTER with disabled priority is shown in Figure 3.

Figure 3 also illustrates the effects that different values of Pl have on the performance of the system. Specifically, the values of Pl = 2 and Pl = 4 are considered. Note that considering a path loss exponent of Pl = 4 results in an immense increase in the required energy for the transmission of a packet compared to the case where Pl = 2. Indeed, a high path loss exponent is closer to the channel conditions found in many practical environments. However, both simulations and numerical methods used to solve the analytical models take considerably more time to perform. As such, the rest of the paper considers a value of Pl = 2. Also consider that the general behavior of the studied protocols is the same for both path loss values. For instance, note that in both cases WALTER outperforms LEACH and TEEN, and the main difference between both models is the extra energy consumption in the system. Finally, a low value of the path loss exponent is closely related to open and obstacle free scenarios, such as the ones considered for mobile detection applications, which is the focus of this work. The study of WALTER is now extended, as we now consider the case when this protocol is capable of detecting events with different priority levels. Event detection is performed using a double sliding window scheme [18] in order to limit transmissions generated by each event. This implies that transmissions are triggered only at the first slot in which an event occurs (in which a sensor lecture increases abruptly or surpasses a specific threshold defined by the user, which may be related to the accuracy level in the specific application). Hence, for a new event-related transmission in a specific node to be triggered, there must be at least a time slot in which no event is detected. Once an event has been detected, priority is assigned using a detection scheme where two threshold values are selected. In case only the first threshold is exceeded, the detected event is labeled as low priority. However, if both thresholds are exceeded, the event is considered to be of high priority. These thresholds can be established at any time by the network operator according to its particular interests. The rationale behind this is as follows. Typically, whenever an event occurs, nodes closer to the coordinates of the origin of the event present a higher value in their sensor lectures and have more detailed information about the event. As such, it is assumed that this data should be transmitted as soon as possible to the sink. On the other hand, nodes that still detect the event but with a

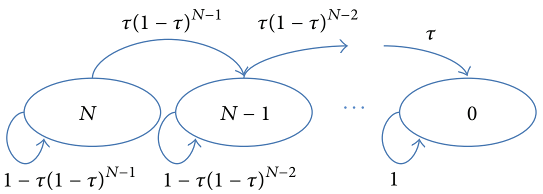


Figure 4: Markov model for nonpriority event reporting.

lower intensity are usually farther from the origin of the event and their data can be transmitted afterwards.

5. Analytical Model

Given the WALTER protocol, a Markov model [19] is used to analyze energy consumption and report delay depending on the average number of nodes attempting transmission. First, the case where no priorities are enabled is considered. The analysis that follows is the same for the CF or event reporting phases. In the former, we start with N nodes, each one transmitting with probability τ_c. In the latter, the corresponding values are N_e and τ_e. Denoting these parameters generically by N and τ (values for τ are selected before deploying the network), let S_n denote the number of sensors that transmit when there are n nodes with a pending event or joint packet transmission. S_n is a binomial random variable with parameters n ($0 \leq n \leq N$) and τ. Then, $P(S_n = j) = \binom{n}{j}\tau^j(1-\tau)^{n-j}$ and $E(S_n) = n\tau$.

The aforementioned system can be modeled as a discrete-time Markov chain, W, depicted in Figure 4, where the states represent the number of nodes that have not yet successfully transmitted their packet.

The state space of W is thus $\{N, N-1, \ldots, 1, 0\}$, with $W(0) = N$. Denoting $p_n = P(S_n = 1) = n\tau(1-\tau)^{n-1}$, for $n \geq 1$, the nonzero transition probabilities are $P(W(t+1) = n-1 \mid W(t) = n) = p_n$ and $P(W(t+1) = n \mid W(t) = n) = 1 - p_n$. To these we add $P(W(t+1) = 0 \mid W(t) = 0) = 1$ (i.e., 0 is an absorbing state).

The time (number of time slots, each lasting 0.05 s) that W spends in state n, x_s, is geometrically distributed: for any state $n \geq 1$ and for $m \geq 1$, $P(T_n = m) = (1-p_n)^{m-1}p_n$. The mean time that the system remains in state n is thus

$$E(T_n) = \frac{1}{p_n}. \tag{5}$$

Therefore, the mean time to form the cluster is

$$E(T_{\text{cluster}}) = \sum_{n=1}^{N}\left[n\tau_c(1-\tau_c)^{n-1}\right]^{-1}. \tag{6}$$

On the other hand, the average event reporting time of all the nodes that sense the event is

$$E(T_{\text{event}}) = \sum_{n=1}^{N_e}\left[n\tau_e(1-\tau_e)^{n-1}\right]^{-1}. \tag{7}$$

Finally, the average energy required for the transition of the system from state n to $n-1$, $E(E_n)$, is computed based on the energetic cost of a transmission attempt, E_{tx}, the required energy for receiving a packet, E_{rx}, and τ. As stated above, x_s is the number of time slots required for transition. Finally, γ is the energetic cost of a failed attempt. Then,

$$E(E_n) = \sum_{j=1}^{\infty} E\left[\frac{E_n}{x_s} = j\right] P(x_s = j), \tag{8}$$

where

$$E\left[\frac{E_n}{x_s} = j\right] = \gamma(j-1) + \left[E_{tx} + (n-1)E_{rx}\right], \tag{9}$$

which gives

$$E(E_n) = \gamma\left(\frac{1-p_n}{p_n}\right) + E_{tx} + (n-1)E_{rx}, \tag{10}$$

where the energetic cost of a failed transmission attempt for each time slot, γ, is

$$\gamma = (E_{tx} - E_{rx})\left(\frac{n\tau - p_n}{1-p_n}\right) + nE_{rx}. \tag{11}$$

By substituting (11) in (10) we obtain the formula for the required energy for the transition from state n to $n-1$ as follows:

$$E(E_n) = \frac{n((E_{tx} - E_{rx})\tau + E_{rx})}{p_n} = \frac{(E_{tx} - E_{rx})\tau + E_{rx}}{\tau(1-\tau)^{n-1}}. \tag{12}$$

This model may be used to calculate energy consumption for both event transmission and cluster formation by adding the consumed energy for each n. As the process of cluster formation requires only one transmission from the CH, the total energy required to form a cluster is

$$E(E_{\text{cluster}}) = E_{txch} + \sum_{n=1}^{N}\frac{(E_{tx} - E_{rx})\tau_c + E_{rx}}{\tau_c(1-\tau_c)^{n-1}}. \tag{13}$$

For event reporting, the CH is set to transmit to the BS (base station) each time a packet containing event data is received; therefore,

$$E(E_{\text{event}}) = NE_{txch} + \sum_{n=1}^{N_e}\frac{(E_{tx} - E_{rx})\tau_e + E_{rx}}{\tau_e(1-\tau_e)^{n-1}}. \tag{14}$$

As stated in Section 3, a single-hop network is considered. This is due to the fact that multihop delivery has been found to present a higher latency when compared to single-hop [9, 20]. To evaluate this statement, consider the following assumptions.

(i) While on TDMA CHs are unable to send data directly to sink, thus assuming that these nodes are able to receive a relay request in these phases, they must wait for RA to send the requested message. The number of TDMA slots left is denoted by ts.

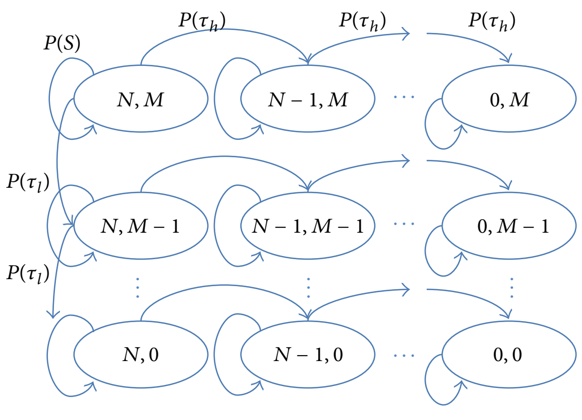


FIGURE 5: Markov model for high and low priority event reporting.

(ii) During steady state phases, a same number of time slots are used for both TDMA and RA. Therefore, the probability of requesting a packet retransmission by a specific CH while being busy (performing TDMA) is $P(\text{TDMA}) = 1/2$.

(iii) The desired number of clusters is set to 5; then, as $N_{\text{tot}} = 100$, each cluster has a mean number of CMs, $\overline{N}_{\text{cm}} = 19$. Then, the average length of TDMA and RA phases is $E(T_{\text{TDMA}}) = \overline{N}_{\text{cm}} + 1 = 20$.

(iv) As the clusters operate independently, the probability of receiving a relay request in each time slot is uniformly distributed. Then, the average increase in report delay due to a n-hop transmission is

$$\begin{aligned} E\left(\Delta T_{nh}\right) &= nh \times \left[P\left(\text{TDMA}\right) \times \sum_{ts=1}^{\overline{N}_{\text{cm}}} \frac{ts}{E\left(T_{\text{TDMA}}\right)} + 1 \right] \\ &= nh \times \left[\frac{P\left(\text{TDMA}\right) \overline{N}_{\text{cm}} \left(\overline{N}_{\text{cm}} + 1\right)}{2E\left(T_{\text{TDMA}}\right)} + 1 \right] \\ &= nh \times \left[\frac{P\left(\text{TDMA}\right) \overline{N}_{\text{cm}}}{2} + 1 \right] = \frac{23}{4} nh. \end{aligned} \tag{15}$$

Here, nh is the number of hops required to reach the sink node. Since we assume that there are packets with high priority to be delivered, the use of a multihop scheme is not suitable for the specific applications considered in this work.

A bidimensional Markov model as the one shown in Figure 5 is used for the case when nodes are able to detect high and low priority events. Here, N represents the number of nodes detecting a high priority event and M represents the number of nodes that detect the low priority event. Recall that nodes that detect a high priority event transmit with probability τ_h, while nodes that detect a low priority event transmit with probability τ_l. Building on this, the system begins at state $W(0) = (N, M)$ and the state $(0, 0)$ is an absorbing state. The nonzero transitions probabilities at state (i, j) are as follows:

(i) to state $(i-1, j)$ with probability $P_s^h(i, j) = i\tau_h(1-\tau_h)^{i-1}(1-\tau_l)^j$,

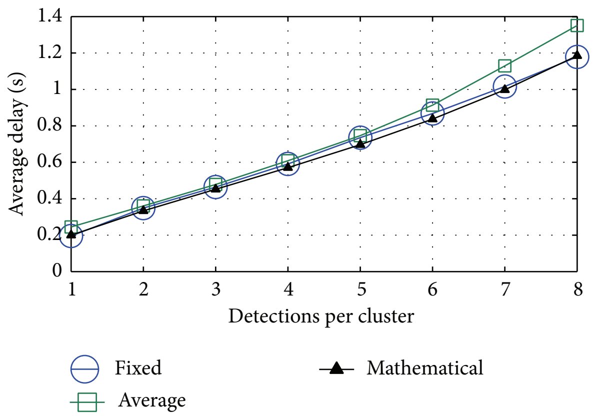


FIGURE 6: Average event transmission delay in a nonpriority environment for different values of reporting nodes per cluster.

(ii) to state $(i, j-1)$ with probability $P_s^l(i, j) = (1-\tau_h)^i j\tau_l(1-\tau_l)^{j-1}$,

(iii) to state (i, j) with probability $1 - P_s^h(i, j) - P_s^l(i, j)$.

Let $V_{i,j}$ represent the average time it takes for the chain to go form state (i, j) to state $(0, 0)$. This can be calculated through the following relationship:

$$V_{i,j}\left[P_s^h(i,j) + P_s^l(i,j)\right] - V_{i-1,j}P_s^h(i,j) - V_{i,j-1}P_s^l(i,j) = 1. \tag{16}$$

Specifically, we are interested in finding the average time that both the nodes that sense the high and low priority events report their data. Hence, we numerically solve (16) for $i = N$ and $j = M$.

Parameters section at the end of the paper depicts the variables used for analyzing WALTER by means of the described Markov model.

6. Numerical Results

6.1. Static Environments: Priority Disabled. QoS analysis is conducted using the Markov model described in Section 5 and a discrete-time event simulator developed in C++. The proposed Markov model is first used to validate simulations when no priorities are assigned; hence the transmission probability per time slot in each node during RA phases is $\tau = 0.25$.

Figure 6 shows the average delay for event packets with different values of detections per event in each cluster. In this case, three ways for achieving event detections for each cluster were considered, as shown in Figure 6: (a) a deterministic approach, called *Fixed* strategy, where a certain number of nodes per cluster are set to perform event detection duties, and then when every node in a cluster is affected by an event, μ detections are triggered; (b) a probabilistic approach, called *Average* strategy, where a number of reporting nodes are chosen randomly and where a random number of nodes uniformly distributed in the ranges 0 and 2μ detect and report the event; and (c) a probabilistic approach, named *Radius* strategy, where events are generated by means of a bidimensional random variable for the event center, $(X_{\mathrm{ev}}, Y_{\mathrm{ev}})$, and a fixed detection radius, R.

In the *Fixed* strategy, it is assumed that the network manager decides the number of nodes that report their sensed data whenever an event occurs (given that the event affects the entire cluster); for example, 10% of the nodes in the cluster may report the event. This represents an efficient option in order to limit the amount of transmissions in the network, but the accuracy of the reported event may be reduced. The specific protocol to select these nodes is not straightforward and we leave this issue for a future work. For the *Average* strategy, each CH broadcasts the value of a uniformly distributed variable $U(0, 2\mu)$ during the setup phase, which determines the percentage of CMs that may report the event. Hence, the number of reporting nodes for each Steady Phase may vary. Note that the mathematical analysis developed above closely matches the *Fixed* strategy results. Indeed, for a deterministic number of nodes there is no variation in the number of reports per event. However, for the *Average* and *Radius* strategies, the number of reporting nodes for each event is selected randomly. Hence, also the reporting delay varies. It is important to notice that this difference is more important for a high number of detections per cluster, where there is a difference of approximately 16% between the mathematical and *Average* results.

By comparing the results in Figure 6 with (15), it may be seen that the expected increase in report delay due to multihop transmissions is significant. In fact, it is higher than the average delay for successfully reporting an event with a single detection, even for $nh = 1$. Then, despite the fact that (15) represents average delay increase for a single event packet, it illustrates that multihop is not a time-efficient strategy for critical-time applications. As a result of this, single-hop transmission is used for the rest of the paper.

Building on this, we study the effect of the event detection radius. Here, uniformly distributed events are generated within the area with center $(X_{\mathrm{ev}}, Y_{\mathrm{ev}})$ and radius R; that is, an event occurring at a distance lower than or equal to R from the node is detected. For this, the distribution of the simultaneous detections for each event radius is calculated. From there, the average number of nodes detecting each event within a cluster is computed. Note that this approach is different from the *Average* strategy, where the average number of nodes reporting the event μ is determined a priori by the user. Figure 7 shows the mean number of simultaneously triggered detections within a cluster for different values of R.

In order to adequately calculate average report delay for randomly distributed events, the proposed Markov model must include the probability of n nodes detecting the event. Then, (7) becomes

$$E\left(T_{\text{event}}(R)\right) = \sum_{n=1}^{N_e} \left[p_d(n) \times n\tau_e \left(1 - \tau_e\right)^{n-1}\right]^{-1}. \qquad (17)$$

By using (17), the error obtained between simulation and mathematical results is acceptable, as shown in Figure 8. However, computing the probability distribution of detecting nodes, $p_d(n)$ is not straightforward, specially for environments containing events with a higher complexity.

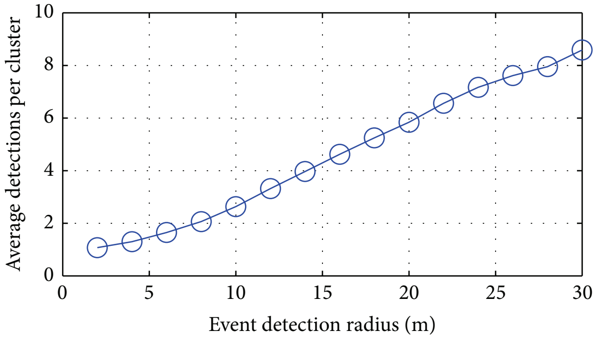


Figure 7: Average detection for each generated event in a nonpriority environment with uniformly distributed events with radius of detection R.

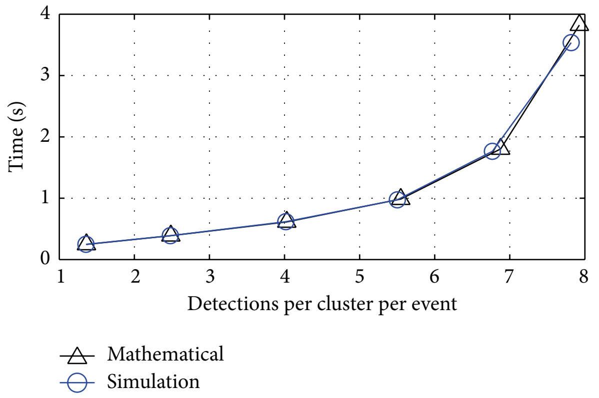


Figure 8: Average event report delay in a nonpriority environment with uniformly distributed events with radius of detection R.

Regarding the detection strategies, it can be seen that the *Fixed* strategy achieves a lower reporting delay than the *Average* and *Radius* strategies. As such, we focus on this strategy for the rest of the results presented in this section, as well as the basis for the mathematical model presented in the previous section, when randomly distributed events are considered. As expected, the average delay increases along with the number of simultaneously triggered transmissions per cluster. It is worth noting that, in order to achieve an adequate performance, a suitable threshold(s) value for event detection duties must be selected. Specifically, when set to a higher value than needed, several events may be overlooked and, when set to a lower level than required, network congestion is likely to occur due to an increased rate of event detections and transmissions. Therefore, the network manager has to carefully select this threshold value. This is of special importance as energy consumption in contention mode highly depends on the number of simultaneously triggered transmissions for both cluster formation and event reporting phases. Figure 9 shows the total energy consumption per event report for different values of nodes detecting the event and a close match between the simulation and mathematical results is observed.

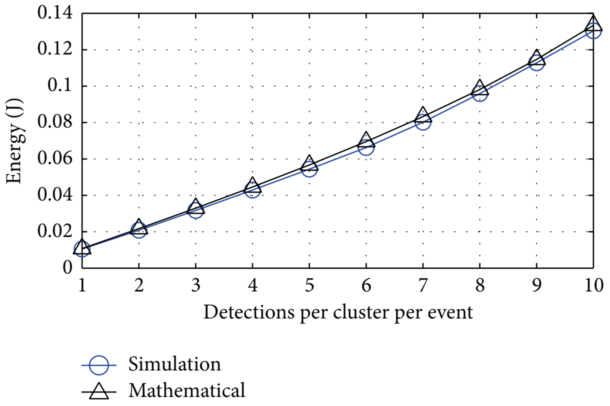


Figure 9: Average energy consumption per event transmission in a nonpriority environment with uniformly distributed events.

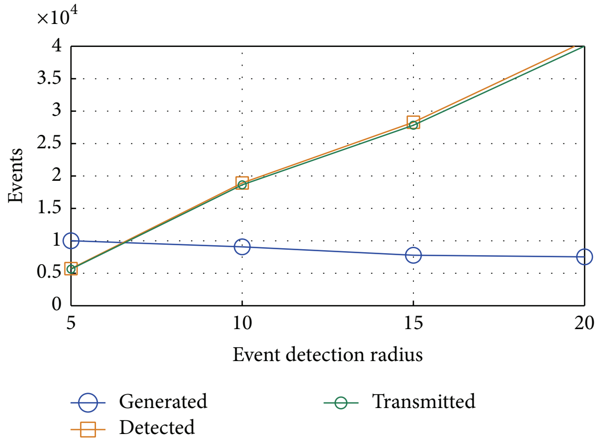


Figure 10: Network activity for a nonpriority environment with uniformly distributed events with radius of detection R.

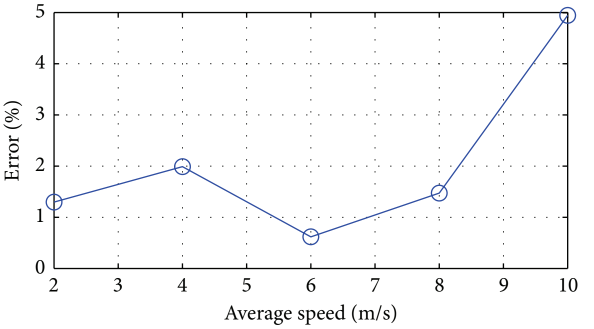


Figure 11: Error percentage when comparing mathematical and simulation results for the hitting time in a network with one mobile object (MO) with detection radius $R = 5$ m, $\overline{L} = 52.14$ m, $\overline{T}_{\text{stop}} = 2$ s, and $N = 100$.

Specifically, the error between mathematical and simulation results is lower than 5% for all cases, which validates the energy consumption model.

Also, this figure shows that energy consumption is not a linear function of the number of simultaneous detections. As such, the network user must be aware of the compromise between reliability and energy economy, achieved by selecting the number of nodes per cluster that perform event detection duties, μ.

When studying the *Radius* strategy, this behavior gets intensified. In Figure 10, several event detection radiuses are set by modifying the threshold value at the input of the sensors of the nodes; that is, for achieving a detection radius of 5 m, the threshold is set to a higher value and for the 20 m radius it is set to a lower value.

From this figure it can be inferred that, by increasing the threshold value, the detection radius decreases, so the probability that an event is overlooked is higher. On the other hand, as the threshold value gets lower, the event detection radius increases along with the probability of generating redundant data. Also, the number of detected and successfully transmitted events is almost identical, which implies that packet loss probability is near to zero. Then, transmission probability is guaranteed, even when operating in environments with high rate of event occurrence.

6.2. Mobile Environments: Priority Disabled. In case that detections within the area are generated by movement or the presence of mobile objects (MO), a nonpriority environment is considered. In this scenario, object position or status is transmitted by means of the RA protocol and environmental parameters are reported during the collision-free TDMA schedule. By using (18) presented in [17], it is possible to determine the expected hitting time (average time between mobile object detections within the network area) for the proposed system. Consider

$$ET_{rd} = \left(\frac{N_{\text{tot}}}{2 * R * \overline{L}}\right)\left(\frac{\overline{L}}{\overline{v}} + \overline{T}_{\text{stop}}\right). \quad (18)$$

Here, N_{tot} is the total number of nodes forming the network, $\overline{L}$ is the mean length of the *epoch*, $\overline{v}$ is the average speed of the MO, and $\overline{T}_{\text{stop}}$ is the mean time the MO remains static for each movement stage. While (18) was proposed for ad hoc networks, the high count of nodes in the proposed WSN and their uniform distribution throughout the network present a similar scenario to that of an ad hoc network.

For validation purposes, hitting time is obtained by simulation and (18), the error obtained when comparing both results, is more than acceptable as shown in Figure 11; then, both methods are considered suitable for calculating hitting time for the selected environment.

Building on this, expected hitting time may be calculated for several values of the average speed of the MO, $\overline{v}$, and detection radius in the sensors by using (18). Results are shown in Figure 12.

As expected, hitting time decreases as the mean speed of the objects, $\overline{v}$, or detection radius, R, increases. For further research, these results may be used to estimate power consumption based on the average number of detections per second, $\text{EDPs}_{rd} = 1/ET_{rd}$, and average energy consumed for reporting an event with 1 simultaneous detection. It is worth noting that (18) has been used when considering one mobile

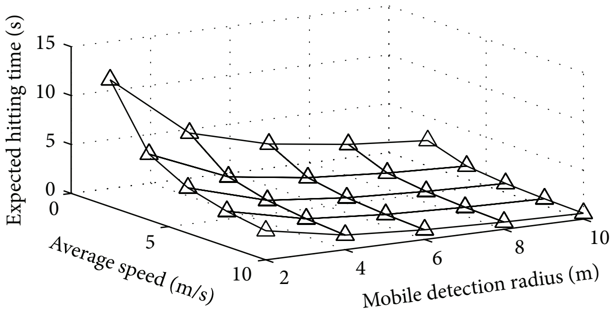


Figure 12: Expected hitting time for different $\overline{v}$ and R obtained with (18).

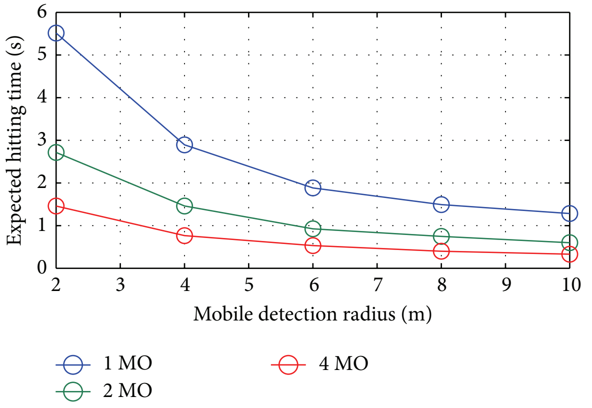


Figure 13: Expected hitting time when considering a number of mobile objects (MO) and detection radiuses with $\overline{v} = 5$ m/s.

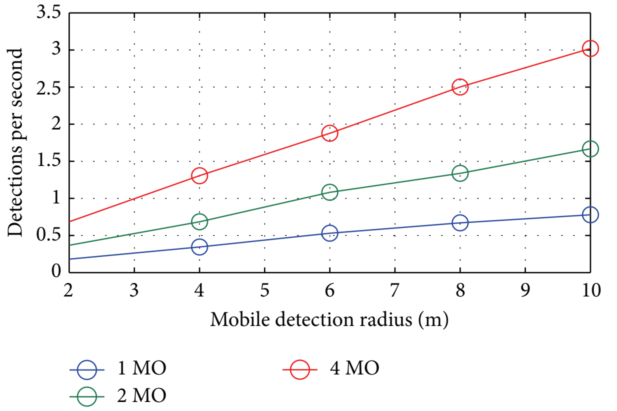


Figure 14: Detections per second when considering a number of mobile objects (MO) and detection radiuses with $\overline{v} = 5$ m/s.

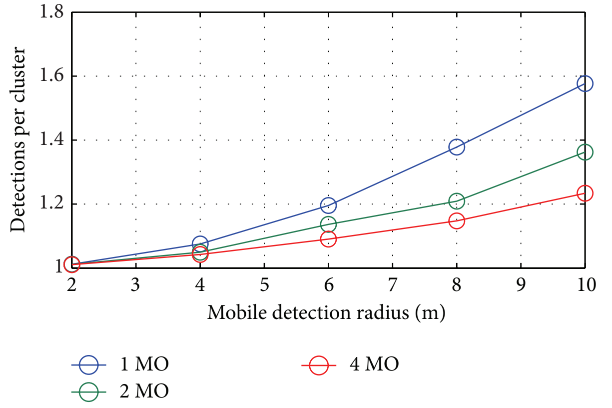


Figure 15: Detections per cluster when considering a number of mobile objects (MO) and detection radiuses with $\overline{v} = 5$ m/s.

object within the area of interest. However, when several mobile objects are included, its use may not be accurate. In Figure 13 several MOs are considered and a direct relation between hitting time for 1, 2, and 4 MOs is not observed.

This behavior occurs due to the fact that when a node has detected an event it is unable to perform a second detection until the event packet has been successfully transmitted. Thus the rate of detection within the simulated network was decreased slightly, an scenario not considered by (18). Building on this, an equation for obtaining the expected hitting time for n mobile nodes is left for future work and, thus, results for environments considering several MOs are obtained by simulation.

Figure 14 shows that as detection radius increases, the detections per second increase, which implies a proportional increase in energy consumption, given that the number of simultaneous detections per cluster remains constant. As the probability of a node entering the detection radius of several nodes in the exact same time slot is significantly low for the given scenario, detections are considered to be simultaneous if a detection is performed while any other node in the cluster is attempting the transmission of a previously detected MO.

Nevertheless, as Figure 15 shows, a significant increase in detections per cluster is not observed; in fact, as more mobile objects are added, this value decreases slightly. This occurs due to the fact that as more objects transit the area, the probability that more than one MO is within the detection range of a node increases. But also, as MOs are added to the network, detections are less spatially concentrated and more spread throughout the network.

It can also be seen from Figure 15 that, as the average speed of the MOs increases, the detections per cluster increase. However, that increase is not significant, which also implies that average report delay is not expected to increase.

This can be seen in Figure 16, where average report delay for an environment containing a single MO is considered.

As stated above, adding MOs to the network does not increase simultaneous detections, so average report delay is expected to remain constant for these scenarios. While report delay is not significantly affected by neither detection radius nor average speed nor the number of objects inside the area of interest, other QoS parameters may be affected, such as the energy consumption in the system.

Due to the fact that RA transmissions are high energy consuming, an increase in the rate of detection of the events would imply an increase in power consumption; this behavior is shown in Figure 17.

It can also be observed that, in this case, energy consumption is mostly affected for scenarios with higher detections

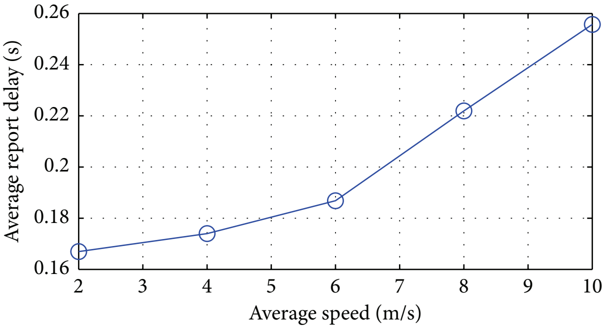


FIGURE 16: Average report delay for an environment with one mobile object for several values of $\bar{v}$.

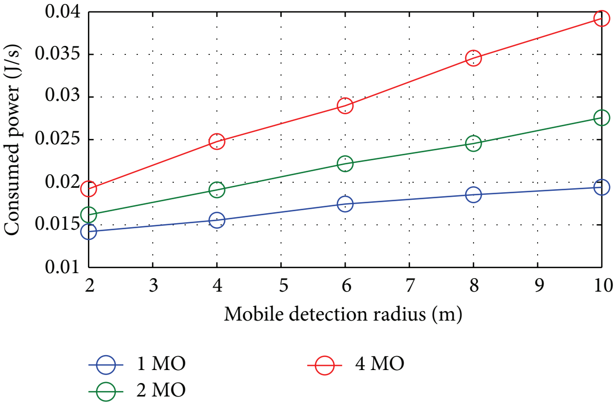


FIGURE 17: Energy consumption for the Steady Phase when considering a number of mobile objects (MO) and detection radiuses with $\bar{v}$ = 5 m/s.

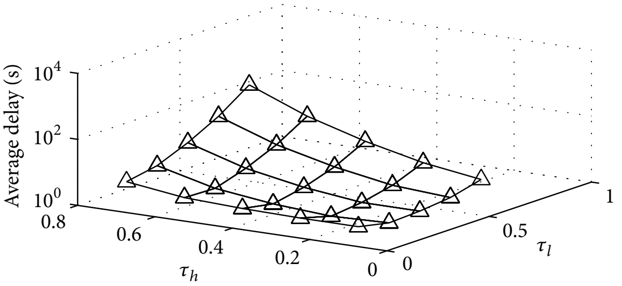


FIGURE 18: Average report delay for $N = 5$ and $M = 5$ for several transmission probabilities, obtained using the Markov model.

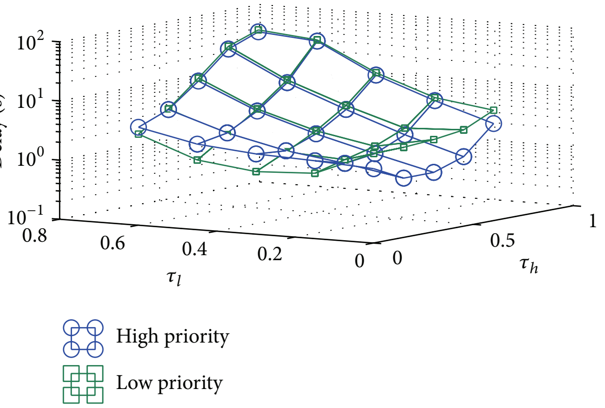


FIGURE 19: Delay improvement for high priority packets for several transmission probabilities.

per second, rather than detections per cluster. Therefore, by increasing the rate of detection, the consumed energy per time unit is also increased. Building on this, the network operator has to achieve a balance between detection accuracy and low energy consumption, which may be controlled by carefully selecting the threshold value for detection, which in turn leads to achieving the desired detection radius, R.

6.3. Static Environments: Priority Enabled. As reviewed previously in this section, randomly distributed events that affect several nodes simultaneously tend to affect report delay and energy consumption (as a result of increased rate of detection/transmission) significantly. Hence, assigning priority to certain packets is expected to enhance the performance of the system. In this work, priority is enabled by assigning different transmission probabilities, namely, τ_h and τ_l, according to the lectures in the sensor in charge of event detection. This is achieved by selecting two different threshold values. For instance, a higher transmission probability, τ_h, is selected when both upper and lower thresholds are exceeded. Conversely, a lower transmission probability, τ_l, is selected when the lower threshold is exceeded but not the upper threshold. We now investigate the effect of the values of τ_h and τ_l on the performance of the system in order to find those that increase performance for the given environment.

In Figure 18, it can be seen that the lowest overall delay is achieved when selecting relatively low values of the transmission probability: near $\tau_h = 0.3$ and $\tau_l = 0.15$. In this case, a number of triggered high and low priority detections, N and M, respectively, are set to be $N = M = 5$. This is in order to recreate a highly collided environment so the system is tested in the worst case scenario. Additionally, Figure 19 shows the average report delay for high and low priority packets. Since the main goal of the priority scheme is to offer a lower reporting delay to high priority packets, the values where $\tau_h < \tau_l$ must not be considered and are shown to demonstrate that only transmission probabilities are being modified. From this figure it is clear that, in order to achieve a lower delay for high priority packets, the appropriate values of the transmission probabilities must be set to $\tau_h = 0.3$ and $\tau_l = 0.15$. As such, these values will be used as default throughout the rest of the paper.

In order to evaluate whether the selected priority handling scheme benefits the transmission of important packets under several conditions, average report delay for high and low priority packets has been obtained for different values of M and N and is shown in Figure 20.

As expected, delay for high priority packets is significantly improved when compared to low priority packets. It is worth noting that average report delay is computed as the time

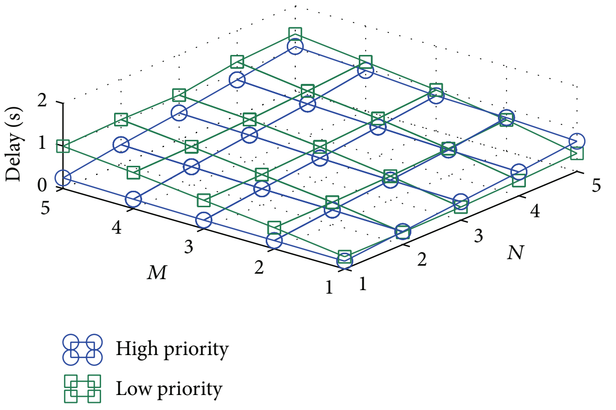


Figure 20: Average report delay for high and low priority packets for different N and M using the Markov model.

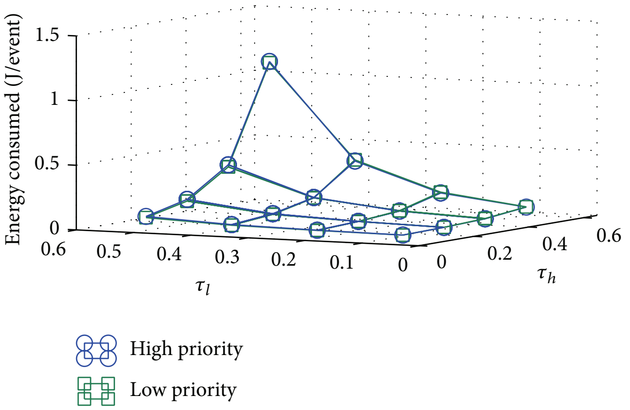


Figure 22: Energy consumed for several transmission probabilities.

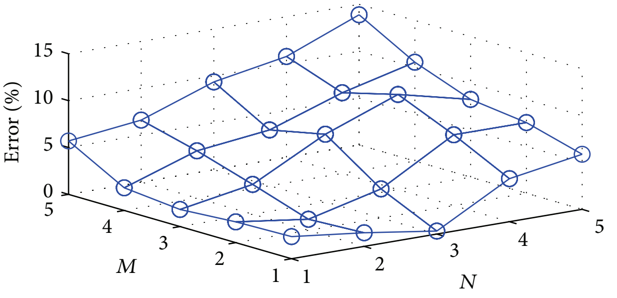


Figure 21: Error percentage for average report delay between the Markov model and simulator.

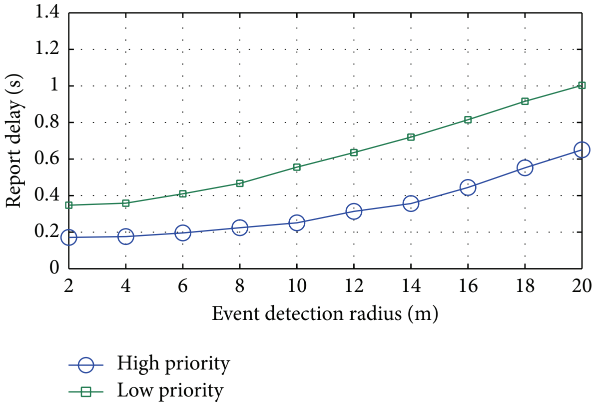


Figure 23: Delay improvement for high priority packets in a concentric detection scheme.

it takes for the transmission of every packet corresponding to each priority level within a cluster, which is the reason for high priority packets experimenting higher delay when $N > M + 2$. In this case, more high priority than low priority packets must be sent for the high priority event to be successfully transmitted. However, this scenario is difficult to occur due to the inherent nature of high priority detections. Hence, the selected priority handling scheme performs as desired. For validating the developed discrete event simulator, both analytical and simulation results regarding report delay have been compared and the obtained error is presented in Figure 21. It can be seen that, in general, there is a good match between the analytical model and simulation results. As the number of nodes in a cluster corresponds to a random variable, as $N + M$ increases, so does the probability that a cluster contains a lower number of CMs; then, when comparing delay results for N and M detecting nodes (obtained by means of the Markov model) and $\min(N, \mathrm{CM})$ and $\min(M, \mathrm{CM}-N)$, results are likely to vary, which is the reason for the increase in the error for high values of N and M. In order to adequately obtain average report delay in these cases, the probability distribution for high and low priority simultaneous detections must be calculated and used in a similar manner as in (17) for the two priority models. In order to verify that the selected priority scheme is economically adequate, energy consumption for high and low priority events is calculated for several values of the transmission probabilities, τ_h and τ_l, and shown in Figure 22.

It is observed that unless these values are set extremely high, that is, τ_h and τ_l above 0.5, which incurs in a highly collided environment, consumed energy per transmission is not significantly affected. Furthermore, the consumed energy by high and low priority transmissions is almost identical. Then, for achieving better performance for both, energy consumption and report delay, lower values for transmission probabilities must be used, specifically, values near $\tau_h = 0.3$ and $\tau_l = 0.15$ as suggested previously. Finally, results for the case where randomly distributed events with detection radius R are considered are shown in Figure 23.

In this case, it is considered that nodes near the event center (selected by means of a bidimensional uniform random variable) are assigned high priority. Specifically, detections within $R/2$ from the selected coordinate are set to be high priority detections. On the other hand, detections between $R/2$ and R are assigned a low priority label. Recall that the rationale behind this priority scheme is that some environment monitoring applications [1] like wildfire [2] and moisture or seismic monitoring may affect large areas. However, their intensity fades as the distance from the initial occurring site increases. As observed, by using the selected

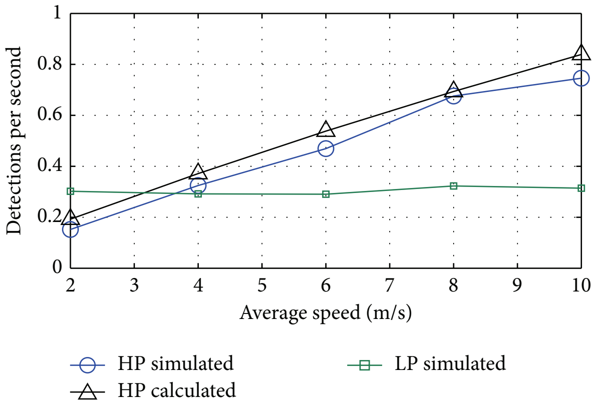


FIGURE 24: Detections per second for high and low priority packets in a hybrid environment.

values for τ_h and τ_l, delay in high priority packets is reduced and shows better tolerance to situations where a high number of transmissions are triggered simultaneously. In this case, it is assumed that every transmission related to an event detection, regardless of the distance, is triggered simultaneously for evaluating the network in the worst case scenario. By doing this, a high packet load environment is created, where high priority packets compete directly with low priority ones for gaining channel usage. This behavior can be found in some specific applications where explosions or chemical leaks can occur. When using the studied protocol for monitoring events with a low spreading rate, high priority packets are likely to be the first ones to trigger transmission, thus leading to better performance.

6.4. Multievent Environments: Priority Enabled. Results shown in previous sections suggest that energy consumption is significantly affected as detections per second increase. This is specially true when $\overline{v}$, R, or even the number of mobile objects (MOs) within the network increases. The following set of results shows the report delay for mobile object detections when considering a hybrid environment containing mobile and randomly distributed events. Due to the importance of accurately reporting the detection or status of a mobile object, packets containing this type of data are considered to be of high priority. Therefore, randomly distributed events with detection radius R are considered to be of low priority. First, an environment where the detection radius for both low priority and the mobile object is fixed with $R = 5\,\text{m}$ is studied.

Figure 24 shows the average expected (mathematical) and obtained (simulation) detections per second for high priority events. Also, low priority detections per second are shown. As observed, specially as average speed increases, low priority events interfere with the detection of the mobile object, leading to loss of information. However, as Figure 25 shows, high priority packets do not experiment a significant increase in report delay, validating the proposed priority model.

Finally, an experiment is conducted by increasing detection radius and defining $\overline{v} = 5\,\text{m/s}$. In this case, the number of low priority detections increases along with R, which creates a more hostile environment for mobile object detections.

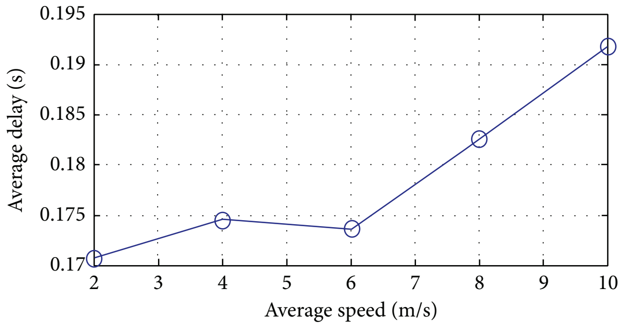


FIGURE 25: Average delay for high priority packets in a hybrid environment.

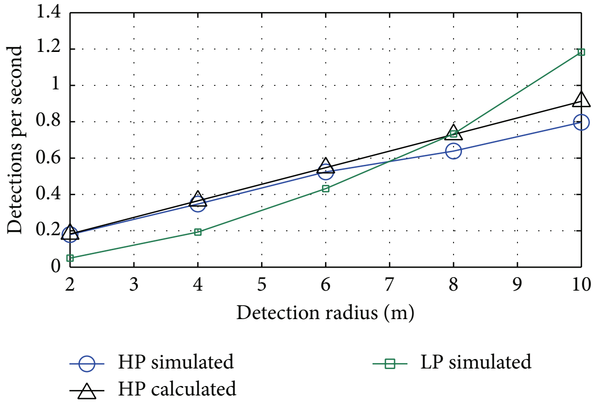


FIGURE 26: Detections per second for high and low priority packets in a hybrid environment.

This can be seen in Figure 26 where several mobile detections are ignored due to the fact that the sensing node is in a busy state (where the node is trying to send information related to a low priority detection).

Despite the fact that the number of detections increases, delay in high priority packets (Figure 27) tends to behave in a similar manner compared to the previous experiment, where low priority detections were defined constant.

This is due to the fact that the probability for a mobile object to be detected inside a busy section of the network, for example, a cluster, is significantly low. Thus, report delay for high priority mobile events is nearly independent from the occurrence of randomly distributed events.

On the other hand, it is likely that some mobile detections are ignored due to the fact that low priority events trigger the transmission of several nodes and while transmission attempts are performed, these nodes are considered unable to detect any other events. It is worth noting then that, in this scenario, important data is lost.

7. Conclusions and Future Work

In this work, WALTER, a nonpreemptive hybrid protocol for WSN, is studied by means of a proposed Markov model

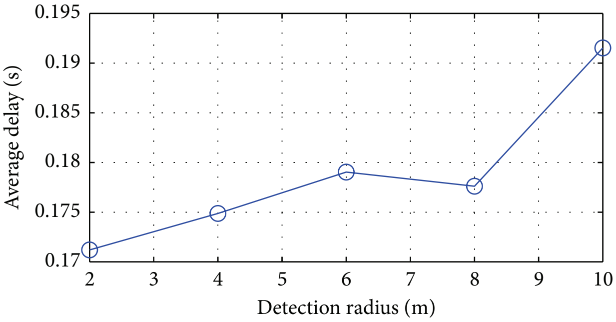


Figure 27: Average delay for high priority packets in a hybrid environment.

and a discrete event simulator. The aforementioned protocol improves network lifetime and energy consumption for low rate event detection environments. Throughout this study, WALTER performance is analyzed under several conditions by defining a wide variety of environments. For each environment, the proposed Markov model may be used to accurately obtain the average report delay and the energy consumed per event transmission. Some key aspects regarding the performance of WALTER are as follows.

(i) Fixed Event Detection Strategy. It is observed that average report delay increases in a nonlinear fashion along with the number of simultaneous detections. By selecting the *Fixed* strategy, average report delay and energy consumption may be calculated by means of a Markov model similar to the one presented. However, as N and M represent the number of CMs set to perform event detection duties, the calculated QoS parameters correspond to scenarios in which events affect the clusters entirely. Whenever an event affects certain section of the cluster, transmission is triggered in a lower number of nodes. This causes event report delay and energy consumption to adopt lower values than calculated. So, the presented performance analysis for the *Fixed* strategy represents the maximum values that can occur for average report delay and energy consumption. In case the network user requires to calculate the exact values for those parameters, the described *Average* and *Radius* strategies may be used. However, it implies that the user must be aware of the probability distribution function of the detecting nodes, which is not straightforward to calculate. Furthermore, given the worst possible performance for a given scenario meets the monitoring needs of the user, performance is guaranteed throughout the operation of the network.

(ii) Radius Event Detection Strategy. As mentioned in previous sections, the studied environments such as concentric detections are modeled in such a manner that when an event occurs, every node involved in the detection process switches to transmission mode simultaneously. This behavior can de found in some military or industrial applications. On the other hand, when considering events that spread slowly within the network, high priority transmissions are expected to occur before low priority detections, hence lowering congestion within the clusters and enhancing performance.

(iii) Mobile Environments. When operating in this type of environments, detections per second can be calculated in order to predict whether an increase in energy consumption is expected. During this study, ET_{rd} is used to model the average time needed for a single mobile object to be detected by a static node, however this equation should not be used when trying to determine the number of detections per second when several mobile objects are considered. This is due to the fact that, for the given detection scheme, events are ignored when several objects are within range of detection or the detecting node has any RA transmission left.

(iv) Multievent Environments. The equation used to calculate ET_{rd} can be used to obtain the percentage of ignored mobile detections when multievent environments are considered. By calculating $1/ET_{rd}$ the average detections per second are modeled. Then, this same parameter is calculated using the simulator. By comparing these values the average missed mobile detections per minute may be obtained, which represents an important QoS parameter for a particular application.

(v) Future Work. Despite the extensive analysis performed during this study, some issues are left for future work, such as developing an algorithm for adequately selecting the reporting nodes in the *Fixed* strategy for enhancing energy consumption and event report delay while maintaining event report probability. As stated above, (18) may be used to accurately obtain hitting time in mobile-only environments containing a single MO. A new equation may be proposed to adequately model this parameter in mobile environments with several MOs or in multievent environments. For this, the equation must consider report delay and the probability of the MO entering to zones affected by other events. However, this is not straightforward.

Parameters

N: Number of nodes attempting transmission, high priority or priority disabled
M: Number of nodes detecting a low priority event
τ: Transmission probability, priority disabled
τ_c: Transmission probability during CF phases
τ_e: Transmission probability during event reporting phases
τ_h: Transmission probability for high priority packets
τ_l: Transmission probability for low priority packets
n: Nodes with a pending event or joint packet transmission
S_n: Nodes attempting transmission for a given time slot
W: State space for the proposed Markov Model
p_n: Successful transmission probability

x_s: Number of time slots the system stays at a given state
T_{cluster}: Number of time slots required to form a cluster
T_{event}: Number of time slots required to send every event packet
E_n: Mean energy required for transition in a given state
E_{tx}: Energy required for a transmission attempt by a CM
E_{txch}: Energy required for a transmission attempt by a CH
E_{rx}: Energy required for receiving a packet
γ: Energetic cost of a failed transmission
$E(\Delta T_{nh})$: Mean increase in report delay for multihop transmissions
nh: Number of hops needed to reach the sink node
$P(\text{TDMA})$: Probability that a cluster is performing TDMA during steady state
$\overline{N}_{\text{cm}}$: Average number of CMs in each cluster
ts: Remaining TDMA slots at a given time slot.
$E(T_{\text{TDMA}})$: Mean number of time slots required for TDMA.
$P_s^h(i, j)$: Transition probability from state (i, j) to $(i - 1, j)$
$P_s^l(i, j)$: Transition probability from state (i, j) to $(i, j - 1)$
$V_{i,j}$: Mean time slots to absorption from state (i, j).

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was partially supported by CONACyT under Project 183370. The research of Vicent Pla has been supported in part by the Ministry of Economy and Competitiveness of Spain under Grant TIN2013-47272-C2-1-R.

References

[1] T. Arampatzis, J. Lygeros, and S. Manesis, “A survey of applications of wireless sensors and wireless sensor networks,” in *Proceedings of the IEEE International Symposium on Mediterrean Conference on Control and Automation*, vol. 1, pp. 719–724, Limassol, Cyprus, June 2005.

[2] C. Ramachandran, S. Misra, and M. S. Obaidat, “A probabilistic zonal approach for swarm-inspired wildfire detection using sensor networks,” *International Journal of Communication Systems*, vol. 21, no. 10, pp. 1047–1073, 2008.

[3] S. Misra, S. Singh, M. Khatua, and M. S. Obaidat, “Extracting mobility pattern from target trajectory in wireless sensor networks,” *International Journal of Communication Systems*, vol. 28, no. 2, pp. 213–230, 2015.

[4] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[5] O. Younis and S. Fahamy, “Distributed clustering in Ad-Hoc sensor networks: a hybrid, energy-efficient approach,” in *Proceedings of the 23th Annual Joint Conference of the IEEE Computer and Communications (INFOCOM ’04)*, IEEE, March 2004.

[6] A. Manjeshwar and D. P. Agrawal, “TEEN: a routing protocol for enhanced efficiency in wireless sensor networks,” in *Proceedings of the 15th International Parallel and Distributed Processing Symposium*, pp. 2009–2015, San Francisco, Calif, USA, April 2000.

[7] A. Manjeshwar and D. P. Agrawal, “APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks,” in *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS ’02)*, Ft. Lauderdale, Fla, USA, April 2002.

[8] I. L. Mayorga, M. E. Rivero-Angeles, C. C. Gutierrez-Torres, J. A. Jimenez-Bernal, R. Rodriguez, and A. Torres-Rivera, “Data transmission strategies for event reporting and continuous monitoring applications in wireless sensor networks,” in *Proceedings of the 7th International Conference on Broadband, Wireless Computing, Communication and Applications*, pp. 1–7, November 2012.

[9] A. Sharif, V. Potdar, and A. J. D. Rathnayaka, “Prioritizing information for achieving QoS control in WSN,” in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA ’10)*, pp. 835–842, IEEE, Perth, Australia, April 2010.

[10] V. J. Alappat, K. Nitish, and K. K. Anoop, “Advanced sensor MAC protocol to support applications having different priority levels in wireless sensor networks,” in *Proceedings of the 6th International ICST Conference on Communications and Networking in China (CHINACOM ’11)*, pp. 340–343, Harbin, China, August 2011.

[11] K. M. Alam, J. Kamruzzaman, G. Karmakar, and M. Murshed, “Priority sensitive event detection in hybrid wireless sensor networks,” in *Proceedings of the 21st International Conference on Computer Communications and Networks (ICCCN ’12)*, Munich, Germany, August 2012.

[12] A. Raja and X. Su, “A mobility adaptive hybrid protocol for wireless sensor networks,” in *Proceedings of the 5th IEEE Consumer Communications and Networking Conference (CCNC ’08)*, pp. 692–696, IEEE, Las Vegas, Nev, USA, January 2008.

[13] B. Srikanth, M. Harish, and R. Bhattacharjee, “An energy efficient hybrid MAC protocol for WSN containing mobile nodes,” in *Proceedings of the 8th International Conference on Information, Communications and Signal Processing (ICICS ’11)*, December 2011.

[14] Y.-D. Lee, D.-U. Jeong, and H.-J. Lee, “Empirical analysis of the reliability of low-rate wireless u-healthcare monitoring applications,” *International Journal of Communication Systems*, vol. 26, no. 4, pp. 505–514, 2013.

[15] K. S. Deepak and A. V. Babu, “Improving energy efficiency of incremental relay based cooperative communications in wireless body area networks,” *International Journal of Communication Systems*, vol. 28, no. 1, pp. 91–111, 2013.

[16] Y. Li, W. Ye, and J. Heidemann, “Energy and latency control in low duty cycle MAC protocols,” in *Proceedings of the IEEE*

Wireless Communications and Networking Conference (WCNC '05), pp. 676–682, March 2005.

[17] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Performance analysis of mobility-assisted routing," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '06)*, pp. 49–60, Florence, Italy, May 2006.

[18] H. Shu and Q. Liang, "Fundamental performance analysis of event detection in wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '06)*, pp. 2187–2192, Las Vegas, Nev, USA, April 2006.

[19] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE International Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2002.

[20] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, pp. 1567–1576, June 2002.

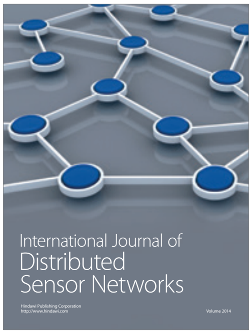

The Scientific World Journal

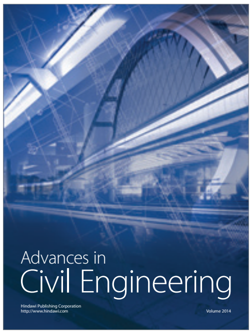

International Journal of Rotating Machinery

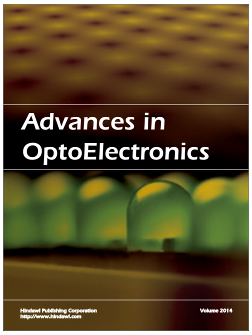

Journal of Engineering


Journal of Sensors

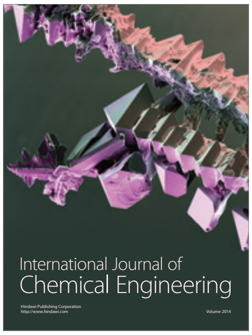

Advances in Mechanical Engineering


International Journal of Distributed Sensor Networks


Advances in Civil Engineering

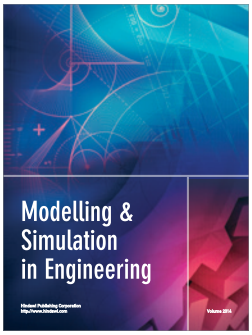

Advances in OptoElectronics

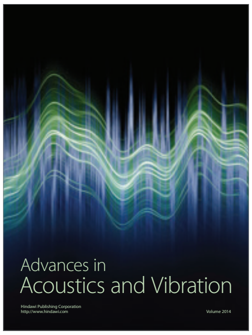

Journal of Robotics


International Journal of Chemical Engineering

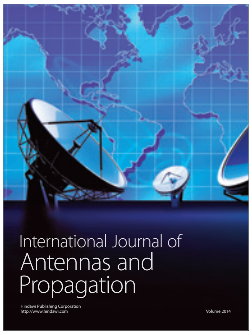

VLSI Design

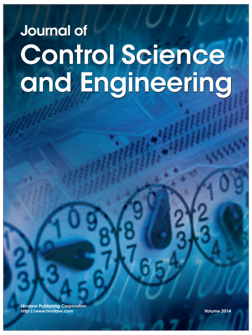

International Journal of Navigation and Observation

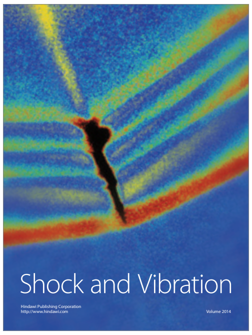

Modelling & Simulation in Engineering


Advances in Acoustics and Vibration


Active and Passive Electronic Components


International Journal of Antennas and Propagation


Journal of Control Science and Engineering


Shock and Vibration


Journal of Electrical and Computer Engineering