Applied Soft Computing 13 (2013) 967–980

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Knowledge-based duty cycle estimation in wireless sensor networks: Application for sound pressure monitoring

J.C. Cuevas-Martinez*, J. Canada-Bago, J.A. Fernandez-Prieto, M.A. Gadeo-Martos

Telecommunication Engineering Department, E.P.S. Linares, University of Jaén, Alfonso X El Sabio, 28, 23700 Linares, Jaén, Spain

ARTICLE INFO

Article history:
Received 27 January 2012
Received in revised form 30 July 2012
Accepted 5 October 2012
Available online 22 October 2012

Keywords:
Sound pressure
Constrained device
Multi-agent system
Wireless sensor network
Fuzzy rule based system

ABSTRACT

Wireless sensor networks comprise an important research area and a near future for industry and communications. Wireless sensor networks contain resource-constrained sensor nodes that are powered by small batteries, limited process and memory and wireless communication. These features give sensors their versatility and drawbacks, such as their limited operating lifetimes. To feasibly deploy wireless sensor networks with isolated motes, several approaches and solutions have been developed; the most common, apart from using alternative power sources such as solar panels, are those that put sensors to sleep for time periods established by the application. We thus propose a fuzzy rule-based system that estimates the next duty cycle, taking the magnitude being tested and battery charge as input. To show how it works, we compare an analytical delta system to our contribution. As an application to test both systems, a sound pressure monitoring application is presented. The results have shown that the fuzzy rule-based system better predicts the evolution of the magnitude by which errors committed by idle periods decrease. This work also shows that application-oriented duty cycle control can be an alternative for measuring systems, thus saving battery and improving sensor node lifetime, with a reasonable loss of precision.

1. Introduction

Wireless sensor networks (WSNs) [1] have become an important study and development area for many applications [2], including distributed processing, video acquisition, intelligent agriculture, industrial control, intelligent buildings, environmental monitoring systems, surveillance, health monitoring and traffic monitoring. A main feature of WSNs is that they contain a variable number of sensor nodes, where each node has a processing unit with limited computational capability and memory, wireless communication capabilities, a set of probes and actuators to measure and modify the environment and they are usually only powered by batteries. Though sensor nodes are resource-constrained devices, several soft computing technologies have been adapted to WSNs, including genetic algorithms [3], particle swarm optimization [4] and fuzzy rule-based systems (FRBSs) [5] to improve the efficiency of the processes that involves sensor nodes in a WSN and avoid deep mathematical calculus.

To prolong sensor lifetime, a common solution [6,7] is to stay as idle as possible and only wake up when necessary. Sensor nodes thus operate in a work cycle in which they first execute the application (e.g. measure, calculate outputs, actuate), decide if it is necessary to transmit any information and go to sleep mode for an interval calculated for each application. This interval generally remains constant, even for MAC layers such as S-MAC [8].

An important topic examined in WSN applications is power consumption optimization [9]. Multiple approaches exist forgetting a suitable solution, as presented by [10]; his algorithm dynamically adjusts the MAC duty cycle, observing the residual energy in the nodes and the network traffic applying FRBS. Estimating the sleep mode interval (SMI) is an important factor in WSN power saving techniques and is usually calculated with coverage and routing restrictions [11,12] or based on environmental evaluation. However, less attention has been paid to estimating the SMI by evaluating the magnitude of the surrounding environment that can be useful for an event-driven WSN. This is explained in further detail below.

This work presents three contributions: an FRBS to estimate the SMI, applying that estimator to an environment-based duty cycle controller and demonstrating that FRBS can be easily setup for this type of monitoring applications instead of analytical solutions.

We first propose a duty cycle scheduler based on an FRBS that estimates the next SMI, taking the values of the magnitude being

* Corresponding author. Tel.: +34 953 648 617; fax: +34 953 648 508.
E-mail addresses: jccuevas@ujaen.es (J.C. Cuevas-Martinez), jcbago@ujaen.es (J. Canada-Bago), jan@ujaen.es (J.A. Fernandez-Prieto), gadeo@ujaen.es (M.A. Gadeo-Martos).

1568-4946/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2012.10.005

tested and battery charge level as input. This system aims to optimize the isolated sensor lifetime for sensing applications using a duty cycle schedule that keeps the node idle when it is not necessary to take measures due to the FRBS estimation. The improved battery lifetime is due to the power drawn in the active mode compared to idle modes. Using Sun Spot WSN technology [13], the current in idle mode falls to 24 mA from a range of 70–120 mA in active mode (400 mA if the sensing board is used) and can be decreased to a minimum of 32 μA in deep sleep mode.

Our second contribution is completing an environment-based duty cycle controller that can help other estimators obtain the best duty cycle. The motivation for using FRBS is that system tuning can be performed only in the rule and fuzzy set setup.

The last contribution of this work is comparing our new FRBS for SMI estimation to an analytical differential system (DS) presented by Cuevas-Martinez et al. [14]. The DS is based on a set of analytical functions that take the absolute value of the magnitude being tested, the difference between the current and previous measures, previous estimated SMI and battery charge as input, resulting in the new estimated SMI. We show that FRBS can obtain good predictions over real magnitudes without a complex model. Moreover, FRBS can suit any other magnitude analysis without any sensor reprogramming; a new knowledge base is instead sent to the affected motes.

With these contributions, we consider the trade-off between magnitude sensing and battery lifetime with low computational cost that can be used in stand-alone or other applications, e.g. ambient intelligence [15], to elongate the sensor lifetime. The impact on battery use for such systems should be low enough to allow long operation periods, i.e. days or weeks, because their estimations must consume less than any sensing or transmission operation. This paper shows that both systems accomplish that condition and can be suitable for long sensing applications due to very low power consumption results.

The remainder of the paper is organized as follows. The following section presents related work. Section 3 describes the FRBS proposed to obtain the SMI on a sensor. Section 4 shows the experimental results with a system adaptation for a sound pressure monitoring application. Section 5 concludes the work.

2. Related work

The multi-agent theory and its applications have been studied for several years and have become a real solution for many complex problems, including industrial applications [16], intelligent buildings [17] and ambient intelligence [18]. Another important area in multi-agent systems is how they consider treating the environment [19]. There are many definitions and theories about what agents really are and what they should do [20]. One theory defines the agent as intentional systems [21] with certain purposes to achieve. In this work, the multi-agent system is embedded inside sensor nodes to manage certain sensor decisions in a smart way to accomplish their tasks efficiently.

WSNs [22] have become a new important area to study due to their possible applications, including easy deployment, distributed processing, mobility, data acquisition and controlling dangerous processes. However, WSNs have significant constraints, mainly when sensor nodes are isolated, e.g. limited power source, computational capacity, wireless interference and routing. Applications with WSNs must consider most of those limiting features; using intelligent systems to manage sensor node behavior is thus common. Moreover, WSNs represent an ideal scenario for integrating intelligent agents that can accomplish complex applications despite the WSNs' constraints [23].

To achieve that premise, new applications have been adapted to WSNs, although sensor nodes have highly constrained resources. One such application is integrating soft computing (SC) technologies into sensor nodes, including fuzzy logic, neuronal networks and FRBSs [24].

FRBSs are considered knowledge-based systems in which system knowledge is represented using a set of "IF–THEN" rules whose antecedents and consequents comprise fuzzy logic statements (fuzzy rules). A main characteristic of these systems is the capacity to incorporate human knowledge by accounting for its lack of accuracy and uncertainty or imprecision. WSNs thus represent an ideal scenario for integrating intelligent agents ruled by FRBS that can accomplish complex applications despite the WSNs' constraints.

Multi-agent architectures have recently been preferentially used over WSNs and with obvious limitations; they can approach tasks for which WSNs are typically applied in a different way. That is the case of WISMAP [25], a WSN application management protocol that defines a special multi-agent based system to build a new framework over WSNs. WISMAP encloses communications, application process, data format, resource hierarchy and agent interaction inside and outside the sensor node. That framework shows that multi-agent systems can be perfectly inserted into a sensor node to efficiently manage WSN resources.

An important handicap of WSNs is that their power sources are limited in most applications; thus, sensor nodes usually remain in power saving modes and enter active mode for very short periods of time. This SMI is one parameter being studied.

There are different approaches to SMI estimation. Most solutions try to schedule communications to avoid periods of unnecessary radio activity and increase node lifetime. One of the first solutions presented is the S-MAC protocol, which adds a slot-based fixed transmission scheduling to the MAC layer to allow sleep periods between radio activity periods. However, S-MAC has high latency rates due to those fixed slots. This step is necessary step for starting the sleep scheduling evolution in the MAC layer ([26] reviews those protocols) that allowed the IEEE 803.15.4 [27] to incorporate some improvements, including the beacon-enabled mode and using sleep scheduling for energy saving in final devices. That scheduling has been recently improved by contributions such as the algorithm proposed by De Paz Alberola [28], where beacon-enabled devices can also fall into sleep mode using a duty cycle learning algorithm (DCLA) that uses traffic to estimate the SMI.

A previous study reviews other power efficiency solutions [29], which are divided between two main classes: topology control and power management. The present paper only considers solutions for power management and sleep/wakeup protocols. Under the sleep/wakeup protocols topic, three types of them appear: on-demand, scheduled rendezvous and asynchronous. Our approach falls into the asynchronous type, similar to a previous approach [30] where each sensor uses a symmetric design for the wakeup schedule function. In our approach, nodes have an independent scheduling based on the measured magnitude and history or past values to allow them to sense for as much time as possible.

A different SMI estimation approach [31] uses stochastic methods, but it needs networks with rigid constraints and does not support node mobility. Although it can achieve good results for the modeled networks, they are based on coverage or latency, not sensing magnitude.

Another type of SMI scheduling is the one used in event-driven WSN, where events detected by sensors produce an important increase in network traffic. This effect is accomplished in the MAC layer [32] by using a non-uniform distribution for the wait time after a collision is detected. This solution tries to minimize the effect of events simultaneously detected by several sensors; it does not estimate when an event is going to occur, but it reacts after an event is detected. Conversely, another previous algorithm [33] estimates a duty cycle by predicting the occurrence of an event by analyzing sensed data and neighbor traffic. This last solution is closer to our approach because the SMI is estimated using external and internal parameters. In that case, the event prediction needs information from neighbor nodes that can derive higher power consumption than our FRBS based on the magnitude being tested.

SMI estimation can be achieved in different ways and with many applications. For our application, we have used an asynchronous scheduler based on the FRBS estimation of the next SMI. The FRBS estimator is described below.

3. SMI estimation based on FRBS

The systems under study, i.e. DS and the new FRBS, estimate the SMI using external and internal parameters to improve battery lifetime for WSN nodes in sensing applications. Both have been developed to be implemented inside the multi-agent architecture explained in the WISMAP framework [25]. Moreover, two FRBS approaches have been tested on two different knowledge bases. The first one tries to save as much battery life as possible while minimizing the error, whereas the second is more conservative and thus infers a longer SMI for the same conditions than the first one. Both knowledge bases are explained in further detail in Section 3.2.

The upgrade to the WISMAP framework, its architecture and the DS are briefly explained below.

3.1. WISMAP framework upgrade

Each sensor node deployed in our WSN runs multi-agent application software. This architecture comprises three agents: management, application control and communication. The management agent aims to control the execution of resources available in the sensor, like the other two agents, but it is not intended to be an operating system. The management agent executes the application control agent, which controls the execution of different sensor applications (e.g. measures probes such temperature or humidity, activates diverse actuators, infer an output in a FRBS, infer and collaborate in FRBS). The communication control agent incorporates the application protocol that allows sensor nodes to communicate with other sensor nodes, neighboring sensors and a base station. The application protocol used is WISMAP [25], which is resource-oriented and specifically designed for WSNs.

3.1.1. Sleep/operation decision subsystem

Two new FRBS estimators have been implemented inside the management agent because it controls the duty cycle in the initial design. The new sleep/operation decision subsystem (SODS) has thus been installed in that agent. When the SODS is active, it controls the sensor sleep–awake cycle. It calculates the new SMI based on the FRBS (described below), programs the sensor node in idle mode and allows it to return to operation mode.

Fig. 1 shows the multi-agent structure and SODS placement.

3.1.2. Differential system

The differential system (DS) [14] to estimate the next SMI is an analytical system based on several parameters: the difference between the variable values measured in the present cycle, its values obtained in previous cycles and battery charge. The range of the measured variable is also divided into several regions, each limited by a set of thresholds. To calculate the SMI, the system considers the difference of the value between different moments and the change in the region.

The method algorithm is comprised of three steps:

First: the sensor node measures the value of the object variable and calculates the difference between the present value and the values obtained in previous cycles.
Second: the sensor node verifies whether the present value belongs to a different region than the previous values. The sensor node may consider whether the present value belongs to a critical region.
Third: considering the difference in values, change of zone, critical regions, battery levels and previous SMIs, the sensor calculates the next SMI.

The way in which the SMI is calculated depends on the application and must be adapted to its objective because parameters such as thresholds are bound to the magnitude being tested.

3.2. FRBS interval estimation

The FRBS contains input and output interfaces, scaling functions, fuzzification and de-fuzzification interfaces, knowledge bases and an adapted inference engine. Each FRBS infers its own output using its inputs and knowledge base (KB). To incorporate this FRBS, it is thus necessary to design a KB, defining the input variables, their value ranges and membership functions, a set of IF–THEN rules and the range and membership functions of the output variable that represent the next cycle sleep interval. Although the KB depends on the application and must be adapted to its objective, the rules and fuzzy sets can be approximately designed, avoiding any detailed physical model of the magnitude at its initial approaches. Only a set of thresholds is used to mark the boundaries of different performance states of the system to measure. To design the knowledge base that runs the FRBS in the sensor, it is necessary to define input and output variables, their fuzzy sets and the rules that manage them.

3.2.1. Input variables

Our approach uses following variables as input for the FRBS that infers the next cycle's SMI:

- *Absolute measured level of the magnitude under study*. The FRBS output changes depending on the absolute value read. Too high or too low values are treated as non-desirable conditions and must be checked in more detail than values read in normal working conditions. This variable is called *sound pressure* and is measured in dBA.
- *Sensor input variations*. The sleep interval may increase if the input variables have small variations and decrease if input variables present large increments. This variable is called *increment* and is measured in dB.
- *Battery level*. In some applications, it is desirable to increase the sleep interval when the battery level is low and prolong the sensor node lifetime, though the sensor node may lose part of the signal being measured. This variable is called *battery* and is measured in percentage.

3.2.2. Output variable

The FRBS output variable is the estimation for the next SMI used in the sensor. This variable is called *SMI* and is measured in milliseconds.

3.2.3. Knowledge base 1 (KB 1): initial approach for inertial signals

The first approach tries to model signals that represent an inertial magnitude, such as sound pressure. The first KB focuses on conservative battery use. When the battery charge is assumed as *HIGH*, the SMI can be inferred as *SHORT* (Table 1). The other main

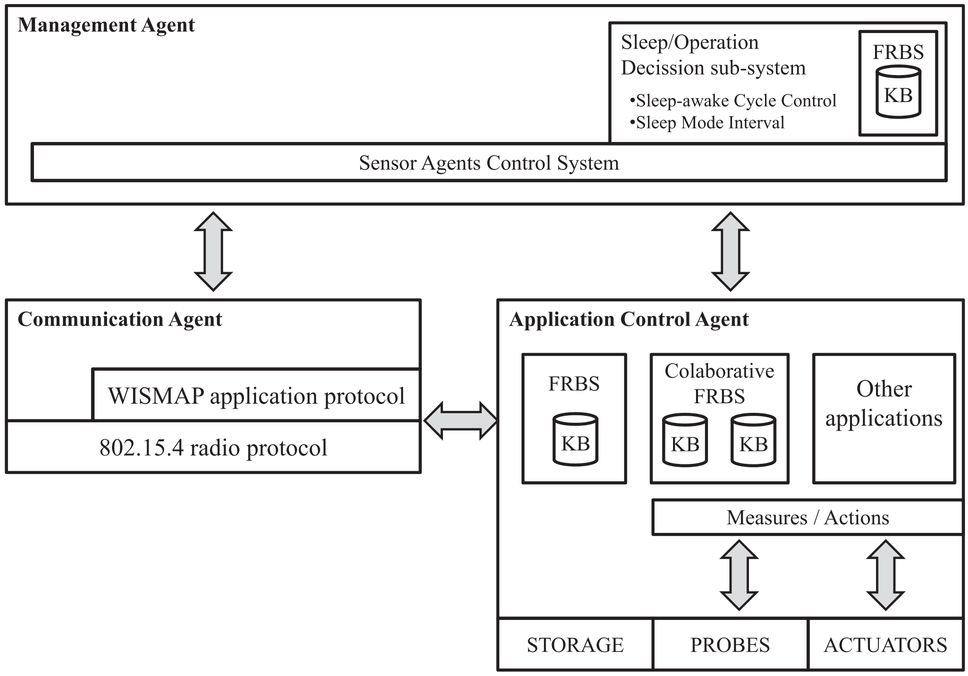


Fig. 1. Multi-agent structure for a sensor with WISMAP framework.

effect, which tries to save as much energy as possible, is the increment between measures; for *LOW* increments, the system tries to give as long an SMI as possible. The third adjustment to SMI is caused due to the absolute level of the *sound pressure* magnitude, which makes shorter SMIs for HIGH magnitude values. For periods of low magnitude variation, the FRBS infers long SMIs to save as much battery as possible. When *Battery* is *HIGH* and the system detects high magnitude variations, it tries to reduce the error and infers shorter SMIs. That behavior is modeled with the rules detailed in Table 1.

Fig. 2 shows the corresponding fuzzy sets of the input and output variables.

The fuzzy sets used and their special features are only examples for our test scenario. Due to the versatility of FRBS, they can be tuned or changed easily to suit any sensing application.

For the *Increment* variable, the chosen difference ranges from 5 to 20 dB. The higher limit of the range is 20 dB because our system aims to be deployed in controlled environments where the instant changes are not usual. The minimum level of 5 dB is employed to avoid minimum differences due to failures in microphone calibration or electrical noise.

Battery charge is considered in two levels that represent a fully charged battery (100%) and a completely depleted one (0%) because more complex fuzzy sets domains have been tested and do not present better results. The rules base grew, making the system slower. To show the battery effect, all tests are repeated for six different battery charges: 5%, 15%, 30%, 75%, 99% and 100%. Those tests are performed because the power consumption of the FRBS and the DS SMI estimator is so low that both can be repeated for days to consume only 1% of the battery load.

The *Sound pressure* variable is divided among three fuzzy sets to model the silence threshold (40 dBA), the sound pressure for a normal working duty (65 dBA) and the threshold of a disturbing noise or a failure (90 dBA). Higher limits could be used, but in this application, we have assumed that measures over 90 dBA are all considered HIGH levels.

The output variable, *SMI*, has been designed with three fuzzy sets to give a versatile estimation, as the fuzzy engine must be implemented into a highly constrained device. The rule base must therefore be designed carefully to avoid unnecessary or redundant rules that model residual behaviors. With these fuzzy sets, the knowledge base can give low SMIs for intervals with measures near the limits or high differences, medium SMI values for transitional periods and long SMIs for measures with values under normal thresholds.

3.2.4. Knowledge base 2 (KB2): adjustment for further battery saving

The rules enumerated above confirm that one of the two KBs has been tested to infer the SMI using FRBS. To detect different behaviors of the magnitude under evaluation, another rule base has been tested. This approach is more conservative: it gives longer sleep periods and shows a different behavior of the estimation system by changing a few bits inside a sensor. The FRBS with KB2 infers a longer SMI than FRBS with KB1 due to rules 3, 4 and 5. Table 2 details the rule base. The fuzzy sets used are the same as in KB1.

Table 1
Rule base for the initial approach (for KB 1).

Rule	Increment	Battery	Sound pressure	SMI
1	Low	Low		Long
2	Low	High	Low	Long
3	Low	High	Medium	Medium
4	Low	High	High	Short
5	High	Low		Medium
6	High	High		Short

Table 2
Rule base for the less conservative approach (KB 2).

Rule	Increment	Battery	Sound pressure	SMI
1	Low	Low		Long
2	Low	High	Low	**Medium**
3	Low	High	Medium	**Long**
4	Low	High	High	**Medium**
5	High	Low		**Long**
6	High	High		Short

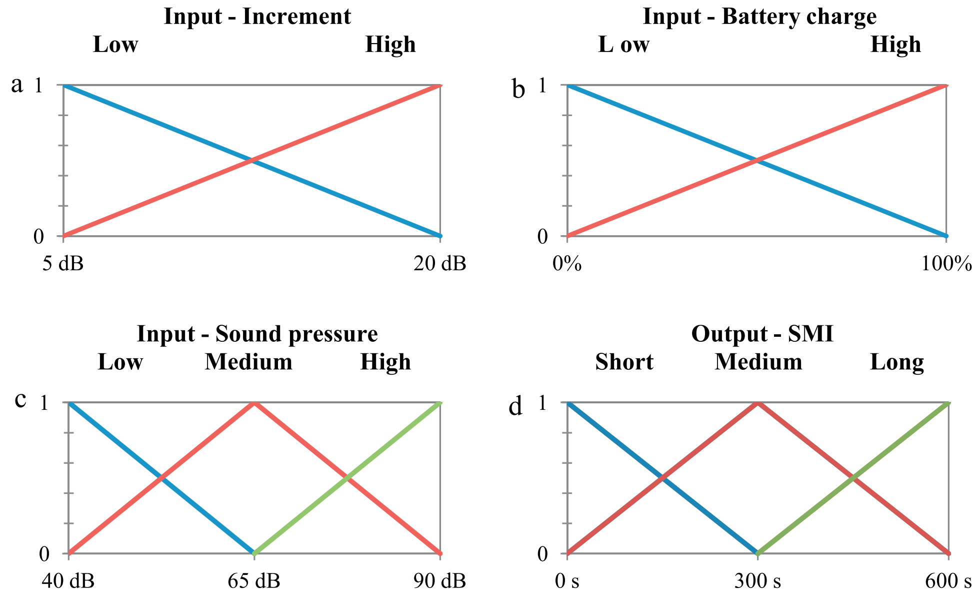


Fig. 2. Fuzzy sets for the three input variables and the output variable SMI: (a) input variable for increment in the magnitude being tested, (b) input variable for battery charge in a percentage of the sensors, (c) input variable for the instant sound pressure and (d) output variable for the next cycle sleep time interval.

Differences between KB1 and KB2 can be explained by battery saving. KB1 tries to save as battery as a priority, but the FRBS with KB2 is more conservative and infers longer SMIs in high variation periods for the magnitude being tested.

3.3. *Advantages of FRBS*

One main advantage of using FRBS over WISMAP is that the system can be adapted to a new environment by deploying a new knowledge base through a WISMAP protocol. This advantage makes this solution more versatile than analytical systems that usually require software re-coding for each platform. Moreover, minimal changes to a KB can provide new capabilities that can adapt the estimation system to new conditions.

4. Experiments and results

The delta system [14] and FRBS have been compared using three different groups of thirty input signals to analyze the error accumulated in the measures when the SMI is not constant. Those signals try to model the same inertial system under slightly different environmental situations. These comparisons seek to demonstrate that the FRBS can obtain enough measurements to detect the evolution of inertial magnitudes without any analytical model and, in some cases, with fewer work cycles than the DS. A good study of the thresholds for the different environment magnitudes used as input is required.

For the error generated due to lost measures, the sampling instant with FRBS is expected to be more accurate than using the DS, mainly due to the difficulties of finding a precise analytical model for the tested magnitudes. The error introduced by FRBS should thus be insignificant compared to the obtained increase in battery duration.

4.1. *Sound pressure monitoring application*

The applications covered by this type of estimation systems have an inertial behavior, e.g. room temperature or the revolutions per minute of an engine. Each magnitude must be studied and modeled separately before it can be used with this application.

4.1.1. *Sound pressure*

The magnitude used to compare the DS with FRBS is sound pressure. This magnitude is used to measure the loudness of sound in one area for a certain period of time. Sound pressure is a typical magnitude in acoustical pollution that WSNs can measure [34]. With the right tuning, it can be used to control other systems, e.g. working machinery using only sound (or vibrations) [35,36]. These mechanical systems produce similar sound patterns when engines or moving parts are working and thus, several sound pressure ranges can be assigned to different workloads or failures.

The parameters of the approach used in these experiments are based on empirical values extracted from studying sound pressure values from different CPU fans. The parameters can be divided into three categories, related to the difference in measured values, sound pressure level and battery level.

4.1.2. *Differential system parameters*

The DS is based on several thresholds that limit the behavior of the functions that estimate the new SMI. The way that FRBS calculates the new SMI must therefore be equivalent. To allow a good comparison between the systems, FRBS has been setup with equivalent values in fuzzy sets and scales as DS, e.g. 20 and 300 s for the minimum and maximum SMI, respectively. The DS parameters can be found in Section 3.1.2 or a previous study [14] in further detail.

4.2. *Experimental description*

This work aims to achieve a system that can predict the evolution of a magnitude to dynamically schedule the sleep time for a sensor, allowing it to stay idle for as much time as possible. Both systems have been tested with three different groups of thirty pseudo-random sound pressure signals based on real sound pressure measures obtained from working computers where the main sound sources are the fans (i.e. power supply, microprocessor, case) and hard disks. These measures have been obtained with a simple analog circuit equipped with an *electret* microphone. This circuit has been incorporated into a Sun SPOT sensor [13]. Systems with spinning parts, as explained in Section 4.1.1, are used to achieve a non-intrusive failure detection system into already deployed machinery, including engines, extractor fans or conveyor belts. A main feature of those systems is that they are highly inertial in normal work duty. Moreover, the sound produced by spinning parts should keep constant pattern overtime, only changing when the working duty increases or there is a failure.

The pseudo-random signals (test signals) consider those features and have been generated with values of sound pressure from 40 to 90 dBA. The normal working sound pressure value measured is approximately 60 dBA; 40 dBA is close to silence and 90 dBA is the louder sound pressure expected due to a failure. These pseudo-random input signals model the sound pressure with values every second that simulate the magnitude that a sensor would read when it wakes up. Those signals have been generated using the following algorithm. To obtain the values that form a pseudo-random signal, this algorithm is repeated to obtain a value at each of P seconds. The random function follows a uniform distribution.

```
i = 0
DO
    If s > S THENn = m + (R*sign(i))
    ELSE n = m + (N*sign(i))
    IF n > T THEN n = T
    IF n < B THEN n = B
    G(i*P) = n
WHILE (i*P < duration)
```

The parameters used to generate those signal are the following:

G, generated signal; P, interval between algorithm executions; duration, time to be generated in seconds; *m*, previous value obtained using the algorithm. The first value is 60 dBA; *n*, next signal value to generate; *s*, random value used for Stability from 0 to 1; *i*, random value used for increment from −0.5 to 0.5; base (*B*), minimum signal value; top (*T*): maximum signal value; Normal increment (*N*), absolute increment for the magnitude to generate when the signal should be stable; Rare increment (*R*), absolute increment for the magnitude to generate when the signal fluctuates; Stability (*S*), the probability (0–1) that marks signal steadiness. Higher values generate steadier signals. For random values lower or equal than Stability, the equation uses the Normal increment to generate the next magnitude value to generate. For random values over the Stability parameter, the equation uses the Rare increment value.

The signals used to test the SMI estimation system have been generated using the following parameter configurations:

Signals type A: N = 0.5, R = 2.0, Steady = 0.9, P = 120 s.
Signals type B: N = 0.2, R = 2.0, Steady = 0.9, P = 25 s.
Signals type C: N = 0.2, R = 0.5, Steady = 0.9, P = 25 s.

4.3. Real system

To measure the sound pressure and avoid the effects of different battery consumption among sensor nodes and distance to the sound source, the same sensor is used to take the real sound pressure samples. The real system used to take sound pressures is an open computer case with an additional fan that can be manipulated to change its spinning speed and simulate failures, including collisions with other parts of the machine, sudden speed changes or lack of lubrication.

Both systems, FRBS and DS, have almost equal processing times, e.g. approximately 730 ms running on a Sun SPOT with the sound pressure calculus included; thus, either could be used with no extra power consumption. The processing time of a sensor node testing the instant sound pressure takes approximately 725 ms and therefore, the penalty of those systems per work cycle is insignificant. These values are later used to check the real reduction in processing time gained or lost due to using an FRBS instead of fixed time sampling.

Table 3
Mean quadratic error of the samples of the input signals obtained with a static duty cycle.

Test signal type	20 s	40 s	150 s	300 s
A	0.0001	0.0008	0.0343	0.1703
B	0.0066	0.0414	0.4765	1.1478
C	0.0010	0.0061	0.0683	0.1661

4.4. Tests

The tests for the FRBS SMI estimation, both knowledge bases and DS have been simulated for ten hours of continuous working. The estimation results for the FRBS are compared to those for the DS for six different initial battery charges: 5%, 15%, 30%, 75%, 99% and 100%. The battery charge values have been used are percentages to avoid differences between sensors and their built-in batteries. The signals used to model the magnitude being tested have been generated using the algorithm explained in Section 4.2. They are grouped into three series of thirty signals. Each group represents different environmental conditions. To have a reference for the obtained error values in the tests, the signals have been sampled every 20, 40, 150 and 300 s to obtain other measures, as if they were taken by a sensor with a static duty cycle. Those sampling periods give 1800, 900, 240 and 120 awake times, respectively. Table 3 shows the mean quadratic error of the thirty signals obtained from interpolating those measure points compared to the test signals used.

4.4.1. Error measures

The error indicator used to evaluate the SMI estimation systems and the statically scheduled system (SSS) is the mean quadratic error, obtained by subtracting the interpolated signal values generated in the measurement points of FRBS, DS and SSS from the test signals. With fewer measures, the possibility of losing significant values increases. In WSNs, the cost of losing measures is more acceptable than excessive power consumption only when the global error introduced by the dynamic SMI estimation system (FRBS-based or DS) is lower than or almost equal to the error produced when the sensor has a statically scheduled duty cycle with a similar number of awake times (e.g. SSS) (Figs. 3–5).

The mean quadratic errors for the both FRBS and DS are shown below. Table 4 also shows the mean quadratic error for SSS with 300 s of sleep time.

To check the significance of the results, an ANOVA is performed for the Error variable using two factors: SMI estimator and battery

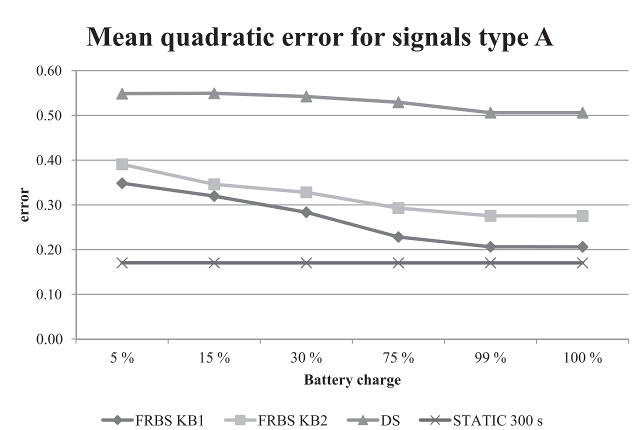


Fig. 3. Mean quadratic error for type A signals of the FRBS SMI estimator with the two knowledge bases, DS and SSS with 300 s of sleep time.

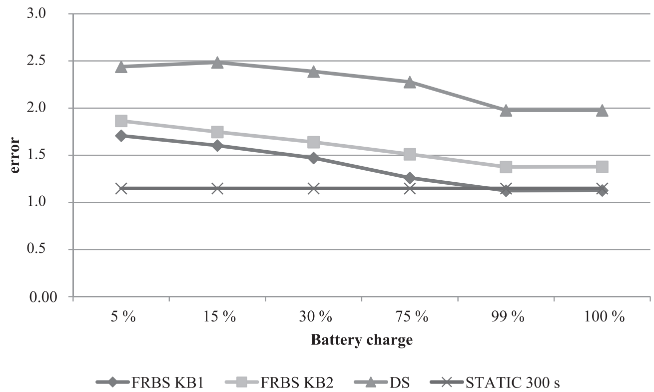


Fig. 4. Mean quadratic error for type B signals of the FRBS SMI estimator with the two knowledge bases, DS and SSS with 300 s of sleep time.

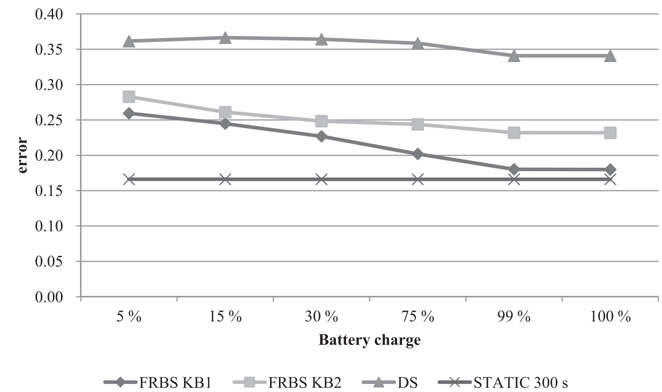


Fig. 5. Mean quadratic error for type C signals of the FRBS SMI estimator with the two knowledge bases, DS and SSS with 300 s of sleep time.

charge. The interaction between the error and SMI estimator shows that one of the estimators has a significantly different mean (Table 5).

To detect the estimator or estimators that have a significantly different mean, the Tukey honestly significant difference (HSD) test is performed. The results (Table 6) show that both FRBS-based estimators achieve a significantly lower error than DS. Table 6 summarizes those results and Figs. 6–8 show the differences in the mean levels for the three signal types.

In some cases with low battery charge, FRBS with KB1 does not have a significantly better mean than FRBS with KB2. FRBS with KB2 with low battery charges thus obtains similar errors but with fewer awake times. Figs. 6–8 present those exceptions, marked with an arrow.

To check the effect on error with different battery charges, the ANOVA shows that the difference between means for some battery levels is also significant. Both systems, FRBS and DS, are battery-dependent; thus, the results can be used to tune them and reduce the error. Table 7 shows the ANOVA results for error variable and battery charge and Table 8 shows the Tukey HSD. The different battery levels denoted as *B* followed by a number represent the percentage of battery charge used in the test, e.g. B05 indicates a 5% battery charge. The comparisons for non-significant factors are summarized in the last row for each signal type. To avoid unnecessary redundancy, the results for a 99% battery charge are used in the analysis due to their similarity to the values for 100% battery charge.

These results are for both methods, FRBS (with two knowledge bases) and DS. The significant values show that only when the sensor battery decreases more than 70%, the effect of the three methods is significant. Therefore, to investigate the real influence of the battery on each method, a new ANOVA is performed using the error variable and battery charge for the error measured for that estimator. To summarize the data, Table 9 only shows the significance of the test for each method for the three signal series.

The above results show that DS does not have a significant mean difference for the different battery charges for signal types A and C. FRBS with both knowledge bases achieves a significant difference

Table 4
Mean quadratic error for sampling the input signals with the proposed FRBS SMI estimator, DS and SSS with 300 s of sleep time.

System	Signal type	Battery charge					
		5%	15%	30% s	75%	99%	100%
FRBS KB1	A	0.3483	0.3196	0.2834	0.2282	0.2062	0.2061
FRBS KB2	A	0.3907	0.346	0.328	0.2929	0.2755	0.2751
DS	A	0.5487	0.5494	0.5423	0.5294	0.5062	0.5062
STATIC 300 s	A	0.1703	0.1703	0.1703	0.1703	0.1703	0.1703
FRBS KB1	B	1.7075	1.6029	1.4711	1.2596	1.124	1.1265
FRBS KB2	B	1.8643	1.7469	1.6388	1.5102	1.3757	1.3784
DS	B	2.4387	2.4849	2.3878	2.2767	1.9764	1.9764
STATIC 300 s	B	1.1478	1.1478	1.1478	1.1478	1.1478	1.1478
FRBS KB1	C	0.2595	0.2448	0.2269	0.2020	0.1802	0.1799
FRBS KB2	C	0.2829	0.261	0.2484	0.2438	0.232	0.2319
DS	C	0.3614	0.3664	0.3642	0.3585	0.3408	0.3408
STATIC 300 s	C	0.1661	0.1661	0.1661	0.1661	0.1661	0.1661

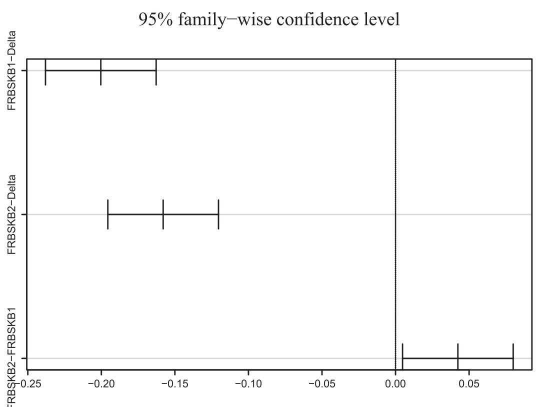
Table 5
ANOVA results for Error using the SMI Estimator factor.

Type of signals		Df	Sum Sq	Mean Sq	*F* value	Pr(>*F*)
A	SMI estimator	2	7.086	3.543	606.9	<2e−16***
	Residuals	537	3.135	0.006		
B	SMI estimator	2	75.52	37.76	330.3	<2e−16***
	Residuals	537	61.40	0.11		
C	SMI estimator	2	1.9096	0.9548	590.9	<2e−16***
	Residuals	537	0.8677	0.0016		

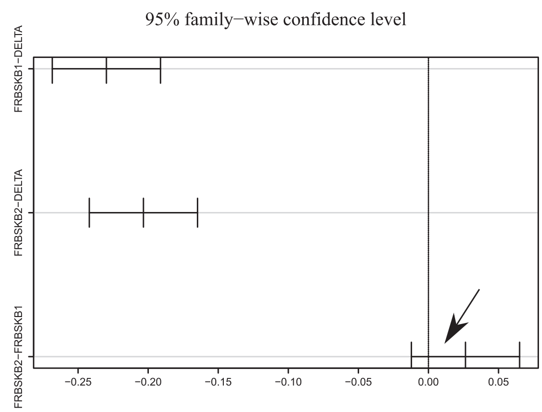
*** Significance codes: 0.

Table 6
Tukey comparisons of means with 95% family-wise confidence level for the three signal groups for the SMI estimator factor.

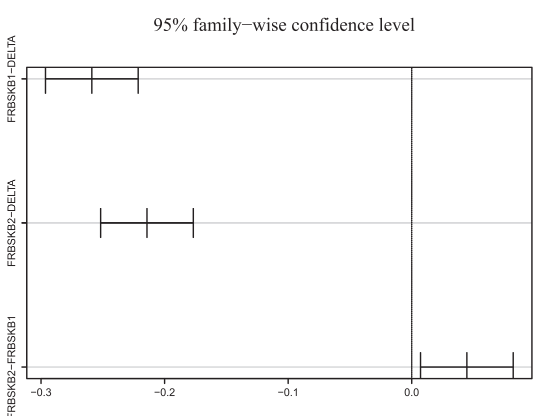
Type of signals	SMI estimator	Difference	Lower	Upper	*p* adj
A	FRBSKB1-DELTA	−0.26505481	−0.28398367	−0.24612595	0
	FRBSKB2-DELTA	−0.21227714	−0.23120600	−0.19334828	0
	FRBSKB2-FRBSKB1	0.05277767	0.03384881	0.07170653	0
B	FRBSKB1-DELTA	−0.8753200	−0.9590903	−0.7915498	0
	FRBSKB2-DELTA	−0.6715803	−0.7553505	−0.5878100	0
	FRBSKB2-FRBSKB1	0.2037398	0.1199695	0.2875100	1e−07
C	FRBSKB1-DELTA	−0.13979145	−0.14975004	−0.12983287	0
	FRBSKB2-DELTA	−0.10535054	−0.11530913	−0.09539196	0
	FRBSKB2-FRBSKB1	0.03444091	0.02448233	0.04439950	0



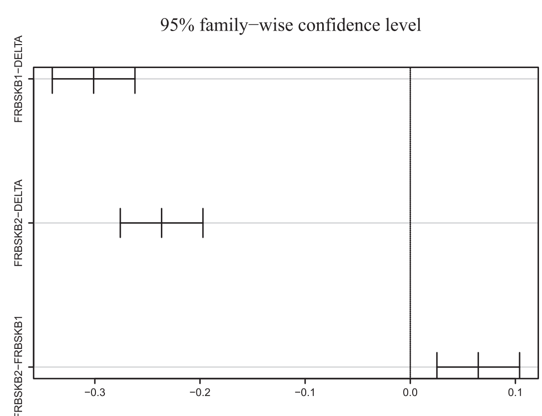
a. Differences in mean levels of System for 5% of battery charge



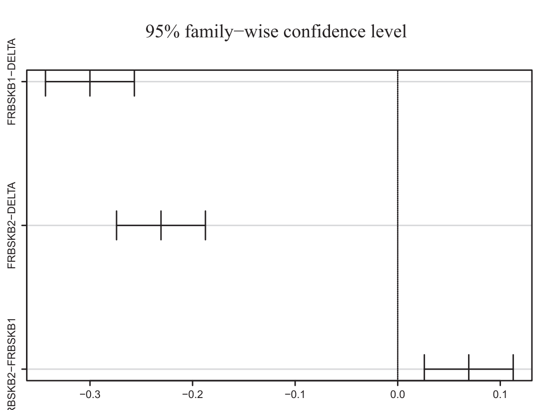
b. Differences in mean levels of System for 15% of battery charge



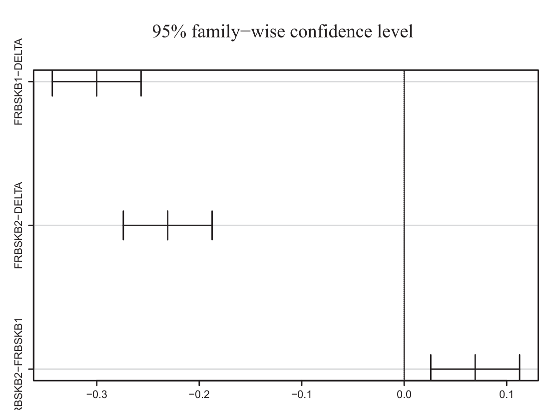
c. Differences in mean levels of System for 30% of battery charge



d. Differences in mean levels of System for 75% of battery charge

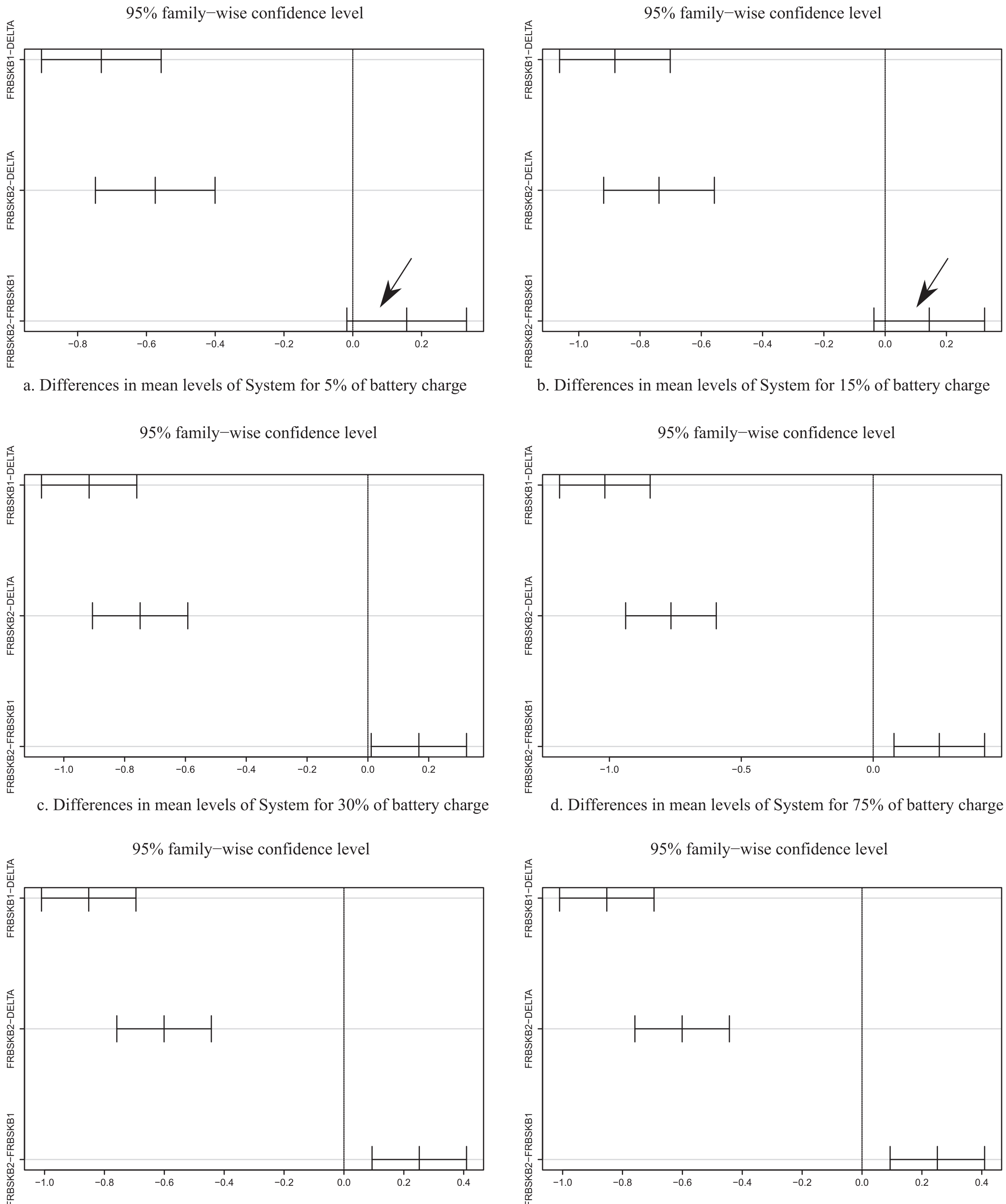


e. Differences in mean levels of System for 99% of battery charge



f. Differences in mean levels of System for 100% of battery charge

Fig. 6. Differences in mean levels of the three estimator systems for Type A signals: (a) 5% battery charge, (b) 15% battery charge, (c) 30% battery charge, (d) 75% battery charge, (e) 99% battery charge and (f) 100% battery charge.



a. Differences in mean levels of System for 5% of battery charge
b. Differences in mean levels of System for 15% of battery charge
c. Differences in mean levels of System for 30% of battery charge
d. Differences in mean levels of System for 75% of battery charge
e. Differences in mean levels of System for 99% of battery charge
f. Differences in mean levels of System for 100% of battery charge

Fig. 7. Differences in mean levels of the three estimator systems for Type B signals: (a) 5% battery charge, (b) 15% battery charge, (c) 30% battery charge, (d) 75% battery charge, (e) 99% battery charge and (f) 100% battery charge.

Table 7
ANOVA results for error with the battery charge factor.

Type of signals		Df	Sum Sq	Mean Sq	F value	Pr(>F)
A	SMI estimator	5	0.778	0.15565	8.802	4.87e−08***
	Residuals	534	9.443	0.01768		
B	SMI estimator	5	22.17	4.433	20.63	<2e−16***
	Residuals	534	114.76	0.215		
C	SMI estimator	5	0.1941	0.03881	8.023	2.62e−07***
	Residuals	534	2.5833	0.00484		

*** Significance codes: 0.

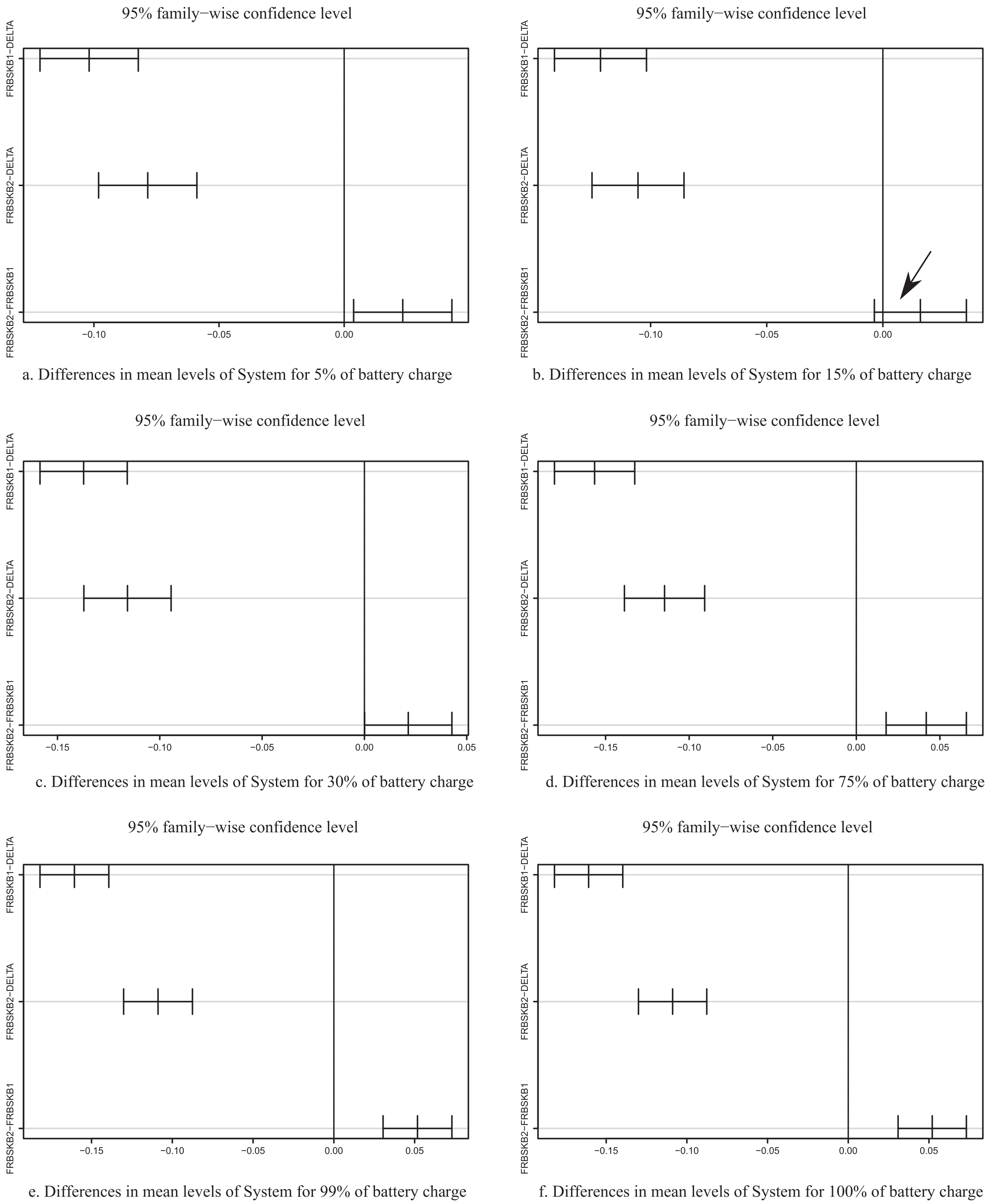


Fig. 8. Differences in mean levels of the three estimator systems for Type C signals: (a) 5% battery charge, (b) 15% battery charge, (c) 30% battery charge, (d) 75% battery charge, (e) 99% battery charge and (f) 100% battery charge.

between battery charges. Table 10 summarizes the significance of the Tukey HSD for all cases. The significant values appear in bold font.

Table 10 shows that FRBS obtains better results than DS, giving significant values for charge differences of only 10%. Moreover FRBS with KB1 achieves better results than FRBS with KB2, in order of significance of its performance with different battery charges for signals Type C, whereas both are almost equivalent for signals Type A or B.

4.5. Explanation of the results

The experimental results show that DS awakes fewer times than both FRBS (Table 11). One reason for this behavior is that DS has more inertia than FRBS and cannot follow a signal with fast changes because DS must use previous SMIs to obtain the new estimation. With FRBS, the estimation is independent of the previous results; FRBS can thus follow the magnitude under control with more accuracy.

Although DS may be sufficient to manage the SMI for a sensor, comparing the error values for similar or fewer awake times of the FRBS (marked with ‘a’ in Table 11) reveals that FRBS is more accurate, as observed in the following figures (Figs. 9–11).

Moreover, the penalty for the larger number of times can be assumed by reducing the error due to a best prediction of the signal evolution and the wider spread of signals that can be modeled with FRBS.

Table 8
Tukey multiple comparisons of means with 95% family-wise confidence level for the three signal groups for the battery charge factor.

Type of signals	SMI estimator	Difference	Lower	Upper	p adj
A	B100-B05	−0.09994576	−0.15664227	−0.04324925	0.0000094
	B75-B05	−0.07908576	−0.13578227	−0.02238925	0.0010595
	B15-B100	0.07574261	0.01904610	0.13243912	0.0020514
	B15-B05 (0.82), B30-B05 (0.21), B30-B100 (0.06), B75-B100 (0.89), B30-B15 (0.90), B75-B15 (0.06) and B75-B30 (0.50)				
B	$B100-B05$	−0.5114560	−0.7091050	−0.3138070	0.0000000
	$B75-B05$	−0.3213377	−0.5189867	−0.1236887	0.0000613
	$B15-B100$	0.4528874	0.2552384	0.6505364	0.0000000
	$B30-B100$	0.3405021	0.1428531	0.5381511	0.0000164
	$B75-B15$	−0.2627691	−0.4604181	−0.0651201	0.0022015
	B15-B05 (0.95), B30-B05 (0.13), B75-B100 (0.06), B30-B15 (0.58) and B75-B30 (0.25)				
C	B100-B05	−0.0504013	−0.0800557	−0.0207469	0.0000227
	B75-B05	−0.0331810	−0.0628353	−0.0035266	0.0181305
	B15-B100	0.0398462	0.0101918	0.0695006	0.0018836
	B15-B05 (0.91), B30-B05 (0.30), B30-B100 (0.06), B75-B100 (0.55), B30-B15 (0.9), B75-B15 (0.24) and B75-B30 (0.86)				

Table 9
ANOVA results for error with factor battery for each estimator separately.

	Signals type A	Signals type B	Signals type C
Method	Pr(>F)	Pr(>F)	Pr(>F)
FRBS with KB1	<2e−16[***]	<2e−16[***]	<2e−16[***]
FRBS with KB2	1.78e−14[***]	1.78e−15[***]	5.03e−10[***]
Delta	0.262	5.9e−07[***]	0.204

[***] Significance codes: 0.

Table 10
Summary of the Tukey HSD for each estimator for the three signal groups for the battery factor.

SMI estimator	Signal type	Battery comparisons				
		B100-B05	B15-B05	B30-B05	B75-B05	B15-B100
FRBS (KB1)	A	**0.000**[*]	0.119	**0.000**[*]	**0.000**[*]	**0.000**[*]
	B	**0.000**[*]	0.378	**0.001**	**0.000**[*]	**0.000**[*]
	C	**0.000**[*]	0.216	**0.000**[*]	**0.000**[*]	**0.000**[*]
FRBS (KB2)	A	**0.000**[*]	**0.011**	**0.000**[*]	**0.000**[*]	**0.000**[*]
	B	**0.000**[*]	0.211	**0.001**	**0.000**[*]	**0.000**[*]
	C	**0.000**[*]	**0.022**	**0.000**[*]	**0.000**[*]	**0.001**
Delta	A	0.314	1.000	0.998	0.907	0.297
	B	**0.000**[*]	0.987	0.982	0.407	**0.000**[*]
	C	0.406	0.993	0.999	0.999	0.195
		B30-B100	B75-B100	B30-B15	B75-B15	B75-B30
FRBS (KB1)	A	**0.000**[*]	0.349	**0.023**	**0.000**[*]	**0.000**[*]
	B	**0.000**[*]	0.141	0.162	**0.000**[*]	**0.003**
	C	**0.000**[*]	**0.015**	0.079	**0.000**[*]	**0.004**
FRBS (KB2)	A	**0.001**	0.699	0.671	**0.001**	0.076
	B	**0.000**[*]	0.109	0.287	**0.000**[*]	0.138
	C	0.156	0.469	0.407	0.125	0.969
Delta	A	0.483	0.834	0.998	0.895	0.978
	B	**0.000**[*]	**0.013**	0.832	0.168	0.752
	C	0.278	0.559	1.000	0.963	0.989

[*] Significance codes: 0.

Table 11
Mean awake times for the proposed FRBS SMI estimator, DS and SSS with 300 s of sleep time.

System	Signal type	Battery charge					
		5%	15%	30%	75%	99%	100%
FRBS KB1	A	82.233	87.633	93.000	106.167	115.000	115.033
FRBS KB2	A	78.000[a]	82.933	86.767	93.733	99.233	99.300
DS	A	64.000	65.133	66.800	70.467	76.967	76.967[b]
STATIC 300 s	A	120.000	120.000	120.000	120.000	120.000	120.000
FRBS KB1	B	85.600[a]	92.267	98.600	116.033	134.133	134.433
FRBS KB2	B	78.100	84.100[a]	88.800	97.200	104.833	104.833
DS	B	64.100	65.400	68.533	76.567	91.567[b]	91.567
STATIC 300 s	B	120.000	120.000	120.000	120.000	120.000	120.000
FRBS KB1	C	81.233	86.467	92.500	106.733	114.200	114.200
FRBS KB2	C	78.000[a]	82.167	85.033	88.567	92.167	92.167
DS	C	64.033	65.233	67.567	71.400	77.467	77.467[b]
STATIC 300 s	C	120.000	120.000	120.000	120.000	120.000	120.000

[a] In this case the FRBS obtain less error than the DS with similar awake times (see Table 4).
[b] Error compared with the FRBS.

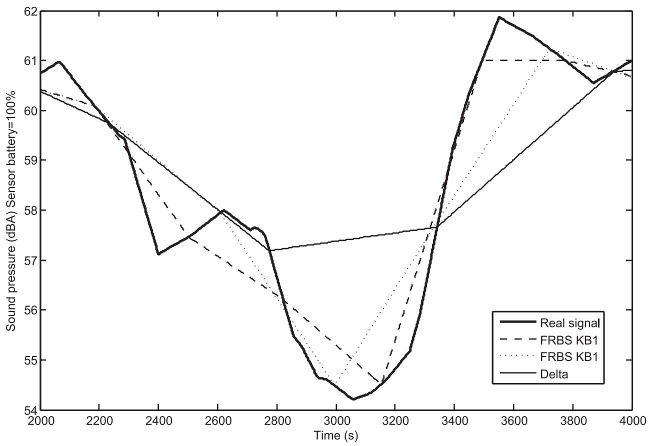


Fig. 9. Interpolation signals for the wake up points for seconds 2000–4000 for measures simulated by signal 2 of the Type A group.

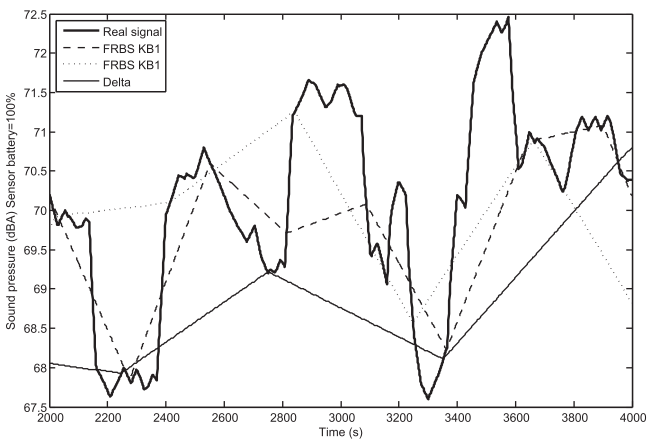


Fig. 10. Interpolation signals for the wake up points for seconds 2000–4000 for measures simulated by signal 2 of the Type B group.

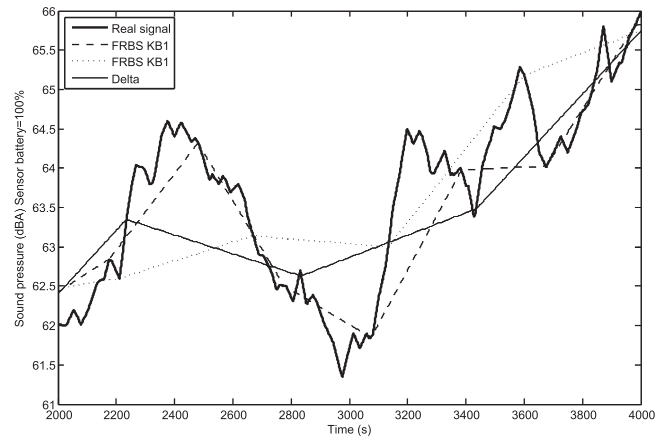


Fig. 11. Interpolation signals for the wake up points for seconds 2000–4000 for measures simulated by signal 2 of the Type C group.

Comparing the results from FRBS to the different KBs shows that each can be applied in different situations to obtain better results. KB1 thus minimizes the error and KB2 minimizes the awake times.

4.6. Energy consumption

To test the effective energy consumption of each method, both systems have been deployed in the same sensor with the same starting battery charge. The node is a Sun SPOT sensor with 720 mAh rechargeable lithium-ion battery and the circuit is presented in Section 4.3. To acquire the real power consumption under the same constraints, both tests obtain the available battery charge after waking from sleep mode, the sensor then calls the SMI estimator (FRBS or DS) based on the sound pressure monitoring and sleeps for a fixed SMI. A fixed SMI is used to compare the effective energy consumption under the same conditions. With fewer awake times,

Table 12
Mean energy consumption and process time for SMI estimators in a Sun SPOT sensor.

	SSS	Delta	FRBS with KB1	FRBS with KB2
Energy consumption (mAh)	0.02135	0.02137	0.02165	0.02159
Process time (ms)	727.11	730.83	737.60	737.29

Table 13
ANOVA results for energy consumption and process time with the SMI estimator factor.

		Df	Sum Sq	Mean Sq	F value	Pr(>F)
Energy consumption	SMI estimator	2	1.620e−06	8.089e−07	1.281	0.282
	Residuals	102	6.442e−05	6.316e−07		
Process time	SMI estimator	2	1169	584.3	1.181	0.311
	Residuals	102	50,464	494.7		

Table 14
Energy saved (mAh) for ten hours of working compared to the SSS with 300 s of sleep time.

System	Signal type	Battery charge				
		5%	15%	30%	75%	100%
FRBS KB1	A	0.78192984	0.6650316	0.54884773	0.26381085	0.07188126
FRBS KB2	A	0.87830119	0.77181201	0.68904706	0.5386713	0.41849589
DS	A	1.19461587	1.17040721	1.13478865	1.05643636	0.91755173
FRBS KB1	B	0.70904162	0.56471559	0.42761993	0.05023343	−0.3480865
FRBS KB2	B	0.87614248	0.74661986	0.64516048	0.46382881	0.29905445
DS	B	1.19247919	1.16470226	1.09775987	0.92609848	0.6055955
FRBS KB1	C	0.80357766	0.69027296	0.55967164	0.25155819	0.08991389
FRBS KB2	C	0.87830119	0.78834773	0.72647909	0.65019027	0.5724767
DS	C	1.19391077	1.16827053	1.11840027	1.03650107	0.9068683

less battery consumption is required. Table 12 shows the result of this experiment.

As Table 12 shows, the power consumption for each SMI estimator is almost equal (battery measurement circuit fluctuations are assumed). To check this finding, an ANOVA is performed for the energy consumption and process time variables with SMI estimator as factors that effectively verify that there are no significant differences between the estimators (Table 13 shows the test results). The power saved in each method depends only on the times that each one awakes and thus, less energy is consumed with fewer awake times.

Table 14 shows the power saved compared to the SSS with 300 s of sleep time in the test period for a fully charged sensor.

Those results can be compared with the energy consumed by a sensor for 5 min of sleep, approximately 0.024 mAh, to achieve an idea of the relative power saving for any of the methods.

5. Conclusions

This work has compared two methods that obtain signal dynamic parameters to estimate the next sleep time to reduce the number of cycles and prolong the sensor node lifetime. The result of the comparison shows that FRBS can be adapted to more types of signals than DS, although DS saves more battery life with less accuracy in the measures. Both systems demonstrate that simple estimation methods with low computational costs can save battery life in continuously sensing applications with a reasonable precision loss and that they can help cover and route estimation methods to synchronize a sleep time for sensors working in clusters that test similar magnitudes.

Future work will focus on designing and developing a FRBS to infer other parameters inside the sensor and collaborate with neighbor nodes to infer better estimations.

Acknowledgments

This work has been partially supported by the University of Jaen and Caja Rural de Jaen (Spain) (project UJA/2010/14/2).

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Computer Networks 38 (2002) 393.
[2] J. Yick, B. Mukherjee, D. Ghosal, Wireless sensor network survey, Computer Networks 52 (2008) 2292–2330.
[3] A.J. Yuste, F.D. Trujillo, A. Triviño, E. Casilari, An adaptive gateway discovery for mobile ad hoc networks, in: MOBIWAC '07, Fith ACM International Workshop on Mobility Management and Wireless Access, 2007, pp. 159–162.
[4] J. Senthilkumar, M. Chandrasekaran, Y. Suresh, S. Arumugam, V. Mohanraj, Advertisement timeout driven bee's mating approach to maintain fair energy level in sensor networks, Applied Soft Computing Journal 11 (2011) 4029–4035.
[5] O. Cordón, F. Herrera, F. Hoffmann, L. Magdalena, Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases, World Scientific Pub Co Inc., 2001.
[6] L. Wang, Y. Xiao, A survey of energy-efficient scheduling mechanisms in sensor networks, Mobile Networks and Applications 11 (2006) 723–740.
[7] I. Dietrich, F. Dressler, On the lifetime of wireless sensor networks, ACM Transactions on Sensor Networks (TOSN) 5 (2009) 5.
[8] W. Ye, J. Heidemann, D. Estrin, An energy-efficient MAC protocol for wireless sensor networks, in: INFOCOM 2002, Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings of IEEE, vol. 3, 2002, pp. 1567–1576.
[9] L. Wang, Y. Xiao, Energy saving mechanisms in sensor networks, in: 2nd International Conference on Broadband Networks BROADNETS 2005, vol. 2005, 2005, pp. 777–785.
[10] S. Misra, D. Mohanta, Adaptive listen for energy-efficient medium access control in wireless sensor networks, Multimedia Tools and Applications 47 (2010) 121–145.
[11] R. Cohen, B. Kapchits, An optimal wake-up scheduling algorithm for minimizing energy consumption while limiting maximum delay in a mesh sensor network, IEEE/ACM Transactions on Networking (TON) 17 (2009) 570–581.
[12] Q. Cao, T. Abdelzaher, T. He, J. Stankovic, Towards optimal sleep scheduling in sensor networks for rare-event detection, in: International Conference on Information Processing in Sensor Networks, 2005, pp. 20–27.
[13] Oracle Labs, SunSPOTWorld—Home, 2012.
[14] J.C. Cuevas-Martinez, J. Canada-Bago, J.A. Fernandez-Prieto, M. Gadeo-Martos, Propagation of agent performance parameters in wireless sensor networks, in: Highlights in Practical Applications of Agents and Multiagent Systems, 2011, pp. 213–221.
[15] A. Muñoz, J.C. Augusto, A. Villa, J.A. Botía, Design and evaluation of an ambient assisted living system based on an argumentative multi-agent system, Personal and Ubiquitous Computing 15 (2011) 377–387.
[16] M. Pěchouček, Multiagent modelling and simulation as a means to wider industrial deployment of agent based computing in air-traffic control, Advances in Intelligent and Soft Computing 70 (2010) 1–3.
[17] B. Qiao, K. Liu, C. Guy, A multi-agent system for building control, in: IAT '06, IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2006, pp. 653–659.
[18] J.C. Augusto, P. McCullagh, Ambient intelligence: concepts and applications, Computer Science and Information Systems 4 (2007) 1–26.
[19] D. Weyns, H. Van Dyke Parunak, F. Michel, T. Holvoet, J. Ferber, Environments for multiagent systems state-of-the-art and research challenges, Environments for Multi-agent Systems 3374 (2005) 1–47.
[20] M.J. Wooldridge, An Introduction to Multiagent Systems, Wiley, 2009.
[21] M. Wooldridge, N. Jennings, Agent theories, architectures, and languages: a survey, Intelligent Agents 890 (1995) 1–39.
[22] H. Karl, A. Willig, Protocols and Architectures for Wireless Sensor Networks, Wiley-Interscience, 2007.
[23] B. Karlsson, O. Bäckström, W. Kulesza, L. Axelsson, Intelligent sensor networks—an agent-oriented approach, in: REALWSN'05, Workshop on Real World Wireless Sensor Networks, 2005.
[24] E. Benoit, R. Dapoigny, L. Foulloy, Fuzzy-based Intelligent Sensors: Modeling, Design, Application, 2009 arXiv:0904.3685.
[25] J.C. Cuevas-Martinez, M.A. Gadeo-Martos, J.A. Fernandez-Prieto, J. Canada-Bago, A.J. Yuste-Delgado, Wireless intelligent sensors management application protocol—WISMAP, Sensors 10 (2010) 8827–8849.
[26] I. Demirkol, C. Ersoy, F. Alagoz, MAC protocols for wireless sensor networks: a survey, Communications Magazine IEEE 44 (2006) 115–121.
[27] IEEE, IEEE 802.15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), IEEE Computer Society, LAN/MAN Standards Committee, 2003.
[28] R. De Paz Alberola, D. Pesch, Duty cycle learning algorithm (DCLA) for IEEE 802.15.4 beacon-enabled wireless sensor networks, Ad Hoc Networks 10 (2012) 664–679.
[29] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, Energy conservation in wireless sensor networks: a survey, Ad Hoc Networks 7 (2009) 537–568.
[30] R. Zheng, J.C. Hou, L. Sha, Asynchronous wakeup for ad hoc networks, in: MOBIHOC'03, Fourth ACM International Symposium on Mobile Ad Hoc Networking & Computing, 2003, pp. 35–45.
[31] Z. Yaxiong, W. Jie, Stochastic sleep scheduling for large scale wireless sensor networks, in: 2010 IEEE International Conference on Communications (ICC), 2010, pp. 1–5.
[32] K. Jamieson, H. Balakrishnan, Y. Tay, Sift: a MAC protocol for event-driven wireless sensor networks, Wireless Sensor Networks 3868 (2006) 260–275.
[33] P.J. Shin, H. Medeiros, J. Park, A. Kak, Predictive duty cycle adaptation for wireless camera networks, in: 2011 Fifth ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC), 2011, pp. 1–6.
[34] L. Filipponi, S. Santini, A. Vitaletti, Data collection in wireless sensor networks for noise pollution monitoring, in: Distributed Computing in Sensor Systems, 2008, pp. 492–497.
[35] P. Vijayraghavan, R. Krishnan, Noise in electric machines: a review, IEEE Industry Applications Conference: Thirty-third IAS Annual Meeting 1 (1998) 251–258.
[36] N. Tandon, A. Choudhury, A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings, Tribology International 32 (1999) 469–480.

Juan Carlos Cuevas-Martinez received his M.S. degree in Telecommunication Engineering from the University of Malaga in 2002. Currently, he works as an Associate Teacher in the Telecommunication Engineering Department of the University of Jaen (Spain). His main research interests include Wireless Sensor Networks, Networking and Soft Computing. Currently, he is a member of the research group Telematics System Engineering Group in the University of Jaen (http://ist.ujaen.es).

Joaquin Canada-Bago received his M.S. and Ph.D. degrees in Telecommunication Engineering from the Polytechnic University of Madrid in 1990 and 2004, respectively. Currently, he works as an Associate Professor in the Telecommunication Engineering Department of the University of Jaen (Spain). His main research interests include Wireless Sensor Networks and Soft Computing. Currently, he is the head of the research group Telematics System Engineering Group in the University of Jaen (http://ist.ujaen.es).

Jose Angel Fernandez-Prieto works as an Associate Professor in the Telecommunication Engineering Department of the University of Jaen (Spain). He received his M.S. degree in Telecommunication Engineering from the Public University of Navarra in 1997 and his Ph.D. degree in Telecommunication Engineering from the University of Alcala in 2009. His main research interests include Soft Computing, Computer Networks and Wireless Sensor Networks. Currently, he is a member of the research group Telematics System Engineering Group in the University of Jaen (http://ist.ujaen.es).

Manuel Angel Gadeo Martos received his M.S. degree in Telecommunication Engineering from the Polytechnic University of Madrid in 1992 and his Ph.D. degree in Telecommunication Engineering from the University of Alcala in 2009. Currently, he works as an Associate Professor in the Telecommunication Engineering Department of the University of Jaen (Spain). His main research interests include Fuzzy Systems, Genetic Algorithms, Wireless Sensor Networks and Soft Computing. Currently, he is a member of the research group Telematics System Engineering Group in the University of Jaen (http://ist.ujaen.es).