

e-LiteSense: Self-adaptive energy-aware data sensing in WSN environments

João Marco Silva¹  | Paulo Carvalho²  | Kalil Araujo Bispo³  | Solange Rito Lima² 

¹INESC TEC, HASLab, Universidade do Minho, Braga, Portugal

²Centro Algoritmi, Universidade do Minho, Braga, Portugal

³Departamento de Computação, Universidade Federal de Sergipe, São Cristóvão, Brazil

Correspondence

Paulo Carvalho, Centro Algoritmi, Universidade do Minho, 4750-057 Braga, Portugal.
Email: pmc@di.uminho.pt

Funding information

Fundação para a Ciência e Tecnologia, Grant/Award Number: UID/CEC/00319/2019; The EU Framework Programme for Research and Innovation 2014-2020, Grant/Award Number: 732505

Summary

Currently deployed in a wide variety of applicational scenarios, wireless sensor networks (WSNs) are typically a resource-constrained infrastructure. Consequently, characteristics such as WSN adaptability, low-overhead, and low-energy consumption are particularly relevant in dynamic and autonomous sensing environments where the measuring requirements change and human intervention is not viable. To tackle this issue, this article proposes e-LiteSense as an adaptive, energy-aware sensing solution for WSNs, capable of auto-regulate how data are sensed, adjusting it to each applicational scenario. The proposed adaptive scheme is able to maintain the sensing accuracy of the physical phenomena, while reducing the overall process overhead. In this way, the adaptive algorithm relies on low-complexity rules to establish the sensing frequency weighting the recent drifts of the physical parameter and the levels of remaining energy in the sensor. Using datasets from WSN operational scenarios, we prove e-LiteSense effectiveness in self-regulating data sensing accurately through a low-overhead process where the WSN energy levels are preserved. This constitutes a step-forward for implementing self-adaptive energy-aware data sensing in dynamic WSN environments.

KEYWORDS

adaptive sensing, data collection, energy-aware sensing, wireless sensor networks

1 | INTRODUCTION

As a key enabler of the Internet of Things (IoT) and Smart Cities paradigms, wireless sensor networks (WSNs) are currently a technology capturing much attention, both from the academic community and industry. Sustained by the development of multifunctional low-cost wireless sensors, WSNs are applied in a large range of scenarios, many of them requiring operation without human intervention.^{1,2} This versatility and difficult maintenance urge for mechanisms of self-management and power saving in order to allow a cost-effective adaptation of the sensing process to the application area, while optimising WSN lifetime.³

Reducing energy consumption of sensor nodes for improving WSN lifetime has been a major topic of research in the last decade, being identified three major subsystems impacting on energy consumption—communication, sensing, and processing. The communication subsystem revealed to be the most demanding, even for devices able of energy harvesting.⁴ This motivated solutions such as data gathering aggregation in order to reduce the number of transmission events.

However, several works revealed that, depending on the operational scenario, acquiring and processing data may be more demanding than communicating.^{5,6} This suggests that a versatile data gathering process should avoid acquiring redundant information, even when data aggregation strategies are in place.

This motivates the present study, which the main objective is to improve the ability of WSNs to efficiently and autonomously adapt to distinct operational environments, through a self-adjustable data gathering process able to balance accuracy in data acquisition with energy conservation. To improve this trade-off, e-LiteSense, an energy-aware adaptive sensing scheme, is proposed to adjust sensing to specific data gathering requirements without human intervention, while increasing network lifetime. In order to reduce the impact on sensing, processing, and communicating subsystems, this scheme resorts to low-complexity rules able to adjust the sensing frequency according to the variation in scalar physical quantities being measured and the residual battery level of sensor devices. The proof of concept of e-LiteSense resorts to real datasets to evaluate the performance and reactivity of the proposed rules when applied to distinct WSNs measuring scenarios. Facing the above, the contributions of this work are summarised as follows:

- e-LiteSense is presented, being discussed important design principles for achieving an efficient data gathering process with low complexity and enlarged scope. The building blocks of its architecture and the proposed sensing algorithms are discussed in detail;
- a software-based prototype designed according to TelosB motes' specifications⁵ was implemented to evaluate e-LiteSense performance and reactivity. This prototype, used for the proof-of-concept, receives real datasets as input and allows to assess a fine-grain power profile of communication, sensing, and processing subsystems;
- the results evince that a high accuracy in capturing the sensing parameters dynamics is achieved while reducing the number of sensing events significantly (from approximately 40% to 70%, according to the test scenario) and that sensor operational lifetime can be extended to five times when comparing with a systematic sensing approach.

This paper is organised as follows: Related work on adaptive sensing is discussed in Section 2; the e-LiteSense design principles, architecture, and algorithms are described in Section 3; the methodology of tests and the evaluation results are presented in Section 4; and main conclusions are summarised in Section 5.

2 | RELATED WORK

Although adaptive sampling (and sensing) has been successfully used in conventional computer networks,⁷ the available solutions can hardly be applied in resource-constrained networks such as WSNs. In fact, by disregarding the energy parameter in the adaptive sampling process, Silva et al⁷ do not reflect the required balance between accuracy and network lifetime.

The definition of WSNs lifespan, although closely related with its battery, may change depending on the application's functionality and behaviour, on the goal function and on the network topology in use. Yetgin et al⁸ provided an overall view of the most relevant WSN lifespan maximisation techniques in the literature, being the optimisation algorithms classified as resource allocation, opportunistic transmissions, sleep-wake scheduling, routing, clustering, mobile relays and sinks, coverage and connectivity, optimal deployment, data gathering, network coding, data correlation, energy harvesting, and beamforming.

e-LiteSense is a technique to perform self-adaptive sensing based on battery levels of sensor nodes and sensing variations being measured and can be combined with other techniques. In particular, data correlation techniques reduce the amount of data transmission and, by reducing/removing redundancy, improve power usage and network lifetime. When these techniques are combined with self-adaptive sensing schemes, data gathering and transmission may be substantially reduced, improving lifespan of WSNs in different scenarios and environments.

Many studies on adaptive sensing have been proposed for WSNs taking into account the processing and energy constraints of these networks. In Yang et al,⁹ an optimal sensing scheduling policy is proposed for a power harvesting system equipped with a finite battery. The objective is to select the best sensing periods so that the average sensing performance is optimised. In that system, the authors assume that performance depends on time duration between two consecutive sensing periods and develop an algorithm where the sensing rates depend on the battery power levels of sensor nodes. However, this study only considers the battery level of sensor nodes disregarding the variability of the observed parameter. Unlike this study, e-LiteSense considers these two parameters (battery level and sensing variability) to perform its technique properly.

In Wimalajeewa,¹⁰ an approach was developed to reduce the network data communication load resorting to compressive sensing (CS). It consists in getting information based on signal sparsity, statistical analysis and other related theory.¹¹ However, in general, the data decompression process in sensor nodes is inefficient.¹² In addition, the sparsity of the sensor data may change significantly due to their time varying nature, affecting the recovery of the measured

signal.¹³ In particular, Mehrjoo and Khunjush¹⁴ also highlight the errors CS may suffer in the reconstruction phase, proposing a distributed and semi-adaptive CS-based data collection method to tackle CS limitations. This method reduces the reconstruction errors considering the remaining energy of the nodes, at cost of a distributed control.

In Quer et al,¹⁵ an analytical study was carried out for sensing, compression, and recovering signals from large datasets. The analysis resorts to probabilistic relations among variables involved in those processes to capture the space and time correlations of the signals of interest. The proposal lowers the number of transmissions and, consequently, the power consumption in sensor nodes. However, this study is mainly focused on reducing the transmission rate, disregarding the accuracy of the sensing process. Later, Gupta and De¹⁶ propose a sparse Bayesian learning-based adaptive sensor selection framework for increasing the lifetime of a densely deployed WSN for monitoring spatiotemporally varying signal. Sensor selection takes into account the trade-off between sensing quality and energy efficient. The analytical optimisation approach, difficult to deploy in a real environment, requires a topological knowledge of the WSN, which, per se, is a limitation for deploying wide WSN scenarios.

Indeed, the complex relation between data analysis and data acquisition in adaptive sensing paradigms can be highly powerful, as it allows a reliable estimation in situations where non-adaptive sensing fails.¹⁷ Castro and Tanczos¹⁷ present a general estimation procedure which may be satisfactory for a variety of cases; however, it is recognised that requiring prior knowledge of some parameters might not be available in a real life setting.

Chou et al¹⁸ establish a framework for collecting data from a WSN based on adaptive compression sensing (as in Quer et al¹⁵). Focusing on adapting the transmission rate, this proposal aims to reduce the power consumption and the amount of information in sensing data. The paper proposes an algorithm for obtaining an approximation of the measurements using the lowest possible energy, in a precise way.

Kerasiotis et al⁶ estimated the battery lifetime based on a prediction model. The methodology adopted relies on the decomposition of the hardware operations by time periods, using sensor nodes datasheets as reference. Antonopoulos et al⁵ propose a system to calculate in runtime power consumption in WSNs. The authors based the evaluation on different pieces of hardware in a isolated way, such as leds, data transmission, transmission power, micro-controller, and CPU instructions.

The above review highlights the need for studying and improving the trade-off between measurement accuracy and sensing events for different scenarios involving WSN, including real scenarios, having present energy concerns.

The present proposal extends previous work of the same authors^{19,20} in the sense that data acquisition rate now is dependent not only on the observed variation in the scalar physical quantities measured but also on the perceived devices residual battery level over time.

3 | E-LITESENSE: PRINCIPLES, ARCHITECTURE, AND ALGORITHMS

First, the main principles underneath e-LiteSense design are identified, then its architecture is presented to provide an overall framing and, finally, the algorithms ruling e-LiteSense behaviour are explained.

3.1 | Main design principles

The overall goal to achieve is to perform adaptive sensing of physical parameters, gauging their values and variations correctly with minor impact on the limited resources sensor nodes have available. Thus, e-LiteSense conception and architecture must follow several main design principles, highlighted in Figure 1 and described below.

At higher level, e-LiteSense must constitute a *versatile* solution. It is well-known that WSNs are application dependent and may include diverse nodes sensing multiple physical phenomena; therefore, the sensing scheme should be capable of operating in heterogeneous WSN contexts, with diverse measuring requirements and corresponding parameters. Other fundamental aspect to cover is energy awareness in order to allow effective *energy saving*. Thus, the sensing mechanism should capture the physical parameters under consideration based on an energy-conserving approach so that the sensing, processing, and communication components save energy.

These two design principles—versatility and energy awareness—demand for a major e-LiteSense argument, which is *self-adaptive* sensing. In fact, the sampling scheme should be able to self-regulate according to variations on the parameters being sensed and the levels of energy available. To achieve this, a proper trade-off among accuracy, overhead, and complexity of the sensing process needs to be found. In more detail, *accuracy* means that the sensing mechanism should be able capture the behaviour of the physical variables, gauging their values for each WSN applicational scenario. “Correct behaviour” is actually an “expected behaviour” as the application may require for more accurate or coarse measurements.

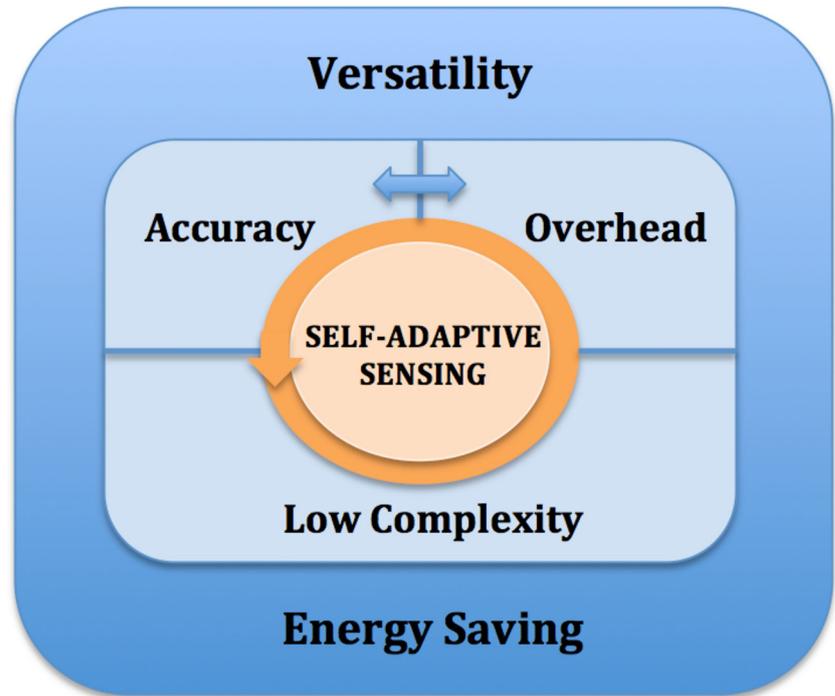


FIGURE 1 e-LiteSense underlying principles

Low overhead means that the sensing process should minimise the burden of sensing, while keeping accuracy within an acceptable range. *Low complexity* means that the expected self-adaptive behaviour of the sensing mechanism should be easy to implement and consume minimal resources.

As mentioned, the *versatility* of the sensing mechanism is understood as the capacity of self-regulate sensing as a function of the intrinsic behaviour of the parameters under control. This implies gauging the variations in the measured parameters, adjusting the sensing frequency as required. Measurement *accuracy* involves detecting both the continuous behaviour of the parameter of interest and events which are critical or sporadic. In Section 3, the accuracy of the measures obtained through e-LiteSense scheme is evaluated comparing visually and statistically the time series involved in systematic and self-adaptive sampling. Despite being a common design goal, a *low-overhead* process assumes an even more relevant role in the presence of resource-constrained devices. Thus, the self-adaptive sensing scheme should exploit parameters steadiness in order to reduce the sensing frequency, reacting to data variations wisely, ie, causing a minor impact on the overall performance. In addition, sensing adaptivity is achieved based on a simple arithmetic process, which is a simpler version of the mechanism transmission control protocol uses to estimate round-trip times,²¹ thus contributing to *low-complexity* processing. In summary, the measures contributing directly to *energy savings* are threefold: (a) reduction of the number of sensing events required to capture each parameter behaviour; (b) adjustment of the sensing frequency through low-cost algorithms; and (c) reduction of the number of transmissions to other sensors or actuators.

3.2 | Architecture

To better understand e-LiteSense role and placement in a WSN deployment scenario, a three plane architecture is envisioned as illustrated in Figure 2 and explained next.

3.2.1 | Application plane

As known, the application plane may encompass a myriad of WSN applicational areas. According to the taxonomy proposed in Yetgin et al,⁸ WSN applications fall within eight broad areas ranging from medical and health to environmental monitoring. Consequently, different measuring requirements have to be specified depending on the required features each application aim to offer, which in turn determine the configuration of sensor nodes. At certain point, the application plane will receive the required measures resulting from single or multiple sensor measurements within the WSN, reflecting, for instance, the spacial or temporal variation of the physical variables under control. Upon receiving the measurement results from the control plane, the application plane may also include processing and user interface units, if applicable.

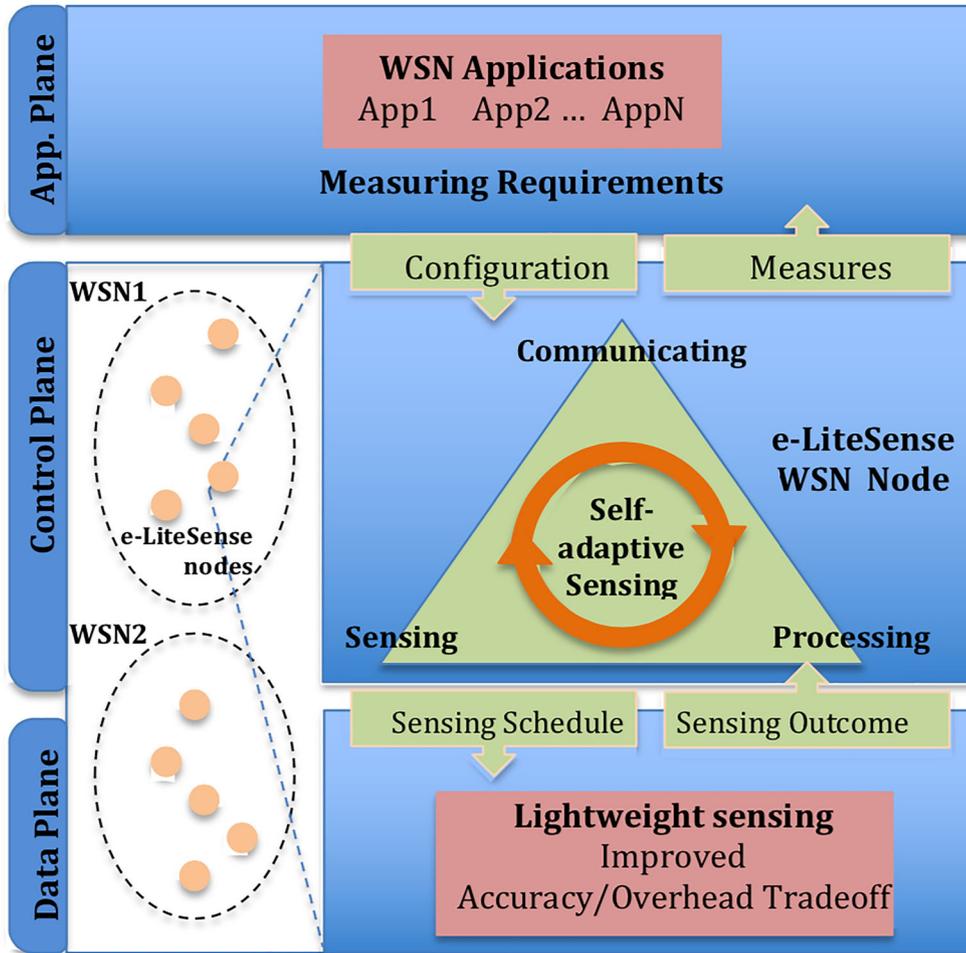


FIGURE 2 e-LiteSense architecture

3.2.2 | Control plane

Directly deployed in WSN sensor nodes, this plane is where e-LiteSense is implemented following the principles drawn in Section 3.1. As detailed in Section 3.3, e-LiteSense determines the occurrence of sensing events autonomously resorting to low-complexity algorithms, which target energy savings within the communicating, processing, and sensing subsystems (vertices of the triangle in Figure 2). As mentioned, the proposed self-adaptive sensing algorithms are aimed at obtaining accurate sensor readings with minor overhead. Balancing the accuracy-overhead trade-off, scheduling of sensing events is configured in runtime depending on how measures vary along the time, conferring a self-adaptive nature to the process. On the other way, the control plane also handles the sensing outcome received from the data plane performing, for instance, data aggregation before passing the results to the application plane.

In practice, control plane functions can be directly instantiated in sensor nodes (see Figure 2) or in any external control component following, for instance, software defined networks (SDN) or edge computing principles.^{22,23} If an external component such as an SDN controller is in place, it would be responsible for configuring e-LiteSense on sensor nodes remotely.

3.2.3 | Data plane

Sensing data are effectively collected within data plane. Taking into consideration the scheduled sensing frequency, sensor readings are collected and sent to the control plane. Recall that e-LiteSense, as a flexible sensing mechanism, allows scheduling the next sensing event according to the application dynamics and measuring requirements.

3.3 | Sensing algorithms

Independently of the application type, it is clear that increasing the number sensing events tends to increase the measurements accuracy. This increase in accuracy has however a major downside: the burden on the sensing, processing, and communicating subsystems regarding energy consumption. In order to find the best balance between measuring accuracy

TABLE 1 Description of Algorithms 1 and 2 elements

S	sensed value of the parameter of interest
\bar{X}_i	moving average of the sensed parameter from current reading.
\bar{X}_{i-1}	moving average of the sensed parameter from previous reading
\bar{V}_i	current mean deviation of the sensed values.
α	weight of S in the moving average \bar{X}_i .
\bar{V}_{i-1}	mean deviation from the last reading.
β	the weight of the current deviation $ \bar{X}_i - S $.
ΔT_{next}	the time interval until the next sensing reading.
$\Delta T_{min} \Delta T_{max}$	lower and upper ΔT bounds.
$getScale()$	function ruling reactivity scale.
$\epsilon_{min} \epsilon_{max}$	scale reactivity bounds.
γ	scale adjustment threshold.

and energy consumption, e-LiteSense takes advantage of the variations in the sensed parameters and the current energy levels to regulate the sensing frequency accordingly. In this way, if data sensing does not exhibit significant variability, the sensing frequency is reduced, ie, the time period until the next sensing event (ΔT_{next}) is increased to save energy and processing resources. This adjustment has to be carefully defined and tuned so that measuring accuracy is yet acceptable. In opposition, if data sensed expresses significant variability, ΔT_{next} is decreased with the aim to increase e-LiteSense reactivity to scalar fluctuations of the parameter of interest. Therefore, defining ΔT_{next} always takes into account battery level of the sensor board.

As mentioned in Section 3.1, e-LiteSense resorts to simple arithmetic rules, pursuing simplicity of implementation and low resource consumption as main design principles. Using intuitive pseudocode, Algorithms 1 and 2 illustrate how e-LiteSense behaves. Table 1 summarises the main constants and variables handled. As illustrated in Algorithm 1, when a sensing event takes place (lines 4,5), the mean value of the observed variable \bar{X}_i is calculated using the moving average between the mean obtained in the previous event and the new sample S , ie, $\bar{X}_i = (1 - \alpha)\bar{X}_{i-1} + \alpha S$ (line 6). The constant α allows to express the weight of new observed value S on \bar{X}_i . Here, for the sake of simplicity, the scalar variation of the parameter of interest is defined as the moving average of the observed parameter \bar{X}_i and its mean deviation \bar{V}_i . This is a simple and acceptable approach to compute an approximation for the standard deviation, as in Shi et al.²¹ Thus, when the sensing event S occurs, the algorithm compares how the new reading has evolved

$$(\bar{X}_{i-1} \xleftarrow{\bar{V}_{i-1}} \bar{X}_i \xleftarrow{\bar{V}_i} S),$$

in order to determine ΔT_{next} , ie, the time period until the next sensing event (lines 8,9). If the deviation between the current and last variations (\bar{V}_i and \bar{V}_{i-1}), in absolute terms, exceeds a threshold γ , the sensed value S has suffered a nonnegligible change; therefore, ΔT_{next} is decreased to capture the new variation trend. Otherwise, if this deviation is minor means that the sensed value S is in a reasonable steady state which allows defining a longer ΔT_{next} to preserve sensor energy.

```

main ()
1   $\bar{X}_{i-1} \leftarrow 0$ 
2   $\bar{V}_{i-1} \leftarrow 0$ 
3  repeat
4     $S \leftarrow getSense()$ 
5     $\Delta T_i \leftarrow \text{timestamp of } S$ 
6     $\bar{X}_i \leftarrow (1 - \alpha)\bar{X}_{i-1} + \alpha S$ 
7     $\bar{V}_i \leftarrow (1 - \beta)\bar{V}_{i-1} + \beta|\bar{X}_i - S|$ 
8     $\Delta T_{next} \leftarrow \Delta T_i + (\Delta T_i \times getScale(\bar{V}_i, \bar{V}_{i-1}))$ 
9     $\Delta T_{next} \leftarrow \max(\min(\Delta T_{max}, \Delta T_{next}), \Delta T_{min})$ 
10    $\bar{X}_{i-1} \leftarrow \bar{X}_i$ 
11    $\bar{V}_{i-1} \leftarrow \bar{V}_i$ 
12    $wait(\Delta T_{next})$ 
13 until end execution;

```

Algorithm 1: Evaluating ΔT_{next}

```

getScale( $\bar{V}_i, \bar{V}_{i-1}$ )
1  if ( $|\bar{V}_i - \bar{V}_{i-1}| \geq \gamma$ ) then
2    return  $-(\text{getBattery}()) \times (\epsilon_{max} - \epsilon_{min}) / 100 + \epsilon_{min}$ 
3  else
4    return  $((100 - \text{getBattery}()) \times (\epsilon_{max} - \epsilon_{min}) / 100 + \epsilon_{min}$ 

```

Algorithm 2: Adjusting e-LiteSense reactivity

The function *getScale()* allows to adjust the scale of e-LiteSense reactivity between sensor readings (see Algorithm 1). As shown, ΔT adjustments are sensible to the current energy in the node's battery. In the presence of high-energy levels, ΔT tend to suffer a fast decrease or slow increase, whereas for low-energy levels, the behaviour is the opposite, ie, ΔT has a slow decrease or fast increase. The constants (ϵ_{min} and ϵ_{max}) are lower and upper bounds to the reactivity scale. These values are adjusted according to each application type and to the corresponding measuring requirements of physical phenomena.

The algorithm also sets important lower and upper bounds on ΔT , namely, ΔT_{min} and ΔT_{max} (see line 9). While ΔT_{max} prevents ΔT from growing indefinitely, thus guaranteeing a minimum number of readings per time unit, (ΔT_{min}) limits the maximum sensing frequency so that the ΔT does not tend to zero, resulting in extreme resource consumption. Again, setting these bounds also depend on the application scenario.

Although not represented in Algorithms 1 and 2, e-LiteSense also provides a critical operation mode which is triggered if a threshold value is observed in the measurements. In this operation mode, sensing events are performed at the highest frequency regardless the sensor's battery level. Monitoring human vital signs or sensitive environments are examples in which such feature is essential, even though redundant batteries are in place.

4 | PROOF-OF-CONCEPT

The main goal in e-LiteSense design is to reduce the overall energy consumed in a WSN while producing accurate readings about the measured environmental phenomena. In this way, the proof-of-concept provides a comprehensive analysis of the proposed adaptive scheme effectiveness, including the impact of adjusting its time boundaries (ΔT_{min} and ΔT_{max}) and sensing reactivity (ϵ_{min} and ϵ_{max}).

4.1 | Test scenarios

The e-LiteSense evaluation resorts to a flexible prototype of TelosB motes designed to allow the deployment of different sensing schemes and provide a detailed power profiling of sensing, processing and communication subsystems considering their underlying energy costs.⁵

The analysis consists of comparing the energy consumed and the accuracy in measuring scalar physical quantities between the proposed scheme and a systematic approach, which represents the most common scheme adopted in WSNs. In order to assess e-LiteSense performance in diverse scenarios, both sensing schemes are applied to pre-collected and public datasets from real environments, described below and summarised in Table 2.

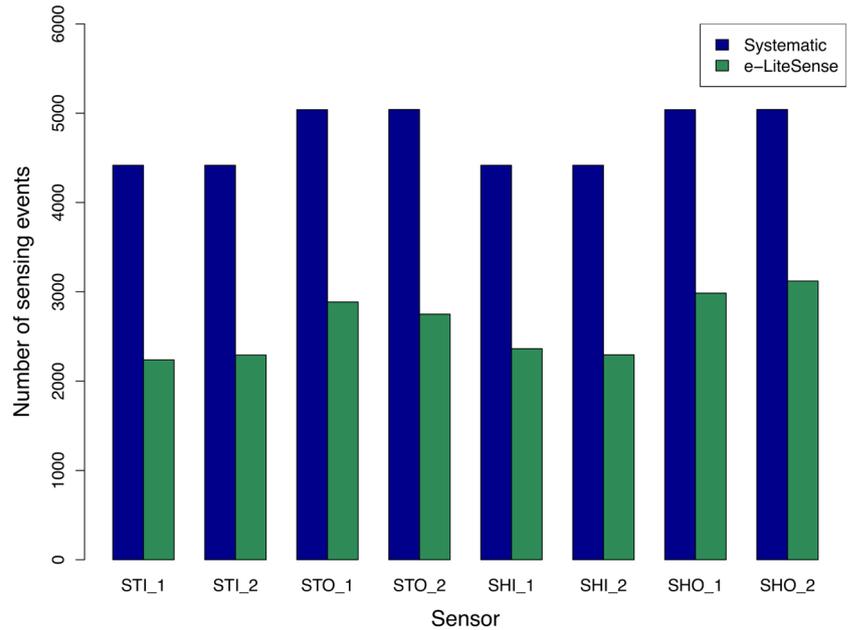
TABLE 2 Identification and description of sensors for each scenario

Sensor ID	Scenario	Description
STI_1	1	Indoor temperature ($^{\circ}$ C)
STI_2	1	Indoor temperature ($^{\circ}$ C), exposed to a steam of hot water
STO_1	1	Outdoor temperature ($^{\circ}$ C)
STO_2	1	Outdoor temperature ($^{\circ}$ C), exposed to a steam of hot water
SHI_1	1	Indoor relativity humidity (%)
SHI_2	1	Indoor relativity humidity (%), exposed to a steam of hot water
SHO_1	1	Outdoor relativity humidity (%)
SHO_2	1	Outdoor relativity humidity (%), exposed to a steam of hot water
LUX_T	2	Four years of indoor temperature measurements ($^{\circ}$ C)
LUX_P	3	One year of atmospheric pressure measurements, (<i>kPa</i>)
LUX_M	3	One year of precipitation measurements (<i>mm</i>)
LUX_H	3	One year of outdoor relative humidity measurements (%)
LUX_I	3	One year of solar irradiance (<i>kW/m²</i>)

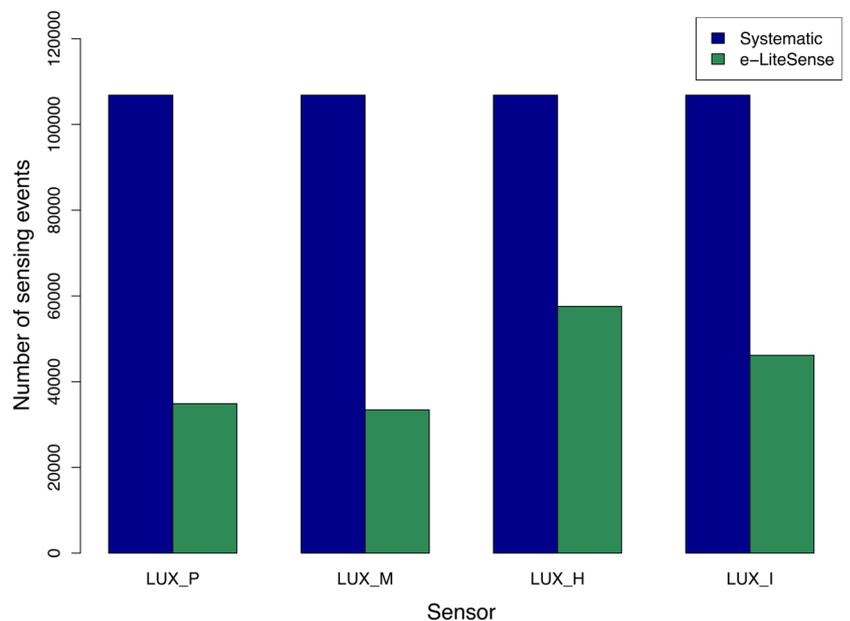
Scenario 1 - eight sensors deployed in indoor and outdoor environments, measuring *temperature* and *relative humidity*, at intervals of five seconds, along six hours.²⁴ Aiming at evaluating e-LiteSense accuracy in identifying sudden changes in the observed parameters, some of these sensors were exposed to a steam of hot water to spark humidity and temperature levels (see Table 2);

Scenario 2 - one indoor sensor, located in Luxembourg City, measuring environmental *temperature*, at intervals of 3 minutes, along with 4 years²⁵;

Scenario 3 - four sensors deployed in a river area weather station in Luxembourg City measuring *atmospheric pressure*, *precipitation*, *relative humidity*, and *solar irradiance*, at intervals of 3 minutes, along with approximately 10 months.²⁶



(A) Scenario 1



(B) Scenario 3

FIGURE 3 Number of sensing events

The performance evaluation consists in assessing (a) the number of sensing events along the monitoring period; (b) the accuracy in identifying the temporal variation of the observed parameters; and (c) the overall energy consumed by all sensor subsystems. Estimating the measurement accuracy provided by e-LiteSense relies on the *correlation coefficient* and the *Mean Squared Error* (MSE) between the sensed values and the real phenomena dynamics, a methodology commonly used to evaluate sampling-based estimators. Furthermore, with the aim of clearly represent the impact of the compared schemes, the energy consumed in each sensing cycle for the smaller datasets was normalised.

The parameterisation of tests, discussed in Section 4.2, considers $\Delta T_{min} = 5$ and $\Delta T_{max} = 60$, meaning that, at least, one sensing event occurs every 60 seconds for *Scenario 1*, and every 60 minutes for *Scenarios 2 and 3*. The reactivity factor is defined as $\epsilon_{min} = 0.05$ and $\epsilon_{max} = 0.30$, meaning that ΔT_{next} will vary between 5% and 30% in each cycle. The overall impact of adjusting these parameters is further discussed in Section 4.3. In addition, the weight of the last observations \bar{X}_{i-1} and \bar{V}_{i-1} on \bar{X}_i and \bar{V}_i (see Algorithm 1) are $\alpha = 0.7$ and $\beta = 0.7$, respectively, and $\gamma = 0$, representing a nonconservative approach in which any deviation observed between \bar{V}_i and \bar{V}_{i-1} will trigger a change in ΔT_{next} .

4.2 | Performance evaluation

As mentioned above, a performance parameter under evaluation is the e-LiteSense ability to reduce the *number of sensing events*. As reference, the systematic sensing frequency in scenario 1 is 30 seconds and in scenario 3 is 15 minutes. For test scenario 1, Figure 3A shows that, for all types of sensor nodes considered, e-LiteSense reduces the number of sensing events significantly, ie, around 39% for STI_1 (worst case) and 50% for SHO_2 (best case) when compared with the number of events observed using the systematic scheme. For test scenario 3, in Figure 3B, the reduction ranges between 47% for sensor LUX_H and 69% for sensor LUX_M.

Although, in order to achieve the expected efficiency, such significant reduction in the number of sensing events must be accompanied by accurate readings, which consists of producing measurements statistically representative when compared to the real distribution of the observed parameters. In this regard, Figure 4 highlights such relation for sensor STO_2, in which, even exhibiting a temperature burst, e-LiteSense was able to provide mostly an overlapped distribution based on 46% of the sensing events compared with the original dataset (systematically sensed every 5 seconds). The visual perception is statistically validated through the low MSE and high correlation coefficient, presented in Table 3.

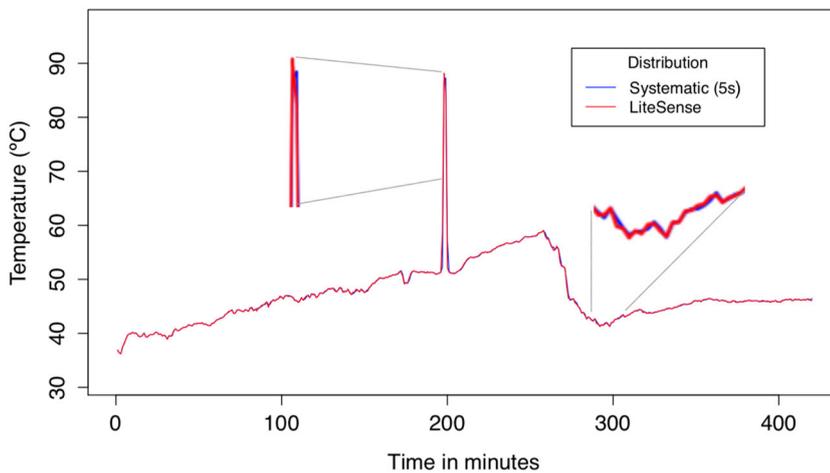


FIGURE 4 Measurements for sensor STO_2

TABLE 3 Measurement accuracy

Scenario 1			Scenario 2			Scenario 3		
Sensor	Correlation	MSE	Sensor	Correlation	MSE	Sensor	Correlation	MSE
STI_1	0.998	0.003	LUX_T	0.999	0.0008	LUX_P	0.999	0.070
STI_2	0.998	0.060				LUX_M	0.690	0.004
STO_1	0.999	0.010				LUX_H	0.992	0.34
STO_2	0.999	0.019				LUX_I	0.988	0.0009
SHI_1	0.996	0.001						
SHI_2	0.955	0.079						
SHO_1	0.999	0.0001						
SHO_2	0.999	0.0009						

Table 3 extends the analysis in Figure 4 for all considered scenarios. The high accuracy achieved for all datasets demonstrates the effectiveness of e-LiteSense self-adaptive sensing behavior according to the dynamics in the observed phenomena.

Due to the reduced number of sensing events, and consequently less data being processed and transmitted, e-LiteSense also promotes high optimization in energy consumption. As presented in Figure 5, for the sensor LUX_T (scenario 2), when compared with the current operation deployment in Luxembourg (ie, one sensing every 3 minutes), the proposed scheme has extended its operational lifetime in approximately five times. Further adjusting the reference approach to a

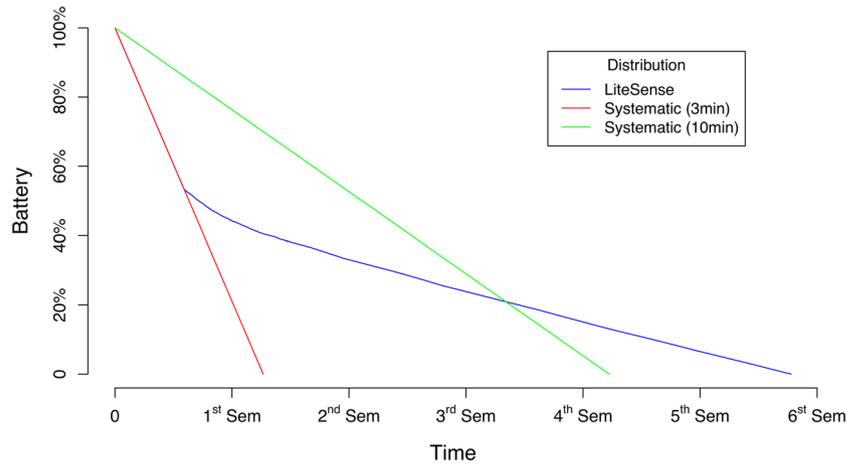


FIGURE 5 Power consumption for scenario 2 (sensor LUX_T)

TABLE 4 Energy saving

Scenario 1							
STI_1	STI_2	STO_1	STO_2	SHI_1	SHI_2	SHO_1	SHO_2
49.35%	51.41%	44.35%	44.95%	47.77%	47.63%	41.02%	38.72%
Scenario 3							
LUX_P	LUX_M	LUX_H	LUX_I				
69.20%	71.25%	47.62%	61.11%				

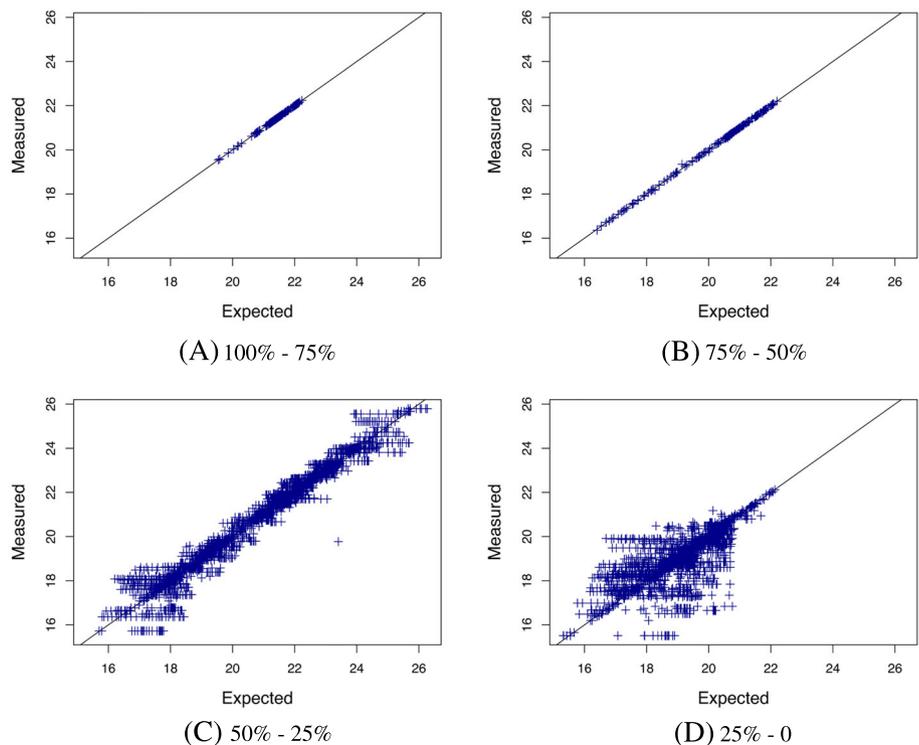


FIGURE 6 Dispersion of measured values for different battery levels

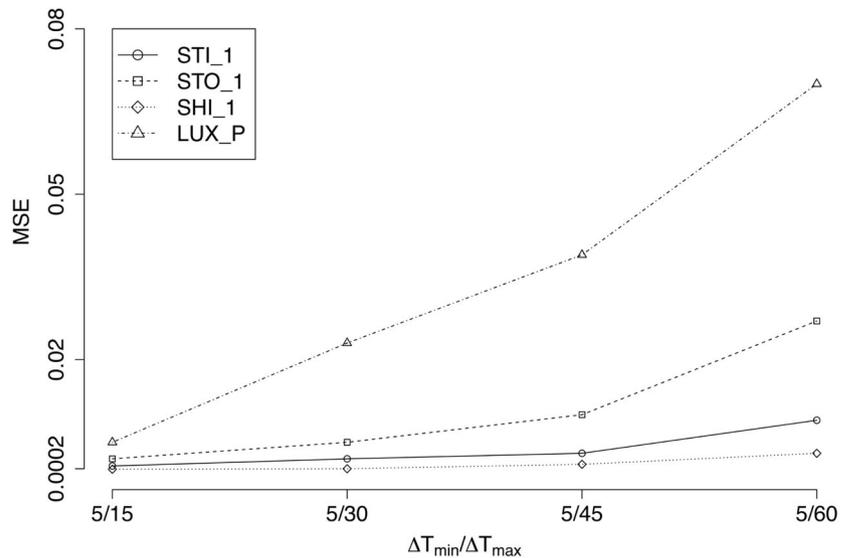
TABLE 5 Impact of adjusting the algorithm reactivity

Sensor	$\epsilon_{min} = 0.05 : \epsilon_{max} = 0.3$			$\epsilon_{min} = 0.05 : \epsilon_{max} = 0.5$			$\epsilon_{min} = 0.05 : \epsilon_{max} = 0.7$			$\epsilon_{min} = 0.05 : \epsilon_{max} = 1.0$		
	Correlation	MSE	Battery									
STI_1	0.997	0.009	48%	0.996	0.011	52%	0.997	0.007	50%	0.999	0.001	52%
STI_2	0.997	0.076	50%	0.998	0.044	50%	0.998	0.056	50%	0.996	0.131	50%
STO_1	0.999	0.027	55%	0.999	0.029	55%	0.999	0.0029	60%	0.999	0.006	60%
STO_2	0.997	0.019	55%	0.999	0.032	56%	0.998	0.072	57%	0.968	2.238	58%
SHI_1	0.993	0.003	52%	0.998	0.0005	55%	0.999	0.0003	55%	0.999	0.0001	55%
SHI_2	0.952	0.086	52%	0.949	0.090	52%	0.960	0.081	52%	0.961	0.070	54%
SHO_1	0.999	0.0006	58%	0.999	0.0008	57%	0.999	0.0002	59%	0.999	0.0002	58%
SHO_2	0.999	0.001	61%	0.999	0.0006	59%	0.999	0.0007	58%	0.999	0.002	58%
LUX_T	0.998	0.011	78%	0.999	0.004	79%	0.999	0.003	81%	0.999	0.001	84%
LUX_P	0.996	0.39	30%	0.998	0.13	28%	0.999	0.09	28%	0.999	0.08	29%
LUX_M	0.468	0.006	28%	0.590	0.005	22%	0.635	0.004	20%	0.656	0.004	19%
LUX_H	0.914	0.40	52%	0.963	0.17	53%	0.983	0.77	55%	0.985	0.69	55%
LUX_I	0.971	0.002	38%	0.989	0.0008	41%	0.994	0.0004	42%	0.995	0.0003	41%

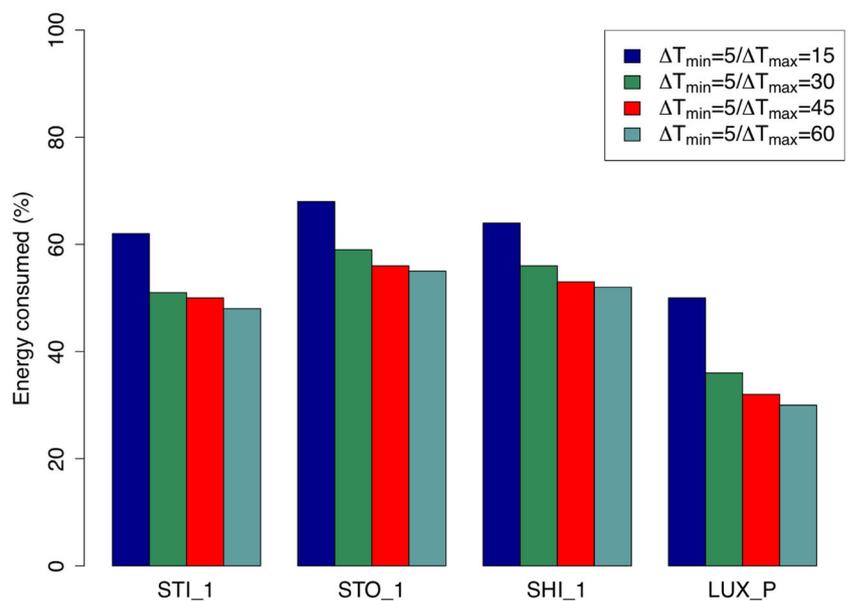
more conservative frequency, ie, systematically sensing every 10 minutes, a sensor running e-LiteSense algorithm would still extend its lifetime around 1 year until full battery depletion.

The overall energy saved by the e-LiteSense approach, when compared with the systematic scenarios described in Section 4.1, is summarised in Table 4. The results show a reduction between 38% (sensor SHO_2) and 71% (sensor LUX_M), confirming its effectiveness, mainly taking into consideration the underlying high accuracy in the yielded measurements (see Table 3).

As described in Section 3.3, the residual energy level in a sensor running e-LiteSense regulates the reactivity used to self-adjust its sensing frequency. Therefore, the accuracy tends to be affected along the energy draining process by increasing the measurement error as the battery is depleted. This behaviour is demonstrated in Figure 6, which shows the accuracy dispersion of the measured temperature per minute in sensor LUX_T, related to the original dataset. In this way, the proposed scheme provides the flexibility for configuring the time boundaries and reactivity in order to balance



(A) Measurement error



(B) Battery consumption

FIGURE 7 Impact of adjusting time boundaries

the extension of the sensor lifetime with the tolerated error for a specific application. The underlying impact of such parameterisation is discussed in Section 4.3.

4.3 | Analysing the algorithm reactivity and time boundaries

The flexibility of adjusting the algorithm reactivity and time boundaries (see Section 3) leverages the variety of scenarios and applications in which e-LiteSense can be applied. Both parameters can be previously defined according to specific requirements or dynamically adjusted by an external entity (eg, an SDN controller) able to react to the WSN measurements.

In this way, Table 5 presents the overall impact of adjusting the algorithm reactivity, ie, ϵ_{min} and ϵ_{max} for all scenarios considered. For these tests, time boundaries are kept fixed ($\Delta T_{min} = 5$ and $\Delta T_{min} = 45$). As observed, even though varying, the overall performance is high for all scenarios, which demonstrates e-LiteSense effectiveness regardless the setup defined. In these cases, the major impact observed is on the sensing accuracy, where MSE varies up to an order of magnitude.

Considering the impact of adjusting the sensing scheme time boundaries, similarly to the observed in Section 4.2, Figure 7 shows a clearer trend regarding the relation between accuracy and energy consumption. In this case, for different sensors, it is possible to observe that increasing ΔT_{max} reduces the energy consumption, with a slight impact on measurement accuracy. As observed for the reactivity assessment, for all considered scenarios, adjusting the time boundaries does not affect the high accuracy of measurements significantly.

5 | CONCLUSIONS

Currently, WSNs sustain a multitude of applicational areas involving the sensing of physical or environmental conditions. A common challenge to this diversity is the confined resources available in sensor nodes, regarding energy, processing, and transmission capacity. To mitigate this problem and improve the adjustment of the data collection process to each specific WSN context, this paper has proposed e-LiteSense, which relies on properties such as self-adaptiveness, energy-awareness and low-complexity to achieve these goals.

An encompassing view of a WSN architecture integrating e-LiteSense sensor nodes has been discussed, detailing the low-complexity rules behind the auto-regulation behaviour of data collection. This process adjusts the sensing frequency according to the variability of the parameters under measurement and the energy levels of sensors, improving the trade-off between accuracy and WSN lifetime.

The proof-of-concept has resorted to a software-based prototype, designed according to the specification of commonly used sensors, and real datasets to evaluate the performance and reactivity of the proposed scheme for distinct test scenarios. The obtained results have demonstrated the high-accuracy levels of e-LiteSense in capturing the behaviour of the parameters under analysis, with a remarkable reduction of sensing events and increase of sensors lifetime.

Following SDN and edge computing concepts, the architecture presented in Figure 2 is currently evolving to adopt edge computing facilities to adjust e-LiteSense configuration parameters dynamically according to the measuring requirements of each deployed WSN scenario.

ACKNOWLEDGEMENTS

This work is funded by EU H2020-The EU Framework Programme for Research and Innovation 2014-2020, grant agreement No. 732505 and by FCT (Fundação para a Ciência e Tecnologia) within the Project Scope: UID/CEC/00319/2019.

ORCID

João Marco Silva  <https://orcid.org/0000-0003-4621-5288>

Paulo Carvalho  <https://orcid.org/0000-0002-6735-7122>

Kalil Araujo Bispo  <https://orcid.org/0000-0001-8878-9293>

Solange Rito Lima  <https://orcid.org/0000-0003-2139-8073>

REFERENCES

1. Atzori L, Iera A, Morabito G. The Internet of Things: a survey. *Comput Netw.* 2010;54(15):2787-2805. <https://doi.org/10.1016/j.comnet.2010.05.010>, <http://linkinghub.elsevier.com/retrieve/pii/S1389128610001568>

2. Gilbert EPK, Kaliaperumal B, Rajsingh EB. Research issues in wireless sensor network applications: a survey. *Int J Inf Electron Eng*. 2012;2(5):702-706. <https://doi.org/10.7763/IJIEE.2012.V2.191>
3. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: a survey. *Comput Netw Int J Comput Telecommun Netw*. 2002;38(4):393-422. [https://doi.org/10.1016/S1389-1286\(01\)00302-4](https://doi.org/10.1016/S1389-1286(01)00302-4)
4. Anastasi G, Conti M, Di Francesco M, Passarella A. Energy conservation in wireless sensor networks. *A survey. Ad Hoc Networks*. 2009;7(3):537-568. <https://doi.org/10.1016/j.adhoc.2008.06.003>, <https://www.sciencedirect.com/science/article/pii/S1570870508000954>
5. Antonopoulos C, Prayati A, Stoyanova T, Koulamas C, Papadopoulos G. Experimental evaluation of a WSN platform power consumption. *2009 IEEE International Symposium on Parallel & Distributed Processing*: IEEE; 2009:1-8. <https://doi.org/10.1109/IPDPS.2009.5161185><http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5161185>
6. Kerasiotis F, Prayati A, Antonopoulos C, Koulamas C, Papadopoulos G. Battery lifetime prediction model for a WSN platform. *2010 Fourth International Conference on Sensor Technologies and Applications*: IEEE; 2010:525-530. <https://doi.org/10.1109/SENSORCOMM.2010.85>. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5558125> <http://ieeexplore.ieee.org/document/5558125/>.
7. Silva JMC, Carvalho P, Lima SR. Inside packet sampling techniques: exploring modularity to enhance network measurements. *Int J Commun Syst*. 2017;30(6):e3135. <https://doi.org/10.1002/dac.3135>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/dac.3135>.
8. Yetgin H, Cheung KTK, El-Hajjar M, Hanzo LH. A survey of network lifetime maximization techniques in wireless sensor networks. *IEEE Commun Surv Tutorials Secondquarter*. 2017;19(2):828-854.
9. Yang J, Wu X, Wu J. Adaptive sensing scheduling for energy harvesting sensors with finite battery. *2015 IEEE International Conference on Communications (ICC)*; 2015:98-103. <https://doi.org/10.1109/ICC.2015.7248305>
10. Wimalajeewa T. Application of compressive sensing techniques in distributed sensor networks: a survey. 2017. :arXiv:1709.10401.
11. Zhang X, Zhang Q, Peng J, Zhao Y, Liu W. A lifetime optimization mobile data gathering strategy with adaptive compressive sensing in WSN. *2017 36th Chinese Control Conference (CCC)*; 2017:8970-8975. <https://doi.org/10.23919/ChiCC.2017.8028784>.
12. Chen Y, Tsou Y. Compressive sensing-based adaptive top-k query over compression domain in wireless sensor networks. *2017 IEEE Wireless Communications and Networking Conference (WCNC)*; 2017:1-6. <https://doi.org/10.1109/WCNC.2017.7925506>
13. Amarlingam M, Mishra PK, Rajalakshmi P, Giluka MK, Tamma BR. Energy efficient wireless sensor networks utilizing adaptive dictionary in compressed sensing. *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*; 2018:383-388. <https://doi.org/10.1109/WF-IoT.2018.8355140>
14. Mehrjoo S, Khunjush F. Distributed semi-adaptive compressive sensing data collection in wireless sensor networks. *Int J Commun Syst*. 2018;31(9):e3546. <https://doi.org/10.1002/dac.3546>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/dac.3546>, e3546IICS-17-0156.R2.
15. Quer G, Masiero R, Pillonetto G, Rossi M, Zorzi M. Sensing, compression, and recovery for WSNs: sparse signal modeling and monitoring framework. *IEEE Trans Wirel Commun*. 2012;11(10):3447-3461. <https://doi.org/10.1109/TWC.2012.081612.110612>
16. Gupta V, De S. Sbl-based adaptive sensing framework for WSN-assisted IOT applications. *IEEE Internet Things J*. 2018;5(6):4598-4612. <https://doi.org/10.1109/JIOT.2018.2870970>
17. Castro RM, Tanczos E. Adaptive sensing for estimation of structured sparse signals. *IEEE Trans Inf Theory*. 2015;61(4):2060-2080. <https://doi.org/10.1109/TIT.2015.2396917>
18. Chou CT, Rana R, Hu W. Energy efficient information collection in wireless sensor networks using adaptive compressive sensing. *2009 IEEE 34th Conference on Local Computer Networks*; 2009:443-450. <https://doi.org/10.1109/LCN.2009.5355162>
19. Silva JMC, Bispo KA, Carvalho P, Lima SR. LiteSense: an adaptive sensing scheme for WSNS. *2017 IEEE Symposium on Computers and Communications (ISCC)*; 2017:1209-1212. <https://doi.org/10.1109/ISCC.2017.8024689>
20. Silva JM, Araujo Bispo K, Carvalho P, Rito Lima S. Flexible WSN data gathering through energy-aware adaptive sensing. *2018 International Conference on Smart Communications in Network Technologies (SaCoNeT)*; 2018:317-322. <https://doi.org/10.1109/SaCoNeT.2018.8585614>
21. Jacobson V. Congestion avoidance and control. *Symposium Proceedings on Communications Architectures and Protocols*. New York, NY, USA: SIGCOMM '88, ACM; 1988:314-329. <https://doi.org/10.1145/52324.52356>, <http://doi.acm.org/10.1145/52324.52356>
22. Kobo HI, Abu-Mahfouz AM, Hancke GPA. Survey on software-defined wireless sensor networks: challenges and design requirements. *IEEE Access*. 2017;5:1872-1899. <https://doi.org/10.1109/ACCESS.2017.2666200>
23. Shi W, Cao J, Zhang Q, Li Y, Xu L. Edge computing: vision and challenges. *IEEE Internet Things J*. 2016;3(5):637-646.
24. Suthaharan S, Alzahrani M, Rajasegarar S, Leckie C, Palaniswami M. Labelled data collection for anomaly detection in wireless sensor networks. *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2010 Sixth International Conference on*: IEEE; 2010:269-274.
25. Luxemburg W, Westhoff M. Maisbich (Luxemburg) temperature data sensor 1209371_house. <https://doi.org/10.4121/UUID:284697EE-9DBE-4F63-B4FB-D5E8E3BC4606>; 2011.
26. Gerrits A. Measured data from sensor meteo at hewelersbach2 for month. <https://doi.org/10.4121/UUID:FE953A23-DA41-498C-A24D-828441C20484>; 2009.

How to cite this article: Silva JM, Carvalho P, Bispo KA, Rito Lima S. e-LiteSense: Self-adaptive energy-aware data sensing in WSN environments. *Int J Commun Syst*. 2019;e4153. <https://doi.org/10.1002/dac.4153>