INTEGRATION, the VLSI journal 94 (2024) 102097

Contents lists available at ScienceDirect

Integration, the VLSI Journal

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

Energy and relevance-aware adaptive monitoring method for wireless sensor nodes with hard energy constraints

David Arnaiz [b,a,*], Francesc Moll [a], Eduard Alarcón [a,c], Xavier Vilajosana [d,b]

[a] *Universitat Politècnica de Catalunya, Spain*
[b] *Worldsensing, Spain*
[c] *NaNoNetworking Center in Catalonia, Spain*
[d] *Universitat Oberta de Catalunya, Spain*

ARTICLE INFO

Keywords:
Wireless sensor networks
Power-awareness
Relevance-awareness
Dynamic energy management
Dual prediction
Adaptive sampling rate
Self-awareness

ABSTRACT

Traditional dynamic energy management methods optimize the energy usage in wireless sensor nodes adjusting their behavior to the operating conditions. However, this comes at the cost of losing the predictability in the operation of the sensor nodes. This loss of predictability is particularly problematic for the battery life, as it determines when the nodes need to be serviced. In this paper, we propose an energy and relevance-aware monitoring method, which leverages the principles of self-awareness to address this challenge. On one hand, the relevance-aware behavior optimizes how the monitoring efforts are allocated to maximize the monitoring accuracy; while on the other hand, the power-aware behavior adjusts the overall energy consumption of the node to achieve the target battery life. The proposed method is able to balance both behaviors so as to achieve the target battery life, at the same time is able to exploit variations in the collected data to maximize the monitoring accuracy. Furthermore, the proposed method coordinates two different adaptive schemes, a dynamic sampling period scheme, and a dual prediction scheme, to adjust the behavior of the sensor node. The evaluation results show that the proposed method consistently meets its battery lifetime goal, even when the operating conditions are artificially changed, and is able to improve the mean square error of the collected signal by up to 20% with respect to the same method with the relevance-aware behavior disabled, and of up to 16% with respect the same algorithm with just the adaptive sampling period or the dual prediction scheme enabled. Consequently showing the ability of the proposed method of making appropriate decisions to balance the competing interest of its two behaviors and coordinate the two adaptive schemes to improve their performance.

1. Introduction

The use of wireless sensor nodes is becoming more prevalent with the passing of time. The increasing adoption of this technology is mainly driven by the ever-growing focus on data analytics [1]. Extracting information from the collected data enables multiple use cases such as predictive maintenance [2], structural health monitoring [3], health care [4], and environmental monitoring [5], between many others. Within the current data-centric mindset, wireless sensors play a facilitator role, allowing the collection of data in a flexible, low-cost, and simple-to-deploy way [6]. These advantages of wireless sensor nodes are obsoleting the more traditional wired counterparts.

Wireless sensor nodes are self-contained devices, and thus, are constrained in terms of available memory, processing power, and energy. Commonly, wireless sensor nodes are battery-powered [7], and extending the battery life is still one of the biggest challenges for this technology. This is particularly the case in applications that require the deployment of sensor nodes in hard-to-reach areas where servicing the nodes to replace the battery is impractical, limiting the lifetime of the node to the lifetime of their batteries [8,9]. To extend the battery life of the node it is paramount to use the available energy as frugally as possible. Usually, sensor nodes operate with a fixed behavior, which is independent of the monitored environment. Nonetheless, the environment is dynamic, and one behavior that may be optimal at a given moment will be suboptimal at a different one. The problem of dynamically adjusting the behavior of the sensor node based on the environmental conditions has attracted attention in the research community, resulting in the definition of a myriad of Dynamic Energy Management (DEM) schemes [7,10,11]. Adjusting the behavior of the

[*] Corresponding author at: Universitat Politècnica de Catalunya, Spain.
E-mail addresses: david.arnaiz@upc.edu (D. Arnaiz), francesc.moll@upc.edu (F. Moll), eduard.alarcon@upc.edu (E. Alarcón), xvilajosana@worldsensing.com (X. Vilajosana).

https://doi.org/10.1016/j.vlsi.2023.102097
Received 30 May 2022; Received in revised form 6 September 2023; Accepted 10 October 2023
Available online 12 October 2023

node to the operating conditions has been shown to increase the efficiency of the sensor nodes, allowing them to extend their battery life with minimal impact on their monitoring accuracy. However, the increase in efficiency comes at the cost of losing predictability. DEM schemes, by their very nature, are unpredictable, since they are driven by environmental conditions.

One of the key parameters for battery-powered nodes is their battery life. The main maintenance cost for battery-powered nodes is the battery replacement process. In some environments such as mining, structural health monitoring, and geotechnical monitoring, the nodes are placed in hard-to-access locations and any maintenance task needs to be planned months in advance. Furthermore, it is expected that the number of nodes per network will continue to increase over time. In this context, it becomes increasingly important to be able to define the battery lifetime of the nodes to plan the maintenance efforts in advance and homogenize their battery lifetime, so that multiple sensors can be serviced with a single visit [12].

The unpredictability of DEM explains why traditional, static, approaches are still the standard in commercial applications, and more particularly so in safety-critical applications. In these applications, the loss in predictability is unacceptable and cannot be justified by increased efficiency. At a first glance, it may seem that there is a trade-off between efficiency and predictability. Nevertheless, a solution to this problem can be devised by applying the notion of self-awareness. For the context of the work presented here, self-awareness can be defined as the ability of a system to obtain knowledge about itself and its environment, use this knowledge to make appropriate decisions autonomously, and act upon these decisions to achieve its goals [13,14]. Self-awareness changes the way environmental uncertainties are handled. Traditionally, the responsibility of defining the device behavior rested on the device designer. Under this view, the device designer has to model the behavior of the device under all possible operating conditions and define a set of configurations for the device, which will result in it accomplishing its operational goals. If the device's behavior is unpredictable, the designer would be unable to model it and define the appropriate configurations. Self-awareness changes this paradigm so that the responsibility of defining the device's appropriate configuration is handed to the sensor node and is performed at runtime [15]. Using its increased level of awareness the sensor node autonomously adjusts itself based on the operating conditions to achieve its goals. The predictability of the sensor node will no longer come from our ability to model its behavior, but it will rather come from the operational goals we define for it. Considering the previous example with the battery lifetime, instead of defining the sampling period required to reach the target battery life, the target battery life can be defined as the goal of the sensor node. In this example, even if the sampling period is unknown at any given moment, we would still be able to know the battery life, since it was defined as a goal.

In this article, we present an energy and relevance-aware monitoring method, which leverages the principles of self-awareness so that once the battery lifetime target has been specified, it autonomously allocates the energy consumption so as to maximize the monitoring accuracy, while complying with the specified battery lifetime goal. Therefore taking advantage of temporal variability of the signal, as DEM methods, without having to sacrifice the predictability of their battery life. Furthermore, the proposed monitoring method has been extended to support two different control parameters of the node, i.e., the sampling period of the sensor node and the rate of samples that are transmitted using the radio interface using a dual prediction scheme. The proposed method models the effect in terms of energy consumption and monitoring accuracy of the two parameters dynamically adjusts them based on their energy and data relevance criteria. The main contributions of this work are summarized as follows:

- we modified the dual predictor scheme to operate on a fixed transmission rate, instead of requiring to use the maximum tolerated transmission error;
- we enhanced the energy and relevance-aware monitoring method to support multiple control variables, specifically the sampling period and the transmission rate of the sensor node, and manage both variables effectively in a coordinated manner;
- the proposed monitoring method has been evaluated using data from three different sources (atmospheric pressure, relative humidity, and light intensity), and two additional tests where the operating conditions were manually altered to stress the monitoring method;

The rest of the article is organized as follows. Section 2 gives an overview of the relevant works related to dual prediction, dynamic sampling, and energy-aware operation in the area of wireless sensor nodes. Section 3 introduces the proposed dynamic monitoring method. The experimental setup and the simulation results are presented in Sections 4 and 5. Finally, Section 6 winds up the article by outlining the main conclusions derived from this study.

2. Related work

2.1. Relevance-awareness

For the context of this article, "relevance" is used as a metric of how much information is conveyed by the observed data. The principle behind these methods is to decrease the monitoring efforts, and thus, save energy when the relevance is low; so as to use this extra energy to increase the monitoring effort when the relevance is high.

2.1.1. Adaptive sampling

Wireless sensor nodes wake up periodically to take a measurement, process it and transmit it through their wireless interface. Adaptive sampling schemes dynamically adjust the period between measurements (i.e., the sampling period) based on the perceived relevance of the data at the current moment. In [16], Das et al. use temporal correlation to calculate the utility of the sensor's observations. Then, they use reinforcement learning to dynamically adjust the sampling rate of the node. Silva et al. introduce LiteSense in [17]. LiteSense uses the mean deviation of the monitored signal to adjust the sampling rate. The mean deviation provides a low overhead approximation of the temporal correlation, making it interesting for its use in resource-constrained sensor nodes. The Kruskal–Wallis test is used in [18] by Tayeh et al. as the base parameter to adjust the sampling rate in a sensor node. The Kruskal–Wallis test provides an estimation of the variance of the measurements, which is used as an estimation of the temporal correlation of the signal. The sampling rate of the sensor node is calculated using a "behavior function" to take into account the risk level of the application. In [19], Lou et al. propose using linear regression to model the signal, and then, the temporal correlation between the samples is calculated based on the median jitter obtained after the line fitting has been updated.

2.1.2. Dual predictor scheme

Dual predictor schemes (DPS) use synchronized sample predictors, one located in the sensor node and the other located in the sink node. When the sensor node takes a measurement, it compares it with the predicted value. The data is only transmitted if the prediction error is outside the tolerated error range, thus reducing the radio usage by allowing some error in the collected data. In [20] Santini et al. presented a Least-Mean-Square (LMS) filter used for dual prediction. In the proposed approach the sink model is only updated using the transmitted data when the prediction error is out of range. Aderohunmu et al. [21] compare a Naive, Fixed-Weighted Moving Average (WMA), LMS, and ARIMA (AutoRegressive Integrated Moving Average) predictors; evaluating their accuracy and computational cost. In their evaluation, Aderohunmu et al. ended up favoring the Naive model over the rest, as it is the simplest method and has comparable characteristics to the rest. A combination of an LMS and an LSTM (Long Short Term Memory) prediction model is proposed by Shu et al. in [22]. The LMS model is able to predict linear changes in the data, while the LSTM model is able to predict non-linear changes. During operation, the node and the sink both use the LMS model. In case the prediction error for the LMS filter gets out of range, the node will try using the LSTM model. If the LSTM model is able to predict the measurement, the node will send a beacon message to the sink as an indication to change the model. If both prediction models fail, the sensor node will send the raw data to the sink.

2.1.3. Adaptive sampling and dual predictor scheme

A combination between DPS and an adaptive sampling scheme is proposed by Tayeh et al. in [18], which was already commented on in Section 2.1.1. The monitoring method proposed by Tayeh et al. adjusts the sampling period of the sensor node based on the signal variability, allowing it to be reduced if the signal is smooth; and the transmission rate of the DPS method depends on how well the signal is predicted. Both schemes operate independently from each other based on their initial configuration settings. In contrast, the monitoring method presented in this paper manages the adaptive sampling and DPS in a cooperative way. The sampling period and the transmission rate of the DPS are dynamically adjusted to comply with the allocated energy budget, and their respective values are calculated taking into account the configuration of the other scheme. This does not only allow controlling both schemes at the same time, but also increases the dynamic range of the node's response as if one of the values reaches its limit, the other value can be recalculated to compensate for this.

2.2. Energy-awareness

All the techniques discussed in Section 2.1 have been shown to outperform conventional systematic sampling strategies by exploiting the temporal correlation in the data. Alternatively, energy-aware methods adjust the behavior of the sensor node based on the characteristics of its energy source. Dai et al. in [23] proposed an energy-aware method to dynamically adjust the charging power of the supercapacitor in a hybrid power supply, based on the available charge, the harvested energy and the modeled event arrival rate to minimize the battery capacity loss while maintaining a high wake-up success rate. More critically, energy-aware methods may consider the available energy budget to guide the adaptive actions. A clear example of this is Heliomote, a solar-powered sensor node proposed by Kansal et al. in [24], which is capable of modeling the available solar energy and adjusting its energy consumption to match the solar energy generation so as to achieve sustainable operation. The energy consumption of Heliomote is adjusted by configuring its sampling period. Being able to model the energy budget of the node is also useful in battery-powered networks to homogenize the battery lifetime for the nodes. In [12], Martínez et al. proposed a lean sensing technique to monitor parking spot availability. The proposed method enables sensor nodes to forecast their power usage based on the activity level of the monitored spot and their sampling period. Then, the sensor nodes are able to adjust their sampling period to homogenize the battery discharge rate with the other sensor nodes in the network. These methods are able to satisfy the energy constraints but are oblivious to the relevance of the observed data.

2.3. Energy and relevance-awareness

Some efforts have been devoted to coordinating relevance-awareness with energy-awareness. In [25,26], Silva et al. discuss e-LiteSense, which is an upgrade of their previous work, LiteSense, that adds some level of energy-awareness to the initial relevance-aware technique. e-LiteSense technique is able to adjust the sampling rate based on the mean deviance of the data, and it also adjusts the reactivity of the adjustments based on the remaining battery charge. It can be noted that there is a clear difference in the energy-awareness level of e-LiteSense and the example methods discussed in Section 2.2. The techniques discussed in the previous section had to satisfy a given constraint. Heliomote had to operate continuously from its harvested energy, and the lean sensing approach had to homogenize the discharge rate of the nodes in the network. However, the energy management done by e-LiteSense does not provide any specific guarantee about its energy usage, as it only attempts to maximize its battery life in a best-effort nature. An effort to manage hard energy constraints while minimizing the monitoring error was proposed by Dang et al. in [27] with QuARES. The method proposed by Dang et al. uses a DPS to reduce the energy consumption of the sensor node and dynamically adjusts the maximum allowed prediction error based on the energy constraints. The goal of the proposed method is to minimize the prediction error, while managing the energy consumption based on the harvested energy. QuARES is partly coordinated by the sink node, which is responsible for predicting the energy generation, and based on that allocates the energy budget for the sensor node. The sensor node monitors the actual energy generation during operation and performs some adjustments to ensure it remains operating sustainably. Even though the main objective of QuARES is to minimize the prediction error, the actual monitored signal is not taken into account to modify the initial energy allocation. Consequently, it is not able to fully exploit the variability of the monitored signal.

The adaptive monitoring method presented in this article is an extension of a previous method presented in a different publication by the same authors [28]. The initially proposed method was capable of managing the sampling period of the sensor node to comply with the battery life target, allocating more resources when the signal was more variable to improve the overall monitoring accuracy. This method is not only able to meet its battery lifetime goals at the same time it outperforms the equivalent systematic monitoring method with the same battery life in terms of monitoring accuracy. Thus, showing it is able to allocate energy more efficiently. This work extends the existing one by introducing a new control variable in the form of the DPS, and the modeling and decision-making capabilities to split the monitoring efforts between the two control parameters. Adding the DPS reduces the energy consumption of the sensor node by reducing the number of radio transmissions while having minimal impact on the monitoring quality. Moreover, managing both control parameters in a coordinated way improves the efficiency of the monitoring method, since the adaptive action can take into account the impact of adjusting both parameters and select the most optimal configuration.

3. Energy and relevance-aware adaptive sampling

This section details the proposed energy and relevance-aware dynamic monitoring method. All notations used through this document can be found in Table 1.

3.1. Architecture

The proposed architecture follows a hierarchical agent-based design, where each agent is responsible for managing one specific task or module of the sensor node. Similar hierarchical agent-based architectures have been previously proposed to implement self-awareness in the context of wireless sensor networks (WSN), [29,30]. This kind of architecture is highly flexible and scalable, as any change in the underlying components just requires an update of their respective agent. It is also simple to scale, as new agents and interactions can be added in the future. A block diagram of the proposed architecture is shown in Fig. 1.

- The *sensor agent* is responsible for managing the data acquisition task. It reads the data from the internal sensor and computes the relevance index, which is used by the decision engine to model the perceived utility of the collected data.

Table 1
Notation references.

Symbol	Meaning
$\overline{X}$	Moving average of the sampled data.
α	Weight for the $\overline{X}$ EMA.
S	Sample taken by the node's internal sensor.
$\overline{V}$	Moving average of the deviation.
β	Weight for the for $\overline{V}_i$ EMA.
V_{max}	Maximum value for the mean deviation.
R	Relevance index.
$\overline{R}$	Moving average of the relevance index.
γ	Weight for the for $\overline{R}$ EMA.
G_r	Gain of the relevance index.
I_{pow}	Discharge rate of the energy-aware behavior.
I_{rel}	Discharge rate of the relevance-aware behavior.
TR	Transmission rate target of the DPS.
TR_{sat}	Minimum transmission rate of the DPS before saturation.
Δt	Sampling period.
Q_{bat}	Remaining battery charge.
Q_{tr}	Charge consumption with radio transmission.
Q_{ntr}	Charge consumption without radio transmission.
η	Weight for the Q_{tr} and Q_{ntr} EMA.
E_{SR}	Estimated error of the sampling period.
E_{TR}	Estimated error of the DPS.
ΔQ_i	Charge consumption since the last iteration.
$\overline{I}_{idle}$	Average idle current.
st	Step size for the DPS.
st_{max}	Maximum step size for the DPS.
P	Proportional Gain of the PI controller.
I	Integral gain of the PI controller.

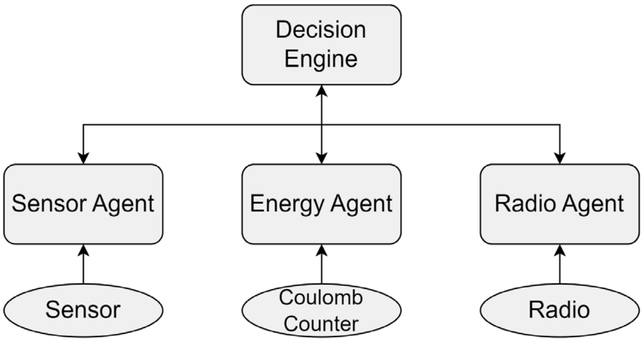

Fig. 1. Proposed agent-based architecture.

- The *Radio agent* is responsible for managing wireless communications. It implements the DPS to reduce the number of transmissions and reports the transmission rate to the decision engine, which is used to model and predict the power consumption of the sensor node.
- The *Energy agent* monitors the power consumption of the node with the use of a coulomb counter. It is also responsible for modeling the battery charge and predicting the effects in terms of energy consumption of any changes in the sampling period and the transmission rate.
- The *Decision engine* coordinates all the agents. The decision engine observes the data from all the agents, then decides the optimal configuration based on the operational goals, current operating conditions, and predicted effect of such configuration in terms of energy consumption and monitoring accuracy. Finally, the selected configurations are passed to the agents to guide their behavior.

3.2. Block diagram

A summary block diagram of the proposed method can be seen in Fig. 2. The overall method is driven by two behavioral tendencies: (1) *Energy-aware behavior*, which is the one responsible for modeling the power consumption and defining the energy budget required to achieve the target battery life. (2) *Relevance-aware behavior*, models the utility of the data, and performs momentary adjustments to the energy budget allocation to maximize the monitoring accuracy, taking advantage of the dynamic nature of the monitored data. The energy budget allocation is defined as a target for the discharge rate of the sensor node. The decision engine then uses this information to compute the optimal sampling period and transmission rate so that the target discharge rate is reached. This process uses the energy consumption profile model maintained by the energy agent to predict the energy consumption based on the selected configuration. Finally, the coulomb counter is used to measure the actual discharge rate of the sensor node and close the control loop by defining a new discharge rate.

The following subsections explain the different parts of the proposed algorithm in detail.

3.3. Energy-aware behavior

The energy-aware behavior monitors the consumption profile of the sensor node, along with the remaining battery life. For simplicity and generality, the model presented here assumes that the actual usable charge of the battery is known, and is used as an input parameter for the proposed model. Every time the sensor node wakes up to take a measurement, the Energy agent reads the accumulated charge measurement from the coulomb counter and calculates the charge consumption since the previous reading.

The remaining battery charge is updated using Eq. (1). The Energy agent also models the energy cost of performing a full sampling cycle $\left[Q_{tr}\right]$, and the energy consumed by the node to perform a sampling cycle when the radio transmission has been avoided $\left[Q_{ntr}\right]$. If there has been a radio transmission since the last time the coulomb counter was read, the Energy agent uses Eq. (2) to update the energy cost of a complete monitoring cycle. Otherwise, it uses Eq. (3) to update the energy cost of the monitoring cycle with no radio transmission. It should be noted that because of the energy consumption in sleep mode, the actual energy cost of a monitoring cycle will depend on the sampling period. This effect has been ignored since the sensor node is expected to operate around a constant sampling period only performing momentary adjustments, and adding this additional parameter would add unnecessary complexity to the model.

$$Q_{bat_i} = Q_{bat_{i-1}} - \Delta Q_{a_i} \tag{1}$$

$$Q_{tr_i} = (1-\eta)\, Q_{tr_{i-1}} + \eta \Delta Q_{a_i} \tag{2}$$

$$Q_{ntr_i} = (1-\eta)\, Q_{ntr_{i-1}} + \eta \Delta Q_{a_i} \tag{3}$$

The Energy agent uses this information to define the required discharge rate to reach the target battery life using Eq. (4).

$$\Delta I_{pow_i} = \frac{Q_{bat_i}}{t_{life_i}} \tag{4}$$

It is possible that the target battery life goal defined for the sensor node cannot be realistically achieved. In this case, even if the control variables are set to the maximum, or minimum, allowed, the sensor node would not be able to operate at the target discharge rate. As a result of the close loop system, the target discharge rate $\left[I_{pow_i}\right]$ would be increased to compensate for this effect on the next iteration. If this effect continues over time, the target discharge rate will start to drift. When the target discharge rate crosses a predefined threshold, the sensor node can send a request to the sink node for a new battery life target and alert that the current target will not be achieved.

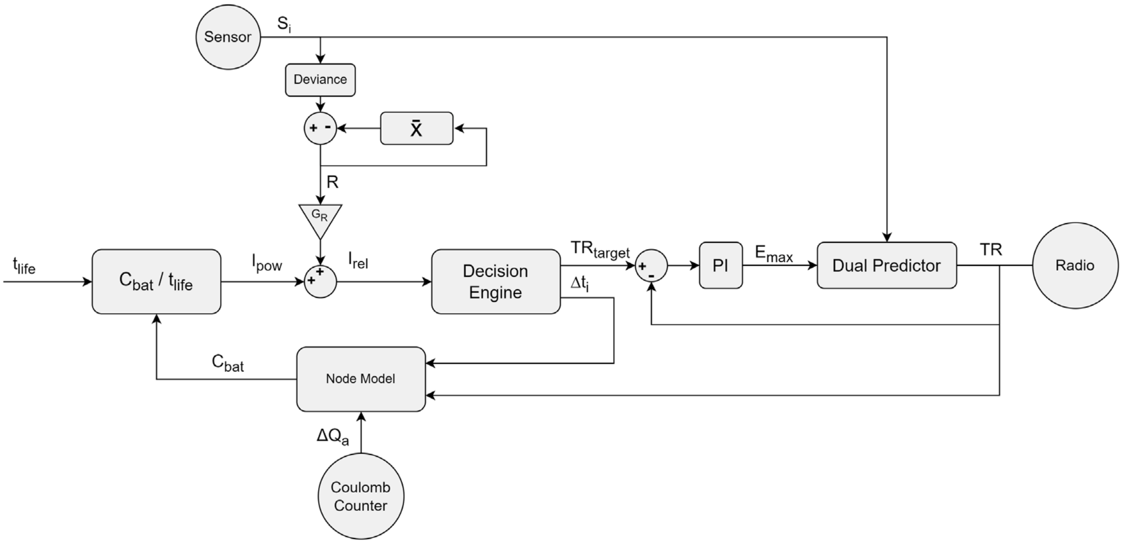

Fig. 2. Simplified block diagram of the proposed adaptive monitoring method.

3.4. Relevance-aware behavior

The relevance-aware behavior allocates the resources based on the perceived utility of the monitored data. The utility of the signal is modeled by the relevance index. The relevance of the signal is calculated using the mean deviation as it is a proven and lightweight method [17]. The mean of the monitored signal and the deviance are calculated using Eqs. (5) and (6) respectively. The Exponential Moving Average (EMA) prevents the need of storing historical data for the moving average.

$$\overline{X}_i = (1-\alpha)\overline{X}_{i-1} + \alpha S_i \tag{5}$$

$$\overline{V}_i = (1-\beta)\overline{V}_{i-1} + \beta\left|\overline{X}_i - S_i\right| \tag{6}$$

The data relevance is calculated as the normalized difference between the current deviation and the average one. For the relevance index in Eq. (7) the data relevance has the mean relevance index subtracted to ensure that the mean data relevance is zero. The γ parameter from Eq. (8) controls the memory period of the relevance index. If there has been a large peak in relevance, it will affect the subsequent measurements as it will have to compensate for the large increase in relevance. The γ parameter defines the length of the compensation period.

$$R_i = \frac{\left|\overline{X}_i - S_i\right| - \overline{V}_i}{V_{max}} - \overline{R}_{i-1} \tag{7}$$

$$\overline{R}_i = (1-\gamma)\overline{R}_{i-1} + \gamma R_i \tag{8}$$

The relevance index is used to adjust the discharge rate target defined by the energy-aware behavior from (4). The modification is performed using Eq. (9).

$$I_{rel_i} = (1 + G_r R_i) I_{pow_i} \tag{9}$$

The sign of the relevance index specifies if the monitored signal is considered more or less important than average, and thus, the sensor node should allocate more or fewer resources at this moment. The magnitude provides an indication of how drastic should the changes be. Since the mean value of the relevance index is zero, it will not impact the average discharge rate of the sensor node. This is the key principle behind the coordination between the relevance-aware and the energy-aware behaviors. While the energy-aware behavior defines the average discharge rate to achieve the lifetime goals; the relevance-aware behavior makes momentary adjustments to take advantage of changes in the utility of the signal, without affecting the average discharge rate. It should be noted that even if the relevance index is not exactly zero, because of the limited memory in the average relevance index, the close loop nature of the proposed method will be able to compensate for any discrepancies.

3.5. Decision engine

The proposed adaptive monitoring method presented here coordinates an adaptive sampling period and a dual prediction scheme. The decision engine takes the energy budget provided by the energy and relevance-aware behaviors and allocates it between the two schemes. The discharge rate of the node can be calculated using Eq. (10). For simplicity, this equation does not take into account the power consumption in idle mode as the node will mostly operate around a constant sampling rate, and thus, it will have very little effect.

$$I_{rel_i} = \frac{TRQ_{tr_i} + (1 - TR) + Q_{ntr_i}}{\Delta t_i} \tag{10}$$

Eq. (10) does not provide enough information to split the effort, and thus, it needs to be complemented by an additional policy. In the work presented here, a heuristic method is used to ensure that the estimated error introduced by both methods is kept equal. Nevertheless, different policies would also be valid.

The maximum error caused by the sampling rate can be calculated as the maximum possible change of the monitored signal in the sampling period.

$$E_{SR_i} = max\left(\frac{\delta S}{\delta t}\right)\Delta t_i \tag{11}$$

The maximum error of the DPS will be the maximum allowed prediction error. The tolerated error range is the input parameter of most DPSs [31]. However, in the presented work, the DPS is modified to use the transmission rate as input instead. This is further described in Section 3.6. Consequently, the maximum error introduced by the DPS is not a priori known. The relation between the transmission rate and the predictor error can be seen in Fig. 3, which shows the relation between the transmission rate and the prediction error for multiple sampling periods. The legend in the figure shows the different represented sampling periods expressed in s. A transmission rate of one means that all the values are transmitted. The graph clearly shows two regions of operation, a linear region when the transmission rate is close to one, and a saturation region for low transmission rates. The saturation region is the section of the curve where small variations of

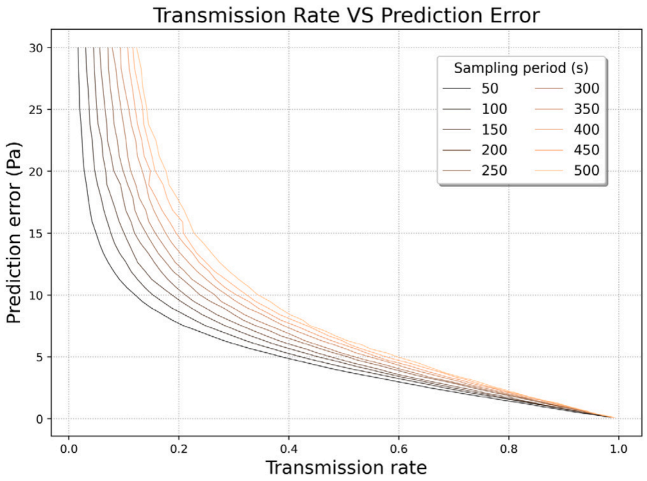

Fig. 3. Transmission rate and prediction error for different sampling periods.

the transmission rate produce large changes in the prediction error. Consequently, the dual predictor cannot be correctly controlled when operating in this region, and thus, it should be avoided.

If the dual predictor is forced to operate in the linear region, the relation between the transmission rate and the predictor error can be modeled as a line. It should be noted that the sampling period also has an effect on this relation. As it can be seen in Fig. 3, when the sampling period increases, the ability of the DPS to predict the signal decreases. However, the effect of the sampling rate is small compared to the effect of the prediction error or the transmission rate. Therefore, To keep the model simple, the relation between the transmission rate and the prediction error will be approximated by a best-fitting line using the median sampling period, resulting in Eq. (12).

$$E_{TR_i} = aTR_i + b \tag{12}$$

The maximum signal derivative used to estimate the error caused by the sampling period and the linear approximation used to estimate the prediction error, are highly dependent on the signal being monitored. To correctly estimate these parameters, the node can be configured to undergo an initial training period. During this period, the sensor node conducts measurements of the monitored signal and calculates its maximum derivative and the best-fitting line that relates the transmission rate to the prediction error.

In the proposed heuristic the estimated errors from Eqs. (11) and (12) are set equal. Solving in Eq. (10), the sampling period and the transmission rate can be calculated with Eq. (13) and (14) respectively.

$$\Delta t_i = \frac{b\left(Q_{tr_i} - Q_{ntr_i}\right) - aQ_{ntr_i}}{max\left(\frac{\delta S}{\delta t}\right)\left(Q_{tr_i} - Q_{ntr_i}\right) - aI_{rel_i}} \tag{13}$$

$$TR = \frac{max\left(\frac{\delta S}{\delta t}\right)\Delta t_i - b}{a} \tag{14}$$

It may happen that the sampling period calculated using Eq. (13) is outside the allowed range defined for the node. Also, the transmission rate calculated with Eq. (14) may be in the saturated region or higher than one. In these cases, the parameter that has gone out of range can be set to the limit value, while the other parameter is adjusted to compensate for this effect. This shows that the proposed adaptive monitoring method can take advantage of having two control parameters to extend its dynamic range. The compensated transmission rate and sampling period can be obtained using Eq. (15) and (16) respectively.

$$\Delta t_i = \frac{TRQ_{tr_i} + (1 - TR) + Q_{ntr_i}}{I_{rel_i}} \tag{15}$$

$$TR = \frac{I_{rel_i}\Delta t_i - Q_{ntr_i}}{Q_{tr_i} - Q_{ntr_i}} \tag{16}$$

In more extreme cases the sampling period and the transmission rate can both hit their limit points at the same time. In such conditions, the sensor node will not be able to follow the target discharge rate I_{pow_i}, causing it to drift over time as the control loop tries to compensate for it. As identified in Section 3.3, if this condition is sustained over time, the sensor node will raise an alarm.

3.5.1. *Adaptive sampling period*

The sensor node remains in sleep mode most of the time, only waking up periodically to take a measurement, process it, and if applicable, transmit it using its wireless interface. The time between one node waking up from the event and the next is the sampling period. The sampling period is managed by setting an alarm in the internal Real-Time Clock (RTC) of the sensor node. When the defined time has passed the RTC generates an interrupt to wake up the main microcontroller of the sensor node, thus, starting a new monitoring cycle. Before going back to sleep mode, the sensor node updates the next waking up time on the RTC using the updated value of the sampling period, so it takes effect immediately.

3.6. *Dual prediction scheme*

Generally, the wireless interface has the greatest contribution to the overall energy consumption of wireless sensor nodes [32]. Dual Predictor Schemes (DPSs) [31] attempt to reduce the number of radio transmissions using two linked data predictors. A local sample predictor in the sensor nodes is kept synchronized with an equivalent predictor located in the server or sink node. When the sensor node collects a measurement, it compares the measurement with the value predicted by the model. If the prediction error is within the acceptable error range, the measurement is not transmitted, so the sink node will use the value predicted with its model. Otherwise, if the prediction error is out of bounds, the sensor node sends its measurement to the sink and both of them update their model with the new value. The model is only updated using the transmitted values, so no additional synchronization messages between the node and the sink node are needed to keep the models aligned. The model update and prediction procedures are detailed in Algorithm 1, the prediction algorithm is a first-order LMS [20] predictor with the maximum step size limited to prevent it from becoming unstable.

Algorithm 1 Common procedures for the dual predictor algorithm

```
  Parameters: $stepsize$, $maxstep$
procedure UpdateModel($\epsilon_i, w$)
    $st \leftarrow \epsilon_i * stepsize$
    $st \leftarrow \max(\min(st, maxstep), -maxstep)$
    $w \leftarrow w + st$
    return $w$
procedure Predict($\Delta t_i, V_c, w$)
    $p \leftarrow V_c + \Delta t_i * w$
    return $p$
```

The control parameter of the DPSs is the tolerated prediction error. However, one of the key aspects of the proposed adaptive monitoring method is that it manages the discharge rate of the node, and thus, it is more convenient to directly control the transmission rate. On top of this, with a conventional DPS, the current transmission rate cannot be known, since it requires averaging the number of transmissions and skips over a period of time. The decision engine requires knowledge about the current transmission rate to estimate the power consumption of the sensor node and select the appropriate adaptive actions. To integrate the DPS in the proposed method, the transmission rate needs to be known and more or less maintained around the target value.

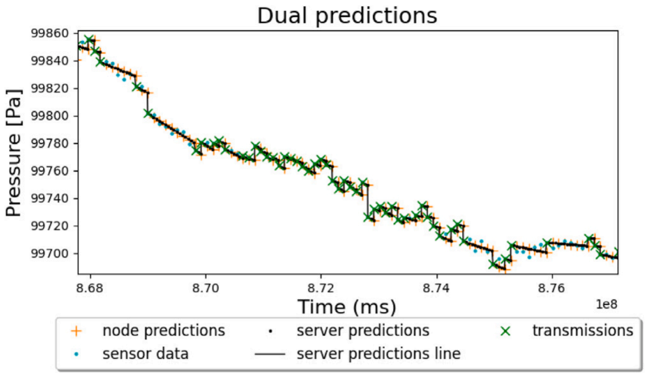

Fig. 4. Visualization of the dual prediction scheme.

In our proposed method, the DPS uses a proportional–integral (PI) controller, which automatically adjusts the tolerated prediction error of the DPS to achieve the target transmission rate. This PI controller can be seen in Fig. 1. The integral component of the PI controller ensures that the accumulated error between the target transmission rate and the obtained one gets compensated over time, and thus, the energy consumption prediction of the DPS is correct. The output of the PI controller is forced to remain within a defined operating range. The operating range is defined between zero, as the prediction error is in absolute value; and the maximum allowed prediction error, which is defined based on the application needs. In some cases, because of the limitation of the tolerated prediction error, the target transmission rate may not be possible to achieve, causing a windup effect on the PI controller. To prevent the PI from becoming unstable, whenever the output of the PI controller has to be limited, the integral error is set to zero. Resetting the integral error implies that the error in the transmission rate is not compensated by the DPS, but as it will have an impact on the energy consumption of the node, it will be later compensated by the energy-aware behavior of the proposed method. It should be noted that the DPS is generally expected to operate far from this limit case, so this will have a limited effect on the overall behavior of the DPS. The dual prediction algorithm that is running on the node is described in Algorithm 2.

Algorithm 2 Dual prediction algorithm for the node

```
Parameters: stepsize, maxstep
$w \leftarrow 0$
$V_c, t_c \leftarrow$ WaitForData()
$\epsilon_c \leftarrow 0$
while Predictor is running do
    $V_i, t_i \leftarrow$ WaitForData()
    $\Delta t_i \leftarrow t_i - t_c$
    $V_p \leftarrow$ Predict $(\Delta t_i, V_c, w)$
    $\epsilon_i \leftarrow V_i - V_p$
    if $|\epsilon_i| \geq \epsilon_c$ then
        $V_c \leftarrow V_i$
        $TR \leftarrow 1$
        $t_c \leftarrow t_i$
        $w \leftarrow$ UpdateModel $(w, \epsilon_i)$
        TransmitData $(V_i)$
    else
        $TR \leftarrow 0$
    $\epsilon_c \leftarrow$ ProporcionalIntegralGetError $(TR, \Delta t_i)$
```

Another characteristic of the proposed adaptive sampling scheme is the dynamic sampling period. The sensor node will adjust its sampling period based on the operating conditions. Consequently, the server or sink node does not have visibility of the sampling period of the sensor node. The server will generate samples periodically using its predictor with its own defined period, which may not be aligned with the sensor node. When a message from the node is received, the prediction model will be updated in an asynchronous way. Setting a fixed *virtual* sampling rate on the server-side, masks the dynamic nature of the sampling period of the sensor node, allowing the final application to operate oblivious to it [18]. The algorithm for the dual prediction scheme running on the sink node is detailed in Algorithm 3.

Table 2
Tested adaptive monitoring schemes.

Label	Description
Systematic: Opt	Systematic with no adaptations.
E & R: Complete	Proposed scheme.
E & R: Only Power	Only energy-aware.
E & R: Only SR	Only adaptive period.
E & R: Only TR	Only with DPS.
E & R: Prev	Prior publication.

Algorithm 3 Dual prediction algorithm for the server

```
Parameters: stepsize, maxstep
$w \leftarrow 0$
$V_c, t_c \leftarrow$ WaitForNodeData()
$\epsilon_c \leftarrow 0$
while Predictor is running do
    $t_i \leftarrow$ WaitForServerDataUpdate()
    while DataAvailableFromNode() do
        $V_i, t_i \leftarrow$ GetNodeData()
        $\Delta t_i \leftarrow t_i - t_c$
        $V_p \leftarrow$ Predict $(\Delta t_i, V_c, w)$
        $\epsilon_i \leftarrow V_c - V_p$
        $t_c \leftarrow t_i$
        $V_c \leftarrow V_i$
        $w \leftarrow$ UpdateModel $(w, \epsilon_i)$
    $\Delta t_i \leftarrow t_i - t_c$
    $V_p \leftarrow$ Predict $(\Delta t_i, V_c, w)$
    UpdateServerData $(V_p)$
```

An example of the DPS in action can be seen in Fig. 4. The figure shows how both models are updated whenever there is a radio transmission, which is represented by the green marker, and it can be seen that they remain perfectly synchronized. In the figure, the server has a virtual sampling period of 10 s, while the sensor node has an adaptive period between 50 and 200 s. It can be seen that even if the virtual sampling period defined on the server side is different from the sampling period of the node, the DPS is still able to work as expected.

4. Experimental setup

This section details the proof-of-concept simulation used to assess the proposed dynamic energy management method. The proposed method was compared with a systematic sampling scheme, which does not apply any kind of adaptations; the equivalent method with no relevance-aware behavior; the equivalent method but only adjusting the sampling rate; the equivalent method, but only adjusting the transmission rate; and the method from our previous publication, which did not use the DPS. The systematic sampling scheme is not able to manage its energy consumption, so the sampling rate for this scheme will be manually fixed so that its battery life matches the target battery life defined for the other methods. For the two methods where one of the control parameters has been disabled (“$E\&R$: $Only\ SR$” and “$E\&R$: $Only\ TR$”), the disabled parameter is manually set to match the mean value obtained by the complete method in the simulation. The reference names used for each of the tested methods are detailed in Table 2

4.1. Datasets

Data from three different sources have been used for the simulation (1) atmospheric pressure, (2) relative humidity, and (3) light intensity. The dataset contains two different sets of measurements for each of the considered sources. Each set of data was collected in the same environment, using the same device, but at different times. One of the sets was used as training data to adjust the model parameters, and the other set of data was used to validate the proposed method. This ensures that the selected parameters have not been overfitted. The dataset was recorded with a sampling period of 10 s over several weeks. This dataset have been made publicly available in [33].

The ability to take advantage of the changes in the relevance of the data is highly dependent on how dynamic is the monitored signal. If the temporal correlation of the signal is more or less constant over time, the relevance-aware behavior will have very little effect on the overall performance of the proposed method. On the contrary, if the signal has drastic changes in the temporal correlation of the data, the relevance-aware behavior will have more room for improvement. The atmospheric pressure data has been selected as it has mostly a constant temporal correlation. The light intensity data has very large variations in the temporal correlation as it has periods of darkness with very little variability, followed by sudden periods of illumination with high variability. Finally, the relative humidity data sets a nice middle point in terms of variability of its temporal correlation.

In the simulation, it may be possible that the sampling period of the evaluated methods is not a multiple of the 10 s sampling period used to record the original dataset. In such cases, the intermediate values were obtained as the linear interpolant between the nearest points of the original data. The interpolant is calculated using Eq. (17), where S_0, S_1 are nearest points in the original time series; and t_0, t_1 are their respective timestamps.

$$S_i = S_0 - (t_i - t_0)\,\frac{S_1 - S_0}{t_1 - t_0} \tag{17}$$

4.2. Node model

The node model in the simulation is used to calculate the energy consumption of the sensor node. The energy consumption of wireless sensor nodes in [34,35] is modeled considering the energy contribution for each of the different tasks of the node (i.e., sleep, sampling and processing, and transmission). In the simulation, the energy consumption of the sensor node is evaluated every time it wakes up to take a measurement using Eq. (18). The charge consumption when there has been a radio transmission is represented by Q_{tr}, and the charge consumption without a radio transmission is represented by Q_{ntr}. The energy consumption of the sensor node is also used to generate the measurement of the coulomb counter measurement, which is an input parameter for the proposed monitoring method. It should be noted that the simulation model does take into consideration the current consumption in sleep mode to accurately model its power consumption. However, the battery model used in the simulation, still only considers the usable charge of the battery, so as to avoid chemistry-dependent effects caused by temperature or discharge rates [36], which would add unnecessary complications to the simulations.

$$\Delta Q_i = \begin{cases} Q_{tr} + Hours\left(\Delta t_i\right)\overline{I}_{idle} & \text{if } Transmission \\ Q_{ntr} + Hours\left(\Delta t_i\right)\overline{I}_{idle} & Otherwise \end{cases} \tag{18}$$

4.3. Computational cost of the energy and relevance-aware monitoring method

The proposed energy and relevance-aware monitoring method runs on the sensor node, and thus the energy required to run this method adds to the total energy consumption of the node. In order for the monitoring method to be practical, the energy saved by the method needs to offset the energy overhead it creates. The reference wireless sensor node utilized in this paper employs a Cortex-M4-based microcontroller with a clock speed of 48 MHz. This microcontroller has a Floating Point Unit (FPU), allowing it to perform one floating point operation per clock cycle. Based on the approximation of less than 1M operations being introduced by the proposed monitoring method; and with a processor active current consumption of 20 mA; the cost of running the proposed monitoring method for a single cycle is 0.115 μAh. The power consumption required to sample and process the measurement (a complete cycle without counting the energy required to transmit the data), for this example sensor node is 0.04 mAh. It should be noted that to collect a measurement, most of the energy consumption of the sensor node is due to the warm-up time of the sensor node and the oversampling of the sensor data, as shown in Fig. 5. The figure is a capture of the node's current consumption during a complete monitoring cycle taken from a power profiler, which shows the different tasks of the node. In general, it can be seen for the reference sensor node considered for this paper, the energy overhead caused by the proposed method is negligible compared to the energy required to acquire the data.

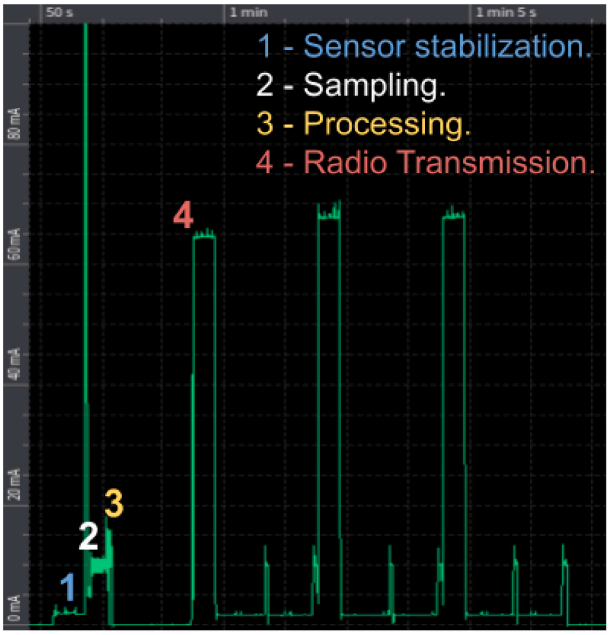

Fig. 5. Typical consumption profile of a wireless sensor node. X-axis shows the time in seconds, and the Y-axis shows the instantaneous current in mA.

4.4. Performance metrics

The battery and node models from Section 4.2 are used to model the remaining battery life and the energy consumption of the nodes. Comparing the battery life achieved in the simulation with the target battery life provides a direct indication of the ability of the evaluated method to reach the target battery life.

Assessing the monitoring accuracy of the node is not so straightforward, more so, considering that for each of the evaluated models, the samples may be collected at different times. To compare the data, it is resampled to match the 10 s period of the original dataset. For the methods that use the DPS, the predictor on the server side was directly used to resample by setting the virtual sampling period to 10 s. For the rest of the models, a Naive predictor [21] was used to resample the data. Finally, the monitoring accuracy is assessed by computing the Mean Squared Error (MSE) and Mean Absolute Error (MAE) between the resampled signal and the original. The quadratic nature of the MSE

Table 3
Common simulation settings.

Symbol	Value	Symbol	Value
α	0.1	$\bar{I}_{idle}$	0.1 mA
β	0.1	$\Delta\bar{Q}_{ntr}$	0.19 mAh
γ	0.0005	$\Delta\bar{Q}_{tr}$	0.04 mAh
G_r	10	η	0.1
Q_{max}	2500 mAh	st	0.000001
Δt_{min}	50 s	st_{max}	0.000005
Δt_{max}	200 s	P	0.05
t_{life_0}	400 h	I	0.1
TR_{sat}	0.2		

makes it more sensitive to large error spikes, while the MAE will be more representative of the average error.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(X_i - \hat{X}_i\right)^2 \tag{19}$$

$$NME = \frac{1}{n}\sum_{i=1}^{n}\left|X_i - \hat{X}_i\right| \tag{20}$$

where X_i is the sample from the evaluated method and $\hat{X}_i$ is the sample from the original dataset.

5. Experimental results

The simulation results are presented in this section. The proposed adaptive monitoring method has been tested in five different tests. The first three tests evaluate the proposed method using data from three different sources of data (atmospheric pressure, relative humidity, and intensity light). The fourth test modifies the operating conditions during the simulation to evaluate the ability of the evaluated method to react to changes. The final test is a more severe version of the fourth test, where the operating conditions are modified to the point that the sensor nodes can no longer meet their battery life. The fifth test demonstrates the ability of the proposed method to recognize this issue and request a new target.

The generic configuration parameters for the simulation are shown in Table 3. This table contains the general parameters for the simulation, such as the power consumption of the sensor node, maximum and minimum sampling period values, etc. Along with some configuration settings of the proposed method. It should be noted that these configuration settings have little effect on the actual performance of the proposed method, and thus, they do not need to be adjusted depending on the data source. On top of these, there are some parameters that are set specifically for each of the evaluated sources of data from Section 4.1.

5.1. Atmospheric pressure data

The specific configuration parameters used for the atmospheric pressure simulations are detailed in Table 4.

The atmospheric pressure data is used as an example of a data source where the data relevance is mostly constant, and therefore, the relevance-aware behavior will have very little effect on the overall result. The results from the simulation are shown in Table 5. As can be seen, all the evaluated methods have a battery life of 400 h. Figs. 6, 7, 8 show the evolution over time of the sampling period, the transmission rate, and the discharge rate respectively. During this test, the control parameters from all the tested schemes are very similar. This is to be expected since the atmospheric pressure data has very little variability in its data relevance. All of the tested methods are able to outperform the $Systematic : Opt$ scheme and the $E\&R : Prev$ methods, which are the only methods that do not implement the DPS. Allocating some error budget to the DPS allows the sensor node to decrease the sampling period, without affecting the monitoring accuracy in a significant way.

Table 4
Configuration settings for the atmospheric pressure simulation.

Symbol	Value
a	−11.915 Pa
b	11.741 Pa
max (dS/dt)	1.766E−05 Pa/ms

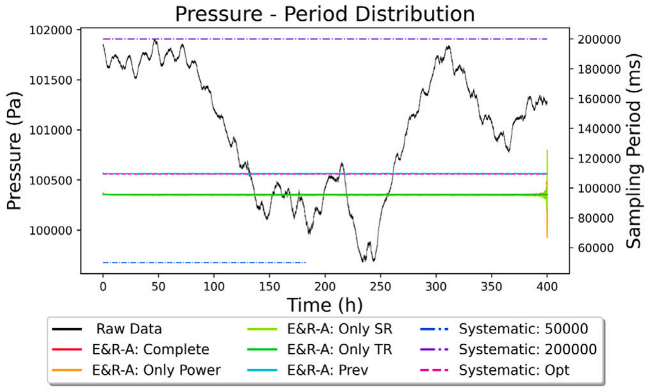

Fig. 6. Sampling period over time for the atmospheric pressure simulation.

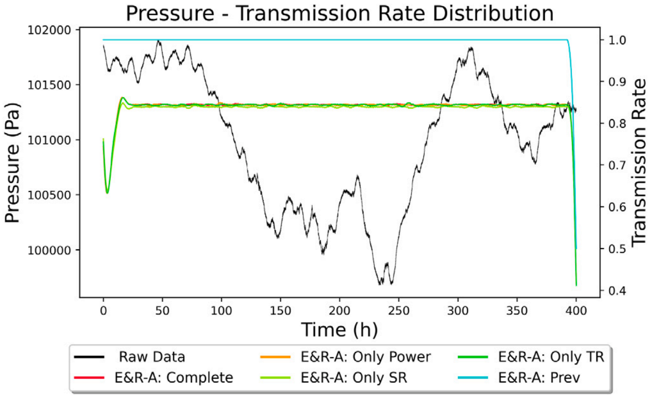

Fig. 7. Transmission rate over time for the atmospheric pressure simulation.

Using our proposed method ($E\&R : Complete$) resulted in a reduction of 2.5% and 1.2% of the MSE and MAE respectively, compared with the systematic method ($Systematic : Opt$); and a reduction of 2.1% and 1.1% of the MSE and MAE respectively compared with our previous method ($E\&R : Prev$). It should be noted that the atmospheric pressure data is presented as a worst-case scenario where the proposed method is expected to have minimal impact.

5.2. Relative humidity data

The specific configuration parameters used for the relative humidity simulations are detailed in Table 6.

The relative humidity data is used as an example of a data source with intermediate variability in its data relevance. The result from the simulation can be seen in Table 7. As in the previous test, all the evaluated methods were able to successfully reach the target lifetime of 400 h. The simulation results show that all the evaluated methods that include the DPS, have an average sampling period close to the minimum allowed, and the transmission rate is close to the saturation point. Therefore, most of the error is allocated to the DPS, indicating that it is much more effective than the adaptive period for this data source. This is further confirmed in Figs. 9–11. For this data source, the complete method and the method only adjusting the SR obtained

Table 5
Atmospheric pressure simulation results.

Scheme	Battery Life [h]	MSE [Pa^2]	MAE [Pa]	Avg Period [ms]	Avg TR
Systematic: Opt	399.46	28.841	4.164	109 300	1.000
E & R: Complete	400.07	28.115	4.113	95 520	0.839
E & R: Only Power	400.08	28.431	4.139	95 510	0.839
E & R: Only SR	400.06	28.371	4.130	95 091	0.834
E & R: Only TR	400.14	28.352	4.140	95 505	0.838
E & R: Prev	400.05	28.707	4.157	109 462	1.000

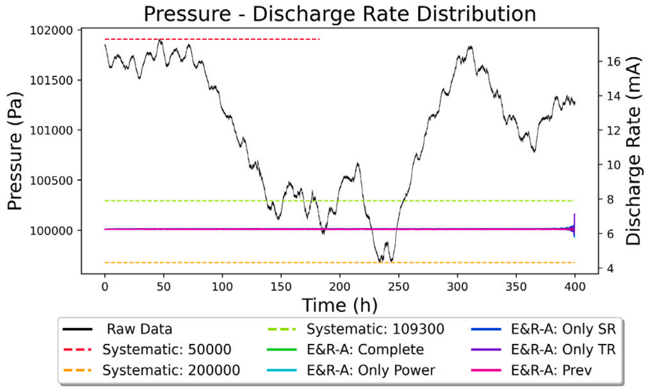

Fig. 8. Discharge rate over time for the atmospheric pressure simulation.

Table 6
Configuration settings for the relative humidity simulation.

Symbol	Value
a	−11.469%
b	4.937%
max (dS/dt)	7.604–05 %/ms

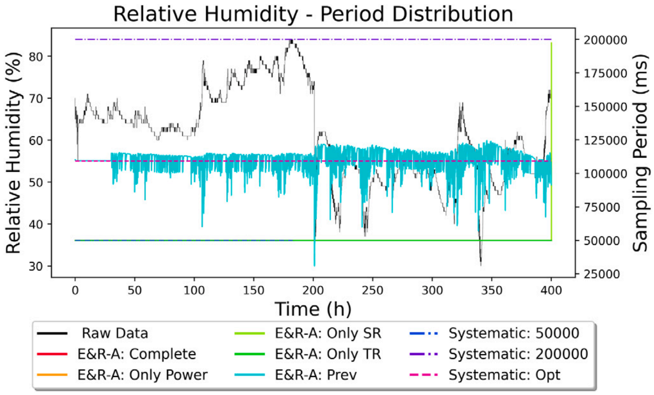

Fig. 9. Sampling period over time for the relative humidity simulation.

the same results, since for this data source both methods maintained the DPS close to its saturation point and only controlled the sampling rate. Using our proposed method ($E\&R$: $Complete$) resulted in a reduction of 17.5% and 18.1% of the MSE and MAE respectively, compared with the systematic method ($Systematic$: Opt); a reduction of 16.8% and 17.3% of the MSE and MAE respectively compared with our previous method ($E\&R$: $Prev$); and a reduction of 19.5% and 12.24% of the MSE and MAE respectively compared with the method without relevance-aware behavior ($E\&R$: $Only\ Power$).

5.3. Light intensity data

The specific configuration parameters used for the light intensity simulations are detailed in Table 8.

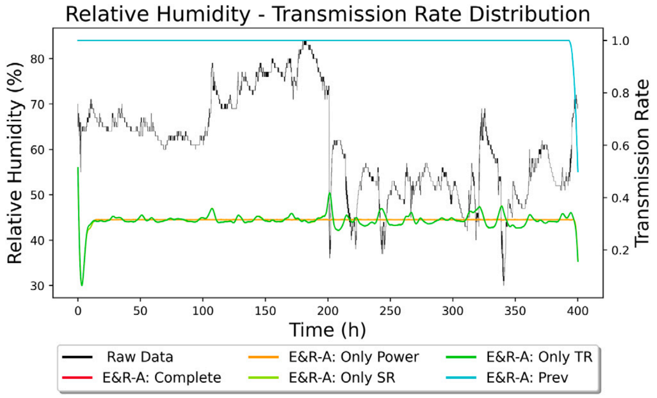

Fig. 10. Transmission rate over time for the relative humidity simulation.

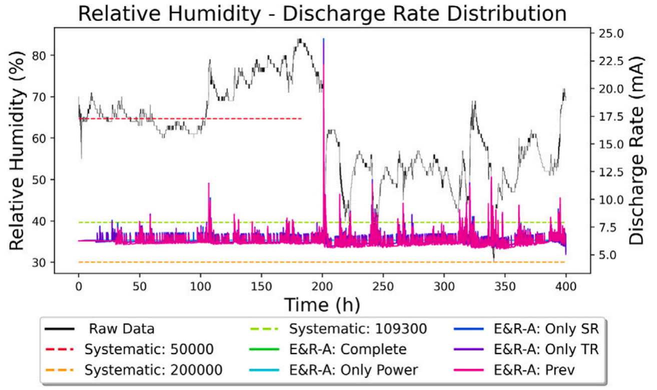

Fig. 11. Discharge rate over time for the relative humidity simulation.

The light intensity data shows large variations in its data relevance, and consequently, it is the data source where the relevance-aware schemes are expected to have the greatest effect. Figs. 12–14 clearly show this, as the period, transmission rate, and discharge rate suffers greater variations on all the methods with the relevance aware behavior enabled. The result from the simulation can be seen in Table 9. In this test, the proposed scheme is able to outperform all the other evaluated methods, including the other relevance-aware schemes. The proposed method ($E\&R$: $Complete$) was able to reduce the MSE and MAE by 41.3% and 22.4% respectively compared with the systematic method ($Sytematic$: Opt); by 22.7% and 9.8% compared with our previous method ($E\&R$: $Prev$); and by 10.1% and 20.7% respectively compared with the method without the relevance-aware behavior ($E\&R$: $Only\ Power$).

In Table 8, it can be seen that the method with only control over the transmission rate ($E\&R$: $Only\ TR$) obtained a higher MSE than the method without the relevance-aware behavior, however, the MAE was lower for this method. For the light intensity data, increasing the sampling period during the periods of high variability decreases the time it takes to detect a change in the light level, which introduce high peak errors and thus increase the MSE. By not being able to increase the

Table 7
Relative humidity simulation results.

Scheme	Battery Life [h]	MSE [$\%^2$]	MAE [%]	Avg Period [ms]	Avg TR
Systematic: Opt	399.46	0.120	0.105	109 300	1.000
E & R: Complete	400.16	0.099	0.086	50 003	0.312
E & R: Only Power	400.13	0.123	0.098	50 003	0.312
E & R: Only SR	400.17	0.099	0.086	50 005	0.312
E & R: Only TR	400.11	0.117	0.108	50 003	0.312
E & R: Prev	400.06	0.119	0.104	109 463	1.000

Table 8
Configuration settings for the light intensity simulation.

Symbol	Value
a	−18.006 lx
b	8.344 lx
max (dS/dt)	10.139−06 Lx/ms

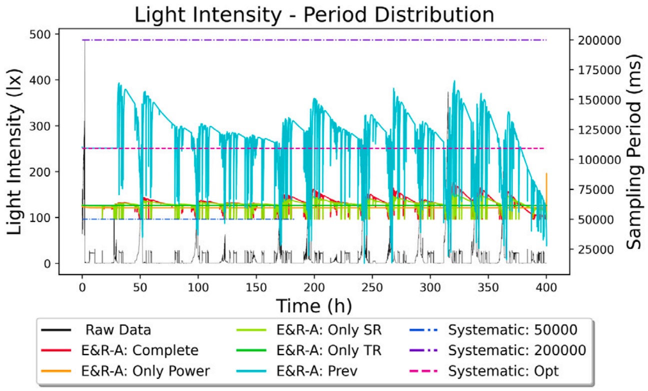

Fig. 12. Sampling period over time for the light intensity simulation.

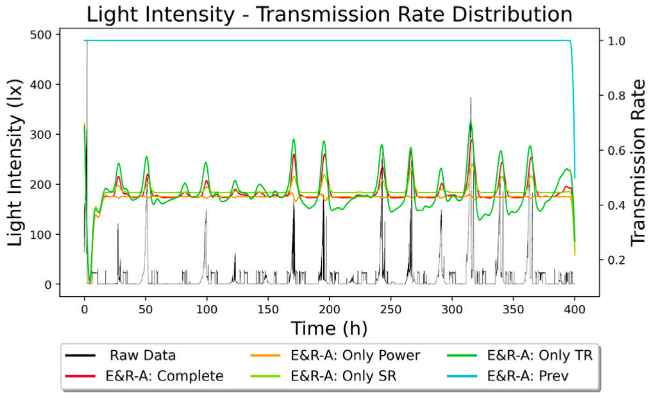

Fig. 13. Transmission rate over time for the light intensity simulation.

sampling periods during these moments, the $E\&R : Only\,TR$ method was not able to reduce the MSE.

5.4. Reaction test

The reaction test evaluates the ability of the proposed monitoring method to react to changes in the operating conditions, while still being able to accomplish the battery life goals and exploit relevance-awareness. The reaction test was performed using the atmospheric pressure data with the relevant configurations from Table 4. During the reaction test, at the 200 h mark, the radio power consumption Q_{tr} was increased from 0.19 mAh to 0.29 mAh. In Figs. 15, 16, and 17, the event time is marked by a horizontal purple line. In Fig. 17 it can be seen that the discharge rate is not visibly affected after the event, however, the transmission rate and the sampling period do change as a consequence of the event. This is the case because the node continuously monitors the energy consumption and recomputes the control parameters with every activation. It should be noted that the transmission rate represented in Fig. 16 appears to react to the change before it takes place. This only appears to be the case because the transmission rate signal represented in the plot has been smoothed using a Gaussian kernel. This needs to be done because the transmission rate is either 1 if there has been a transmission or 0 if it was skipped, and the relevant parameter is the average transmission rate. The results from the simulation are shown in Table 10. As can be seen, all the energy-aware schemes evaluated are able to achieve the target battery life. Consequently, showing the ability of these schemes to successfully react to the event. The results also show that the proposed monitoring method is able to outperform the rest by obtaining a lower MSE and MAE. It can be noted that for this test case, the systematic method obtained lower MSE and MAE values, but this is only the case because it used more energy than the other methods, as can be seen by its lower battery life.

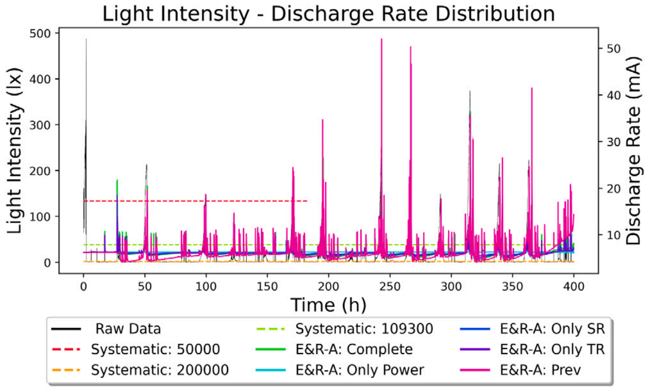

Fig. 14. Discharge rate over time for the light intensity simulation.

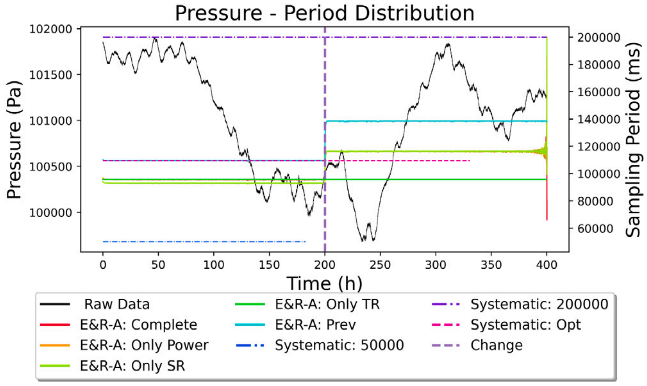

Fig. 15. Sampling period over time for the event simulation.

Table 9
Light intensity simulation results.

Scheme	Battery Life [h]	MSE [lx^2]	MAE [lx]	Avg Period [ms]	Avg TR
Systematic: Opt	399.46	40.651	0.901	109 300	1.000
E & R: Complete	400.11	23.857	0.699	61 491	0.445
E & R: Only Power	400.12	26.528	0.882	59 761	0.425
E & R: Only SR	400.10	25.029	0.758	63 423	0.467
E & R: Only TR	400.10	28.468	0.726	61 770	0.448
E & R: Prev	400.09	30.844	0.775	109 465	1.000

Table 10
Event simulation results.

Scheme	Battery life [h]	MSE [Pa^2]	MAE [Pa]	Avg period [ms]	Avg TR
Systematic: Opt	330.69	28.823	4.155	109 300	1.000
E & R: Complete	400.09	29.119	4.188	104 848	0.824
E & R: Only Power	400.04	29.144	4.193	104 866	0.824
E & R: Only SR	400.09	29.309	4.208	103 248	0.807
E & R: Only TR	399.90	29.977	4.252	95 600	0.671
E & R: Prev	400.08	29.823	4.232	122 193	1.000

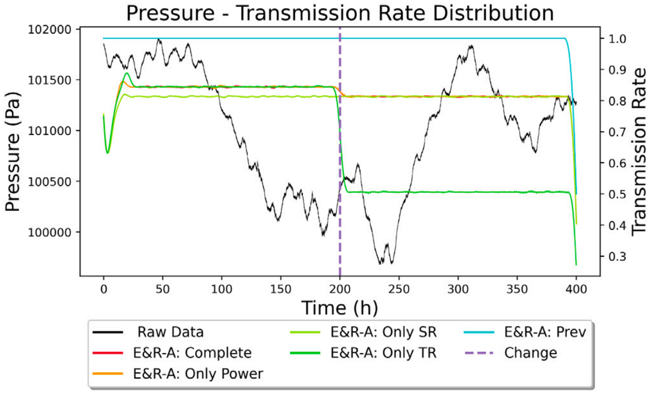

Fig. 16. Transmission rate over time for the event simulation.

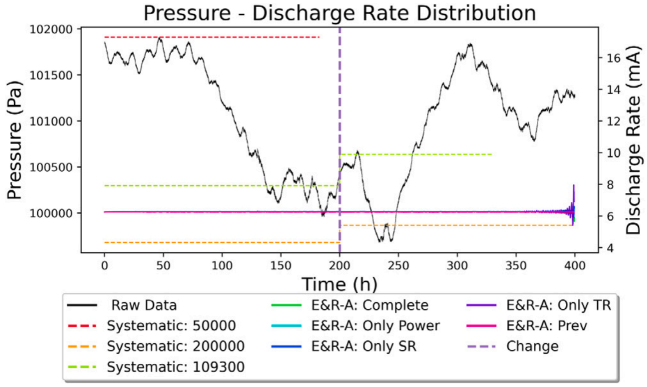

Fig. 17. Discharge rate over time for the event simulation.

5.5. Extreme reaction test

There may be situations where achieving the target battery life is not feasible within the allowed operating range. In such cases, the energy-aware behavior should detect this condition and promptly notify the server to request a more realistic battery life target. The extreme reaction test assesses the effectiveness of the proposed monitoring method in recognizing when the battery life cannot be met, requesting a new battery life target, and making the appropriate adjustments to comply with the new goal. As in the previous test, it uses the atmospheric pressure data with the relevant configurations from Table 4. For the extreme reaction test, at the 200 h mark, the radio power consumption Q_{tr} was increased from 0.19 mAh to 2 mAh. When the target battery life cannot be achieved, the power index will start drifting. For this simulation, the sensor nodes were programmed to raise an alarm at any point of the simulation if the power index decreased below −100. After this alarm, the target battery life was updated from 400 h to 250 h.

Fig. 19 shows the remaining battery charge through the simulation, once the event takes place, which is marked by the vertical purple line, the remaining battery charge starts to decrease rapidly. As can be seen in Fig. 18, not all the methods detect the alarm at the same time. The time at which the alarm is raised can be seen by the sudden increase in their discharge rate, as the monitoring methods adjust for the new battery life target. The complete and only power-aware methods take the longest to raise the alarm since they are both able to maintain the previous discharge rate for some time, even after the event takes place. In contrast, our previously proposed method is the first to raise the alarm, since it is unable to maintain the current discharge rate. This shows that being able to adjust the sampling period and the transmission rate extends the range at which the proposed monitoring method can adapt. Moreover, this is further confirmed by the fact that the $E\&R : Only\ SR$ and our previously proposed method cannot correctly adjust even after the new battery life target is passed, and start to drift at the end of the simulation. Overall, the results demonstrate that the proposed monitoring method is not only capable of detecting when the battery life target cannot be realistically achieved, but also dynamically adjusting when the battery life target is changed at runtime.

5.6. Analysis of the experimental results

The DPS used in the proposed monitoring method was modified so that the transmission rate is kept at the specified value using a PI controller. The ability of the PI to control and maintain a fixed transmission rate can be seen in Figs. 7, 10 and 16, as the transmission rate for the proposed method with only the adaptive sampling rate enabled, is kept approximately constant. For the light intensity data, Fig. 13, it can be seen that there is some variability in the transmission rate since this signal experiences large changes in variability when the sensor is suddenly illuminated. During this period, the required prediction error to maintain the specified transmission rate becomes larger than the maximum error setting of the DPS, and thus it cannot be achieved. At this point the anti-windup mechanism acts, allowing it to recover once the period of high variability ends.

One of the key aspects of the proposed monitoring method is its ability to comply with hard energy constraints, by being able to achieve

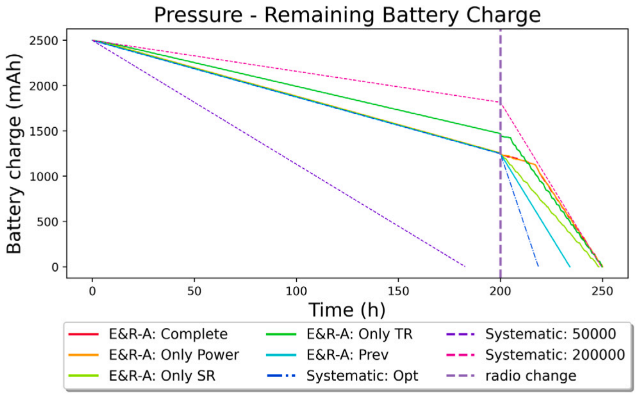

Fig. 18. Discharge rates measured during the extreme event simulation.

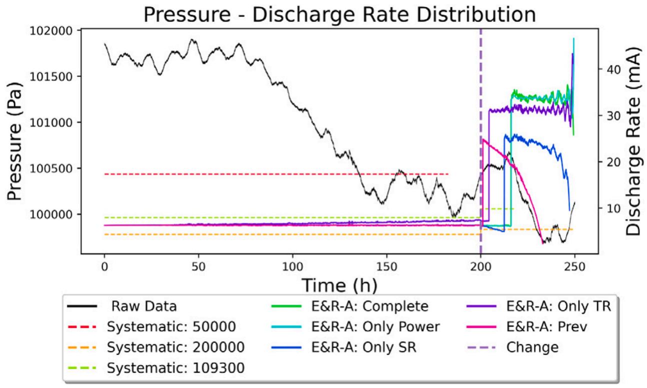

Fig. 19. Discharge rate over time for the extreme event simulation.

the target battery life. This is managed by the energy-aware behavior of the proposed monitoring method. The tests demonstrate that the proposed method is able to successfully meet the battery life target for the three different data sources, even when the operating conditions are artificially modified. The extreme event test shows that when the target battery life target is unachievable with the current operating constraints (e.g., power consumption of the sensor node, maximum sampling period, etc.), the proposed monitoring method is able to detect this condition and raise an alarm. The extreme event test also illustrates that the proposed method is able to adjust to changes in its operational goals at runtime.

The other key aspect of the proposed monitoring method is its relevance-aware behavior, which is responsible for efficiently allocating energy in order to increase the monitoring accuracy without impacting the energy-aware behavior. The tests show that enabling the relevance-aware behavior resulted in a reduction between 1.1 and 19.5% of the MSE and between 0.6 and 20.7% of the MAE; and an improvement between 2.5 and 41.3% in the MSE and between 1.7 and 28.9% in the MAE with respect to the systematic scheme.

This paper proposed an improved monitoring method that builds upon a previously presented approach by introducing cooperative management of two energy management schemes, namely an adaptive sampling rate and a DPS. By adding an additional energy management method, our newly proposed method enhanced the monitoring accuracy with a reduction between 2.1 and 22.7% in the MSE and a reduction between 1.1 and 17.3% of the MAE compared with our previously proposed method. Furthermore, The experimental results reveal that this joint control approach enhances the monitoring accuracy, with up to 4.7% reduction of the MSE and up to 7.8% of the MAE compared to the method that adjusts only the sampling period. Similarly, compared to the method that adjusts only the transmission rate, the proposed approach obtained up to 16.2% reduction of the MSE and 20.3% reduction of the MAE. During the tests, the disabled control variable of the partial methods was set to the average value for such variable obtained in the simulation of the complete energy and relevance-aware monitoring method. Hence, the improvement in accuracy is solely attributed to the coordinated management of both control variables. Moreover, the coordination of both control parameters allows compensating for the effect of one of them reaching its control limit by adjusting the over parameter. This effectively extends the adaptive range of the sensor node. The extended adaptive range was evident in the extreme event test case, as the monitoring method with only the adaptive sampling period enabled, and our previously proposed method were unable to adjust to the new operating conditions.

6. Conclusions

In this paper, we propose a lightweight energy and relevance-aware monitoring method, which leverages the principles of self-awareness so the sensor node is able to comply with a battery lifetime goal, while it efficiently allocates its resources based on the operating conditions to improve the monitoring accuracy. We demonstrate the ability of our proposed monitoring method to meet the battery life target, even when the operating conditions are changed at runtime. Moreover, our monitoring method achieved a reduction of up to 19.5% in the MSE of the collected signal, compared to the proposed method with the relevance-aware behavior disabled. Thus, showing the ability to exploit changes in the temporal correlation of the collected data to improve its energy allocation. The monitoring method proposed in this paper Builds upon our existing method [28], which only considered one control parameter (i.e., the sampling period), by adding a new control parameter (i.e., a DPS) and the ability to jointly manage both schemes. Our newly proposed method was able to achieve a reduction in the MSE of the collected signal by up to 22.6% compared with our original method. We modified the DPS so that it maintains a specified transmission rate, making it possible to estimate its expected energy consumption and its integration with the proposed monitoring method. The join control of both operation parameters showed to improve not only the dynamic response of our proposed method allowing it to adjust a wider set of operating conditions, but it also resulted in a reduction of up to 16% of the MSE compared to the monitoring methods where one of the control parameters was set static at a predefined value.

One of the main limitations of the monitoring method presented in this paper is the need for a training period to estimate the error contribution of the adaptive sampling period and the DPS. Future versions of this method will create and maintain their own models at runtime, eliminating the need for a training period. Another future work is to evaluate this model using a non-linear battery model, where there may be some uncertainty in the amount of usable charge. Furthermore, having an accurate model of the battery will allow the node to optimize the discharge based on chemistry-specific properties [8], adding a supercapacitor in a hybrid power supply configuration [37], or even using an energy harvesting device so the energy budget changes over time [24].

Overall, the proposed method supposes a further step in the introduction of self-awareness at the wireless sensor node level, by enabling the monitoring method to decide between multiple adaptive actions. To do so, the monitoring method has to be able to predict the impact of each action in terms of energy consumption and monitoring accuracy that the new configuration has on the node, and select the best option. Currently, the proposed monitoring method prioritizes the battery lifetime and optimizes the monitoring accuracy on a best-effort basis. a higher level of awareness will be achieved if the node was able to manage multiple competing goals and use its predictive capabilities to prioritize between them. Future publications will focus on enhancing the level of awareness in wireless sensor nodes, guiding the transition toward self-awareness.

CRediT authorship contribution statement

David Arnaiz: Conceptualization, Methodology, Software, Writing – original draft. **Francesc Moll:** Conceptualization, Writing – review & editing. **Eduard Alarcón:** Conceptualization, Writing – review & editing. **Xavier Vilajosana:** Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data has been already made public and is linked on the manuscript.

Acknowledgments

This study was supported by the Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR 2019 DI 075 to David Arnaiz). The founder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

[1] K.K. Mohbey, S. Kumar, The impact of big data in predictive analytics towards technological development in cloud computing, Int. J. Eng. Syst. Model. Simul. 13 (1) (2022) 61–75.

[2] M. Compare, P. Baraldi, E. Zio, Challenges to IoT-enabled predictive maintenance for industry 4.0, IEEE Internet Things J. 7 (5) (2019) 4585–4597.

[3] M. Abdulkarem, K. Samsudin, F.Z. Rokhani, M.F. A Rasid, Wireless sensor network for structural health monitoring: a contemporary review of technologies, challenges, and future direction, Struct. Health Monit. 19 (3) (2020) 693–735.

[4] M.M. Saleh, WSNs and IoT their challenges and applications for healthcare and agriculture: A survey, Iraqi J. Electric. Electron. Eng. (2020).

[5] S.L. Ullo, G. Sinha, Advances in smart environment monitoring systems using IoT and sensors, Sensors 20 (11) (2020) 3113.

[6] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, M. Turon, Health monitoring of civil infrastructures using wireless sensor networks, in: Proceedings of the 6th International Conference on Information Processing in Sensor Networks, 2007, pp. 254–263.

[7] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, Energy conservation in wireless sensor networks: A survey, Ad Hoc Netw. 7 (3) (2009) 537–568.

[8] S. Chanagala, Z.J. Khan, A battery and environmental aware approach to maximize the lifetime of wireless sensor network, in: 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), IEEE, 2017, pp. 431–438.

[9] J.M. Parenreng, A. Kitagawa, Resource optimization techniques and security levels for wireless sensor networks based on the ARSy framework, Sensors 18 (5) (2018) 1594.

[10] J. Singh, R. Kaur, D. Singh, A survey and taxonomy on energy management schemes in wireless sensor networks, J. Syst. Archit. 111 (2020) 101782.

[11] C. Alippi, G. Anastasi, M. Di Francesco, M. Roveri, Energy management in wireless sensor networks with energy-hungry sensors, IEEE Instrum. Meas. Mag. 12 (2) (2009) 16–23.

[12] B. Martinez, X. Vilajosana, I. Vilajosana, M. Dohler, Lean sensing: Exploiting contextual information for most energy-efficient sensing, IEEE Trans. Ind. Inform. 11 (5) (2015) 1156–1165.

[13] S. Kounev, P. Lewis, K.L. Bellman, N. Bencomo, J. Camara, A. Diaconescu, L. Esterle, K. Geihs, H. Giese, S. Götz, et al., The notion of self-aware computing, in: Self-Aware Computing Systems, Springer, 2017, pp. 3–16.

[14] S. Kounev, X. Zhu, J.O. Kephart, M. Kwiatkowska, Model-driven Algorithms and Architectures for Self-Aware Computing Systems (Dagstuhl Seminar 15041), in: S. Kounev, X. Zhu, J.O. Kephart, M. Kwiatkowska (Eds.), Dagstuhl Rep. 5 (1) (2015) 164–196, http://dx.doi.org/10.4230/DagRep.5.1.164, URL http://drops.dagstuhl.de/opus/volltexte/2015/5038.

[15] K. Bellman, C. Landauer, N. Dutt, L. Esterle, A. Herkersdorf, A. Jantsch, N. Taherinejad, P.R. Lewis, M. Platzner, K. Tammemäe, Self-aware cyber-physical systems, ACM Trans. Cyber-Phys. Syst. 4 (4) (2020) 1–26.

[16] S.N. Das, S. Misra, B.E. Wolfinger, M.S. Obaidat, Temporal-correlation-aware dynamic self-management of wireless sensor networks, IEEE Trans. Ind. Inform. 12 (6) (2016) 2127–2138.

[17] J.M.C. Silva, K.A. Bispo, P. Carvalho, S.R. Lima, LiteSense: An adaptive sensing scheme for WSNs, in: 2017 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2017, pp. 1209–1212.

[18] G.B. Tayeh, A. Makhoul, D. Laiymani, J. Demerjian, A distributed real-time data prediction and adaptive sensing approach for wireless sensor networks, Pervasive Mob. Comput. 49 (2018) 62–75.

[19] P. Lou, L. Shi, X. Zhang, Z. Xiao, J. Yan, A data-driven adaptive sampling method based on edge computing, Sensors 20 (8) (2020) 2174.

[20] S. Santini, K. Romer, An adaptive strategy for quality-based data reduction in wireless sensor networks, in: Proceedings of the 3rd International Conference on Networked Sensing Systems (INSS 2006), TRF Chicago, IL, 2006, pp. 29–36.

[21] F.A. Aderohunmu, G. Paci, D. Brunelli, J.D. Deng, L. Benini, M. Purvis, An application-specific forecasting algorithm for extending wsn lifetime, in: 2013 IEEE International Conference on Distributed Computing in Sensor Systems, IEEE, 2013, pp. 374–381.

[22] T. Shu, J. Chen, V.K. Bhargava, C.W. de Silva, An energy-efficient dual prediction scheme using LMS filter and LSTM in wireless sensor networks for environment monitoring, IEEE Internet Things J. 6 (4) (2019) 6736–6747.

[23] K. Dai, C. Zhang, Q. Li, X. Huang, H. Zhang, An adaptive energy management strategy for simultaneous long life and high wake-up success rate of wireless sensor network nodes, Energy Technol. 9 (10) (2021) 2100522.

[24] A. Kansal, D. Potter, M.B. Srivastava, Performance aware tasking for environmentally powered sensor networks, in: Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems, 2004, pp. 223–234.

[25] J.M.C. Silva, K.A. Bispo, P. Carvalho, S.R. Lima, Flexible WSN data gathering through energy-aware adaptive sensing, in: 2018 International Conference on Smart Communications in Network Technologies (SaCoNeT), IEEE, 2018, pp. 317–322.

[26] J.M. Silva, P. Carvalho, K.A. Bispo, S. Rito Lima, E-LiteSense: Self-adaptive energy-aware data sensing in WSN environments, Int. J. Commun. Syst. 33 (10) (2020) e4153.

[27] N. Dang, E. Bozorgzadeh, N. Venkatasubramanian, Quares: Quality-aware data collection in energy harvesting sensor networks, in: 2011 International Green Computing Conference and Workshops, IEEE, 2011, pp. 1–9.

[28] D. Arnaiz, F. Moll, E. Alarcón, X. Vilajosana, Data relevance-aware dynamic sensing technique with battery lifetime guarantee for wireless sensor nodes, in: 2021 XXXVI Conference on Design of Circuits and Integrated Systems (DCIS), IEEE, 2021, pp. 1–6.

[29] M. Götzinger, A. Anzanpour, I. Azimi, N. Taherinejad, A.M. Rahmani, Enhancing the self-aware early warning score system through fuzzified data reliability assessment, in: International Conference on Wireless Mobile Communication and Healthcare, Springer, 2017, pp. 3–11.

[30] M. Götzinger, D. Juhász, N. Taherinejad, E. Willegger, B. Tutzer, P. Liljeberg, A. Jantsch, A.M. Rahmani, Rosa: A framework for modeling self-awareness in cyber-physical systems, IEEE Access 8 (2020) 141373–141394.

[31] G.M. Dias, B. Bellalta, S. Oechsner, A survey about prediction-based data reduction in wireless sensor networks, ACM Comput. Surv. 49 (3) (2016) 1–35.

[32] T. Bouguera, J.-F. Diouris, J.-J. Chaillout, R. Jaouadi, G. Andrieux, Energy consumption model for sensor nodes based on LoRa and LoRaWAN, Sensors 18 (7) (2018) 2104.

[33] D. Arnaiz, Raw atmospheric pressure, relative humidity and light intensity data timeseries, 2022, http://dx.doi.org/10.17632/ghjkcjn4dj.1, Mendeley Data, v1.

[34] D. Jung, T. Teixeira, A. Barton-Sweeney, A. Savvides, Model-based design exploration of wireless sensor node lifetimes, in: European Conference on Wireless Sensor Networks, Springer, 2007, pp. 277–292.

[35] A.H. Pereira, C.B. Margi, Energy management for wireless sensor networks, in: Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, 2012, pp. 329–330.

[36] S. Cheng, B. Li, Z. Yuan, F. Zhang, J. Liu, Development of a lifetime prediction model for lithium thionyl chloride batteries based on an accelerated degradation test, Microelectron. Reliab. 65 (2016) 274–279.

[37] A. Mirhoseini, F. Koushanfar, HypoEnergy. hybrid supercapacitor-battery power-supply optimization for energy efficiency, in: 2011 Design, Automation & Test in Europe, IEEE, 2011, pp. 1–4.