Optimization of Sensor Configurations for Fault Identification in Smart Buildings

Naveed Ahmad, Malcolm Egan, Jean-Marie Gorce, Jilles Dibangoye, Frédéric Le Mouël

OPTIMIZATION OF SENSOR CONFIGURATIONS FOR FAULT IDENTIFICATION IN SMART BUILDINGS

Naveed Ahmad[⋆†], *Malcolm Egan*[⋆], *Jean-Marie Gorce*[⋆], *Jilles S. Dibangoye*[⋆], *and Frédéric Le Mouël*[⋆]

[⋆] Univ Lyon, INSA Lyon, Inria, CITI, France
[†]Univ of Engineering and Technology, Peshawar, Pakistan

ABSTRACT

In predictive maintenance an important problem is to optimize the quantity of information to be transmitted at the control center to guarantee reliable fault detection while limiting sensor power consumption. This problem relies directly on the sensor configurations (e.g., sampling rate, coding, quantization) and the fault detection algorithm. To address this question, we introduce a codesign framework and an algorithm for joint optimization of the sensor configurations and the accuracy of the fault detection classifier. In a use case based on a dataset consisting of multiple sensor measurements and heating power levels known as the Twin House Experiment, we show that our algorithm can find efficient tradeoffs between sensor power consumption and classifier accuracy.

***Index Terms*—** predictive maintenance, fault identification, sensor optimization, GEM classifier

1. INTRODUCTION

As increasing numbers of sensors are utilized to monitor buildings and homes, predictive maintenance is a key strategy to identify degradation or faults in critical systems [1]; e.g., heating, ventilation and air conditioning. In principle, a large number of sensors can yield data capable of supporting effective predictive maintenance. However, the utility of sensors is constrained by their lifetimes and it is often desirable for battery operated sensors to operate on the order of 10 years.

For battery powered sensors to operate for long periods of time, it is necessary to ensure that sensor functions (e.g., data collection, processing and communication) are limited to those strictly necessary for effective predictive maintenance. In general, not all sensors collect and communicate data in the same fashion to a data analytics system for fault detection. As a consequence, the data analytics system must adapt, selecting fault classifiers tailored to the sensor data available at a given time. The predictive maintenance system therefore requires *codesign* of sensor configurations and the classifier utilized by the data analytics system.

In existing sensor and data analytics systems for predictive maintenance, either data analytics *or* sensor configuration are typically optimized, but not both. One exception is the work in [2], which considered tradeoffs between energy consumption and the accuracy of the classifier in a generic manufacturing context, however, it does not offer a systematic way of reducing the communication load. On the other hand, in the context of smart buildings, the focus has been on optimization of the classifier [3, 4, 5, 6] without considering how sensor configurations should be optimized to guarantee sufficient lifetimes.

In this paper, we propose a strategy to codesign communication and data analytics for predictive maintenance of smart buildings. The main contributions are:

- We formalize communication and data analytics codesign in smart buildings for a network of low cost sensors that wirelessly communicate with the cloud via a single access point. The resulting codesign problem is a bi-level optimization problem yielding a tradeoff between classifier accuracy and energy consumption, where both sensor configurations (which unlike [2] include sensor sampling rates, data compression, and sensor activity) and the classifier are optimized.

- The algorithm is evaluated on a use case with experimentally obtained temperature measurements from the Twin House Experiment [7], which may correspond to "normal" operation, where the heating is functioning, and "abnormal" operation, where the heating power source is turned off. Numerical results show rapid convergence to sensor configurations that yield a high accuracy and low battery power consumption and the impact of varying classifier training data.

2. PROBLEM FORMULATION

Consider a building operated by a data service provider and equipped with N low-cost and battery-operated sensors that transmit data to an access point via wireless links. The purpose of the sensor network is to identify degradation of the building. In general, the sensors may be utilized to detect degradation in a range of systems; e.g., HVAC, lighting or water. In Sec. 5, we will focus on a heating system use case.

2.1. Sensor Configurations

Each sensor j is capable of collecting and communicating data every $T_c^j \in \mathcal{T}_c$ seconds, where the data is assumed to be a scalar quantity and $|\mathcal{T}_c| < \infty$. That is, every T_c^j seconds, each sensor can collect a single sample of a physical observable (e.g., temperature, humidity, light intensity) and send it to the access point. Sensor j is also capable of adapting the duration of each data transmission, with transmissions consisting of $n^j \in \mathcal{N}$ bits with $|\mathcal{N}| < \infty$.

Each bit has a constant and known energy cost of E_b Joules. The E_b Joules includes energy consumed in every step such as sensing, processing and transmission. As such, the average power consumption of sensor j is given by

$$P_j(n^j, T_c^j) = E_b \cdot n^j \cdot T_c^j \text{ J/s}. \tag{1}$$

We note that a more sophisticated model of power consumption can be utilized with the codesign framework in Sec. 3.

The duration of a transmission, n^j, is constrained by the level of data compression at sensor j. In particular, if the data packet consists of n information bits, then at most 2^n quantization levels are available. When the data samples lie in an interval $\mathcal{I} = [d_{\min}, d_{\max}]$ of length $L = d_{\max} - d_{\min}$ and 2^n quantization levels are available, the quantization levels are separated by a distance $\Delta_j = \frac{L}{2^n - 1}$.

Given a data sample $d_{t,j} \in \mathcal{I}$ at time t, sensor j transmits the compressed data $\hat{d}_{t,j} = k^* \Delta_j + d_{\min}$, where $k^* = \arg\min_{k=0,1,\ldots,2^n-1} |d_{t,j} - d_{\min} + k\Delta_j|$.

If all sensors are utilized, then the total transmit power is given by $\sum_{j=1}^{N} P_j(n^j, T_c^j)$. However, over a given time period, not all the sensors may provide useful information suggesting that it may be desirable to switch off some of the sensors for a period of time. To this end, each sensor j is assigned an activity variable $x_j \in \{0, 1\}$, which indicates the sensor is off when $x_j = 0$. The total power is then defined as

$$P_{\text{tot}}(\mathbf{x}, \mathbf{n}, \mathbf{T}_c) = \sum_{j=1}^{N} x_j P_j(n^j, T_c^j), \tag{2}$$

where $\mathbf{x} = (x_1, \ldots, x_N)$, $\mathbf{n} = (n^1, \ldots, n^N)$, and $\mathbf{T}_c = (T_c^1, \ldots, T_c^N)$.

2.2. Fault Identification Problem

Given data available at the access point, the key problem is to identify whether or not it is anomalous. In particular, suppose at time t that sensor j transmits compressed data $\hat{d}_{t,j}$. The access point is then assumed to have an error-free observation of $\hat{\mathbf{d}}_t = (\hat{d}_{t,1}, \ldots, \hat{d}_{t,N})$, where $\hat{d}_{t,j} = 0$ if sensor j is not active (i.e., $x_j = 0$).

Based on $\hat{\mathbf{d}}_t$, the access point makes a decision as to whether or not an anomaly is present. This is achieved via a classifier $\Phi : \mathbb{R}^N \to \{1, \ldots, M\}$, $\hat{\mathbf{d}}_t \mapsto \hat{\ell}_t$. In particular, if $\hat{\ell}_t = \Phi(\hat{\mathbf{d}}_t) \neq 1$, then an anomaly is detected. Often, labeled data $\{(\mathbf{d}_t, \ell_t)\}$, where ℓ_t is the true label for sample t, is only available for "normal operation". Additional unlabeled data, potentially containing anomalous events, may also be available. As a consequence, semi-supervised classification [8] methods are required. The choice of the classifier depends on the sensor configurations. For example, if the quantity of sensors used to collect temperature measurements is reduced, the corresponding classifier in general differs from the case when all sensors are utilized.

3. CODESIGN BI-LEVEL OPTIMIZATION PROBLEM

In this section, we introduce a framework for optimizing the configuration of the sensors and the classifier in order to provide an efficient tradeoff between performance of the classifier and the power consumption of the sensor configuration.

3.1. Master Classifier

Initially available to the access point is a set of training data $\{\mathbf{d}_t\}_{t \in \mathcal{D}_{\text{train}}}$ obtained from the full set of sensors at the lowest compression rate. This data is assumed to be obtained from "normal" operation. In the context of fault identification, this is a reasonable assumption as the majority of the time, the system is functional.

Classification based on the data $\{\mathbf{d}_t\}_{t \in \mathcal{D}_{\text{train}}}$ can be performed via geometric entropy minimization techniques, such as K-kNN graphs [9], for a fixed false alarm rate $\alpha \in (0, 1)$. To label additional test data $\mathcal{D}_{\text{test}}$ that may contain "abnormal" operation events, we introduce a master classifier based on the bipartite K-kNNG method in [9]. We denote the labeled test data by $\{\hat{\mathbf{d}}_t, \hat{\ell}_t\}_{t \in \mathcal{D}_{\text{test}}}$, where $\hat{\ell}_t \in \{0, 1\}$ with $\hat{\ell}_t = 1$ denoting an abnormal operation event. The data labeled via the master classifier can then be used to evaluate the performance of classifiers based on data from a subset of the sensors, which are called induced classifiers.

3.2. Induced Classifiers

With the initial data $\{\mathbf{d}_t\}_{t \in \mathcal{D}_{\text{train}}}$, the classifier can also tune a set of *induced* classifiers $\{\Phi_z\}$ corresponding to different configurations $z = (\mathbf{x}, \mathbf{n}, \mathbf{T}_c)$ of the sensors. As for the master classifier, the induced classifiers are obtained via the K-kNNG classifier in [9].

The performance of the induced classifiers is computed using the test data labeled by the master classifier; namely, $\{\hat{\mathbf{d}}_t, \hat{\ell}_t\}_{t \in \mathcal{D}_{\text{test}}}$. The accuracy of the induced classifer Φ_z is given by

$$P_{\text{acc}}(\mathbf{x}, \mathbf{n}, \mathbf{T}_c) = \beta(\mathbf{x}, \mathbf{n}, \mathbf{T}_c), \tag{3}$$

where $\beta(\mathbf{x}, \mathbf{n}, \mathbf{T}_c)$ is the proportion of false negatives for the configuration $(\mathbf{x}, \mathbf{n}, \mathbf{T}_c)$ obtained from the test dataset labeled by the master classifier.

3.3. Codesign Optimization Problem

We formalize the codesign problem for optimization of the sensor configurations as

$$(\mathbf{x}^*, \mathbf{T}_c^*, \mathbf{n}^*) = \arg \max_{(\mathbf{x}, \mathbf{T}_c, \mathbf{n}) \in \{0,1\}^N \times \mathcal{T}_c \times \mathcal{N}} -\lambda_1 P_{\mathrm{tot}}(\mathbf{x}, \mathbf{n}, \mathbf{T}_c) + \lambda_2 P_{\mathrm{acc}}(\mathbf{x}, \mathbf{T}_c, \mathbf{n}), \tag{4}$$

where $\lambda_1, \lambda_2 > 0$ are penalty parameters. Due to the need to compute an induced classifier for each configuration of the sensors, the optimization problem in (4) is a *bi-level optimization problem*. Also note that $\mathbf{x} \in \{0,1\}^N$ and $\mathbf{T}_c, \mathbf{n}$ lie in finite sets leading to a discrete optimization problem. We introduce an efficient heuristic method to optimize the sensor configuration in the following section.

4. SENSOR CONFIGURATION OPTIMIZATION ALGORITHM

Suppose that the set of feasible configurations $\mathcal{C} = \{(\mathbf{x}, \mathbf{n}, \mathbf{T}_c)\} = \{0,1\}^N \times \mathcal{T}_c^N \times \mathcal{N}^N$ has a cardinality of S. We denote the i-th element of the set of feasible configurations by $\mathbf{c}(i)$.

Although the optimization problem in (4) is discrete, it can be approximated by a smooth problem as follows. Let $\boldsymbol{\alpha} \in \mathbb{R}_+^S$ and consider the optimization problem

$$\boldsymbol{\alpha}^* = \arg \max_{\boldsymbol{\alpha} \in \mathbb{R}_+^S} \sum_{i=1}^{S} f(i) \frac{\alpha_i}{\sum_{j=1}^{S} \alpha_j}, \tag{5}$$

where

$$f(i) = -\lambda_1 P_{\mathrm{tot}}(\mathbf{c}(i)) + \lambda_2 P_{\mathrm{acc}}(\mathbf{c}(i)). \tag{6}$$

Note that in order to compute $P_{\mathrm{acc}}(\mathbf{c})$ it is necessary to train the corresponding induced classifier described in Sec. 3.2. An approximate solution to (4) can then be obtained via $i^* = \arg\max_{i=1,\dots,S} \alpha_i$.

While the problem in (5) can in principle be solved by gradient descent, a drawback is that when S is large there are many terms in the sum. As such, computing each gradient is computationally expensive. Moreover, the computation of $f(i)$ requires optimization of the classifier to account for the data available to the access point under configuration $\mathbf{c}(i)$.

This issue can be resolved by noting that

$$\boldsymbol{\alpha}^* = \arg \max_{\boldsymbol{\alpha} \in \mathbb{R}_+^S} \frac{1}{S} \sum_{i=1}^{S} f(i) \frac{\alpha_i}{\sum_{j=1}^{S} \alpha_j}, \tag{7}$$

which can be viewed as a stochastic optimization problem. A standard method to solve the problem in (7) is projected stochastic gradient descent. Let $\mathcal{H} = \{\boldsymbol{\alpha} \in \mathbb{R}^N : \alpha_i \geq 0,\ i = 1, 2, \dots, S\}$. The parameter $\boldsymbol{\alpha}$ is then updated recursively in Line 3 of Algorithm 1, where $\Pi_{\mathcal{H}}\{\cdot\}$ denotes the Euclidean projection onto $\mathcal{H}$.

Algorithm 1 Algorithm for Optimization of Sensor Configurations

Input: $\boldsymbol{\alpha}_0 \in \mathbb{R}_+^S$, $t = 0$, step-size sequence $\{\epsilon_t\}$.
Output: Configuration i^*.

1: **while** Not Stopped **do**
2: Sample i_{t+1} uniformly from $\{1, \dots, S\}$.
3: Compute $\boldsymbol{\alpha}_{t+1} = \Pi_{\mathcal{H}} \left\{ \boldsymbol{\alpha}_t + \epsilon_{t+1} f(i_{t+1}) \nabla_{\boldsymbol{\alpha}} \frac{\alpha_{i_{t+1}}}{\sum_{j=1}^{S} \alpha_j} \Big|_{\boldsymbol{\alpha} = \boldsymbol{\alpha}_t} \right\}$, where $f(i)$ is defined in (6), computed using the samples in $\mathcal{D}_{\mathrm{test}}$ based on the induced classifier obtained from the samples in $\mathcal{D}_{\mathrm{train}}$.
4: $t \to t + 1$.
5: **end while**
6: **return** $i^* = \arg\max_{i=1,\dots,S} \alpha_{t,i}$.

5. USE CASE: POWER OUTAGE DETECTION IN THE TWIN HOUSE EXPERIMENT

5.1. Use Case Description

The Twin House Experiment [7] was performed on two stand alone houses installed with a number of sensors in each room. Each house consisted of a living room, kitchen, childrens' room, bedroom, doorway, corridor, wan attic, and basement. The experiments involved heating the houses at different levels and measuring variables such as the indoor temperature evolution, the power consumption, humidity, and heat losses from the walls.

The experiment was conducted for a period of 41 days, with temperature data recorded every 10 minutes. The corresponding data set consists of 5609 samples obtained from 8 sensors. During part of this time period, the heating system was switched off. These measurements (from the indoor sensors) along with the description of the houses are provided in [10]. The temperature measurements and corresponding power levels for heating for several of the rooms are illustrated in Fig. 1.

5.2. Numerical Results

We utilized Alg. 1 to optimize the sensor configurations in order to solve (4). The sensor configuration space is determined by which of the 8 sensors are active, the set of compression levels $n = \{4, 3, 2, 1\}$, where the numbers show the number of digits after the decimal, and the set of sampling periods $T_c = \{2947, 2210, 1473, 736\}$. The master and the induced classifiers detailed in Sec. 3 utilized the bipartite K-kNNG method in [9] using $k = 6$ neighbours and a power weighting $\gamma = 0.9$ (for details of these parameters see [9]). The temperature sensors data with power ON is considered as normal (training data set) whereas sensors data with power OFF is considered an anomalous (anomalous) data. The maximum

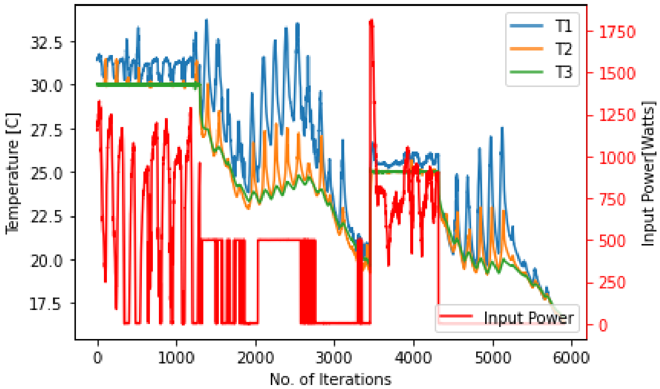

Fig. 1. The temperature evolution in different rooms of the Twin House. The data corresponding to T1, T2, T3 are the temperature data from the living room, kitchen, doorway respectively. The input power is the heating power in the living room, the sequence of power in other rooms is similar.

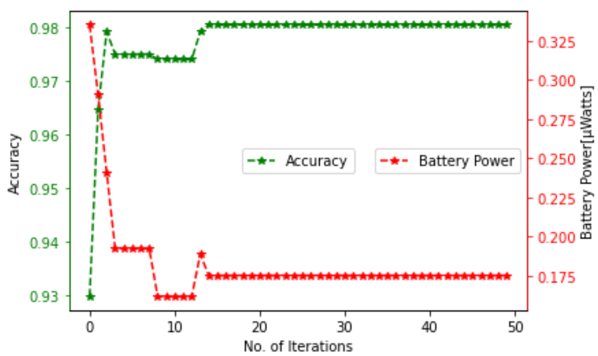

Fig. 2. Improvement in accuracy and battery power with the number of iterations of Alg. 1. The accuracy and battery curves are obtained by averaging 30 runs of Alg. 1.

battery power consumed with all the set of sensors communicating at full sampling rate and zero compression is approximately $0.7\ \mu W$.

Fig. 2 plots the average accuracy and battery power of the sensor configurations obtained from 10 runs of Alg. 1. The algorithm was initialized with $\alpha_{0,j} = 0.5,\ j = 1, \dots, S$ and utilized a constant step-size $\epsilon_t = 10,\ t > 0$. The accuracy is determined in terms of the ratio of the number of anomalies detected by the algorithm to the total number of anomalies present. Observe that the algorithm rapidly improves the average accuracy and battery power, and refines the sensor configurations as the number of iterations increases.

In case of residential buildings/homes the comfortable range of temperature changes with the time of day. To study the impact of quantity of time period from which the "normal" training data was drawn, we divided the data into three zones D_1, D_2 and D_3. In particular, D_1 corresponds to samples $\{0, \dots, 1200\}$, $D_2 = \{1001, \dots, 2500\}$ and $D_3 = \{1600, \dots, 2900\}$. Fig. 3 and Fig. 4 plot the average accuracy and battery power, respectively, of the sensor configurations obtained from Alg. 1 over 10 runs. Observe that the accuracy for all data zones is similar. However, there are significant differences in the battery power levels utilized with different training data. This is due to a larger number of active sensors with training data D_2.

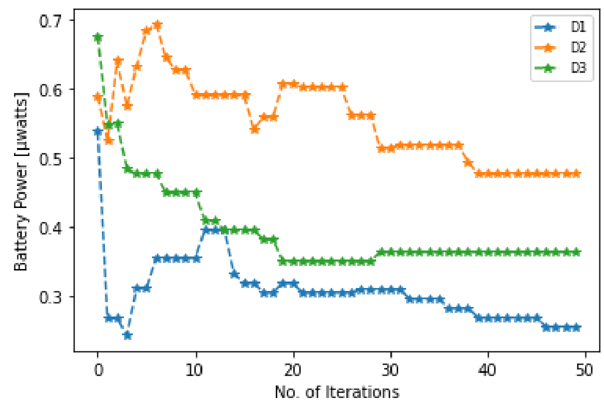

Fig. 3. Battery Power selected by Alg. 1 averaged over 10 runs. D_1, D_2 and D_3 correspond to different training datasets.

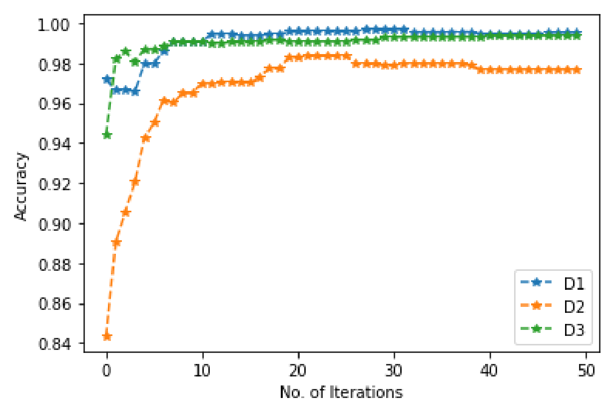

Fig. 4. Accuracy obtained by running Alg. 1, the accuracy score averaged over 10 runs. D_1, D_2 and D_3 correspond to different training datasets.

6. CONCLUSION

We introduced a codesign framework which provides a means of jointly optimizing the quality of the classifier and sensor configurations. A numerical study for the Twin House Experiment use case suggests that our method can find efficient tradeoffs between power consumption of the sensors and the accuracy of the classifier.

7. REFERENCES

[1] R.K. Mobley, *An Introduction to Predictive Maintenance*, Elsevier Science, Woburn, MA, 2002.

[2] Y. Teoh, S. Gill, and A. Parlikad, "IoT and fog computing based predictive maintenance model for effective asset management in industry 4.0 using machine learning," *IEEE Internet of Things Journal*, 2021.

[3] M. Razali et al., "Big data analytics for predictive maintenance in maintenance management," *Property Management*, 2020.

[4] S. Malik et al., "A feature selection-based predictive-learning framework for optimal actuator control in smart homes," *Actuators*, vol. 10, no. 4, 2021.

[5] Y. Bouabdallaoui et al., "Predictive maintenance in building facilities: a machine learning-based appraoch," *Sensors*, vol. 21, no. 4, 2021.

[6] M. Castangia et al., "Anomaly detection on househould appliances based on variational autoencoders," *Sustainable Energy, Grids and Networks*, vol. 32, 2022.

[7] Matthias Kersken and Paul Strachan, "Twin house experiment IEA EBC annex 71 validation of building energy simulation tools-specifications and dataset," 2020.

[8] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," in *Proc. International Workshop on Artificial Intelligence and Statistics*, 2005.

[9] K. Sricharan and A. Hero, "Efficient anomaly detection using bipartite k-nn graphs," in *Proc. Advances in Neural Information Processing Systems*, 2011.

[10] Strachan, "Iea annex 58: Full-scale empirical validation of detailed thermal simulation programs," *Energy Procedia*, vol. 78, pp. 3288–3293, 2015.