sensors

MDPI

Article

An Intelligent Failure Detection on a Wireless Sensor Network for Indoor Climate Conditions

Sebastián Gutiérrez [1] and Hiram Ponce [2,*]

[1] Facultad de Ingeniería, Universidad Panamericana, Josemaría Escrivá de Balaguer 101, Aguascalientes, Aguascalientes 20290, Mexico; jsgutierrez@up.edu.mx

[2] Facultad de Ingeniería, Universidad Panamericana, Augusto Rodin 498, Ciudad de México 03920, Mexico

* Correspondence: hponce@up.edu.mx; Tel.: +52-55-5482-1600

Received: 17 December 2018; Accepted: 5 February 2019; Published: 19 February 2019

check for updates

Abstract: Wireless sensor networks (WSN) involve large number of sensor nodes distributed at diverse locations. The collected data are prone to be inaccurate and faulty due to internal or external influences, such as, environmental interference or sensor aging. Intelligent failure detection is necessary for the effective functioning of the sensor network. In this paper, we propose a supervised learning method that is named artificial hydrocarbon networks (AHN), to predict temperature in a remote location and detect failures in sensors. It allows predicting the temperature and detecting failure in sensor node of remote locations using information from a web service comparing it with field temperature sensors. For experimentation, we implemented a small WSN to test our sensor in order to measure failure detection, identification and accommodation proposal. In our experiments, 94.18% of the testing data were recovered and accommodated allowing of validation our proposed approach that is based on AHN, which detects, identify and accommodate sensor failures accurately.

Keywords: artificial organic networks; artificial hydrocarbon networks; distributed services architecture; failure detection; internet-of-things; machine learning; weather web services; sensor networks

1. Introduction

Nowadays, weather conditions affect the daily life of many people in the world [1]. The need of environmental information for the development of activities around the cities, farms and communities is increasing as technology of meteorological predictions becomes available. Likewise, the grown of development of meteorological stations for reports is increasing [2–7]. The weather monitoring stations help to predict and understand the weather conditions in order to monitor and track weather changes. These stations uses multiple meteorological sensors to monitoring weather conditions by sensing weather variables such as temperature, humidity, atmospheric pressure, wind speed, wind direction and rainfall, with the objective to give accurate information to users.

In fact, some institutions and companies, and in some cases, buildings and homes, have their own weather stations to collect weather data. With a right distribution of meteorological sensors, a prediction based on acquired data with information obtained in real time can be created. This information is used to their own ends, and in some cases, shared between third parties through a new rising technology like the Internet of Things (IoT) to keep in hand the needed information to anticipate and be alert to the weather and climate conditions.

In this context, IoT is a new part of a novel generation of information technology that allows the connection between devices to share multiple data to control, automate and centralize systems of a variety of application domains. This technology has several application domains such as environmental monitoring, smart cities, smart business and product management, smart homes, smart building management, healthcare and security and surveillance. To achieve this, many protocols for communication and management must allow the connection with sensors, cloud-based systems, control devices and monitoring [8–10]. Moreover, it is important to mention that the failures from sensor nodes to receive messages must be taken into account, i.e., sensors can have a poor connection to the base station, meaning that the data transmission from that sensor has a high risk to fail. In addition, due to the limited energy capacity, sensor nodes with run out battery may cause disconnected links which will result in failed data transmission in IoT [11–13].

The implementation of weather monitoring station based on IoT system allows to obtain information from weather web services and physical multi-sensors connected into IoT network, attaining better accuracy of data used to predict and show weather statistics [14–17]. Moreover, the problem of failure detection in sensor nodes is essential since this issue can destabilize the normal operation of measurement. In addition, artificial intelligence (AI), i.e., machine learning and data mining, enhance predictive climate conditions actions over IoT systems [18]. And, some approaches have been developed on sensor failure using AI.

In particular, sensor failure detection, identification and accommodation (SFDIA) approaches are those schemes used to preserve robustness and high reliability of sensor-based systems [19]. For instance, sensor accommodation (SA) refers to the ability of the system to replace the malfunctioning sensor with another sensor or estimator. A typical scheme for SA is redundancy in hardware using multiple sensors for one measurement, in which, if one of them has different behavior than the others, one of the other sensors replaces it [19]. Sensor failure detection and identification (SFDI) refers to determine an anomaly in the system and to identify where it takes place. Literature reports SFDI schemes is mainly applied with the comparison between the real sensor value and an estimation of it [19,20]. For estimation, linear and nonlinear models have been applied [19]. Recently, AI has been applied to solve SFDIA systems. Another approach, mainly for wireless sensor networks, is the check-point recover algorithm which measures the energy level of a sensor and sends that information through all the network. If one sensor is not able to send this energy level, then it is assumed as a fault [21]. In literature [19–22], some proposal for failure detection in sensors through neural networks learning algorithms and support vector machines have been reported.

In this context, this paper proposes to apply artificial intelligence by means of a supervised learning method namely artificial hydrocarbon networks (AHN), to predict temperature in a remote location and detect failures in sensors. For this work, a small wireless sensor network over IoT protocols is developed. This network consists on three temperature sensors, remotely located indoors, and a temperature web service. At each node of the network, an AHN-model for predicting temperature is trained, but also the architecture of the AHN considers a failure detection method. In this sense the system is able to identify a failure in sensors over the wireless network and to promote SFDIA. For validation purposes, a simulated and a real data sets of large number of sensor nodes were previously tested.

The contribution of this work considers: (i) the development of a wireless sensor network for indoor temperature estimation using AHN, and (ii) the proposed sensor failure detection module using AHN. To the best of our knowledge, this is the first time AHN is implemented for failure detection systems.

The rest of the paper is organized as follows. Section 2 summarizes the prototype development of the IoT system. Section 3 describes the proposed intelligent sensor failure detection module using artificial hydrocarbon networks. Section 4 shows the applicability of the proposed strategy in large wireless sensor networks. Section 5 presents experimental results and discussion about the proposal, and Section 6 concludes this work.

2. Prototype Development

The prototype implementation of the proposed system uses a weather web service and physical temperature sensors under DSA (Distributed Services Architecture) open source platform for IoT [23]. The main goal of the DSA is to unify multiple devices, services and application in real time, structured and adaptable data model. To achieve this, we employed an embedded system Raspberry Pi 3 as a DSA server to control the system using a distributed server link (DSLink) to enable data exchange with other connected sensor nodes, as a weather web service in the cloud that uses Yahoo Weather API [24], and a communication protocol for receiving physical sensors data by means of EnOcean®protocol [25–27]. Figure 1 shows the IoT architecture of the prototype development.

In addition to the embedded system, the DGLux5®program, integrated in the DSA server, allows to work with real time data to ensure the design and deployment process can be shown satisfactory on the different temperatures from physical sensors and the weather web service. The visual programming language to designing the front-end was performed connecting blocks that represent different operations. In the dataflow, to display the various temperatures and save the values in a database, the blocks were dragged and placed to be connected.

As shown in Figure 1, the DSA obtains the temperatures from the weather web service connected to the cloud via internet and collects the information from the physical sensors using the EnOcean®protocol. The temperature data obtained are shown in a graphical interface of DGLux5®program, where each of the temperatures appears, and where it is possible to download a CSV file of each one of them, as shown in Figure 2.

For this prototype implementation, the temperature sensors were located physically in three different locations inside a building of the Faculty of Engineering in Universidad Panamericana (Aguascalientes, Mexico). Sensor 1 was located on an electronic laboratory; sensor 2 was placed in a manufacturing room; and finally, sensor 3 was positioned on an office. Figure 3 shows the locations of the three sensors inside the building, while Figure 4 shows the inside enclosures where the sensors were placed.

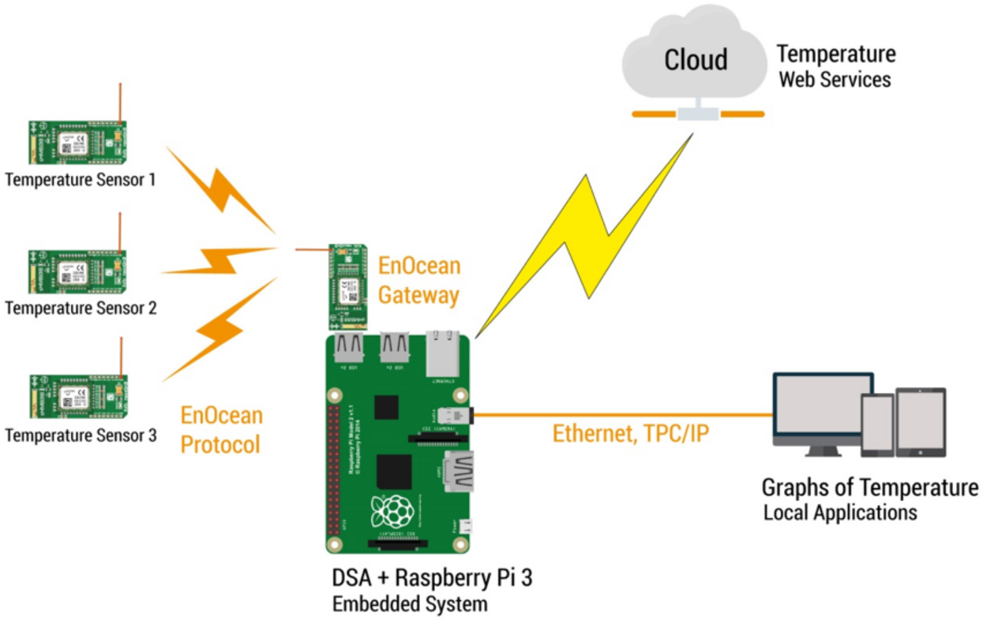
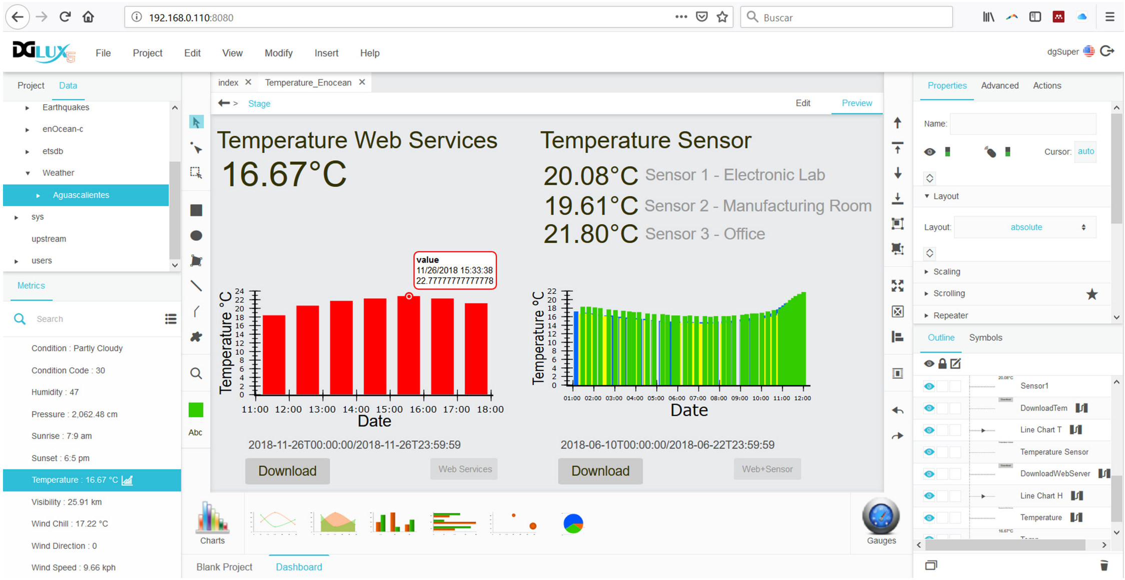


Figure 1. Architecture of the IoT system.



Condition : Partly Cloudy

Condition Code : 30

Humidity : 47

Pressure : 2,062.48 cm

Sunrise : 7:9 am

Sunset : 6:5 pm

Temperature : 16.67 °C

Visibility : 25.91 km

Wind Chill : 17.22 °C

Wind Direction : 0

Wind Speed : 9.66 kph

Temperature Web Services

16.67°C

value
11/26/2018 15:33:38
22.7777777777778

2018-11-26T00:00:00/2018-11-26T23:59:59

Temperature Sensor

20.08°C Sensor 1 - Electronic Lab

19.61°C Sensor 2 - Manufacturing Room

21.80°C Sensor 3 - Office

2018-06-10T00:00:00/2018-06-22T23:59:59

Figure 2. Snapshot of the front-end developer of DGLux5 ®program.

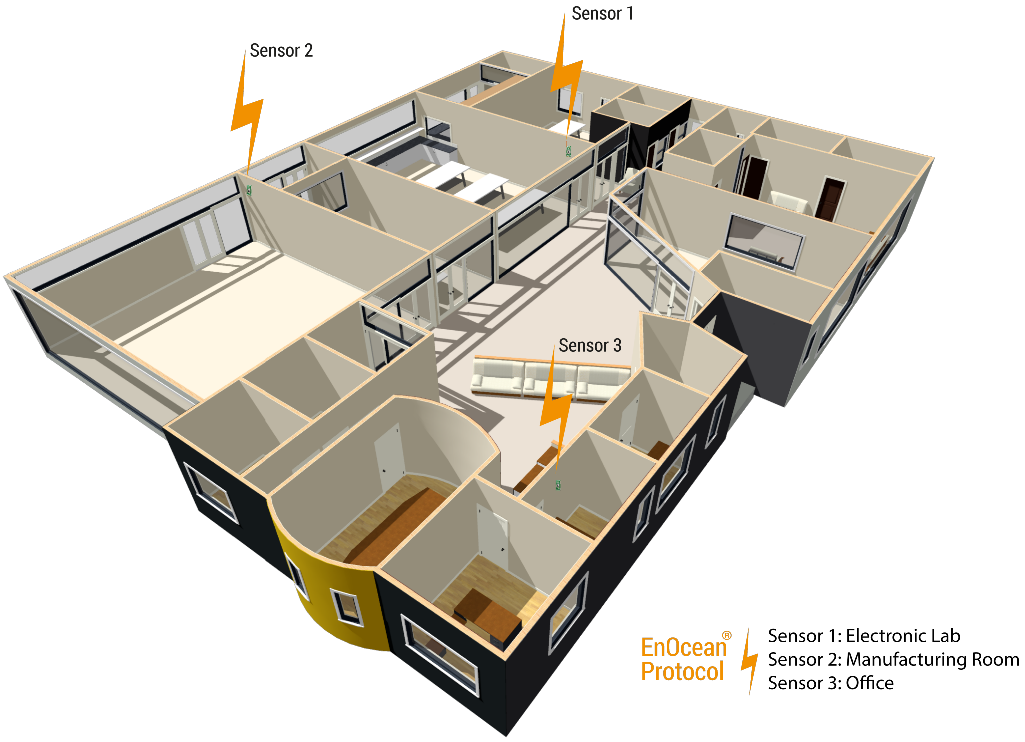


Figure 3. Layout of the building showing the location of the three temperature sensors.

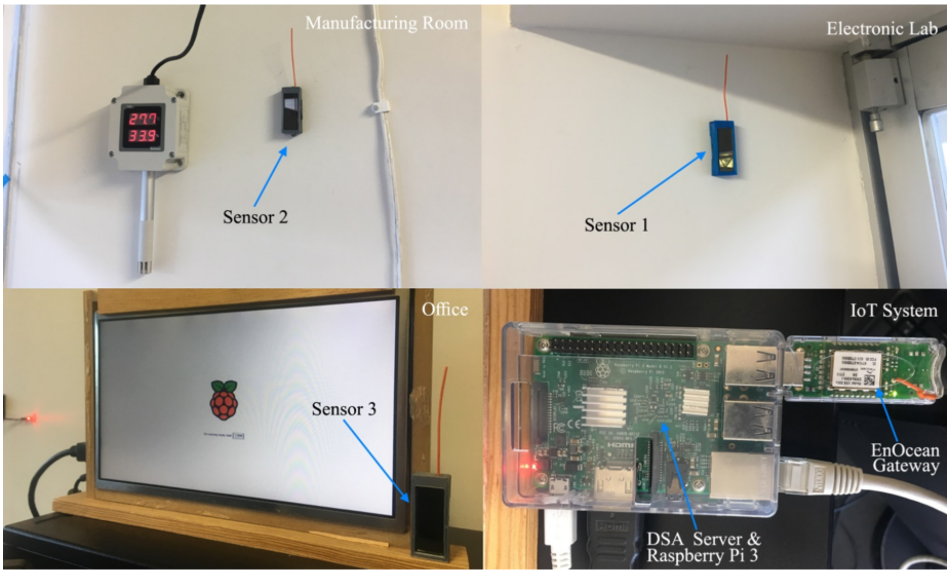


Figure 4. Location of the three temperature sensors inside enclosures.

3. Development of the Temperature Estimation and Sensor Failure Detection System

In order to perform an intelligent system that can predict temperature changes and failure detection in sensors through an IoT system, the AHN supervised learning method is proposed. In particular, this work is focused solely on the prediction of the temperature and failure detection sensors using the IoT system presented in Section 2. The proposed intelligent system is shown in Figure 5. The ensuing is a general description of wireless sensor network architecture, the method of artificial hydrocarbon networks and the description of the intelligent module.

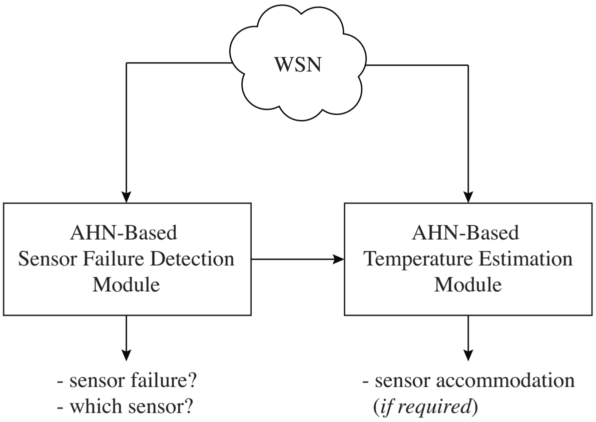

WSN
AHN-Based Sensor Failure Detection Module
AHN-Based Temperature Estimation Module
- sensor failure?
- which sensor?
- sensor accommodation
(if required)

Figure 5. Proposed intelligent system for temperature estimation and sensor failure detection.

3.1. Wireless Sensor Networks

A wireless sensor network (WSN) is composed of a large number of sensor nodes that cover an even larger set of application areas [28], such as, military and surveillance [29–31], environment [3,32,33], health [34–36], home and building monitoring [37–39]. On the network, sensor nodes send information at regular time intervals from inside enclosures to be monitored, and ensure that the nodes in the network operate faultlessly. Therefore, fault detection in sensors nodes play an important role in network monitoring since an error can have a negative effect on the system's efficiency and reliability. Figure 6 shows our WSN architecture that is applied in temperature monitoring which contains sensor nodes, gateway and user. Sensor nodes will communicate with each other and transmit the processed data to gateway over a wireless communication. The gateway collects data from all the sensor nodes, transmits the analyzed data to user via Internet and receives data from temperature web services.

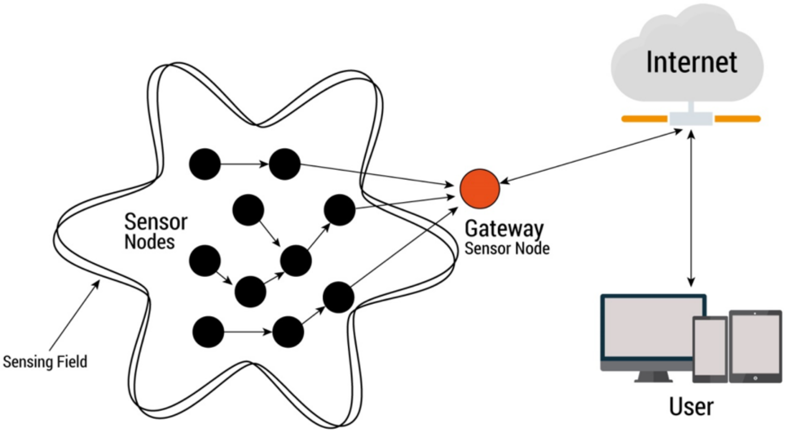


Figure 6. Wireless sensor network architecture of the IoT system.

3.2. Artificial Hydrocarbon Networks

In machine learning, AHN is a supervised learning method. It aims to model data using the inspiration of carbon networks, simulating the chemical rules involved within organic molecules to represent the structure and behavior of data [40,41]. Thus, the method loosely simulates the chemical interactions of hydrocarbon molecules.

The basic unit of information in AHN is namely CH-molecule or simply molecule. From inspiration on nature, a molecule is formed with one carbon atom that is linked to up to four hydrogen atoms. Generally speaking, one molecule models a chunk of data in its parameters (hydrogen and carbon atoms) and configuration. The molecule has a structural representation, a graphical model as shown in Figure 7, and a chemical behavior. Mathematically, the behavior φ of a molecule with k hydrogen atoms is expressed as in (1); where, $\sigma \in \mathbb{R}^n$ is called the carbon value, $H_i \in \mathbb{R}^n$ is the i-th hydrogen atom attached to the carbon atom, and $x = (x_1, \ldots, x_n)$ is the input vector with n features.

$$\varphi(x,k) = \sum_{r=1}^{n} \sigma_r \sum_{i=1}^{k \leq 4} H_{ir} x^k \tag{1}$$

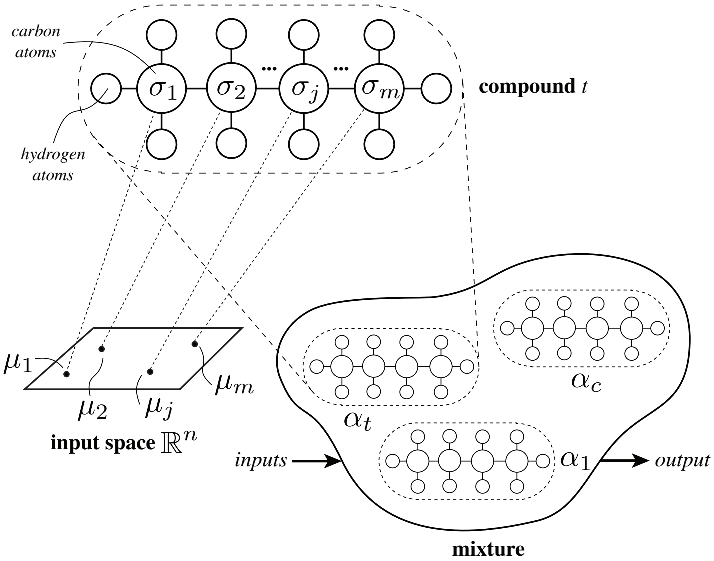


Figure 7. Schematic of an artificial hydrocarbon network.

Two or more unsaturated molecules, i.e., those with $k < 4$ hydrogen atoms, can be joined together. In AHN, this structure is called a compound. Different compounds have been defined in literature [40,42,43]. The simplest one is the saturated and linear chain of m molecules. It is composed structurally of two molecules with three hydrogen atoms and $(m-2)$ molecules with two hydrogen atoms. The behavior ψ of a saturated-and-linear compound is defined as (2); where, φ_j is the behavior of the jth associated molecule that represents a subset Σ_j of the input x such that $\Sigma_j = \{x | \arg\min_j (x - \mu_j) = j\}$, and $\mu_j \in \mathbb{R}^n$ is the center of the jth molecule [44,45]. In fact, $\Sigma_{j_1} \cap \Sigma_{j_2} = \emptyset$ if $j_1 \neq j_2$. In this regard, literature reports extensive usage of AHN with a single saturated-and-linear compound [40,41,43–46].

$$\psi(x) = \begin{cases} \varphi_1(x,3) & x \in \Sigma_1 \\ \varphi_2(x,2) & x \in \Sigma_2 \\ \cdots & \cdots \\ \varphi_{m-1}(x,2) & x \in \Sigma_{m-1} \\ \varphi_m(x,3) & x \in \Sigma_m \end{cases} \tag{2}$$

Compounds can interact among them in definite ratios α_t, so-called stoichiometric coefficients or weights, forming a mixture $S(x)$ as shown in (3); where, c represents the number of compounds in the mixture and α_t is the weighted factor of the t-th compound [40].

$$S(x) = \sum_{t=1}^{c} \alpha_t \psi_t(x) \tag{3}$$

Formally, an AHN is a mixture of compounds (see Figure 7) each one computed using a chemical-based heuristic rule, expressed in its own training algorithm, as depicted in Algorithm 1 [40,42,46]. At first, it initializes the structure of a saturated and linear compound with m molecules, and the centers of molecules $\{\mu_j\}$ are randomly set. While a stop criterion is not reached, the compound is computed and updated as follows. First, for each molecule, the training dataset Σ is partitioned in subsets Σ_j such that every input x is near to μ_j. Then, the hydrogen and carbon values of each molecule is computed independently using the least squares estimates (LSE) method, and the error E_j between the output response of the j-th molecule and the actual targets y of the j-th subset is calculated. If any of the subsets is empty, then these subsets require a new position, i.e., update their centers. This relocation is made by simply changing the center of the empty subset randomly close to one molecule with large error. Later on, the center of molecules are updated via a gradient descent approach with learning rate $0 < \eta < 1$, previously chosen, as shown in (4) with $E_0 = 0$. Lastly, the compound ψ is updated with the behaviors of molecules calculated so far.

$$\mu_j \leftarrow \mu_j - \eta(E_{j-1} - E_j) \tag{4}$$

A detailed description of the AHN-algorithm can be found in [40]. Information about properties and comparisons with other supervised learning methods can be seen in [40,44,45].

3.3. Temperature Estimation and Sensor Failure Detection Module

For our proposal, we model the dynamics of the temperature sensors in the WSN to provide an estimate using AHN. In addition, this estimation is then compared with the actual sensor values, and treated statistically for failure detection. In the following, the entire module is explained.

Algorithm 1 Simple AHN training algorithm

```
Data: the training data set $\Sigma = (x, y)$, the number of molecules in the compound $m \geq 2$ and the
      learning rate $0 < \eta < 1$.
Result: the trained compound $\psi$.
 1. Create the structure of a new compound with $m$ molecules.
 2. Randomly initialize a set of molecular centers, $\mu_j$.
 while not reach a stop criterion do
     for $j = 1$ to $m$ do
         3. $\Sigma_j \leftarrow$ do a subset of $\Sigma$ using $\mu_j$.
         4. $\{H_j, \sigma_j\} \leftarrow$ calculate molecular parameters of $\varphi_j$ using LSE method.
         5. $E_j \leftarrow$ compute the error in molecule.
     end
     for $j = 1$ to $m$ do
         if $\Sigma_j = \emptyset$ then
             6. $\mu_j \leftarrow$ relocate the center of molecule near to other $\mu_k$ such that $E_k$ is very large.
         end
     end
     7. Update all centers using $\mu_j = \mu_j - \eta(E_{j-1} - E_j)$.
 end
 8. Update the behavior of compound $\psi$ using all $\varphi_j$ already calculated.
 9. Return $\psi$.
```

3.3.1. Temperature Estimation Module

Consider a WSN with N sensors as nodes, s_i for all $i = 1, \ldots, N$. Also, let say that sensor s_i fails. Then, we propose a sensor accommodation using the estimations of an AHN-based model $\hat{fs}_{\omega}^{(i)}$ of sensor s_i, as expressed in (5); where, $\hat{s}_i(t) \in \mathbb{R}$ is the estimation of sensor s_i in time t and $s_{wsn}(t) \in \mathbb{R}^{(N-1)}$ is a vector of values of the $(N-1)$ sensors $s_{wsn}(t) = (s_1(t), \ldots, s_k(t), \ldots, s_{(N-1)}(t))$ and $i \neq k$ in time t.

$$\hat{s}_i(t) = \hat{fs}_{\omega}^{(i)}(s_{wsn}(t), s_{wsn}(t-1)) \tag{5}$$

This sensor model $\hat{fs}_{\omega}^{(i)}$ is proposed to be based on AHN with a vector of parameters ω that represents the hydrogen, carbon and molecular centers over the hydrocarbon compound, such that $\omega = \{H_{ir}, \sigma_r, \mu_j\}$. For training the AHN-model, we use tuples of the form $(s_{wsn}(t), s_{wsn}(t-1)) \in \mathbb{R}^{2(N-1)}$ as training inputs and $s_i(t) \in \mathbb{R}$ as training outputs, collected from the WSN for a given period of time. In this regard, the sensor model $\hat{fs}_{\omega}^{(i)}$ is trained by minimizing the error function described in (6) using the gradient descent method.

$$E(\omega) = \frac{1}{2} \sum_{s_{wsn}(t), s_{wsn}(t-1) \in \mathbb{R}^{2*(N-1)}} \| s_i(t) - \hat{fs}_{\omega}^{(i)}(s_{wsn}(t), s_{wsn}(t-1)) \|^2 \tag{6}$$

3.3.2. Failure Detection Module

The aim of this module is to detect sensor failures through implicit estimations of the dynamic system. In general, this process considers two steps: (i) compute the error between the dynamic of the system and the estimates, and (ii) compute a metric for failure detection. This is described below.

Consider a WSN with N sensors as nodes, s_i for all $i = 1, \ldots, N$. Moreover, consider the dynamics function model $\hat{f}_{\omega}$ of a sensor as (7), where $s_i(t) \in \mathbb{R}$ is the value of sensor s_i in time t, $s_{wsn}(t) \in \mathbb{R}^{(N-1)}$ is a vector of values of the $(N-1)$ sensors $s_{wsn}(t) = (s_1(t), \ldots, s_k(t), \ldots, s_{(N-1)}(t))$ and $i \neq k$ in time t, and $\Delta\hat{s}_{wsn}(t)$ represents the changes in sensors values s_{wsn} in time t.

$$\Delta\hat{s}_{wsn}(t) = \hat{f}_{\omega}(s_{wsn}(t), s_i(t)) \tag{7}$$

The dynamic function $\hat{f}_\omega$ is proposed to be parameterized with AHN, where the vector of parameters ω represents the hydrogen, carbon and molecular centers over the hydrocarbon compound, i.e., $\omega = \{H_{ir}, \sigma_r, \mu_j\}$. After training the AHN-model $\hat{f}_\omega$ and using (7), it is easy to compute the predicted next values of sensors s_{wsn} for time $t+1$ as expressed in (8).

$$\hat{s}_{wsn}(t+1) = s_{wsn}(t) + \hat{f}_\omega(s_{wsn}(t), s_i(t)) \tag{8}$$

For training the AHN-model, we use tuples of the form $(s_{wsn}(t-1), s_i(t-1)) \in \mathbb{R}^N$ as training inputs and differences $\Delta s_{wsn}(t) = s_{wsn}(t) - s_{wsn}(t-1) \in \mathbb{R}^{N-1}$ as training outputs, collected from the WSN for a given period of time, and all these data were randomly shuffled. Thus, we train the dynamics function model $\hat{f}_\omega$ by minimizing the error (9) using the gradient descent method.

$$E(\omega) = \frac{1}{2} \sum_{s_{wsn}(t-1), s_i(t-1) \in \mathbb{R}^N} \|\Delta s_{wsn}(t) - \hat{f}_\omega(s_{wsn}(t-1), s_i(t-1))\|^2 \tag{9}$$

It is remarkable to say that using the definition from (7), it is possible to observe that changes of sensors values $\Delta\hat{s}_{wsn}$ in t are impacted by the actual values of the sensor $s_i(t)$. Thus, if $s_i(t)$ fails, then it will impact on the changes of sensors values and also in the prediction of the next values $\hat{s}_{wsn}(t+1)$.

From the above, it is possible to detect a failure in the system comparing the actual values of sensors $s_{wsn}(t)$ and the estimation $\hat{s}_{wsn}(t)$ as shown in (10), where E_i is the error between the actual and estimate values of the sensors using the function model excited from s_i.

$$E_i(t) = \|s_{wsn}(t) - \hat{s}_{wsn}(t)\| \tag{10}$$

Since an actual failure of s_i affects on all the function models built for each sensor in the WSN, then E_j for all $j = 1, \ldots, N$ are affected differently. To take into account all these variations, an overall error (11) is computed.

$$E(t) = \sum_{j=1}^{N} E_j(t) \tag{11}$$

It would be easy to use a threshold scheme for sensor failure detection [19]. However, tuning the threshold is highly dependent on the behavior of the WSN. Hence, we propose to normalize the overall error online, obtaining its mean μ_E and standard deviation σ_E from zero to t, such a confidence value α for failure detection can be selected. To this end, the failure detection function $fail(t)$ can be expressed as in (12).

$$fail(t) = \begin{cases} 1 & \left(\frac{E(t)-\mu_E}{\sigma_E}\right)^2 > \alpha \\ 0 & otherwise \end{cases} \tag{12}$$

3.3.3. Sensor Identification

The aim of this phase is to determine which sensor fails in a given time t. From the above description, if a sensor failure is produced, then it is possible to detect it comparing the dynamic system with the estimation of the dynamics. Since the estimation dynamics is computed for each sensor node, then the one with the highest error would be identified as the failure one, as described below.

To identify a failed sensor s_i over a WSN of N sensor nodes, it is possible to simply compare the actual values $s_k(t)$ and the estimation values $\hat{s}_k(t)$ of all sensors (using the sensor model $\hat{fs}_\omega^{(k)}$), and determine which sensor k obtains the maximum difference, as calculated in (13).

$$i = \max_k \|s_k(t) - \hat{s}_k(t)\| \tag{13}$$

3.3.4. SFDIA Using AHN

To this end, a complete SFDIA is proposed over a WSN of N sensor nodes by employing AHN at different levels of the approach. Algorithm 2 summarizes the whole SFDIA strategy.

It consists in five phases: (i) a training step that consists on collecting healthy data for building all AHN-models required for the strategy (sensor models and dynamic function models), (ii) a failure detection step that determines if there exists or not a sensor failure, (iii) a sensor identification step where the system recognizes which sensor failed, (iv) a sensor accommodation step that replaces the sensor failed by the sensor estimations done with its related sensor model, and (v) an update statistics step that recalculates the statistical values of overall error that allows to define a failure (c.f. Figure 5).

Algorithm 2 Sensor failure, identification and accommodation strategy using AHN

```
Data: the WSN with $N$ sensor nodes and failure confidence value $\alpha$.
Result: SFDIA strategy.
 Training Phase:
 1. Collect healthy data from WSN.
 2. Train all sensor models $\hat{fs}_{\omega}^{(k)}$ based on AHN.
 3. Train dynamic function models $\hat{f}_{\omega}$ based on AHN.
 while true do
     Failure Detection Phase:
     4. Compute the error in dynamic function models using (10).
     5. Calculate the overall error in dynamic function models using (11).
     6. Compute the failure function $fail$ using (12).
   if fail then
       Sensor Identification Phase:
       7. Detect the sensor failed $s_i$ using (13).
       Sensor Accommodation Phase:
       8. Replace sensor $s_i$ by estimation $\hat{s}_i$ using $\hat{fs}_{\omega}^{(i)}$.
   else
       Update Phase:
       9. Calculate the online statistics $\mu_E$ and $\sigma_E$ of overall error.
   end
end
```

4. Applicability of the Proposed SFDIA with AHN

We conducted a preliminary experimentation using the SFDIA strategy including AHN in order to test its performance over large WSN. Two experiments using both simulated and real WSN were done, as described in the following section.

4.1. SFDIA with AHN on Simulated WSN

We simulated three WSN with different number of nodes, ranging from $N = 20, 50$ and 100. In this case, each sensor node simulates the measurement of indoor temperature in different locations. Data were simulated using a pseudo-stochastic periodic function and random values in the frequency and translation parameters. Figure 8 shows thumbnails of sensor nodes in the experiment with $N = 20$, only in the testing data.

For training sensor models and dynamic function models, we generated 10,000 samples, 70% of them for training and 30% for testing. The training set was considered as healthy information. We built all AHN models using the following parameters: molecules $m = 5$, learning rate $\eta = 0.1$ and a maximum of 1000 iterations as stop criterion. For suitability, data were normalized uniformly before using at training step.

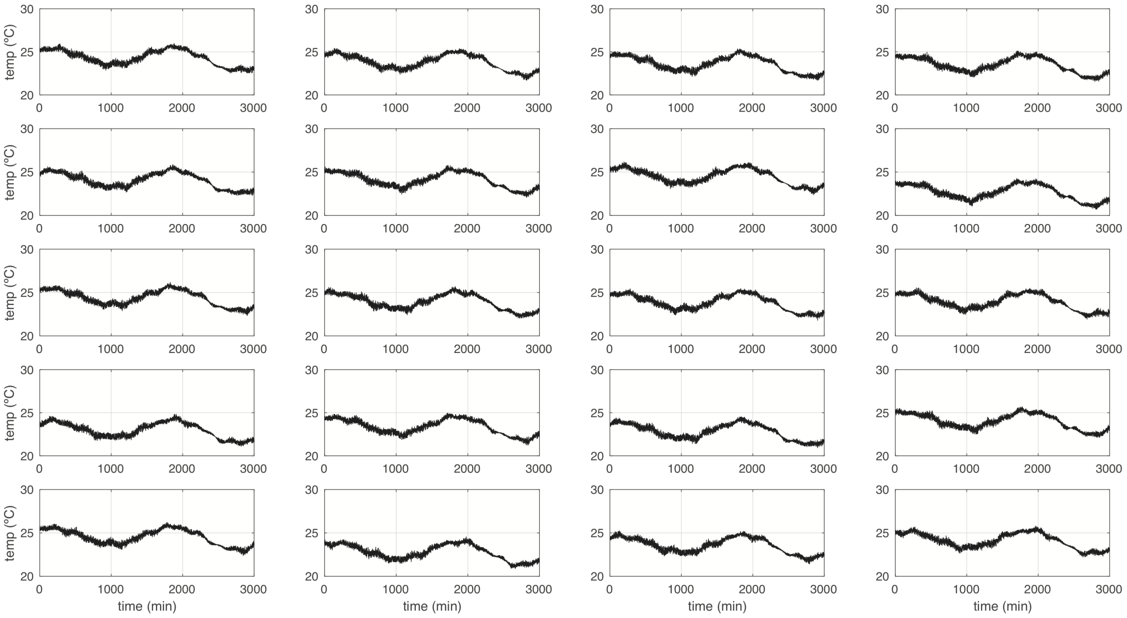


Figure 8. Thumbnails of sensor nodes in a simulated WSN with $N = 20$.

We then polluted sensor data artificially during two windows of time. These windows were chosen to be fixed in positions for comparative purposes. The windows for failure sensors were in ranges $(110, 150)$ min and $(900, 1250)$ min. After applying our proposed SFDIA strategy, of Algorithm 2, over the different scenarios ($N = 20, 50, 100$) in WSN, we obtained the results summarized in Table 1. As shown, these results report the accuracy of the failure detection phase, using (12), in comparison with the ground-truth. Figure 9 depicts an example when $N = 20$ of the comparison between the estimated failure detection and the real failure.

It is shown in Figure 9 that the first part of the estimation in failure detection computes several false-positive. However, this is not an issue assuming that in these regions, the actual values of the identified failure sensor will be replaced by estimated values using the corresponding sensor model. Moreover, from Table 1 it can be observed that increasing the number of nodes does not affect significantly the accuracy of failure detection, e.g., mean accuracy of 89.42%. Thus, our proposed SFDIA with AHN can deal with different number of sensor nodes in WSN.

Table 1. Accuracy in failure detection in different simulated WSN.

N	Accuracy (%)
20	92.13
50	88.09
100	88.03

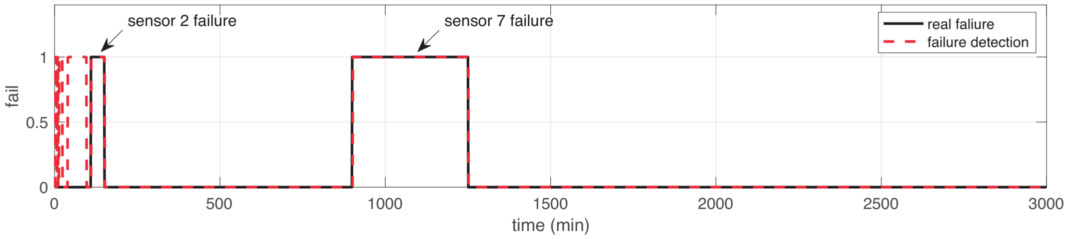


Figure 9. Failure detection response in the system, for $N = 20$.

4.2. SFDIA With AHN on Real WSN

We tested our proposed SFDIA with AHN on a real WSN. We employed the Intel Lab data set [47] that comprises 54 sensors retrieving weather data such as: temperature, relative humidity and light, as well as the voltage of the battery. Data were gathered in the Intel Lab, as shown in Figure 10, between 28 February and 5 April 2004.

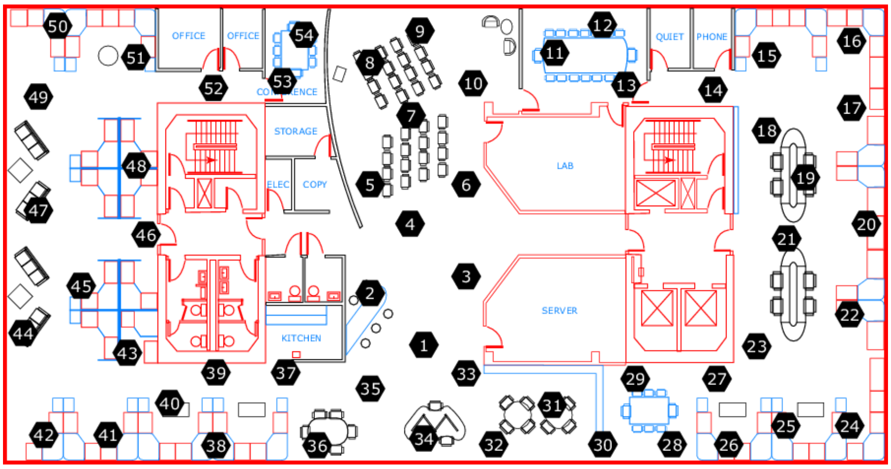


Figure 10. Location of the sensor nodes in the Intel Lab. Layout taken from [47].

For training sensor models and dynamic function models, we extracted 12651 samples for training and 22,251 for testing. These data only contain temperature values as continuous time series. The training set was considered as healthy information, as depicted in Figure 11. In this case, testing data contain altered values (e.g., far from the mean value of time series) from the original data set. We built all AHN models using the following parameters: molecules $m = 5$, learning rate $\eta = 0.1$ and a maximum of 1000 iterations as stop criterion. For suitability, data were normalized uniformly before using at training step.

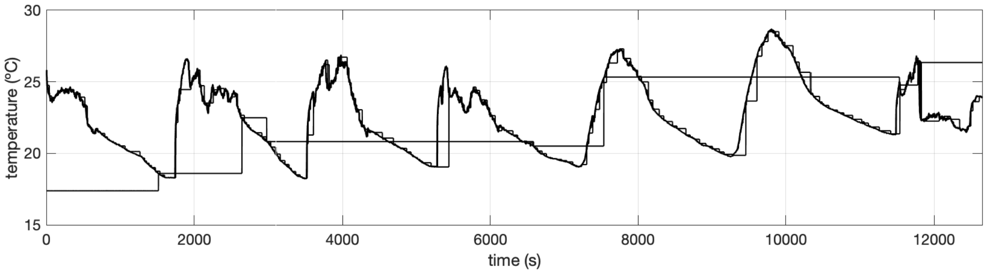


Figure 11. Training data set obtained from the Intel Lab WSN with $N = 54$ sensor nodes.

We applied the proposed SFDIA with AHN of Algorithm 2. After completion of the SFDIA strategy, this experiment reported an accuracy of 98.30% in failure detection phase. Figure 12 shows the testing data set with some failures, and the comparison between the estimated failure detection and the real failure. It can be observed that some regions of false-negative values are estimated. Unfortunately, the proposed SFDIA strategy cannot correct sensor failure detection in the latter regions. Nevertheless, this experiment validates the feasibility to detect sensor failures in an accurate way over real data.

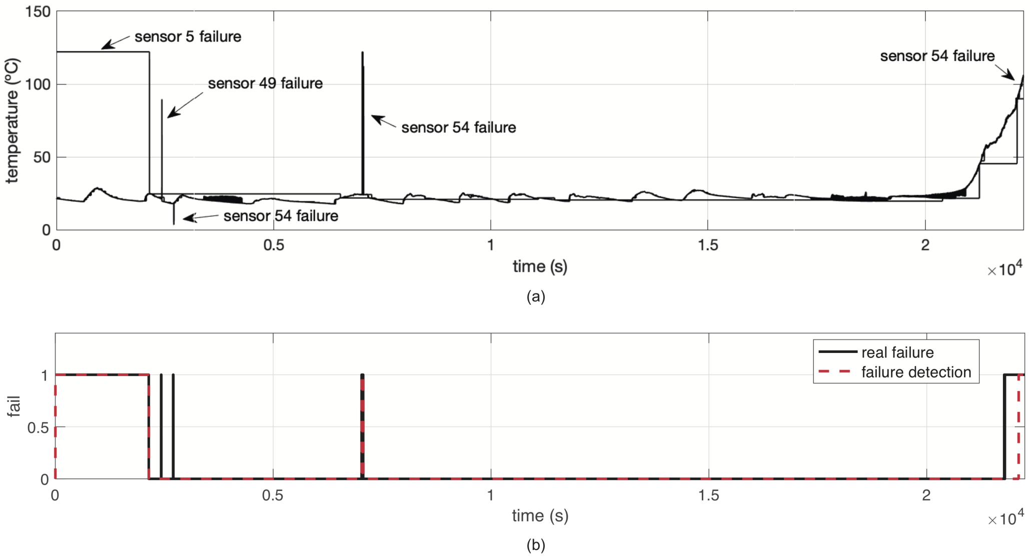


Figure 12. (**a**) Sensor failures in testing data and (**b**) failure detection response in the Intel Lab WSN with $N = 54$ sensor nodes.

5. Experimental Results and Discussion

We conducted a series of experiments over the WSN, described in Section 2, to evaluate the performance of our proposed SFDIA strategy based on AHN. Three enclosed temperature sensors (s_1, s_2, s_3) and a temperature remote web service (s_4) were considered as sensor nodes $(N = 4)$. We collected 2409 temperature samples from all the sensor nodes during ten days in a non-uniform way (no sample rate defined).

5.1. Training Phase

We considered the first seven days of data (1687 samples) as healthy information retrieved from the sensor nodes. So, these data were employed for training all the sensor and dynamic function AHN-models.

For the temperature sensor estimators, we configured the AHN models with the following parameters: molecules $m = 5$, learning rate $\eta = 0.1$ and a maximum of 1000 iterations as stop criterion. For the dynamic function estimators, the AHN models were configured with the parameters: molecules $m = 5$, learning rate $\eta = 0.1$ and a maximum of 1000 iterations as stop criterion. Notice that we only trained models related to the physical sensors in our WSN: s_1, s_2 and s_3. For suitability, data were normalized uniformly before using for training the AHN-models.

Figure 13 shows the output response of the temperature sensor AHN-models. As observed, the left part depicts estimation over the training data and the right part depicts estimation over the testing data (722 samples representing three days). Table 2 summarizes the root-mean-squared error (RMSE) in the estimation for both training and testing in all the physical sensor nodes.

For the dynamic function estimators, we configured the AHN models with the following parameters: molecules $m = 5$, learning rate $\eta = 0.1$ and a maximum of 1000 iterations as stop criterion. For the dynamic function estimators, the AHN models were configured with the parameters: molecules $m = 5$, learning rate $\eta = 0.1$ and a maximum of 1000 iterations as stop criterion. In this case, we only trained models related to the sensor nodes: s_1, s_2 and s_3. Notice that these models shape the dynamics of other sensor nodes, except for s_i. We normalized data uniformly before using for training the AHN-models.

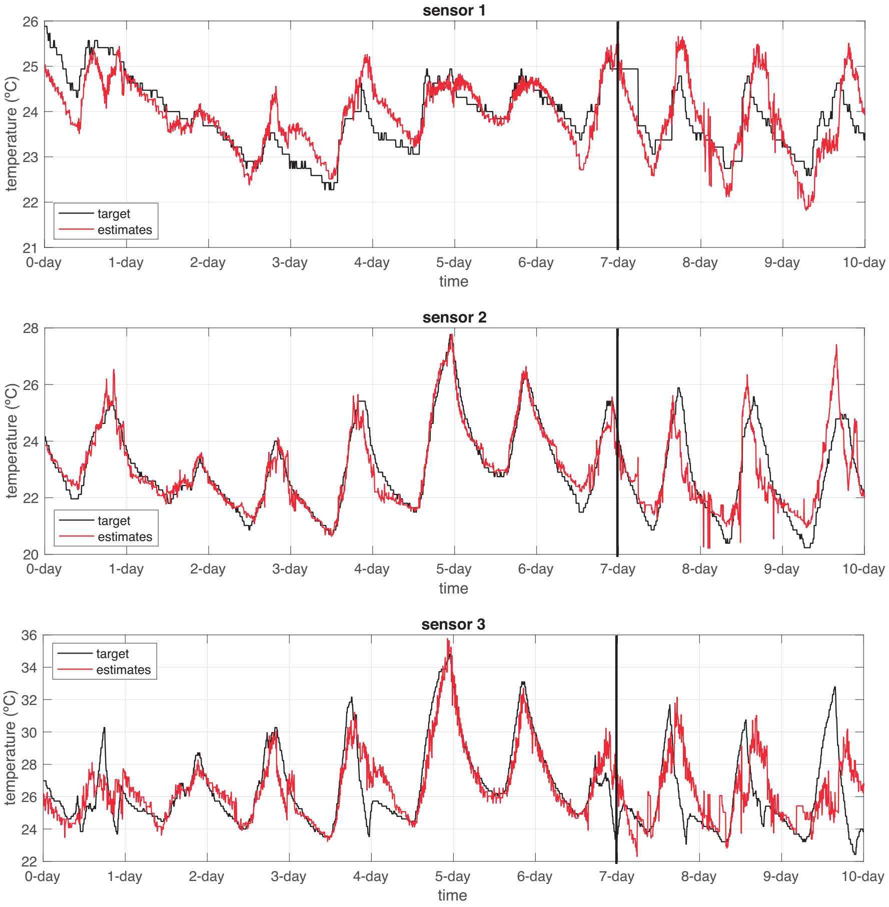


Figure 13. Temperature estimation associated to each physical sensor node s_1, s_2 and s_3 in comparison with the actual values. Left side of the vertical line refers to training data and right side refers to the testing data.

Table 2. Output response of the temperature estimations in training and testing, using AHN-models.

Temperature Sensor Model	RMSE (°C) in Training	RMSE (°C) in Testing
$\hat{s}_1$	0.4482	0.6986
$\hat{s}_2$	0.4042	0.9739
$\hat{s}_3$	1.1986	2.7036

Figure 14 shows the absolute error between the output response of the dynamic function AHN-models and the actual dynamics system. Again, the left part depicts estimation over the training data and the right part depicts estimation over the testing data. Table 3 summarizes the RMSE in the estimation for both training and testing in all the dynamic functions. Notice that these RMSE results are for each of the sensor node signals related in the dynamic function.

Table 3. Output response of the dynamic functions estimations in training and testing, using AHN-models.

Dynamic Function Model	RMSE (°C) in Training	RMSE (°C) in Testing
$\hat{f}_{\omega}^{(1)}$	(0.0196,0.0579,0.0403)	(0.0307,0.0838,0.1042)
$\hat{f}_{\omega}^{(2)}$	(0.0087,0.0583,0.0404)	(0.0187,0.0824,0.1045)
$\hat{f}_{\omega}^{(3)}$	(0.0097,0.0213,0.0352)	(0.0161,0.0225,0.0846)

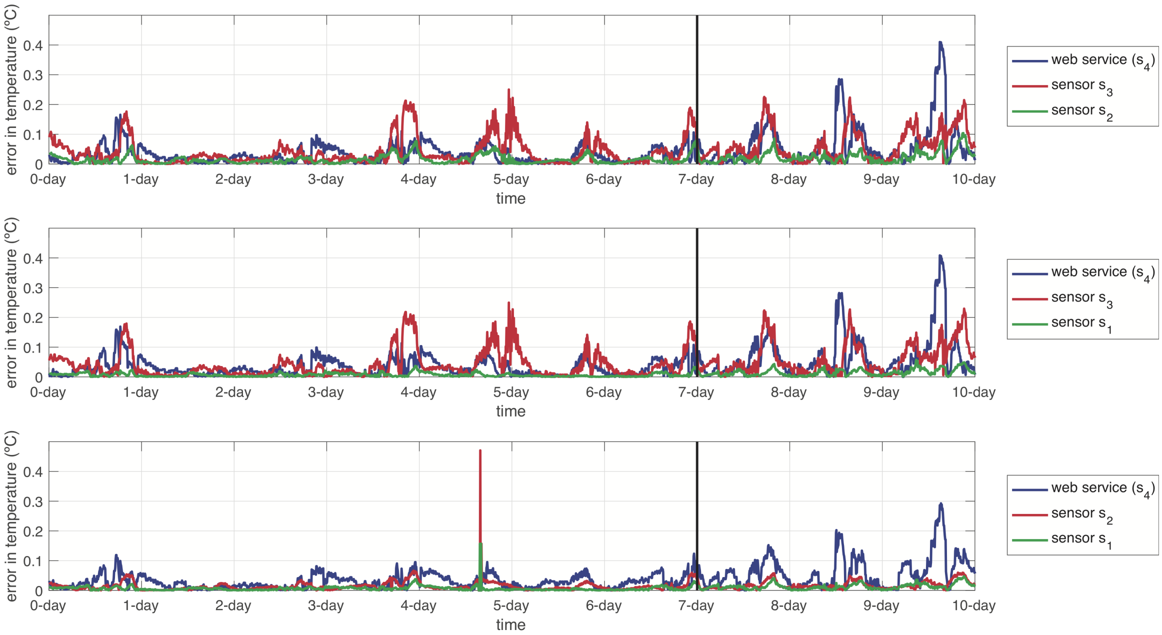


Figure 14. Absolute error between the dynamic function estimations associated to each sensor node s_1, s_2 and s_3 in comparison with the actual values. Left side of the vertical line refers to training data and right side refers to the testing data.

5.2. Failure Detection Phase

Using the above AHN-models, we conducted an experiment on sensor failure, so we can test and evaluate the performance of our SFDIA proposal. To do so, we collected data from the WSN in the three-day period (i.e., the testing set), and then we changed some sensor values to artificially produce sensor failures. The sensor failures where done by changing the real value with a random constant value below the mean values of sensors (measured in the training data). Then, we simulate an online failure detection.

At each simulation step, we computed the error between the dynamic function estimation $\hat{f}_{\omega}^{(i)}$ and the actual values of the sensors in WSN, except the sensor i, such that (10) is calculated. After that, we obtained the overall error by adding up the error in the dynamic function estimations, as expressed in (11). Lastly, we computed the failure function $fail$ using (12) with a confidence value $\alpha = 1.65$ representing 95.05% of confidence in failure detection. Figure 15 shows the output response of this sensor failure detection. As observed, the failure detection is done in 100%, but until now, the sensor failed is not detected.

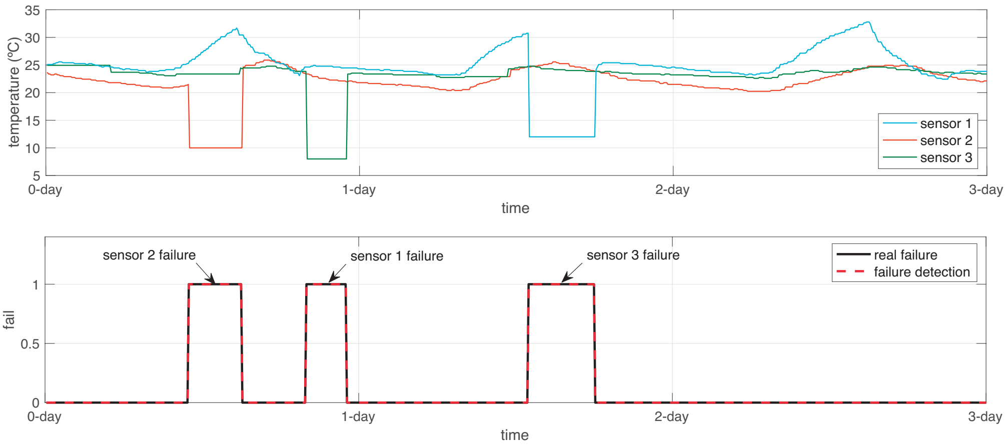


Figure 15. Failure detection response in the system.

5.3. Sensor Identification and Accommodation Phases

In order to identify the sensor who failed, the proposed SFDIA computes a comparison between the actual value of one sensor s_k and the estimation provided by the sensor AHN-model $\hat{s}_k = \hat{f}s_{\omega}^{(k)}$, as expressed in (13). Figure 16 shows the comparison between the actual and the estimated sensor failed. For this experiment, the accuracy of the sensor identification over the testing set was 94.18%.

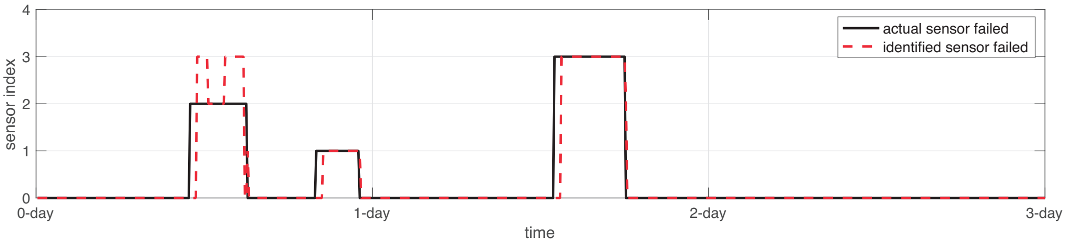


Figure 16. Sensor failure identification response in the system.

To this end, the sensor identification served for sensor accommodation. In this regard, failed values from sensors s_k were changed by estimations from sensors $\hat{s}_k$, as shown in Figure 17. Notice that only 5.82% of testing data failed to be recovered.

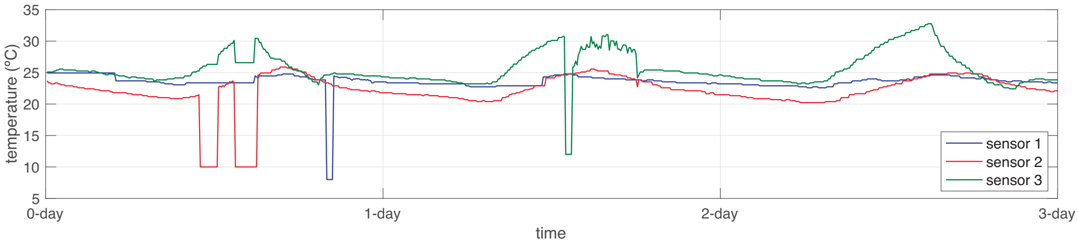


Figure 17. Sensor accommodation and final results over the WSN using the proposed SFDIA approach.

5.4. Discussion

From the previous results, it can be seen that AHN-models can detect failures, identify and accommodate sensor nodes with accuracy over the testing set of 94.18%. In this context, the proposed method of intelligent failure detection to discover malfunctions has the following advantages: the response of the dynamic functions estimations in training and testing, using AHN-models proved high accuracy data; the system had the capability to detect the failure of sensor with a confidence of 95.05% in failure detection; in order to identify the sensor that fails, the proposed SFDIA provided a good accuracy in the detection response with 100% correct appreciation rate; furthermore, the sensor accommodation was obtained from this identification where only 5.82% of testing data failed to be recuperated.

Nonetheless, a number of limitations have been identified in the failure detection method that will need to be addressed in order to improve the efficiency in the system. It requires more data acquisition from sensor nodes in order to perform better accuracy to train the AHN-models, it needs more sensor nodes for failure detection in each one of them, and it also needs retraining to minimize the error in time and to collect healthy data. Also, it is important to highlight that the threshold parameter α was set experimentally. Thus, a further study on how to obtain this parameter is required. One possibility refers to compute α by simulating fake sensor failures during the training phase.

6. Conclusions

In this work, we proposed a sensor failure detection, identification and accommodation approach based on artificial hydrocarbon networks over a wireless sensor network. This approach consisted in two modules: the temperature estimation and the dynamic function models. In addition, an algorithm (c.f. Algorithm 2) was introduced as the implementation of this method.

After experimentation over large WSN with simulated and real data sets and a real small WSN, we observed that our proposal can detect sensor failures in a high accurate way. Moreover, sensor failure identification and accommodation could be done in 94.18% of accuracy over the testing period. This validates that our proposal using AHN can be used as an SFDIA over WSN.

For future work, we are considering extending our proposal for more sensors in the network as well as to provide real-time execution of the SFDIA based on AHN. We are also implementing this system in a real-world application for better analysis of our proposed SFDIA system. To this end, a study on obtaining the threshold parameter α is also required.

Author Contributions: Conceptualization, S.G. and H.P.; Investigation, S.G. and H.P.; Methodology, S.G. and H.P.; Validation, S.G. and H.P.; Writing—original draft, S.G. and H.P.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Denissen, J.; Butalid, L.; Penke, L.; van Aken, M. The Effects of Weather on Daily Mood: A Multilevel Approach. *Emotion* **2008**, *8*, 662–667. [CrossRef] [PubMed]
2. Zhao, S.; Fan, L. Reports the current weather conditions on cell phones using web services. In Proceedings of the 9th IEEE International Conference on Cognitive Informatics, Beijing, China, 7–9 July 2010.
3. Othman, M.F.; Shazali, K. Wireless sensor network applications: A study in environment monitoring system. *Procedia Eng.* **2012**, *41*, 1204–1210. [CrossRef]
4. Ferdoush, S.; Li, X. Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications. *Procedia Comput. Sci.* **2014**, *34*, 103–110. [CrossRef]
5. Alavinia, R.; Zhu, Z.; Zhang, S. Design and Simulation of a Meteorological Data Monitoring System Based on a Wireless Sensor. *Int. J. Online Eng.* **2016**, *12*, 27. [CrossRef]
6. Devaraju, J.T.; Suhas, K.R.; Mohana, H.K.; Patil, V.A. Wireless Portable Microcontroller based Weather Monitoring Station. *Measurement* **2015**, *76*, 189–200. [CrossRef]
7. Xiu, D.; Sun, Q.; Ge, C.; Lao, Y.; Du, Y.; Wang, Y.; Shi, Y. Design of Mobile Meteorological Monitor Based on Wireless Sensor Network. *Appl. Mech. Mater.* **2013**, *303–306*, 938–944. [CrossRef]
8. Miorandi, D.; Sicari, S.; Pellegrini, F.D.; Chlamtac, I. Internet of things: Vision, applications and research challenges. *Ad Hoc Netw.* **2012**, *10*, 1497–1516. [CrossRef]
9. Ray, P.P. A Survey of IoT Cloud Platforms. *Future Comput. Inf. J.* **2017**, *1*, 35–46. [CrossRef]
10. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
11. Lazarescu, M.T. Design and field test of a WSN platform prototype for long-term environmental monitoring. *Sensors* **2015**, *15*, 9481–9518. [CrossRef] [PubMed]
12. Bijarbooneh, F.H.; Du, W.; Ngai, E.C.H.; Fu, X.; Liu, J. Cloud-Assisted Data Fusion and Sensor Selection for Internet of Things. *IEEE Internet Things J.* **2016**, *3*, 257–268. [CrossRef]
13. Luo, J.; Wu, D.; Pan, C.; Zha, J. Optimal Energy Strategy for Node Selection and Data Relay in WSN-based IoT. *Mob. Netw. Appl.* **2015**, *20*, 169–180. [CrossRef]
14. Guanochanga, B.; Cachipuendo, R.; Fuertes, W.; Salvador, S.; Benitez, D.; Toulkeridis, T.; Torres, J.; Villacis, C.; Tapia, F.; Meneses, F. Real-Time Air Pollution Monitoring Systems Using Wireless Sensor Networks Connected in a Cloud-Computing, Wrapped up Web Services. In Proceedings of the Future Technologies Conference (FTC), Vancouver, BC, Canada, 15–16 November 2018.
15. Ram, K.S.S.; Gupta, A.N.P.S. IoT based Data Logger System for weather monitoring using Wireless sensor networks. *Int. J. Eng. Trends Technol.* **2016**, *32*, 71–75. [CrossRef]
16. Shah, J.; Mishra, B. IoT enabled environmental monitoring system for smart cities. In Proceedings of the International Conference on Internet of Things and Applications (IOTA), Pune, India, 22–24 January 2016.

17. Soliman, M.; Abiodun, T.; Hamouda, T.; Zhou, J.; Lung, C.H. Smart home: Integrating internet of things with web services and cloud computing. In Proceedings of the IEEE 5th International Conference on Cloud Computing Technology and Science, Bristol, UK, 2–5 December 2013.
18. Ponce, H.; Gutierrez, S. An indoor predicting climate conditions approach using Internet-of-Things and artificial hydrocarbon networks. *Measurement* **2019**, *135*, 170–179. [CrossRef]
19. Hussain, S.; Mokhtar, M.; Howe, J.M. Sensor failure detection, identification, and accommodation using fully connected cascade neural network. *IEEE Trans. Ind. Electron.* **2015**, *62*, 1683–1692. [CrossRef]
20. Nagarajan, S.; Kayalvizhi, S.; Karthikeyan, B. Neural network based intelligent sensor fault detection in a three tanks interacting level process. In Proceedings of the International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 3–5 March 2016; pp. 2429–2434.
21. Singh, S.S.; Jinila, Y.B. Sensor node failure detection using check point recovery algorithm. In Proceedings of the International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, India, 8–9 April 2016; pp. 3–6.
22. Zidi, S.; Moulahi, T.; Alaya, B. Fault detection in wireless sensor networks through SVM classifier. *IEEE Sens. J.* **2018**, *18*, 340–347. [CrossRef]
23. DSA Initiative. Distributed Services Architecture (DSA). Available Online: http://iot-dsa.org/(accessed on 1 October 2018).
24. Yahoo. Yahoo Weather API. Available online: https://developer.yahoo.com/weather/ (accessed on 1 October 2018).
25. EnOcean Alliance Inc. EnOcean Alliance. Avalible online: https://www.enocean-alliance.org/ (accessed on 1 October 2018).
26. Ploennigs, J.; Ryssel, U.; Kabitzsch, K. Performance analysis of the EnOcean wireless sensor network protocol. In Proceedings of the IEEE 15th Conference on Emerging Technologies & Factory Automation, (ETFA 2010), Bilbao, Spain, 13–16 September 2010.
27. Lopez-Iturri, P.; Celaya-Echarri, M.; Azpilicueta, L.; Aguirre, E.; Astrain, J.; Villadangos, J.; Falcone, F. Integration of Autonomous Wireless Sensor Networks in Academic School Gardens. *Sensors* **2018**, *18*, 3621. [CrossRef] [PubMed]
28. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
29. Lee, S.H.; Lee, S.; Song, H.; Lee, H.S. Wireless sensor network design for tactical military applications: Remote large-scale environments. In Proceedings of the MILCOM 2009—2009 IEEE Military Communications Conference, Boston, MA, USA, 18–21 October 2009.
30. Mahamuni, C.V. A military surveillance system based on wireless sensor networks with extended coverage life. In Proceedings of the Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC), Jalgaon, India, 22–24 December 2016.
31. Madhu, A.; Sreekumar, A. Wireless Sensor Network Security in Military Application Using Unmanned Vehicle. Available online: https://pdfs.semanticscholar.org/e0aa/0e813ccb502c37f03a7b91b477e35336c63d.pdf (accessed on 5 February 2019).
32. Lazarescu, M.T. Design of a WSN platform for long-term environmental monitoring for IoT applications. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2013**, *3*, 45–54. [CrossRef]
33. Cao-hoang, T.; Duy, C.N. Environment monitoring system for agricultural application based on wireless sensor network. In Proceedings of the Seventh International Conference on Information Science and Technology, Da Nang, Vietnam, 16–19 April 2017.
34. Haque, S.A.; Rahman, M.; Aziz, S.M. Sensor anomaly detection in wireless sensor networks for healthcare. *Sensors* **2015**, *15*, 8764–8786. [CrossRef] [PubMed]
35. Zhu, Z.; Liu, T.; Li, G.; Li, T.; Inoue, Y. Wearable sensor systems for infants. *Sensors* **2015**, *15*, 3721–3749. [CrossRef] [PubMed]
36. Magana-Espinoza, P.; Aquino-Santos, R.; Cardenas-Benitez, N.; Aguilar-Velasco, J.; Buenrostro-Segura, C.; Edwards-Block, A.; Medina-Cass, A. Wisph: A wireless sensor network-based home care monitoring system. *Sensors* **2014**, *14*, 7096–7119. [CrossRef] [PubMed]
37. Erol-Kantarci, M.; Mouftah, H.T. Wireless sensor networks for cost-efficient residential energy management in the smart grid. *IEEE Trans. Smart Grid* **2011**, *2*, 314–325. [CrossRef]

38. Li, M.; Lin, H.J. Design and implementation of smart home control systems based on wireless sensor networks and power line communications. *IEEE Trans. Ind. Electron.* **2015**, *62*, 4430–4442. [CrossRef]
39. Suryadevara, N.K.; Mukhopadhyay, S.C. Wireless sensor network based home monitoring system for wellness determination of elderly. *IEEE Sens. J.* **2012**, *12*, 1965–1972. [CrossRef]
40. Ponce, H.; Ponce, P.; Molina, A. *Artificial Organic Networks: Artificial Intelligence Based on Carbon Networks*; *Studies in Computational Intelligence*, Springer: New York, NY, USA, 2014.
41. Ponce, H.; Martínez-Villasenor, L.; Miralles-Pechuán, L. A Novel Wearable Sensor-Based Human Activity Recognition Approach Using Artificial Hydrocarbon Networks. *Sensors* **2016**, *16*, 1033. [CrossRef] [PubMed]
42. Ponce, H.; Ponce, P.; Molina, A. The development of an artificial organic networks toolkit for LabVIEW. *J. Comput. Chem.* **2015**, *36*, 478–492. [CrossRef]
43. Ponce, H.; Ponce, P.; Molina, A. Adaptive noise filtering based on artificial hydrocarbon networks: An application to audio signals. *Expert Syst. Appl.* **2014**, *41*, 6512–6523. [CrossRef]
44. Ponce, H.; Miralles-Pechuán, L.; Martínez-Villasenor, L. A Flexible Approach For Human Activity Recognition Using Artificial Hydrocarbon Networks. *Sensors* **2016**, *16*, 1715. [CrossRef]
45. Ponce, H.; Martínez-Villasenor, L. Interpretability of Artificial Hydrocarbon Networks for Breast Cancer Classification. In Proceedings of the 30th International Joint Conference on Neural Networks, Anchorage, AK, USA, 14–19 May 2017.
46. Ponce, H.; Ponce, P. Artificial Organic Networks. In Proceedings of the Electronics, Robotics and Automotive Mechanics Conference (CERMA), Cuernavaca, Morelos, Mexico, 15–18 November 2011; doi:10.1109/CERMA.2011.12. [CrossRef]
47. Bodik, P.; Hong, W.; Guestrin, C.; Madden, S.; Paskin, M.; Thibaux, R. Intel Lab Data. Available online:. http://db.csail.mit.edu/labdata/labdata.html(accessed on 25 January 2019).