

RESEARCH ARTICLE

A game theory approach for cooperative control to improve data quality and false data detection in WSN

Roberto Casado-Vara¹  | Francisco Prieto-Castrillo^{1,2,3} | Juan M. Corchado^{1,4,5}

¹BISITE Research Group, IoT Digital Innovation Hub, Edificio Multiusos I+D+i, University of Salamanca, 37007, Salamanca, Spain

²Media Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts

³Harvard T.H. Chan School of Public Health, Harvard University, Boston, Massachusetts

⁴Department of Electronics, Information and Communication, Faculty of Engineering, Osaka Institute of Technology, Osaka, Japan

⁵Pusat Komputeran dan Informatik, Universiti Malaysia Kelantan, Kota Bharu, Malaysia

Correspondence

Roberto Casado-Vara, BISITE Research Group, IoT Digital Innovation Hub, Edificio Multiusos I+D+i, University of Salamanca, 37007, Salamanca, Spain.
Email: rober@usal.es

Juan M. Corchado, BISITE Research Group, IoT Digital Innovation Hub, Edificio Multiusos I+D+i, University of Salamanca, 37007, Salamanca, Spain.
Email: corchado@usal.es

Funding information

European Regional Development Fund (ERDF) within the framework of the Interreg program V-A Spain-Portugal 2014-2020 (PocTep), Grant/Award Number: 0123_IOTEC_3_E

Summary

New solutions are required for the management of heterogeneous distributed sensor networks in order to address the problem of data quality and false data detection in wireless sensor networks (WSNs). In this paper, we present a nonlinear cooperative control algorithm based on game theory. Here, a new model is proposed for the automatic processing and management of information in heterogeneous distributed WSNs. We apply our algorithm to a case study with the aim of improving the quality of temperature data collected from indoor surfaces by a WSN. Unlike the classic unsupervised methods, in the proposed algorithm, it is not necessary to define the number of clusters beforehand. Once the game reaches the game equilibrium, the resulting number of clusters can be used as input for the unsupervised classification analysis. Anomalous temperature values are corrected according to their neighborhood, without modifying the temperature clusters.

KEYWORDS

data quality and false data detection, game theory, nonlinear cooperative control, nonlinear models and systems, wireless sensors network

1 | INTRODUCTION

Wireless sensor networks (WSNs) have become important in the last years and are now present in, practically, all the sectors of our society.¹ Their capacity to gather large amounts of data facilitates the construction of smart environments, allowing for a flexible analysis of processes that occur in the environment and the services offered to users. There are many

Abbreviations: WSN, Wireless Sensors Network; GT, Game Theory.

advances in WSN architectures. However, the efficient management of the data collected by them is still a challenging aspect. Data management is complicated because different factors may cause data to be inconsistent (ie, it is difficult to determine if data are reliable or if sensors are accurate). However, the techniques that are currently available to us allow to store and manage big data. Siddia et al² conduct a survey that provides a detailed overview of the latest techniques and technologies for the efficient management of big data. On the other hand, the authors studied 48 data storage and management techniques and compared them with 60 related articles in the work of Gani et al.³ Therefore, there is a growing need for new WSN architectures that would merge data from heterogeneous sensors and manage the collected data intelligently.

The state of the art contains some architectures that allow to merge data from WSNs.^{4,5} There are also some novel frameworks that define methods for integrating dynamic and self-adaptable heterogeneous WSNs⁶ and for the management of data obtained from WSNs.⁷ Other architecture proposals have demonstrated that the accuracy of WSNs can be improved with the use of artificial neural networks.⁸ A popular practice is the use of hybrid multiagent systems for WSN coordination, which automatically process and merge data in heterogeneous distributed WSNs.⁹ Moreover, multiagent systems can improve WSN control and security. Shamshirband et al present a survey of the most recent designs of multiagent systems for detection of intrusion in WSNs.¹⁰ Other works focusing on the detection of intruders in WSNs use different hybrid control techniques in WSNs. There are several hybrid techniques that improve the control of the WSN in solving different problems. In line with the research that we conducted in this paper, a density-based fuzzy imperialist competitive clustering algorithm that employs clustering techniques to detect intruders in WSN is presented in the work of Shamshirband et al.¹¹ In addition, an interesting WSN control system that uses GT is presented in the aforementioned work.¹¹ However, some frameworks are designed for very specific purposes and their functionality is limited.¹²⁻¹⁴ It is also difficult to merge and manage the data obtained from heterogeneous WSN.¹⁵

In this paper, we present an algorithm that will ensure the robustness and reliability of the data collected by WSN. In our approach, we apply game theory (GT) to the data obtained from a WSN. The state of the art on WSNs and GT shows a research gap in this area. To the best of the authors' knowledge, no other study proposed the application of GT to the problem of data quality and false data detection. Thus, in this paper, we propose a game to solve the problem of data reliability. Our game is distributed and self-organized so that it can work in a WSN regardless of the number of sensors, the architecture of the WSN or the type of sensors to which the game is applied. The game is designed in such way that it provides the following features: it is capable of recognizing the neighborhood in which it is implemented (the environment of the sensors, which is defined further on in this paper). The game also identifies the possible coalitions that can be formed between the neighbors. Finally, the temperature is determined by the winning coalition of the sensor to which the game has been applied. Furthermore, the convergence of the game is monitored in order to find the number of iterations required to reach to the correct temperature.

This paper is organized as follows. Section 2 reviews the related work. Section 3 describes the proposed algorithm and the case study that tests the feasibility of the game. Section 4 presents empirical results and discussion. Furthermore, it outlines several industrial applications, and finally, Section 5 discusses the results and future work.

2 | BACKGROUND: GT AND WSN

The GT is a branch of applied mathematics that is used to study how players can interact with each other with the aim of obtaining a fair and stable distribution of useful resources within the studied problem. This way, GT analyzes the interaction between independent and self-interested players to find a solution to a particular problem. Saghezchi et al¹⁶ provide a general review of GT and demonstrate how it is applied to networking and signal processing problems. On the other hand, a WSN is a network of sensors whose communications are transmitted by wireless signals. WSNs are used to collect and study a wide range of magnitudes, such as sound, humidity, temperature, and many others. WSNs are the most suitable architectures for sensor systems, but their limited lifetime of operation is a disadvantage. Unlike other sensor networks with physical support (wire), the use of WSNs is limited by the amount of energy they can store, their calculation capacities, memory, information flow, and communication distance (see the work of Zhou and Mu¹⁷). Since, in most cases, sensors are deployed on a surface, it is difficult and expensive to replace a faulty sensor manually by checking the entire WSN. In addition, the sensors may lack global information of the entire network and the network topology can change with time. In the work of Tapia et al,¹³ a distributed sensor architecture is presented, which allows the sensors to operate autonomously without the need for a central node, whereas Corchado et al¹⁸ provide a practical case of using heterogeneous distributed sensor networks.

On the other hand, WSNs and GT are areas undergoing intense study, Han et al¹⁹ focused on finding innovative solutions to the challenges presented by next-generation WSNs. Since GT is an ideal tool for designing efficient and robust distributed algorithms, its use in the design and analysis of WSN information is attracting increased attention.²⁰ This survey looks at how GT is currently being used in WSNs; from classic to evolved games, from cooperative to noncooperative, and the NE in these cases. Shi et al²¹ make a general classification of the different uses of GT in WSNs; they are classified into the following groups: network management, communication, network security, and applications. Our proposal can be included in the applications category, within the data collection subgroup. This is one of the most important parts of a WSN, although in our case, instead of focusing on the operation of the network itself and its information transmission capacities, we will focus on the collected data.

WSN research has many potential applications including environment and building monitoring, industrial process control, infrastructure and WSN security, and automatic transport (see the work of Kottapalli et al²²). In this regard, Lorincz et al²³ study public building monitoring and propose various research lines for the use of WSNs in event detection. Su et al²⁴ used this study as a basis for creating a fuzzy system for the monitoring of temperature in a given area and for computing that area in the WSN. In this study, they present a fuzzy algorithm for temperature control in which the distance between the sensors is a key issue.

Mishra et al present a method addressing the problem of energy efficiency using WSN, GT, and ant colony-based approach.²⁵ Initially, several clusters were formed without using the WSN, and later on, additional clusters were created using the coalitions formed by the game. The algorithm proposed in this game overcomes the spatial correlation of the harvested data generated by neighboring nodes to form coalitions within clusters. Afsar²⁶ presents an approach to the problem of grouping sensor nodes using game coalitions and clustering theories (coalition-games theoretic clustering). In the work of Schmidt,²⁷ GT-based coalitions are presented for the analysis of the clustering methods-based economy. Finally, Xu et al²⁸ presents a work that can be used in different fields (economic, biological, wireless communications, etc) where decisions are based on cooperative strategies.

The methods presented in the reviewed literature overcome different challenges, such as energy efficiency and economic analysis using WSNs and GT, respectively, through the use of clustering techniques to form coalitions. Distributed sensor networks are also presented depending on the network topology and sensor neighborhood. All the investigations found in the state of the art present approaches of hybrid WSN systems with other techniques and technologies to provide solutions to different problems (eg, intrusion detection in WSNs). However, we observed a gap in the literature review since there are no algorithms that could automatically validate the data collected by a heterogeneous WSN. In our work, coalitions of neighbors are created by using clustering techniques. This distributed and self-organized (overall temperature equilibrium arises from local game interactions between sensors of an initially disordered temperature system) game is designed to provide reliability and robustness to the data collected by a WSN. It identifies sensors gathering defective or inaccurate measurements and detects areas with similar temperatures. This paper tackles the problem of reliability of WSN data from the point of view of GT and probability, which is a novel approach in this field.

3 | PROPOSAL AND CASE STUDY: TEMPERATURE SENSOR EVALUATION

In this section, we provide a mathematical justification of our work. We show how to build the temperature matrix. We also describe the placement of sensors, a mathematical formalization of the designed game, and finally its convergence and NE. We apply our game in a case of study and select temperature sensors to test our game.

This section is organized as follows. (1) This subsection describes the experiment. (2) The construction of the temperature matrix is described in this subsection. (3) The next subsections justify the presented game in mathematical terms. (4) The mathematical formulation of the game and the way in which coalitions are formed. (5) The characteristic function of game is described in next subsection. In the last subsections, this paper shows the temperature of the winning coalition, diffuse convergence, and the game equilibrium (GE). The mathematical justification is shown as follows.

3.1 | General description of the experiment

To test our model, we have chosen an indoor surface. At the time the sensors measured the temperature, the thermostat of the room showed 21°C. A mesh was used to place the sensors in the surface (Figure 1). With the help of laser levels, the sensors were placed vertically every half a meter and horizontally every 1 meter. A total of 81 sensors were deployed.



FIGURE 1 Indoor surface for testing our model and sensor [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Statistical table of measurements of the wireless sensor network sensors

Timestamp Start	Total Timestamp	Min Temp	Max Temp	Mean	Standard Deviation
2017-07-26T12:53:01.367Z	00:00:04.334Z	20	22.6	21.19	0.49

The sensor used is a combination of the ESP8266 microcontroller in its commercial version “ESP-01” and a DHT11 temperature and humidity sensor (Figure 1). The sum of both allows us for greater flexibility when collecting data and adaptability to the case study, since the DHT11 sensor is oriented to indoor (has an operating range of between 0°C and 50°C) according to its datasheet (<http://www.micropik.com/PDF/dht11.pdf>). The microcontroller collects data from this sensor through a one-wire protocol and communicates it to the environment via Wi-Fi using HTTP standards and GET/POST requests. The device is programmed using the ESP-IDF programming environment provided by the microcontroller manufacturer.

The sensors have been collecting data for one week at two-hour intervals. For our analysis, we selected the data collected by the sensors during the next timestamp, the start of the measurement was 2017-07-26T12:53:01.367Z and the end was 2017-07-26T12:53:05.701Z. A specific moment has been selected since the game that has been defined is a static and not a dynamic game (ie, it does not handle the data in a temporal evolution). In the following, a statistical summary of the measurements that were made with the sensors is presented in Table 1

3.2 | Building temperature data matrix

Let $M \in \mathbb{R}^3$ be a regular known surface on which the temperature will be measured, and let $U \in \mathbb{R}^2$ and $V \in M$ be open sets. Then, there is a local parameterization $\varphi : U \rightarrow V$, such that (V, φ^{-1}) is a chart from M . Then, a mesh is created in U , and since $\varphi(U) = V$, the mesh can be exported to V . In this way, the points at which the sensors will be placed are obtained as the mesh nodes. Once the sensors are correctly distributed, they begin to collect the temperature of the surface that is being monitored. Then, an area of $M \in \mathbb{R}^3$ and data collection carried out by the sensors $S = \{s_i\}_{i \in \mathbb{N}}$ (ie, the temperature measurements taken by each sensor) are stored in a matrix in an orderly fashion with respect to the mesh that has been created in $U \in \mathbb{R}^2$ (see Figure 2) and $\{p_{i,j}\}_{i,j \in \mathbb{N}}$ are the mesh nodes corresponding with the matrix elements. Each matrix element $s_{i,j}$ corresponds to point $p_{i,j}$ of the ordered mesh in the surface (ie, $s_{1,1} \rightarrow p_{1,1}$). In the opposite direction, it is also verified.

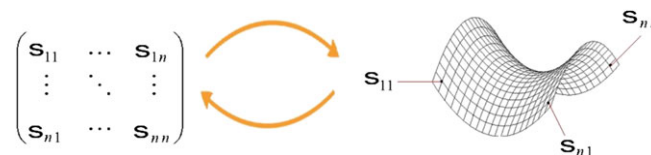


FIGURE 2 Matrix of sensors and position of the sensors on the surface [Colour figure can be viewed at wileyonlinelibrary.com]

For each sensor location $p_{i,j} \in M$, $Tp_{i,j}M$ is the tangent space to M in $p_{i,j}$, and $U \in Tp_{i,j}M$ is an open set. Then, we take a sensor $s_{p_{i,j}}$ and a neighbor sensor $s_{p_{i+1,j}}$, and repeat the same process, then an open set $V \in Tp_{i+1,j}M$ so that $U \cap V \neq \emptyset$. The U and V charts are compatible, since they meet the compatibility conditions. Now, let us look at the temperatures associated with each sensor in Equation (1)

$$T_{n,n} = \begin{pmatrix} t_{s_{1,1}} & \cdots & t_{s_{1,n}} \\ \vdots & \ddots & \vdots \\ t_{s_{n,1}} & \cdots & t_{s_{n,n}} \end{pmatrix}. \quad (1)$$

The temperature matrix was interpolated through grade 1 splines, with the new $p_{i,j}$ points created at the intersections of the different openings U_i and V_j (with $U_i \cap V_j$), the interpolation will be done by rows and columns.

Assume that there are $n + 1$ points in row i , $t_0 = p_{s_{i,1}}$, \dots , $t_n = p_{s_{i,n}}$, such that $t_0 \leq \dots \leq t_n$ (ie, the points are ordered). The intervals $[t_{i-1}, t_i)$ do not intersect each other, so there is no ambiguity in defining the nodes, the grade 1 spline that will be used to interpolate the new temperatures is as follows:

$$S(x) = \begin{cases} s_0(x) = a_0x + b_0 & \text{si } x \in [t_0, t_1) \\ \vdots \\ s_n(x) = a_nx + b_n & \text{si } x \in [t_{n-1}, t_n). \end{cases} \quad (2)$$

From this spline function in Equation (2), the following transformation can be defined in Equation (3)

$$F_i : T_{n,n} \longrightarrow T_{2^i n-1, 2^i n-1}^i, \quad (3)$$

such that

$$\begin{pmatrix} t_{s_{1,1}} & \cdots & t_{s_{1,n}} \\ \vdots & \ddots & \vdots \\ t_{s_{n,1}} & \cdots & t_{s_{n,n}} \end{pmatrix} \xrightarrow{F_1} \begin{pmatrix} t_{s_{1,1}} & s(t_{s_{1,1}}, t_{s_{1,2}}) & \cdots & s(t_{s_{1,n-1}}, t_{s_{1,n}}) & t_{s_{1,n}} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ t_{s_{n,1}} & s(t_{s_{n,1}}, t_{s_{n,2}}) & \cdots & s(t_{s_{n,n-1}}, t_{s_{n,n}}) & t_{s_{n,n}} \end{pmatrix}. \quad (4)$$

Then, the following chain of transformations allows to obtain the new temperature matrices sequentially:

$$T_{n,n} \xrightarrow{F_1} T_{2n-1, 2n-1}^1 \xrightarrow{F_2} T_{4n-1, 4n-1}^2 \xrightarrow{F_3} \cdots \xrightarrow{F_i} T_{2^i n-1, 2^i n-1}^i. \quad (5)$$

Therefore, the temperature matrix $T_{n,n}$ is transformed by F_i in the matrix $T_{2^i n-1, 2^i n-1}^i$, where the temperature at the new points is calculated with the interpolation defined by the spline.

GT is a branch of mathematics that is dedicated to the study of conflicting situations that occur when a group of agents with opposing or nonconcordant interests, must make their own, individual decisions which affect everyone. Traditionally, this conflict is called game and the player agents. The games almost always present a competition between them. When the rules of the game allow it, players can try to solve the competition through cooperation.

The theory of cooperative games focuses, not on the possible strategies that players can carry out but on the possibilities that coalitions (a coalition is any subset from the total set of players) of these players would have, in a game of N players, there are exactly 2^N coalitions.

3.3 | Mathematical formulation of the game

Let $n \geq 2$ denote the number of players in the game, numbered from 1 to n , and let $N = \{1, 2, \dots, n\}$ denote the set of players. A coalition, S , is defined to be a subset of N , $S \subseteq N$, and the set of all coalitions is denoted by \mathbb{S} . A cooperative game in N is a function u (characteristic feature of the game) that assigns to each coalition $S_i \subseteq \mathbb{S}$ a real number $u(S_i)$. In addition one has the condition $u(\emptyset) = 0$. In our case, the game will be nonnegative (the values of the characteristic function are always positive), monotonous (if more players are added to the coalition the value of the expected characteristic function does not change), simple and zero-normalized (players are obliged to cooperate with each other since individually they will obtain zero benefit).

In our case, the set of players is the set of ordered sensors S and the characteristic function u is defined as

$$u : 2^n \rightarrow \{0, 1\}, \quad (6)$$

such that, for each coalition of sensors, $u = 1$ or 0 if that particular coalition can vote or not, respectively (see Equations (6), (7)).

$$\mathbb{S} \ni S_i \longrightarrow u(S_i) = \{0, 1\} \in \mathbb{R}. \quad (7)$$

3.3.1 | Cooperative sensor coalitions

Cooperative games are defined by the fact that players can cooperate with each other in order to achieve a mutual benefit. Once the players have agreed to cooperate among themselves, a coalition should be formed. A coalition can be formed by any number of players. In our game, a single sensor cannot determine if its temperature is correct. Therefore, sensors are forced to cooperate in order to evaluate whether the temperature of the central sensor is correct in relation to their neighborhood. Sensors work together in coalitions in order to cooperate for a mutual benefit (ie, to verify that their temperatures are correct and to self-correct them if this is not the case). Sensors will work together by majority rule-based voting among the coalition members in the game presented in this paper. However, not all sensors can form coalitions, so it is necessary to establish limitations for the allowed coalitions that could potentially be formed regarding the respective central sensor.

The possible coalitions that the sensors will form will be limited by their position, that is, the coalitions can only be formed by neighboring sensors. Let us consider the matrix of the sensors and a pair of sensors $s_{i,j}$ y $s_{k,m}$ will be in the same neighborhood if and only if

$$\|(i - k)^2 - (j - m)^2\| \leq 1, \quad (8)$$

that is, if each sensor to which the game is applied is the center of a Von Neumann neighborhood, its neighbors are those lying within a Manhattan distance (in the matrix) equal to one. In addition, the following conditions have to be fulfilled by the allowed coalitions.

1. Coalition sensors have to be in the same neighborhood as defined in Equation (8).
2. Coalitions cannot be formed by a single sensor.

Assuming that the game is applied to sensor $s_{i,j}$, the allowed coalitions are as follows.

- $S_1 = \{s_{i,j}, s_{i+1,j}\}$
- $S_2 = \{s_{i,j}, s_{i-1,j}\}$
- $S_3 = \{s_{i,j}, s_{i,j-1}\}$
- $S_4 = \{s_{i,j}, s_{i,j+1}\}$
- $S_5 = \{s_{i,j}, s_{i,j+1}, s_{i+1,j}\}$
- $S_6 = \{s_{i,j}, s_{i,j+1}, s_{i-1,j}\}$
- $S_7 = \{s_{i,j}, s_{i-1,j}, s_{i,j-1}\}$
- $S_8 = \{s_{i,j}, s_{i-1,j}, s_{i,j-1}\}$
- $S_9 = \{s_{i,j}, s_{i,j-1}, s_{i+1,j}\}$
- $S_{10} = \{s_{i,j}, s_{i,j-1}, s_{i,j+1}\}$
- $S_{11} = \{s_{i,j}, s_{i-1,j}, s_{i+1,j}\}$
- $S_{12} = \{s_{i,j}, s_{i-1,j}, s_{i+1,j}, s_{i,j+1}\}$
- $S_{13} = \{s_{i,j}, s_{i-1,j}, s_{i+1,j}, s_{i,j-1}\}$
- $S_{14} = \{s_{i,j}, s_{i,j+1}, s_{i,j-1}, s_{i+1,j}\}$
- $S_{15} = \{s_{i,j}, s_{i,j+1}, s_{i,j-1}, s_{i-1,j}\}$
- $S_{16} = \{s_{i,j}, s_{i,j+1}, s_{i,j-1}, s_{i-1,j}, s_{i+1,j}\}$

Therefore, the set of allowed coalitions is $\mathbb{S} = \{s_i\}_{1 \leq i \leq 16}$. Thus, the game can be expressed as a majority game that is weighted according to the formed coalitions in the following way:

$$u \equiv \left\{ \frac{n}{2} + 1; s_1, \dots, s_n \right\}, \text{ where } s_i \text{ are all neighbors and } s_i \in \mathbb{S}. \quad (9)$$

Figure 3 illustrates a flowchart showing the steps involved in deciding if a coalition is allowed or not allowed.

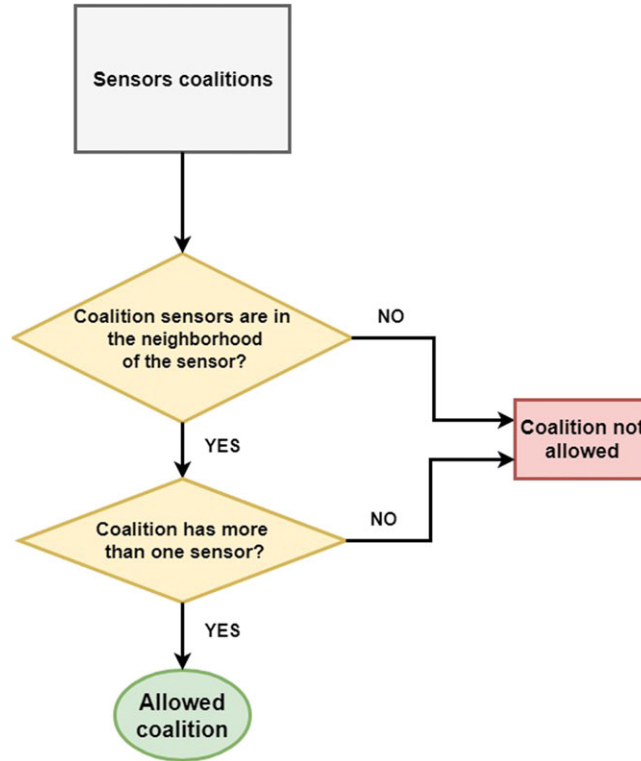


FIGURE 3 Allowed coalition decision-making flowchart [Colour figure can be viewed at wileyonlinelibrary.com]

3.3.2 | A characteristic function to find cooperative temperatures

In the proposed game, we want the neighborhood coalitions to democratically decide the temperature of the main sensor. To do this, they will form coalitions that will decide on the final temperature of the sensor, which will be determined by whether they can vote or not in the process. From the characteristic function defined in Equation (6), if the value is 1(0), the coalition can vote (not vote), respectively. s_i is the main sensor with its associated temperature t_{s_i} , and the characteristic function is built in the following way.

1. First, the average temperature of all the sensors is calculated

$$T_{s_i}^k = \frac{1}{V} \sum_i^V t_{s_i}; \quad (10)$$

here, $T_{s_i}^1$ represents the average temperature of the sensors's neighborhood s_i (including it) in the first iteration of the game and V is the number of neighbors in the coalition.

2. The next step is to compute an absolute value for the temperature difference between the temperatures of each sensor and the average temperature

$$\bar{T}_{s_i}^k = \left(\frac{1}{V} \sum_i^V |t_{s_i} - T_{s_i}^k|^2 \right)^{\frac{1}{2}}. \quad (11)$$

3. Using the differences in temperature values with regard to the average temperature $\bar{T}_{s_i}^k$ (see Equation (11)), a confidence interval is created and defined as follows:

$$I_{s_i}^k = \left(T_{s_i}^k \pm t_{(V-1, \frac{\alpha}{2})} \frac{\bar{T}_{s_i}^k}{\sqrt{V}} \right), \quad (12)$$

in Equation (12), we use the Student's t-distribution with an error of 1%.

4. In this step, we use a hypothesis test. If the temperature of the sensor lies in the interval $I_{s_i}^k$, it belongs to the voting coalition, otherwise, it is not in the voting coalition

$$u^k(s_1, \dots, s_n) = \begin{cases} 1, & \text{if } t_{s_i} \in I_{s_i}^k \\ 0, & \text{if } t_{s_i} \notin I_{s_i}^k. \end{cases} \quad (13)$$

5. The characteristic function will repeat this process iteratively (k is the number of the iteration) until all the sensors in that iteration belong to the voting coalition. In each iteration k , the following payoff vector (PV) of the coalition S_j (with $1 \leq j \leq n$, where n is the number of sensors in the coalition) in the step k ($PV(S_j^k)$) is available:

$$PV(S_j^k) = (u^k(s_1), \dots, u^k(s_n)), \text{ where } \sum_i^n u^k(s_i) \leq n. \quad (14)$$

The stop condition of the game iterations is $PV(S_j^k) = PV(S_j^{k+1})$ the process end. That is, let $PV(S_j^k) = (u^k(s_1), \dots, u^k(s_n))$ and let $PV(S_j^{k+1}) = (u^{k+1}(s_1), \dots, u^{k+1}(s_n))$. The iteration process ends when both PVs contain the same elements. This process is shown in the following equation:

$$\begin{cases} u^k(s_1) = u^{k+1}(s_1) \\ \vdots \\ u^k(s_n) = u^{k+1}(s_n). \end{cases} \quad (15)$$

Then, the game can find the solution that is defined in the following subsection.

3.3.3 | Solution concept of the cooperative game

Once the characteristic function has been applied to all sensors involved in this step of the game a PV in the step k is available (see Equation (14)). Since the proposed game is a cooperative game, the solution concept is a coalition of players that we have called GE. The GE of the proposed game is defined as the minimal coalition with more than half of the votes cast. In the following, we summarize the conditions that must be met by the winning coalition for the game to reach the GE. Let n be the number of players involved in this step of the game. Winning coalition must satisfy the following conditions.

1. Sum of the elements of the coalition PV must be higher than half plus 1 of the votes cast:

$$\sum_i^n u^k(s_i) \geq \frac{n}{2} + 1. \quad (16)$$

2. The coalition is maximal (ie, coalition with the greatest number of elements, different from 0, in its payoff vector $PV(S_j^k)$).

Therefore, the solution to the proposed game is the coalition that verifies both conditions from among all possible coalitions that are formed at each step k of the game.

3.3.4 | Temperatures of the winning coalition

Once the characteristic function decides which is the winning coalition, it is possible to calculate the temperature of the main sensor. Let $\{s_1, \dots, s_j\}$ be the winning coalition's sensors and $\{t_{s_1}, \dots, t_{s_j}\}$ be their associated temperature. The temperature that the game has voted to be the main sensor's temperature (MST) is calculated as follows:

$$MST = \max_{j \in |S_{\text{winner}}|} \{j * t_{s_i}\}_{s_i \in S_{\text{winner}}}, \quad (17)$$

where $|S|$ is the number of elements in the winning coalition. Therefore, the MST will be the maximum temperature that has the highest relative frequency. In case of a draw, it is resolved by the Lagrange criterion.

3.3.5 | Diffuse convergence

In each game iteration, there is a matrix with temperature (see Equation (1)). Hence, we define a sequence of arrays $\{M_n\}_{n \in \mathbb{N}}$, where the M_i element corresponds to the temperature matrix in step i of the game. Therefore, it can be said that the sequence of matrices is convergent if

$$\forall \epsilon > 0, \text{ there is } N \in \mathbb{N} \text{ such that } |M_{i-1} - M_i| \leq \epsilon \forall i \in \mathbb{N}, \forall i \geq N. \quad (18)$$

That is, if the element $m_{n,m}^{i-1} \in M_{i-1}$ and the element $m_{n,m}^i \in M_i$ are set and the convergence criterion is applied, we have

$$\forall \epsilon_{n,m} > 0, \text{ there is } N \in \mathbb{N} \text{ such that } \left| m_{n,m}^{i-1} - m_{n,m}^i \right| \leq \epsilon_{n,m} \quad (19)$$

$$\forall i \in \mathbb{N}, \forall i \geq N \text{ and } m_{n,m}^{i-1} \in M_{i-1}, m_{n,m}^i \in M_i.$$

Therefore, by applying the criterion of convergence in Equation (19) to all the elements, a new matrix is obtained with the temperature differences between the temperatures obtained in previous and in the next step of the game.

$$\begin{pmatrix} \left| m_{1,1}^{i-1} - m_{1,1}^i \right| & \cdots & \left| m_{1,m}^{i-1} - m_{1,m}^i \right| \\ \vdots & \ddots & \vdots \\ \left| m_{n,1}^{i-1} - m_{n,1}^i \right| & \cdots & \left| m_{n,m}^{i-1} - m_{n,m}^i \right| \end{pmatrix} \quad (20)$$

For the succession of matrices to be convergent, each of the sequences of elements that are formed with the $|m_{n,m}^{i-1} - m_{n,m}^i|$ must be less than the fixed $\epsilon > 0$. In this work, it is established that $\epsilon = 0.01$.

Furthermore, the game must incorporate fuzzy logic, since it makes decisions about the temperature of the main sensor. To this end, we introduce the concept of Fuzzy Matrix Convergence (FMC), starting from the previous definition of matrix convergence, we will say that a matrix is FMC if at least 80% of the elements of the matrix are convergent. This allows us to improve the efficiency with which the temperature matrix reaches convergence. In addition, this allows for the convergence of the game to be quite fast in its respective iterations. When the matrix reaches the FMC, temperatures have been self-corrected in a distributed and self-organized way.

3.4 | Distributed and self-organized justification

One of the main goals of our work is that the game is distributed and self-organized. It is distributed because the game can run without the need for a central node (ie, when game is applied to every single node, this node is the *central node* in this step, but in next step, other nodes will be the *central node*). On the other hand, the game is self-organized because in each step, the game uses the output of the previous step as the input of the next step. In this way, game players (ie, WSN nodes) interact in every game step between them to form temperature clusters with their neighboring sensors.

3.5 | Algorithm architecture

Figure 4 presents the architecture of the algorithm proposed in this paper. The architecture has several layers. First is the WSN where the sensors that collect the smart building temperatures are located. In the next layer is the proposed

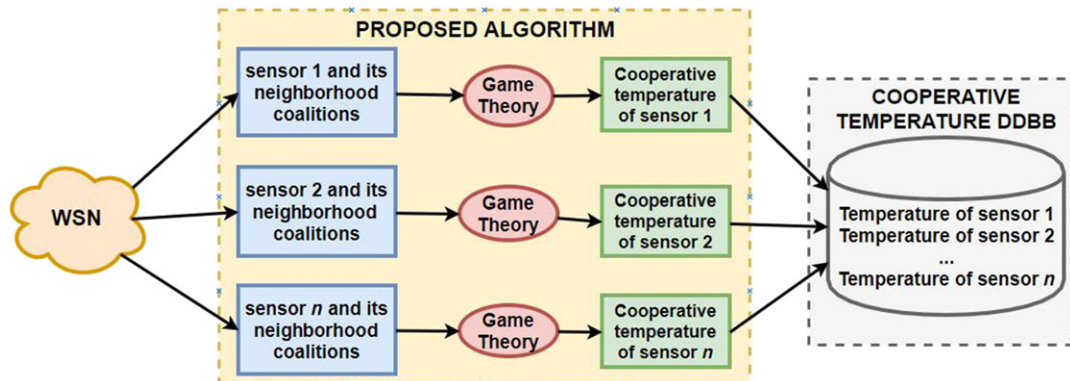


FIGURE 4 Architecture of the proposed algorithm. WSN, wireless sensor networks [Colour figure can be viewed at wileyonlinelibrary.com]

algorithm. This algorithm has three steps. (1) In the first step, the allowed coalitions of the sensors are formed. (2) In the second step, the game we have created for this algorithm is applied and the winning coalition is voted by majority rule. (3) In the last step, the temperature of the winning coalition is calculated, which is the cooperative temperature of the sensor to which the game is applied.

4 | EMPIRICAL RESULTS AND DISCUSSION

As highlighted throughout this paper, this proposal focuses on the application of a distributed and self-organized GT game to the temperature data collected by a WSN from an indoor surface. Our main goal is to analyze the temperature data provided by each sensor and to verify the quality of this data assuming an error of 1%. Our game, was applied to a matrix containing the temperatures collected in a time $t = t_0$ by the WSN. However, here, we only focus on a time t_0 (ie, we considered a static system). One of the main benefits of our game is that it is distributed and self-organized, which is a great advantage when dealing with data obtained by a WSN.

In Figure 5, the initial temperature is shown in the first image, and in the rest of the images, the iterations of the game until the GE are reached. In the successive images, the temperature clusters are being formed, this can be observed by the changes in the color gradient. It can also be seen that some areas with inaccurate temperatures are smoothly self-corrected on the basis of the temperatures in their environment. This is the intended process, since the game is executing its iterations depending on the environment surrounding the sensor. This makes sense because the temperature of the sensor will be similar to the average temperature of the environment in which it is located. In addition, we show the evolution of the temperature on the surface. To this end, we have represented the temperature on the z -axis to facilitate visualization in the form of a surface. The first image shows the temperature measured by the sensors. The game is applied iteratively to every image until it reaches the GE. Notice that the proposed game transforms the temperature data according to its environment. If we consider this as a knowledge data discovery process, the temperature collected by the sensors would be the target data, the game performs the preprocessing and data transformation to the target data simultaneously. Resulting in some transformed data, which, when the game reaches the NE are the surface temperatures in the GE. In addition, since the game is self-organized and distributed, it can be applied to the data in the ETL (extract, transform, and load) process without having to go through a central node first.

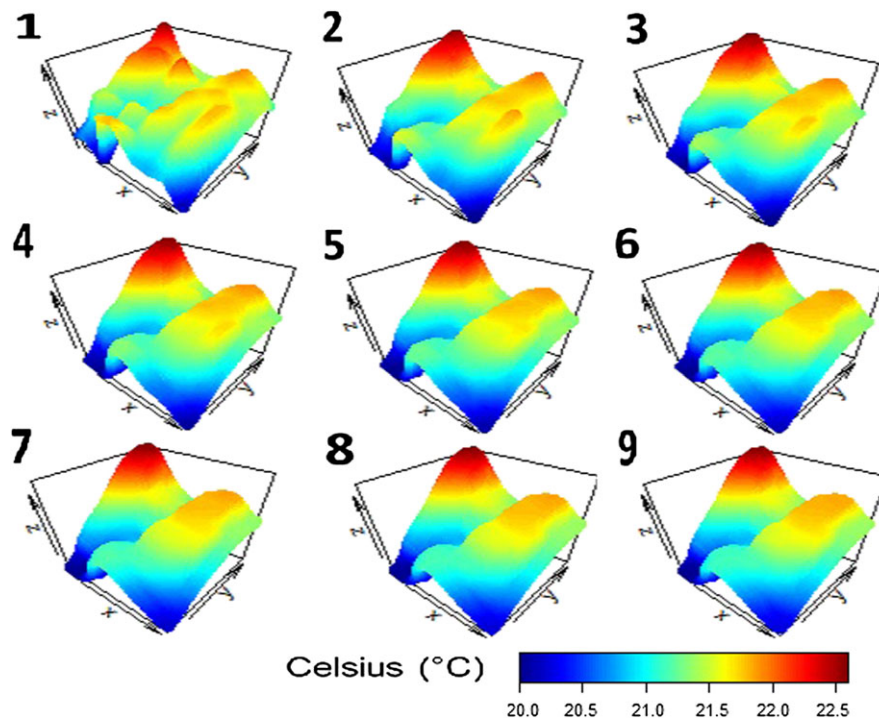


FIGURE 5 Evolution of surface temperatures over the different steps in the game until reaching game equilibrium [Colour figure can be viewed at wileyonlinelibrary.com]

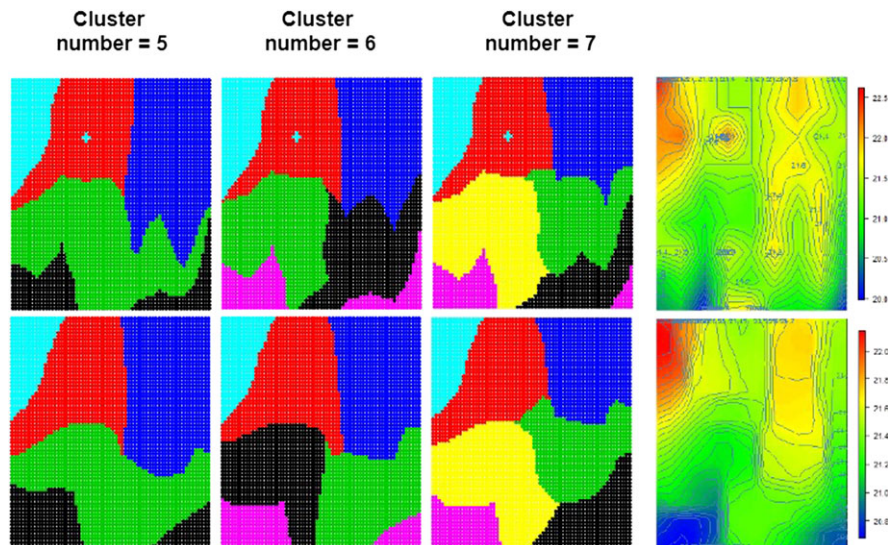


FIGURE 6 Data before applying the algorithm (top row) and after applying the algorithm (bottom row). K-means is used to validate the effectiveness of the proposed algorithm for self-correcting the data. In the first column, k-means with 5 clusters is used, and in the second and third columns, k-means with 6 and 7 clusters is used, respectively [Colour figure can be viewed at wileyonlinelibrary.com]

Once the GE is reached in the game, it can be seen that temperature clusters are formed. In Figure 6, we compare our method with K-means. The clusters can be observed, in the top row with the initial temperatures, and in the bottom row, the temperatures in the GE of the game. The method is more robust and adaptable than some unsupervised classification methods, such as K-means. One of the main advantages of the developed algorithm over K-means is that, before searching, the clusters in the data K-means needs to have the number of clusters. On the other hand, our algorithm can find the number of clusters by itself. This is a remarkable advantage in the exploratory analysis of data, since in an unsupervised way, it will give us very valuable information about the zones with similar temperatures. This data can be used to find the number of optimal centroids for the unsupervised classifiers (eg, K-means).

In Figure 6, we show an execution of K-means for the initial temperatures (first row) and for the temperatures changed by the game into equilibrium (second row). The first three images in each row are the execution of k-means on the temperatures, varying the number of centroids between 5, 6, and 7. A representation of a heat map with the isothermal lines is added to each of the rows. Once the GE has been reached in the game, the isotherms show that the number of suitable clusters is 6. In fact, looking at the execution of k-means for 6 centroids, it was found that the formation of clusters done by K-means is very similar to that on the isotherm map. Another result is that anomalous temperatures are self-corrected (on the basis of their environment) when the game is executed. We can notice in the first image in Figure 5 that some temperature values get smoothed as the game iterations increase (both maximum and minimum). In this way, similar temperatures are achieved, which is correct, since the appearance of critical points in the data that differ significantly from their environment suggests that they may not be reliable. In Figure 7, it is shown how the game self-corrects these points.

In Figure 8, a panel of two rows of images are presented. In the upper row are the images of different clustering methods applied to the temperatures collected by the WSN. In the lower row, the same clustering methods are applied but the data have already been self-corrected by the algorithm proposed in this paper. In Figure 8, different clustering methods have been used to see how the proposed algorithm behaves compared to different clustering methods. In order, the clustering methods used for this comparison are as follows: (1) K-means, (2) Forgy clustering method, (3) MacQueen clustering method, and (4) Hartigan-Wong clustering method. In the application of these clustering methods, we have used the optimal number of clusters for the data we have used in the experiment, the optimal number of clusters is 6. In the four columns, you can see how the proposed algorithm autocorrects the data without making substantial changes in the clusters that make up each of the chosen methods. However, it can be noted that the proposed algorithm autocorrects the inaccurate data and validates the data for later use. One of the novelties of our proposed algorithm is that it autocorrects inaccurate data and minimally modifies clusters. In fact, in Figure 8, this novelty has been validated with four different cluster methods.

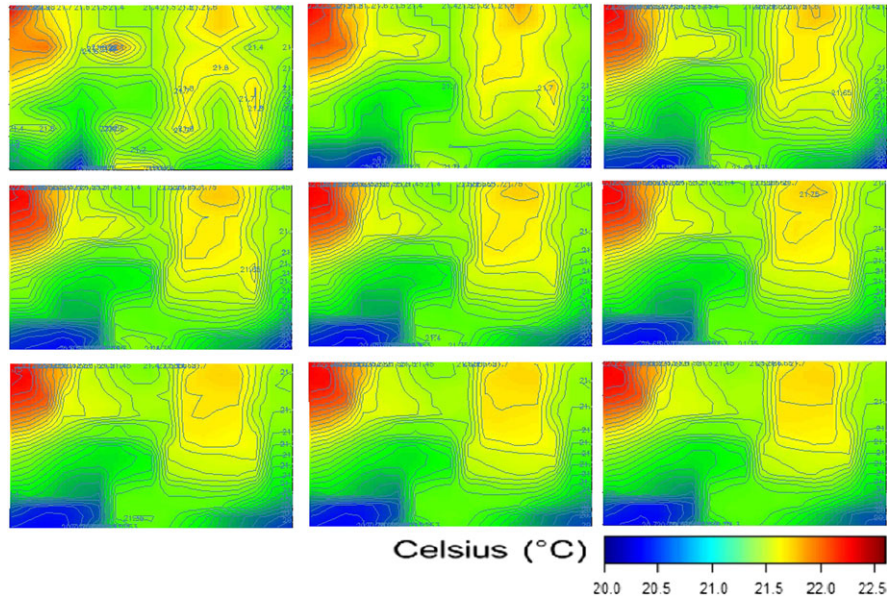


FIGURE 7 Evolution of surface temperatures in the different steps of the game until reaching game equilibrium in a map of isothermal lines. The images are ordered in rows from left to right. The first image shows the initial temperatures, and in the remaining images, we see each of the game iterations until game equilibrium is reached [Colour figure can be viewed at wileyonlinelibrary.com]

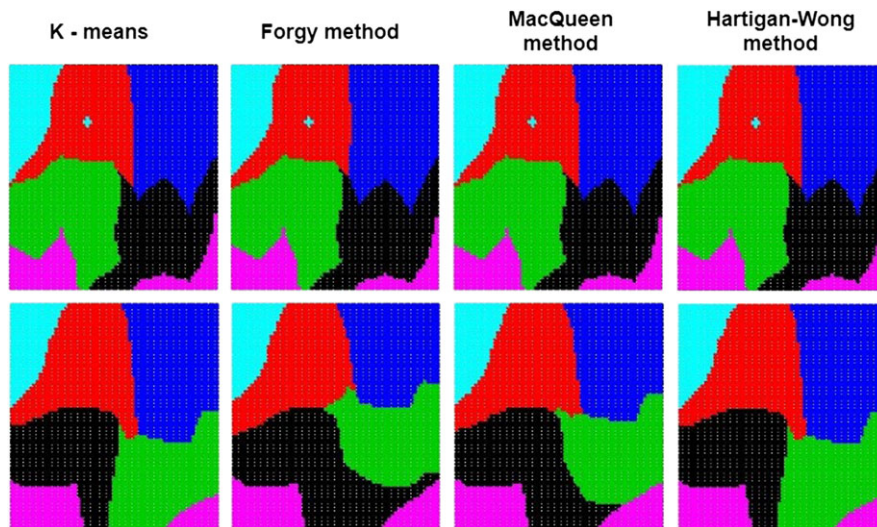


FIGURE 8 Data before applying the algorithm (top row) and after applying the algorithm (bottom row). Four clustering methods are used to validate that the proposed algorithm self-corrects data. In addition, the Figure shows how the proposed algorithm interacts with different clustering methods [Colour figure can be viewed at wileyonlinelibrary.com]

Assume $\epsilon = 0.01$ for the defined fuzzy convergence criterion, in Figure 9, we show the evolution of the elements that converge in each of the stages of the game, starting from $n \geq 2$. With the convergence speed equal to or greater than the order of $O(x^6)$, we can state that, in a few iterations of the game, the fuzzy convergence criterion is reached. In our case, there are nine iterations until convergence.

5 | INDUSTRIAL APPLICATIONS

The self-correction of the temperature values collected by the WSN is a notable outcome of our research. This self-correction is performed on the basis of the average temperature in the surrounding environment. WSNs will provide a temperature matrix for a time $t = t_0$. The advantage of self-correction is that the data provided by the WSN will be

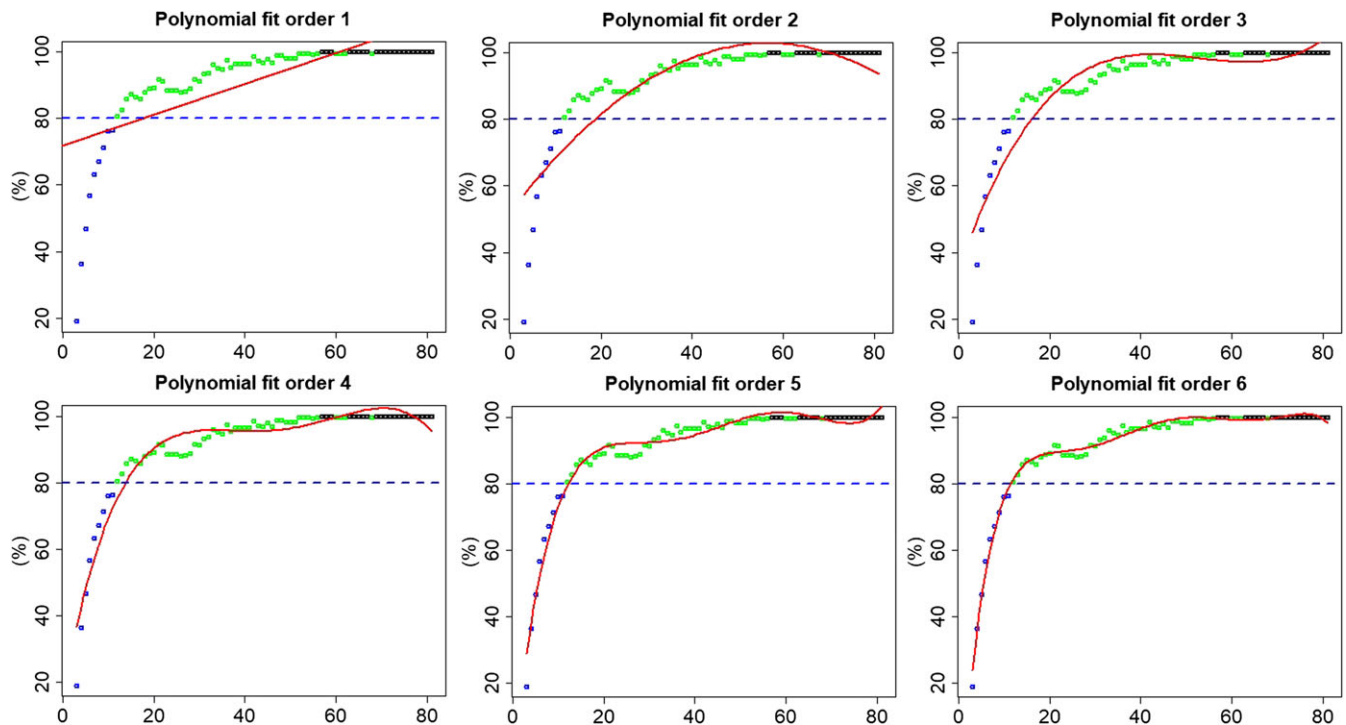


FIGURE 9 Panel with a linear, quadratic and cubic adjustment of percentage of temperature matrix elements convergent with the evolution of the iterations of the game. A blue dashed line has been placed on each of these images marking the defined fuzzy convergence zone. Points below the convergence zone are also blue, the points above the area of convergence, and black when the convergence is 100%. It can be noticed that the game converges to nine iterations for the fixed ϵ . In addition, with the help of the regression, lines highlight the degree of convergence is of the order of $O(x^6)$ [Colour figure can be viewed at wileyonlinelibrary.com]

reliable; there will be no anomalous data, such as temperature peaks (maximum or minimum), which differ significantly from the temperature in the neighborhood. This helps to eliminate (to a large extent) the possible thermal noise introduced by the sensors that are functioning incorrectly or inaccurately, facilitating the task of monitoring the temperature of a surface.

The proposed algorithm provides an excellent transformation in the ETL process, in the data flow itself, we can apply our work as a transformation included in the ETL process for the generation of new temperature data, that is already self-corrected and ready to be used. A large part of the relative thermal noise brought by the data arriving from the sensor, is eliminated (noise is created when sensors are defective or not accurate). Figure 10 shows the number of sensors (in percentages) with thermal noise for each iteration of the game. It can be noticed that by switching the precision of the sensors from 0.01° to 0.3° , the obtained results are quite different. However, if changes to 0.01° , 15% of the sensors had thermal noise, and in a few (<10) iterations, the noise was reduced to less than 5%. When the allowed relative error was increased, the percentage of sensors having some thermal noise also increased. For example, with 0.3° of relative error, 95% of the sensors had thermal noise, and as the iterations increase, it was reduced to less than 75%. However, at some point, the noise began to plateau. These sensors will continue to have some noise for the chosen error (Table 2).

There are two other very useful applications of our work. (1) The identification of sensors that provide false data and the calibration of new sensors by injecting them into the WSN. (2) The intelligent false data detection in a WSN is an important issue, since it allows for the predictive maintenance of the WSN, as well as a high level of data quality. In addition, predictive maintenance allows to maintain the WSN functioning properly. Since the malfunctioning sensors are identified, the maintenance cost reduces significantly as the technician can focus on defective sensors only. On the other hand, the level of confidence that the game provides to the data is very high, since it eliminates the defective and/or inaccurate sensors together with a large part of the noise. This provides us with high quality and fairly accurate data.

Another application of our work is the location of zones with similar temperature (clusters). This is very useful in the field of intelligent monitoring of buildings since hot and cold bulbs can be detected. Furthermore, heat leak can be identified with the proposed game and predictive measures can be taken (such as reinforcing the materials with thermal insulators or modifying the heating in the building in order to adjust the temperature in areas with heat leaks). Another

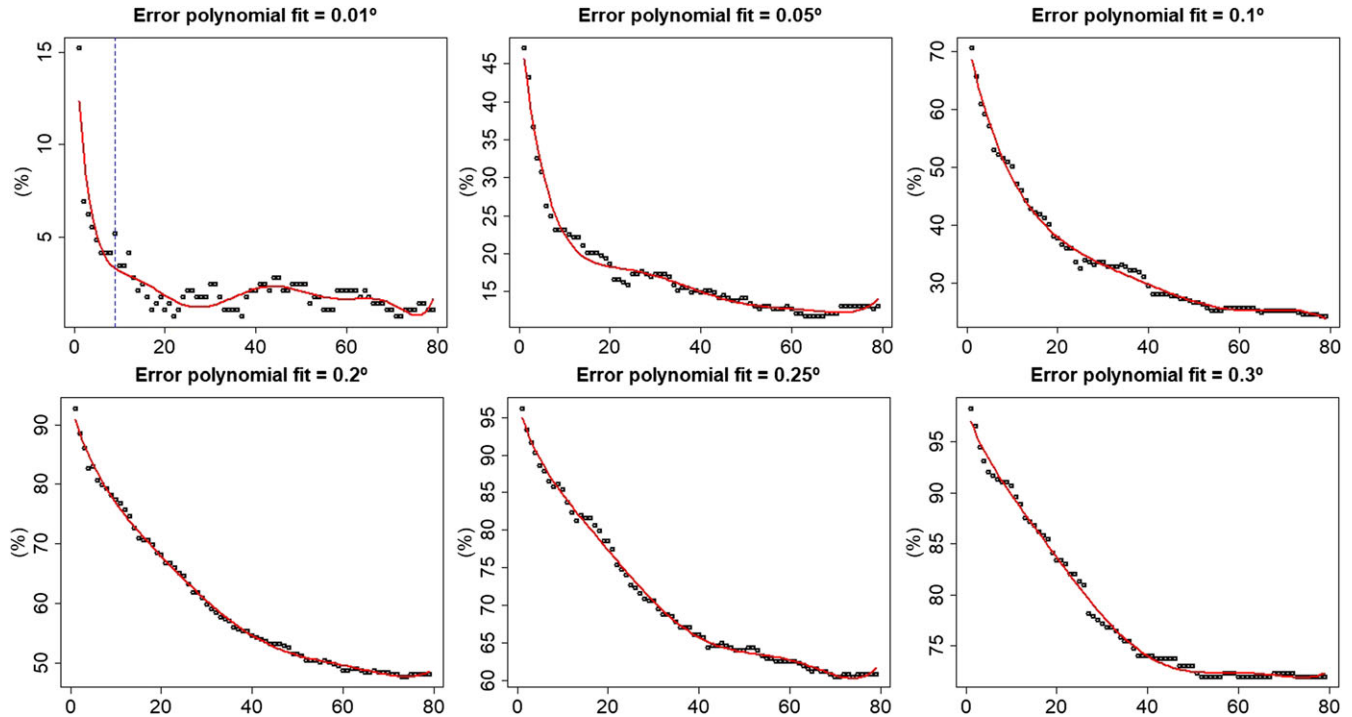


FIGURE 10 Panel with the decrease of thermal noise in the evolution of the game with different margins of confidence from 0.01°C to 0.3°C . In the image panel, the percentage noise in the temperature matrix is presented versus the number of iterations. In each of them, the permitted margin of error for the temperature collected by the sensors is varied. It can be seen that, as the permitted margin of error increases, the thermal noise in the temperature matrix also increases [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 2 Table with the different permitted errors and percentage noise before and after the application of the game

Allowed Error ($^{\circ}\text{C}$)	Sensors With Noise at the Beginning (%)	Sensors With Noise at the End (%)
0.01	15.23%	1.04%
0.05	47.06%	13.25%
0.1	70.59%	24.22%
0.2	92.74%	48.10%
0.25	96.20%	60.90%
0.3	98.27%	71.98%

possible use of the game would be smart agriculture. For example, a crop field with a WSN monitoring temperature, artificial intelligence techniques can be applied to find very hot and cool areas, in order to decide if they need watering or some other type of treatment. This type of predictive maintenance can be very attractive for private industry.

After discussing the results and possible applications of our work, we stress that the major novelty of our research in the field of WSN is that we do not focus on the WSN itself but on the treatment of the data collected by the WSN. As described in the results section, the proposed game is distributed and self-organized. This allows to treat the data obtained from the WSN in order to make them reliable. Identified clusters can discover defective or inaccurate sensors, which allows for predictive maintenance of the WSN.

6 | CONCLUSIONS AND FUTURE WORK

This work proposes a distributed and self-organized cooperative algorithm using GT. The algorithm has been applied to the data collected by a WSN from an indoor surface.

The main goal of the game is to make WSN monitoring more robust through consensus temperature monitoring. Furthermore, the presented work achieves and ensures data quality, false data detection (ie, inaccurate sensors), and

temperature data optimization to improve energy efficiency in cooperative WSNs. The most significant results obtained in this work are listed as follows.

The game is distributed; a central node is not required to run. The game is also self-organized since WSN nodes interact with each other to generate self-corrected data in relation of their surrounding environment.

Unlike the classic unsupervised methods, our work does not require the number of clusters beforehand. Once the game reaches the GE, the resulting number of clusters can be used as input for the unsupervised classification analysis.

On the other hand, anomalous temperature values are corrected according to their surroundings, without modifying the temperature clusters. That is, the game self-corrects without changing the cluster structure that is obtained directly by the WSN in the temperature matrix. Finally, with the established diffuse convergence criteria, the game converges in very few iterations (9) to the GE.

Here, we summarize the assumptions that have been made in the case study in order to reduce complexity and avoid confusion. The surface on which the sensors were placed was regular and known. This is a very significant limitation, since the target surface will not always be a regular surface. Moreover, we have assumed that the data are collected in a time $t = t_0$ and that the game is static. This limits our model since it does not evolve over time. Finally, only a Von Neumann neighborhood was used with $r = 1$, so the environment is fairly small and could slightly affect the rate of convergence of the game.

Despite these limitations, our work provides a novel, distributed, and self-organized algorithm, which allows to self-correct temperature data collected by the sensors according to their surrounding temperatures. We also address some interesting results and some quite promising industrial applications. Future lines of work will be focused on extending the game to larger topological manifolds and we will study these manifolds dynamically.

ACKNOWLEDGEMENT

This project has been funded by the European Regional Development Fund (ERDF) within the framework of the Interreg program V-A Spain-Portugal 2014-2020 (PocTep) grant 0123 IOTEC_3_E (project IOTEC).

ORCID

Roberto Casado-Vara  <http://orcid.org/0000-0003-0198-696X>

REFERENCES

1. He J. *Design and Implementation of a Wireless Sensor Network Testbed* [master's thesis]. Delft, The Netherlands: Delft University of Technology; 2009.
2. Siddia A, Hashem IAT, Yaqoob I, et al. A survey of big data management: taxonomy and state-of-the-art. *J Netw Comput Appl*. 2016;71:151-166.
3. Gani A, Siddiq A, Shamshirband S, Hanum F. A survey on indexing techniques for big data: taxonomy and performance evaluation. *Knowl Inf Syst*. 2016;46(2):241-284.
4. Patel SV, Pandey K. Design of SOA based framework for collaborative cloud computing in wireless sensor networks. *Int J Grid High Perform Comput*. 2010;2(3):60-73. <https://doi.org/10.4018/jghpc.2010070105>
5. Gungor VC, Hancke GP. Industrial wireless sensor networks: challenges, design principles, and technical approaches. *IEEE Trans Ind Electron*. 2009;56(10):4258-4265. <https://doi.org/10.1109/TIE.2009.2015754>
6. Tapia DI, Alonso RS, De la Prieta F, et al. SYLPH: an ambient intelligence based platform for integrating heterogeneous wireless sensor networks. Paper presented at: International Conference on Fuzzy Systems; 2010; Barcelona, Spain.
7. Rodríguez S, De Paz JF, Villarrubia G, Zato C, Bajo J, Corchado JM. Multi-agent information fusion system to manage data from a WSN in a residential home. *Inf Fusion*. 2015;23:43-57.
8. de Paz JF, Tapia DI, Alonso RS, Pinzón CI, Bajo J, Corchado JM. Mitigation of the ground reflection effect in real-time locating systems based on wireless sensor networks by using artificial neural networks. *Knowl Inf Syst*. 2013;34(1):193-217.
9. Bajo J, De Paz JF, Villarrubia G, Corchado JM. Self-organizing architecture for information fusion in distributed sensor networks. *Int J Distrib Sens Netw*. 2015;11:1-13.
10. Shamshirband S, Anuar NB, Kiah MLM, Patel A. An appraisal and design of a multi-agent system based cooperative wireless intrusion detection computational intelligence technique. *Eng Appl Artif Intell*. 2013;26(9):2105-2127. <https://doi.org/10.1016/j.engappai.2013.04.010>
11. Shamshirband S, Patel A, Anuar NB, Kiah MLM, Abraham A. Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks. *Eng Appl Artif Intell*. 2014;32:228-241. <https://doi.org/10.1016/j.engappai.2014.02.001>

12. Tapia DI, Abraham A, Corchado JM, Alonso RS. Agents and ambient intelligence: case studies. *J Ambient Intell Humaniz Comput*. 2010;1(2):85-93.
13. Tapia DI, Alonso RS, Paz JFD, Corchado JM. Introducing a distributed architecture for heterogeneous wireless sensor networks. Paper presented at: 10th International Work-Conference on Artificial Neural Networks; 2009; Salamanca, Spain.
14. Alonso RS, Tapia DI, Bajo J, García Ó, De Paz JF, Corchado JM. Implementing a hardware-embedded reactive agents platform based on a service-oriented architecture over heterogeneous wireless sensor networks. *Ad Hoc Netw*. 2013;11(1):151-166. <https://doi.org/10.1016/j.adhoc.2012.04.013>
15. Steinberg AN, Bowman CL, White FE. Revisions to the JDL data fusion model. Paper presented at: AAIA Missile Sciences Conference; 1998; Monterey, CA.
16. Saghezchi FB, Radwan A, Rodríguez J. Energy-aware relay selection in cooperative wireless networks: an assignment game approach. *Ad Hoc Netw*. 2017;56:96-108. <https://doi.org/10.1016/j.adhoc.2016.12.001>
17. Zhou J, Mu C. Density domination of QoS Control with localized information in wireless sensor networks. Paper presented at: 2006 6th International Conference on ITS Telecommunications; 2006; Chengdu, China.
18. Corchado JM, Bajo J, Tapia DI, Abraham A. Using heterogeneous wireless sensor networks in a telemonitoring system for healthcare. *IEEE Trans Inf Technol Biomed*. 2010;14(2):234-40. <http://www.ncbi.nlm.nih.gov/pubmed/19858034>
19. Han Z, Niyato D, Saad W, Başar T, Hjørungnes A. *Game Theory in Wireless and Communication Networks: Theory, Models and Applications*. Cambridge, UK: Cambridge University Press; 2011.
20. Moura JA, Hutchison D. Survey of game theory and future trends with application to emerging wireless data communication networks. *IEEE Commun Surv Tutor*. 2017:1-50. <http://arxiv.org/abs/1704.00323>
21. Shi H-Y, Wang W-L, Kwok N-M, Chen S-Y. Game theory for wireless sensor networks: a survey. *Sensors*. 2012;12:9055-9097.
22. Kottapalli VA, Kiremidjian AS, Lynch JP, et al. Two-tiered wireless sensor network architecture for structural health monitoring. In: Proceedings Volume 5057, Smart Structures and Materials 2003: Smart Systems and Nondestructive Evaluation for Civil Infrastructures; 2003; San Diego, CA.
23. Lorincz K, Malan DJ, Fulford-Jones TRF, et al. Sensor networks for emergency response: challenges and opportunities. *IEEE Pervasive Comput*. 2004;3(4):16-23.
24. Su I-J, Tsai C-C, Sung W-T. Area temperature system monitoring and computing based on adaptive fuzzy logic in wireless sensor networks. *Appl Soft Comput*. 2012;12(5):1532-1541. <http://linkinghub.elsevier.com/retrieve/pii/S1568494612000026>
25. Mishra R, Jha V, Tripathi RK, Sharma AK. Energy efficient approach in wireless sensor networks using game theoretic approach and ant colony optimization. *Wirel Pers Commun*. 2017;95(3):3333-3355.
26. Afsar M. Energy-efficient coalition formation in sensor networks: a game-theoretic approach. 2015. <http://arxiv.org/abs/1512.08019>
27. Schmidt C, ed. *Game Theory and Economic Analysis*. London, UK: Routledge; 2002.
28. Xu Z, Yin Y, Chen X, Wang J. A game-theory based clustering approach for wireless sensor networks. Paper presented at: The 2nd International Conference on Next Generation Computer and Information Technology; 2013; Qingdao, China.

How to cite this article: Casado-Vara R, Prieto-Castrillo F, Corchado JM. A game theory approach for cooperative control to improve data quality and false data detection in WSN. *Int J Robust Nonlinear Control*. 2018;1-16. <https://doi.org/10.1002/rnc.4306>