Int. J. Fuzzy Syst. (2022) 24(7):3327–3347
https://doi.org/10.1007/s40815-022-01349-1

An Enhanced Anomaly Forecasting in Distributed Wireless Sensor Network Using Fuzzy Model

R. Yasir Abdullah[1,2] · A. Mary Posonia[3] · Usman Barakkath Nisha[4]

Received: 13 January 2022 / Revised: 21 April 2022 / Accepted: 6 June 2022 / Published online: 17 August 2022

Abstract Wireless sensor networks (WSNs) grieve from a wide range of limitations and aberrations which hinder their smooth functioning. Detecting anomaly in WSNs is a decisive research area, which emphasizes making sensor nodes to be reliable in handling data. Owing to energy restrictions and less computation capability of sensor networks, anomaly detection ought to concentrate on the fundamental limitations of sensor networks. Anomaly detection and dipping noisy data transmission are essential to recover the network life span of sensor networks by promising data integrity. Henceforth, academicians and researchers are frequently getting motivated in finding methods to improve the accuracy of data held by the sensor nodes. In such surroundings, the investigators focus on semi-supervised anomaly detection which uses real data to distinguish incidences that are conflicting with the widely-held data. The proposed idea is acquainted with a fuzzy-based anomaly detection model for semi-supervised anomaly detection in an effort to recover accuracy. The proposed model is compared with other leading existing procedures and methodology over robustness and other substantial metrics. Comparatively, our proposed prototype achieves a high-performance score with 99.70% accuracy, 99.14% precision, 99.27% detection rate, 98.56% specificity, 98.78% *F*1 score, and 0.8 correlation coefficient. It is detected that our proposed model diminishes false alarms up to 1.20% through detection which is a key concern in WSNs.

Keywords Anomaly · Cluster · Outlier · Correlation · Self-organization map · Mamdani fuzzy model · Wireless sensor networks

1 Introduction

Wireless sensor networks (WSNs) involve numerous sensor nodes arrayed in the region of interest for application-specific areas. WSNs are diverse networks having their standing in areas such as logistics, healthcare, weather forecasting, military applications, robotics, security, and surveillance. These nodes are usually of small size with computational energy, communication ports, and sensing capabilities. Sensor nodes communicate via short-range wireless signals and team up among themselves to accomplish mutual tasks.

Sensor nodes have inadequate bandwidth, power, meager memory capacity, limited processing resources, and measurable lifetime. The core task of the sensor node is to sense the attributes like light, heat, and temperature [1]. The analyzed report will then be forwarded to the sink or host controller in the format specified by the administrator. Consumption of energy by the WSNs to compute the data

✉ R. Yasir Abdullah
yasirsince1984@gmail.com

A. Mary Posonia
maryposonia.cse@sathyabama.ac.in

Usman Barakkath Nisha
ubnisha@gmail.com

[1] Research Scholar, Sathyabama Institute of Science & Technology, Chennai, India

[2] Department of CSBS, Sri Krishna College of Engineering & Technology, Coimbatore, India

[3] Department of Computer Science and Engineering, Sathyabama Institute of Science & Technology, Chennai, India

[4] Department of Information Technology, Sri Krishna College of Engineering & Technology, Coimbatore, India

is less compared to the transmission of data. Energy utilization could be minimized using the collected data and aggregation will be performed using functions like sum(), avg(), etc. Aggregation of data aka data aggregation is the procedure used to collect and combine the necessary required information related to its specific applications. The efficiency of the data aggregation techniques could be improved by the success rate of the communication between nodes. Data aggregation is an effective tool accepted by maximum researchers to reduce the energy consumed by the individual nodes thereby saving resources that are available in a limited manner. This technique also has the accuracy in enhancing the network lifetime and energy efficiency.

Data aggregation is one of the ways for conserving energy in dense WSNs by minimizing redundant data transfer. In the sensor deployment area, aggregators remove the unnecessary data and send only fused information to the base station. Hence, it removes unwanted data transmission of incorrect data in the network and it preserves the energy which will be utilized for the same. Small sensor nodes are integrated to form any sort of WSN. Within the specified range, communication happens between the sensor nodes. As far as the power of the sensor node is concerned, it is crucial to reduce the energy consumption that will be consumed by the sensor nodes for specific applications which effectively will increase the lifetime of WSNs [2, 3]. Data redundancy reduction will help in achieving the above-said lifetime of the nodes since the maximum power is consumed by processing the superfluous data. Therefore, the best solution to improve the lifetime of the network as on date is to remove the redundancy occurring in WSNs. The technique named data aggregation is widely adopted by the maximum number of applications that result in gathering and aggregation of data while considering the energy efficiency and enhancing the lifetime of the nodes.

Albeit removing the redundancy results in the increase of the lifetime of the nodes, utmost care is needed in fetching accurate results in WSNs. Hence, both reliability and conservation of energy should be treated with equal weightage. This paper mainly focuses on solving the issues like redundancy, accuracy, communication, and computational complexity that occur in the process of aggregation of data. The information collected from the sensor nodes will be fused using the wireless sensor nodes that would be deployed in certain applications. Generally, sensors are not dependent on variable quantities which exist in the raw dataset. Sensor nodes deployed close to one another generally have the same attributes [4]. Therefore, it is crucial to cluster the sensor data and the data that were aggregated using different algorithms. Monitoring the observation retrieved from the above-said techniques, two phases are essential for discovering and substituting inconsistent data to recover data accuracy and the data quality in the sensor network [5, 6].

The first phase of the proposed architecture starts with candidate model construction, where a customized self-organizing network partition is employed and it ends by grouping similar and dissimilar datasets separately. In the second phase, an adaptive fuzzy inference system (FIS) analyses the candidate input space from the data cluster and applies fuzzy logic rules to detect the anomaly, and also appropriate data replacement is done to avoid data loss and improve data quality in the sensor network.

The proposed methodology will be capable of achieving three main objectives:

(1) efficiently clustering the sensor networks based on data using a customized self-organizing map (CSOM),
(2) extending network lifetime by reducing redundant data transmission with localized data aggregation, and
(3) increasing aggregated data accuracy by using an efficient fuzzy-based classification of normal and abnormal sensory values with data imputation.

The remaining part of this paper is organized as follows. In Sect. 2, we elucidate the existing significant ideas related to our anticipated method. Section 3 elucidates the system model and problem assertion. In Sect. 4, the proposed anomaly detection with the fuzzy inference model is explained. In Sect. 5, we experimented with the related activities to compare the performance. In Sect. 6, we confer the conclusion.

2 Related Work

To improve the dependability of WSN data, it is critical to detect outliers both spontaneously and correctly. In general, outlier identification methods are of the following: statistical-based methods, closest neighbor-based methods, clustering-based methods, classification-based methods, and spectral decomposition-based methods [7, 8]. Statistical approaches capture the data distribution and assess how the data instance matches the model. If the model's prediction likelihood for a data sample is abnormally low, the data instance is classified as an outlier. Moreover, because of the large volume of contacts among neighbors, closest neighbor-based approaches require unnecessary resources at each sensor to detect outliers, potentially reducing the lifetime of WSNs [28]. Clustering-based methods aggregate data examples with comparable behaviors into the very same cluster and define an outlier as an instance that cannot be categorized into any cluster. Because cluster-based approaches can only do clustering after collecting the entire dataset. Classification-based approaches use previous data to train the classification model, which is then used to assess new collections online. Outlier identification using spectral-based approaches focuses on high-dimensionality data processing, which may add to the system's computing complexity.

Feng et al. [9] present a distributed outlier detection strategy constructed on trustworthiness response. The method is broken down into three stages: initial credibility of sensor nodes will be assessed, final credibility using credibility feedback, and the Bayesian theorem will also be assessed. As far as the message complexity study is concerned, this model eats a lot of energy. Abid et al. introduce a real-time outlier identification approach for WSNs [10]. The method has the advantage of requiring no previous information on the data distribution. The quality of the generated clusters, on the other hand, has a significant impact on outlier identification performance. The Bayesian classifier is utilized in the first layer at each sensor node, and the choices of different nodes are then integrated into the second layer to identify if an outlier is present in the various sensors [11]. Its goal is to provide an accurate outlier classification system that is both computationally and communicationally simple.

A hierarchical anomaly detection methodology is suggested in [12] for distributed large-scale sensor networks. To cope with anomalies in data collected by the defective sensor, it uses principal component analysis (PCA). The approach creates a model for sensor information, which is then used to find outliers in a sensor node by looking at neighboring sensor nodes. The approach is computationally costly because it necessitates the selection of an appropriate model. A WSN outlier identification approach based on *k*-nearest neighbor (NN) is proposed in [13]. The term "hyper-grid" comes to mind when thinking about this method. The computational cost is significantly reduced by normalizing abnormality from a hypersphere detection area to a hypercube detection area. However, it has to be tested on a larger number of datasets with data stream distributions. Focusing on a fuzzy clustering technique that is almost hybrid, the authors deliver a regimented fuzzy clustering style for Takagi–Sugeno (TS) fuzzy modeling in [14]. For developing an effective T–S fuzzy model from sample data, the process goes through many phases. In [15], authors projected an active associated fuzzy system with the temporal, attribute, and spatial correlations for detecting multivariate outliers in sensor networks. They have employed outlier detection during data aggregation in the cluster head.

For distributed anomaly identification in different actual datasets, Heshan Kumaragea et al. [16] advocated fuzzy data modeling. When a high number of nodes are used, scalability and sensitivity are reduced. Sensor networks that work on the principle of routing algorithm based on subtractive clustering are used in [17] to pick and create clusters in areas with a high node density. Their findings show that an appropriate cluster configuration, a prolonged life for the initial node, and an unfluctuating continued lifetime for the system are necessary to stabilize the energy utilization of every node in the network. In this case, the data density is estimated using Euclidean distance, which does not result in an ideal cluster head. The technique in [18] has a high false alarm rate and a low $F1$ score. It used subtractive clustering in conjunction with a Sugeno FIS to detect anomalies in a sensor network. The relative correlation clustering technique is used to identify anomalies. This model combines the comparative cascaded correlation approach with clustering and re-clustering [19]. Vesanto and Alhoniemi also proposed and examined various approaches for clustering the primitive. These approaches simply treat each model as a single feature vector, performing clustering on small-scale variables derived by SOM [20]. The U-matrix can also be used to cluster the primitive. It has been verified to outperform conventional clustering approaches while requiring less computing time [21].

Erfani et al. [22] suggested a high-dimensional anomaly identification method to lower the data's dimensionality for finding outliers. However, this fits best when the reduced outliers are evenly scattered among the regular occurrences; otherwise, accuracy degrades. In [23], An adaptive mountain clustering technique with data dimensionality measures is deployed for finding outlier data that prevents undesired transmission of data to the base station while improving data accuracy. They concluded that on average, the proposed system has a precision of 98.96%. Also, while increasing the sample size from 5000 to 20,000, accuracy remains consistent.

The approach reliably identifies anomalies while producing a low false positive. The author established an adaptive neural swarm practice for anomaly detection in [24]. The method incorporates agent-based swarm intelligence in a decentralized collaborative method. The swarm agents create modular neural network models independently, and reinforcement learning is used in the training phase on the way to support a learning environment that is supposed to be unsupervised. The authors of [25] proposed a least squares support vector machine that is sliding window-based for detecting outliers in a dynamic environment. This method employs a replicating kernel Hilbert space kernel which contains a radial basis function.

The authors of [26] proposed an anomaly detection algorithm using DBSCAN and SVM. A density-based spatial clustering algorithm was used for detecting anomalies that are marked in low-density regions, and then the model is trained to classify normal and anomaly by using an SVM classifier. The authors compared their performance in each cluster. The overall accuracy was 95.5% and they failed to focus on running time fluctuation while increasing cluster size. The authors focused on clustering technology-based ODCASC algorithm to forecast anomalous node data while considering the spatial correlation between node data for decision making. This ODCASC algorithm focused only on spatial correlation. It fails to focus on attribute relationships while considering multivariate data analysis. Hence, accuracy is less when compared with our proposed method. Moreover, computational complexity is also not considered in performance while considering a large dataset [27].

Each above-mentioned research work has been applied to anomaly detection in WSN but these models are not able to produce results with multivariate attribute correlation analysis. Moreover, these systems lack in producing anomaly detection with various performance indicators. None of them explained an efficient prediction of data classification by applying fuzzy classification models and computational and memory complexity are also not addressed properly. The anomaly detection in the cluster head to make the proposed system acts proactively before the aggregated data transmission process in the network. The proposed model is designed to discover anomalies to address some of the research difficulties in terms of various performance indicators. Our proposed approach produces fewer false alarms without compromising the computing time of the model, which reflects the communication complexity in the cluster head. Furthermore, the suggested model is found to be appropriate for high-dimensional huge datasets with good accuracy and $F1$ score with self-organization map in fuzzy decisions.

3 Network Model and Problem Declaration

The forthcoming segment addresses both the network model and the proposed problem. The main purpose of this paper is to propose a forecasting model based on anomalies that transmit the data without any loss or interference. Numerous research areas are evolving exponentially that represents models for different sensor networks. By applying the hypothesis, we contemplate a sensor network environment in which every node intelligences its neighbors periodically by forwarding the data to a base station located at different places. Sensor nodes remain diverse and concentrate more on their energy parameters. Sensor nodes understand better about their location. Sensor nodes are always volatile, whereas the base station is not volatile and is fixed. Sensor nodes act as watchdog in controlling communication dynamically using the radio waves and their boundaries are

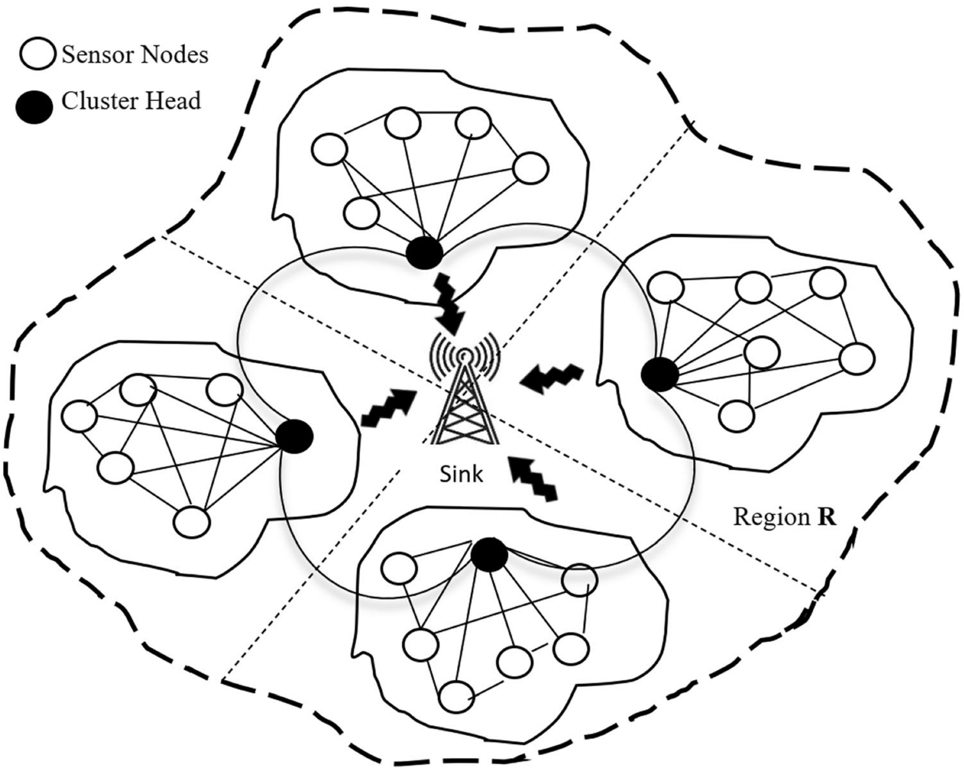

Fig. 1 Network topology and system model

limited. Aggregation or fusing the data is typically used to diminish the number of messages in the network. We believe that combining α identical packets of size k yields a single packet of size n as an alternative to size αn.

Consider S as the set of sensor nodes deployed randomly at the required region R. $\|S\|$ is supposed to be the entire amount of sensor nodes. Figure 1 depicts that a comprehensive connected network is created with probable numbers of distributed clusters. Every cluster would be guided by the cluster head by eventually performing data aggregation. Each cluster head has an anomaly forecasting model that is used to remove anomalies before transmitting merged data to the base station. Because a cluster is made up of sensor nodes that can sense their surroundings, data accuracy is ensured in each cluster head by incorporating the proposed anomaly forecasting model, and accuracy at the sink node is also verified at the end of the user query.

To construct the framework in this paper, the following assumptions are taken into account.

- *Nonhomogenous sensor network* sensor network applications for wireless mesh network infrastructure are assumed to consider sensor nodes with unequal initial energy as well as dissimilar capabilities. The proposed method is also suitable for homogeneous networks by deploying sensors with high energy for performing sensing, computation, sending, and receiving processes. In a homogeneous network, all sensor nodes are having equal initial energy and the aggregator performs the majority of the task. However, it drops its energy and leads to inconsistent computation.
- *Tracking and event sensor report* the nodes in this network adjust their sensors and transmitters regularly by sensing the environment and transmitting the data of interest in response to abrupt and extreme deviations in the value of an identified attribute.
- Closer sensors may receive more correlated data since nodes are positioned to sense overlain zones. Sensor networks require a way to accurately detect not each node's data but also aggregated data in localized aggregation, which involves sensors in the immediate area cooperating to cluster data and transfer aggregated data to a sink.

4 Proposed Methodology

The proposed methodology is comprised of two parts: candidate space partitioning and anomaly detection. In the first part, aiming at each sensor node, the original perceived data can be separated into multiple groups by analyzing its spatial correlation. By testing the conventional SOM, it is inferred that it groups the data points based on Euclidian distance and does not concentrate on analyzing the multivariate data correlation. Hence, conventional SOM considers the NN's distance without considering correlation metrics among input variables. To resolve this issue, Mahalanobis distance is put into practice with spatial correlation for grouping dissimilar regions for examining anomalies that are most deviated from its space in the outer layer [29].

The workflow of the anomaly forecasting process is shown in Fig. 2. This first level of defense identified large deviated data points by grouping them into clusters. Then the sensor node's data are labeled as most disbelieving

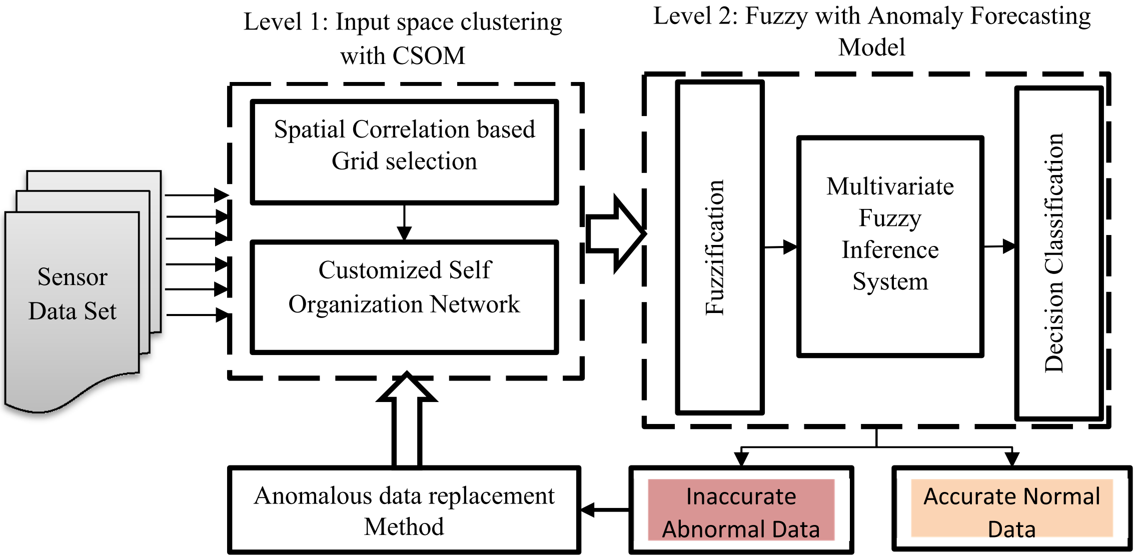
Level 1: Input space clustering with CSOM
Level 2: Fuzzy with Anomaly Forecasting Model
Sensor Data Set
Spatial Correlation based Grid selection
Customized Self Organization Network
Fuzzification
Multivariate Fuzzy Inference System
Decision Classification
Anomalous data replacement Method
Inaccurate Abnormal Data
Accurate Normal Data

Fig. 2 Workflow of anomaly forecasting Process

nodes and the remaining data points sensor nodes are labeled as believing nodes for data aggregation under the guidance of the cluster head. In the second part, each cluster data point is analyzed by a well-defined fuzzy anomaly detection model. Two groups of nodes are taken into consideration labeled as disbelieving and believing nodes. The uncertainty fuzzy inputs are considered as candidate input space for the anomaly detection model. They may counter noise and channel interference or sensor node fault so that some data are lost or damaged during delivery. In the third part, according to received data, we design an ensemble mechanism to impute the original perceived data for improving data accuracy.

4.1 Customized Self-organization Map

SOM is among the popular neural network methods for cluster analysis [31] for two reasons. First, SOM has the features of both auto-organized and structure-preserving topology. Closed data in the input data space will be anticipated onto closed prototype vectors on the grid next to the training phase so that two input vectors that were projected onto closed prototypes belong to the same cluster. The second states that SOM is very useful for dimensionality reduction and clustering high-dimensional data, and SOM has prominent visualization properties.

First, we will take a look at the conventional SOM. SOM is an unsupervised learning method that accepts high-dimensional inputs and produces lower-dimensional output by grouping comparable input data indices together based on their proximity [32]. The model self-organizes based on learning rules and interactions. The main functionality of conventional SOM can be described as follows:

a. Choose a collection of $\underset{\wedge}{N}$ nodes that calculate efficient separation features of random high-dimensional incoming inputs $S(t)$. Each node has a d dimensional weight vector.
b. Initialize the weight factor w=(w_{i1}, w_{i2},...,w_{id}), where i =1, 2,…,m, which has the same size as the input dataset.
c. Select the nodes with the largest output and set them as tentative winner nodes.
d. Selected nodes maintain proximity relationships with their neighbors when they are close to the input vector.

CSOM creates network clusters based on available data sent out by sensor nodes and transmits only aggregated data to sink nodes. CSOM is used for creating a candidate model structure that emphasis topological sensor node arrangements based on the relationship between candidate data and feature elements in the map. The Mahalanobis distance is a useful multiple variable distance metric for determining the distance between two points. It is a really useful statistic with a high degree of efficiency in multivariate anomaly identification. In this topological structure, candidate sensor data points similarities are analyzed and represented in the output map using the feature element characteristics.

The CSOM consists of $\underset{\wedge}{N}$ nodes located in the candidate feature surface. Nodes are connected with their neighbors according to topological arrangements. Topological network arrangements are created to connect $\underset{\wedge}{N}$ nodes with their neighbor nodes. Figure 3 shows the nodes at the specified range distances of the winner nodes' one-hop neighbor nodes with each input data. These winner nodes are labeled as near and far nodes for that specific input. The proposed CSOM can be described in the subsequent steps.

4.1.1 *Step 1: Stretch Out Input Dataset*

Candidate input patterns are the data points collected from the sensor nodes denoted as $S(t)$={S_i: i=1, 2, 3…,d}, where d is the scale indicating the number of the input vector, and the related weights between the input pair (i, j), the resultant layer can be denoted as $\acute{W}_j(S_i)$={$\acute{W}_{ji}$: j=1, 2, 3,…,n & i=1, 2, 3,…,d}, where n number of nodes associated weights are analyzed in the feature map. We have data point S_i from the sensor nodes candidate region mapping to points $M(S_i)$ in the resultant output region. Every data item $M(S_i)$ in the output region would be associated with a weight $\acute{W}(M_i)$ in the candidate region. An input pattern $S(t)$ is chosen informally and assigned concurrently to all nodes.

4.1.2 *Step 2: Weight Factor Initialization*

Initially, all the weight vectors $\acute{W}_j(S_i) \in \underset{\wedge}{N}$ nodes with a set of $n * d$ weight vectors are assigned. Then, the winning occurrence factor $\mathring{w}_\mathrm{f}$=0 is initialized and the strength of topological links between each node is initialized as TC(i, j)=0.

4.1.3 *Step 3: Selection of Winning Node with the Best Matching Score*

The winner node ⓦ is selected using Eq. (1). The distance between the input $S(t)$ and the weight vector $\acute{W}_j(S_i)$ is calculated, and each node is given a position value $\mathcal{P}_i$, where i=0, 1,…,$\underset{\wedge}{N}$. Because of the winning node distance to the input S_i, the position value $\mathcal{P}_i$ is assumed to be 0. The winner node ⓦ winning occurrence factor $\mathring{w}_\mathrm{f}$=0 is incremented by one.

$$\text{ⓦ} = \arg\left(\min_{1\leq i\leq \underset{\wedge}{N}, j\leq \underset{\wedge}{N}} \left\lfloor \sqrt{\left(s_i - w_{ji}\right)' \delta_m \left(s_i - w_{ji}\right)} \right\rfloor\right), \qquad (1)$$

where $\delta_\mathrm{m} = \sqrt{\frac{1}{n-1}\sum_{i,j=1}^{n}\left(s_i - w_{ji}\right)^2}$ and $\lfloor . \rfloor$ denotes Mahalanobis distance measure.

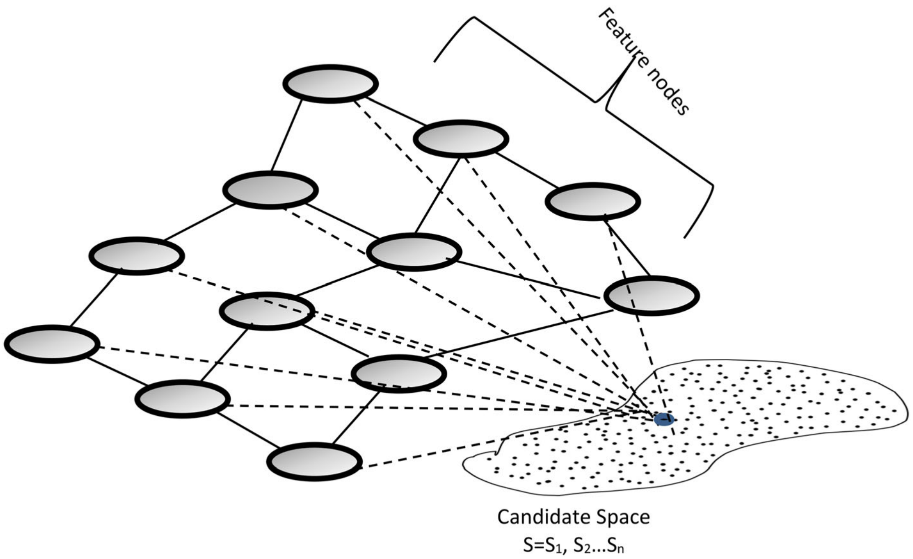

Fig. 3 CSOM neighborhood input space portioning

Here, s_i and w_{ji} are the input data and weight vector of nodes $\underset{\wedge}{\mathrm{N}}$ with t repetitions in different time intervals.

4.1.4 Step 4: Estimation of Neighbor Nodes Spatial Correlation

Among $\acute{W}_j(S_i)$ inputs, tentative winning node's $\textcircled{w}$ neighbor's spatial sensor nodes ($\underset{\wedge}{N}$) correlation proximity is considered. The correlation between the node in the feature map and its neighbor node readings is analyzed for calculating spatial correlation. In the spatial correlation model, all the received sensor data from reporting node SN_iandSN_j locations have been experimented and the threshold distance C_θ and the variance σ^2 are determined. Spatial correlation SCorr can be expressed as follows:

$$\mathrm{SCorr}\left\{\mathrm{SN}_i, \mathrm{SN}_j\right\} = \sigma^2 C_\theta \|\mathrm{SN}_i - \mathrm{SN}_j\|. \tag{2}$$

The tentative winning node's $\textcircled{w}$ neighbor's $N(i)$ is found in the feature map and the new winning node is calculated for updating weight in $\acute{W}_j(S_i)$ in spatially correlated sensor node's data $S(t)$.

4.1.5 Step 5: Wining Node Weight Adjustment Based on Spatial Correlation

The topological connection strength between the winning node and node i is increased by using Eq. (3). The relative winning occurrence factor w_{fi} of the node is updated using Eq. (4).

$$TC_{(\textcircled{w},S_i)} = TC_{(\textcircled{w},S_i)} + 1 \tag{3}$$

$$\mathring{w}'_{fi} = {}^{\mathring{w}_{fi}}\Big/_{\sum_{j=1}^{\underset{\wedge}{\mathrm{N}}} \mathring{w}_{fj}} \tag{4}$$

The following equation is used to update the weight vectors of the winner node and its neighbor with high $\mathrm{SCorr}\left\{\mathrm{SN}_i, \mathrm{SN}_j\right\}$.

$$\acute{W}_{ji}(t+1) = \acute{W}_{ji}(t) + Y_{(\textcircled{w},S_i)}(t)\|S(t) - \acute{W}_{ji}(t)\| \tag{5}$$

Here, $\|.\|$ is Euclidian distance and $Y_{(,S_i)}(t)$ is the neighbor functions described as follows:

$$Y_{(\textcircled{w},S_i)}(t) = \eta(t)(1 - \mathring{w}_{fi}).\exp\left({}^{-\varphi_{(\textcircled{w},S_i)}}\Big/_{2\alpha^2(t)}\right) \tag{6}$$

$$\varphi_{(\textcircled{w},S_i)} = \mathcal{P}_i + \{\|\mathcal{P}_{\textcircled{w}} - \mathcal{P}_i\|^2 + TC_{(\textcircled{w},S_i)}\} \tag{7}$$

where $\mathcal{P}_i$ is the current position value of the node on the feature map, $\eta(t)$ is the training rate, and $\alpha(t)$ is the distance measure of the neighbor node's radius. Both $\eta(t)$ and $\alpha(t)$ reduce at time t with the training period L by using Eqs. (8) and (9).

$$\eta(t) = \eta(0).\left(\frac{\eta(L)}{\eta(0)}\right)^{t/L}, \tag{8}$$

$$\alpha(t) = \alpha(0).\left(\frac{\alpha(L)}{\alpha(0)}\right)^{\frac{t}{L}}. \quad (9)$$

4.1.6 Step 6: Collaboration and Adaptation with Nearest Wining Nodes

Through appropriate adjustment of the related connection weights, the stimulated nodes reduce their values of the dependent variable concerning the input pattern, enhancing the reaction of the winning node to the relevant improvement of a similar input. The nearest node's weight vector is updated using the following equation:

$$\acute{W}_k(t+1) = \acute{W}_k(t) + Y_{(\textcircled{w},S_k)}(t)\left\|S(t) - \acute{W}_k(t)\right\| \quad (10)$$

Here, the function $Y_{(,S_k)}(t)$ is the spatially correlated neighbor function and described in Eqs. (11) and (12).

$$Y_{(\textcircled{w},S_k)}(t) = \eta(t)\left(1 - \mathring{w}_{fi}\right).\exp\left({}^{-G_{(\textcircled{w},S_k)}}\!\Big/_{2\alpha^2(t)}\right) \quad (11)$$

$$\mathrm{G}_{(\textcircled{w},S_k)} = P_i + \left\{\left\|\acute{W}_k(t) - S_i(t)\right\|^2 + TC_{(\textcircled{w},S_i)}/S(t)\right\} \quad (12)$$

Steps 3 to 6 are repeated for all input $S(t)$.

4.2 Multivariate Mamdani Fuzzy Inference System

FIS are approaches that communicate knowledge and erroneous data in a way that is extremely similar to how people think. As a result, this strategy is better used to address problems with a dependent variable rather than observable data. It denotes the existence of a nonlinear connection between one or more input parameters and one or more output variables. This can be used as a beginning point for decision making.

The standard inference engine can accept any fuzzy or crisp input, although it invariably creates fuzzy sets as output [30]. Crisp output can be required, particularly in situations when a FIS is utilized as a controller, as shown in Fig. 4. As a result, we will need a defuzzification method for obtaining a crisp output from a fuzzy set. FIS outfits a nonlinear plotting from its contribution space to production space. Many fuzzy if–then rules are used to perform this mapping.

Multivariate Mamdani FIS (MMFIS) involves a collection of fuzzy rules where its premise part is fuzzy and the consequent part is also fuzzy. Every stage of the inference process is defined in detail to provide a thorough understanding of how a simple FIS works. The system receives input parameters such as temperature and humidity, which are fuzzified and processed by several IF–THEN rules before being deputized to represent the optimal physical phenomenon values.

The IF–THEN rules are proposed by Mamdani's model with a set of guidelines. The following are the definitions of an input matrix and an output vector.

$$F = [f_1, f_2, \cdots f_n]^T \begin{bmatrix} f_{11} & f_{11} & \cdots & f_{1n} \\ f_{21} & f_{22} & \cdots & f_{2n} \\ f_{n1} & f_{n2} & \cdots & f_{nn} \end{bmatrix},$$

$$G = [g_1, g_2, \cdots g],$$

where f_1, f_2, f_n are antecedents and h_n is the consequent, g_1, g_2,...,g_n are fuzzy membership set, and R_n is the number of rules. MFIS functionality is described by a set of fuzzy IF–THEN rules that denote input–output functions of inference engine. Rule R_i of the nonstop Mamdani model is of the following form:

If f_1 is g_1 and f_2 is g_2 .... and f_n is g_n Then h_1 is g_{1k}

The MMFIS is meant to take four maximum inputs and produce a single output/multi-output after fuzzification. A Mamdani FIS approach was chosen because it is easy to manipulate, has a well-understood rule basis, and, most significantly, is suitable for human input because it considers states that are not clearly described by computing standards. For comparison, our MMFIS anomaly detection process is put up against a Sugeno-based anomaly detection system in [18]. Higher-order Sugeno fuzzy models are also used in effective classification and decision-making processes, however, nonlinear input parameters can add significant computational complexity to the system, lowering its overall efficiency. The various stages of the MMFIS are presented using real data and sample input variables.

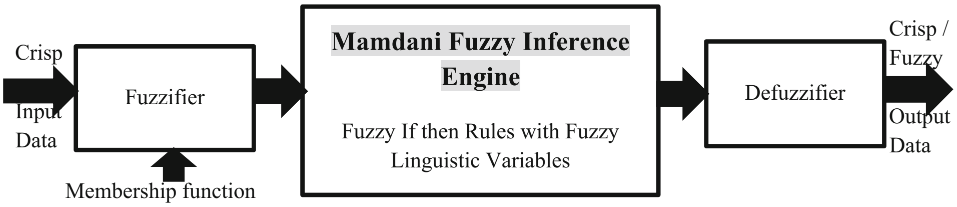
Crisp
Input
Data
Fuzzifier
Membership function
Mamdani Fuzzy Inference Engine
Fuzzy If then Rules with Fuzzy Linguistic Variables
Defuzzifier
Crisp / Fuzzy
Output
Data

Fig. 4 MMFIS logic process

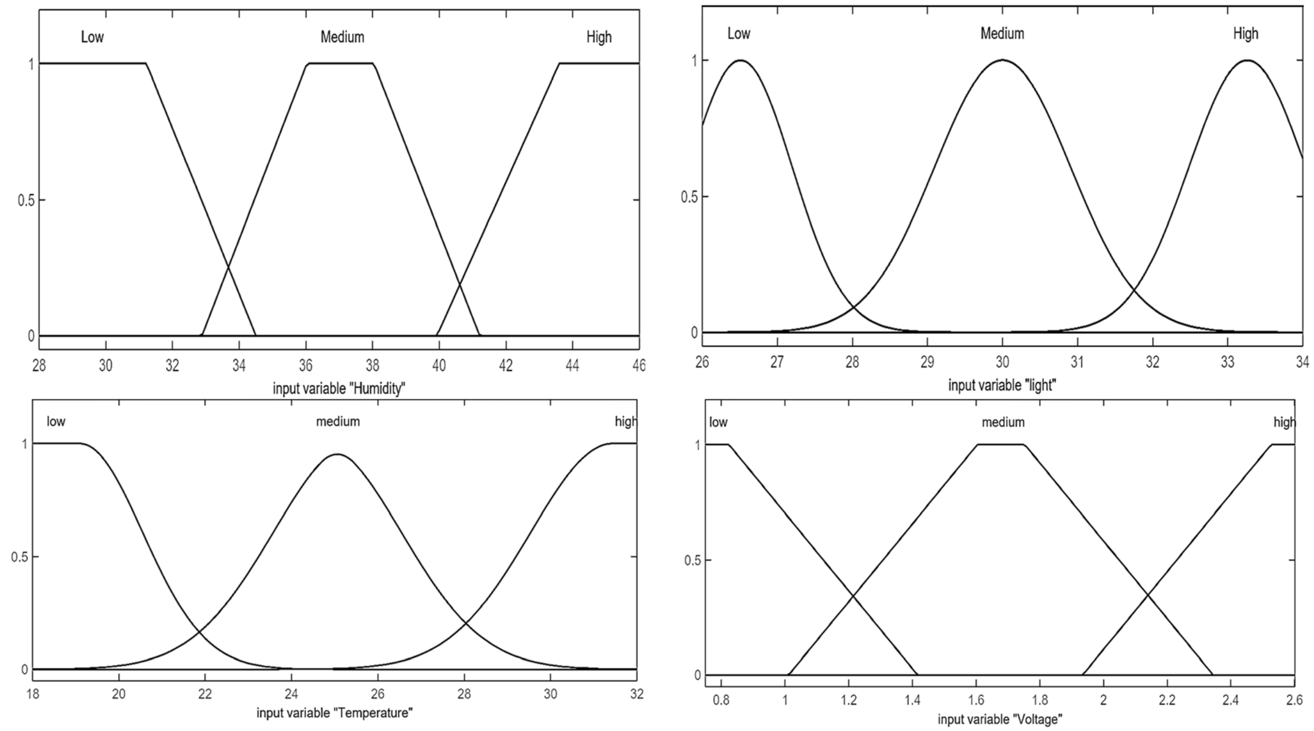

Fig. 5 Input memberships of IBRL dataset

4.2.1 Stage 1: Fuzzification of Input Parameters

The linguistic variables and the linguistic membership values are defined using fuzzification. This is the system's method of using membership functions to determine suitable membership degree inputs corresponding to fuzzy set.

Figures 5 and 6 illustrate the input memberships with the defined values corresponding to the linguistic limits. This is also a set of precise numerical inputs. The crisp input is fuzzified by assigning linguistic variables (low, medium, high) by membership degree in the fuzzification process.

4.2.2 Stage 2: Fuzzy Rule Base and Inference Mechanism

The foundation of the rules defines a fuzzy set's input and output. The degree of belonging, absence of engagement

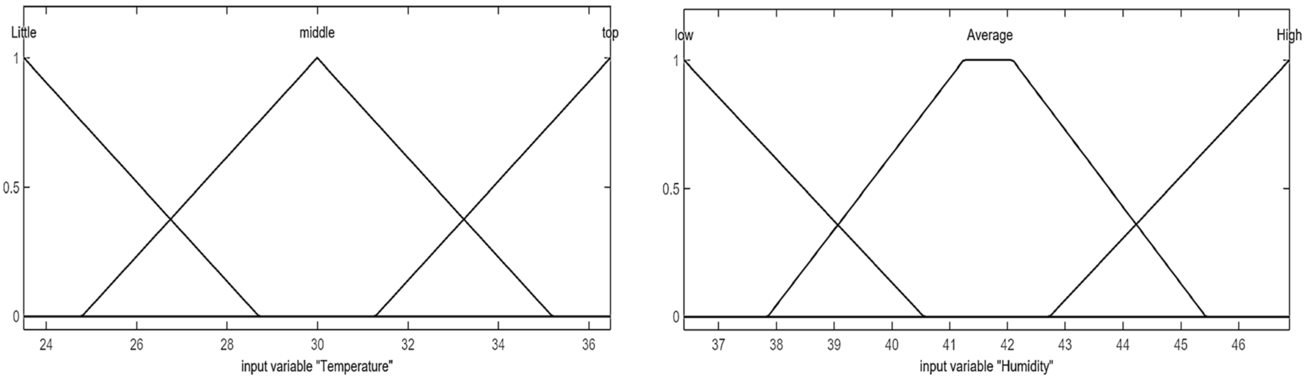

Fig. 6 Input memberships of ISSNIP dataset

Table 1 Rule base inferences

Rule base inferences	Sample rule structure
Rule base I (Attribute Equivalence)	IF (Temperature is High) and (Humidity is Low) and (Light is Medium) and (Voltage is High) THEN (Attribute Equivalence is High Similarity)
Rule base II (Spatial Equivalence)	IF (Distance is Short) and (cluster is Same) THEN (Spatial Equivalence is Near Neighbouring CSOM)
Rule base III (Time Equivalence)	IF (Time is Long) and (cluster is overlapped) THEN (Time Equivalence is Medium Interval)

between elements from different sets is represented by fuzzy relationships. Linguistic variables are used as antecedents and consequents in the fuzzy rules system [33].

The rule list in MMFIS-designed systems can be modified by the user to meet their logical needs. Rule base I is created with 'IF–Then' statements are conditional, as shown in Table 1 the rule that conditions that 'IF Temperature is…,' which translates to "If Temperature is 25–32, Humidity is 26–35, light is 30–34, and voltage is 1.9–2.6 then Attribute Equivalence=60–100' (High Similarity index)." Rule base II is created with conditions that 'IF Distance is …,' which translates to "If Distance is 2–5, sensor data points cluster is in 8–10 best matching score CSOM cluster then Spatial Equivalence=60–100' (Near Neighboring CSOM). Rule base III is created with conditions that 'IF Time is …,' which translates to "If Distance is 2–5, sensor data points cluster is in 8–10 best matching score CSOM cluster then Spatial Equivalence=60–100' (Near Neighboring CSOM). When the rule list is developed, it holds logical rules tailored to the system. The rule

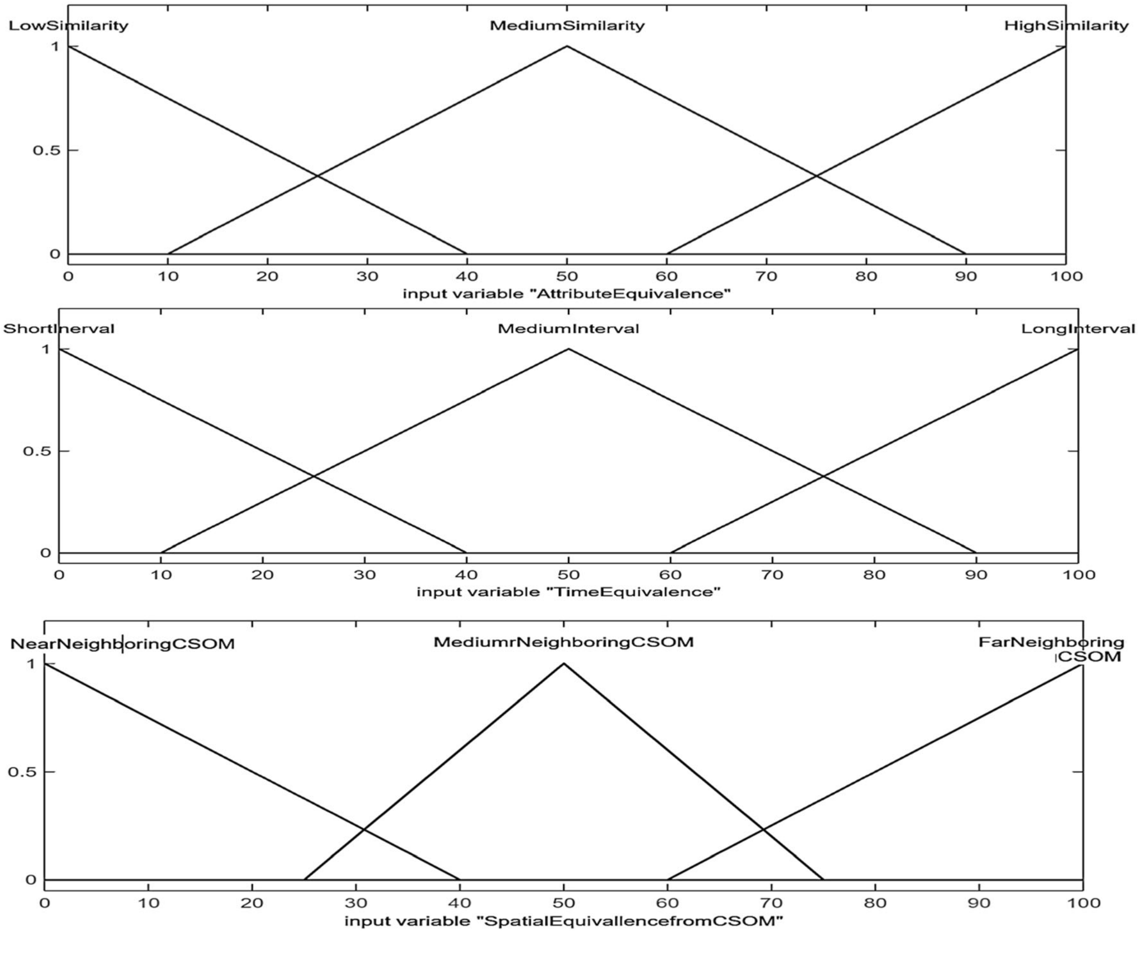

Fig. 7 Fuzzy inference membership functions

Table 2 Decision classification of anomaly detection

Decision classification rule database
1 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
2 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Abnormal)
3 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Abnormal)
4 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
5 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Abnormal)
6 **IF** (Spatial_Equivallence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Abnormal)
7 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
8 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Abnormal)
9 **IF** (Spatial_Equivalence_from_CSOM is Near_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Abnormal)
10 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
11 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Normal)
12 **IF** (Spatial_Equivallence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is short_Inerval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Normal)
13 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
14 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Normal)
15 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Normal)
16 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
17 **IF** (Spatial_Equivallence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Normal)
18 **IF** (Spatial_Equivalence_from_CSOM is Mediumr_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Normal)
19 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
20 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Normal)
21 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is short_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Normal)
22 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
23 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is Medium_Similarity) **THEN** (Decision_Classification is Normal)
24 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is Medium_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Normal)
25 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is Low_Similarity) **THEN** (Decision_Classification is Abnormal)
26 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is Medium_Similaritan exactClassification is Normal)
27 **IF** (Spatial_Equivalence_from_CSOM is Far_Neighboring_CSOM) and (Time_Equivalence is Long_Interval) and (Attribute_Equivalence is High_Similarity) **THEN** (Decision_Classification is Normal)

Table 3 Experimental setup of IBRL–ISSNIP datasets

Dataset	Platform	Sensor type	Quantity	Data type	Features
IBRL	TinyOS	Mica2Dot	54	Labeled and aggregated	Weatherboard stations sensor dataset with deployment location and aggregated raw data values are gathered
ISSNIP	TinyOS	TeleoSB	4	Labeled and normalized	Indoor and outdoor deployment dataset with single-hop and multi-hop sensor nodes data are gathered

Table 4 IBRL dataset outline structure

Date	Time	Epoch	Mote ID	Temperature	Humidity	Light	Voltage
28-02-2004 to 05-04-2004	Every 31 s	Iterative sequence	1–54	Degrees in Celsius	0–100%	Lux	Volts

set is then applied to each permutation of the fuzzified inputs and the appropriate linguist phrases are allocated. Figure 7 illustrates the fuzzy inference membership function for spatial, temporal, and attribute equivalences.

Table 2 confirms spatial equivalence, time equivalence, and attribute equivalence as antecedents for stating results in an anomaly detection system. The suggested system's anomaly diagnosis decision levels are classified as *consistent* and *Inconsistent*. Consistent data are sent to the cluster head or base station. Inconsistent data are disconnected, and the associated attributed data are replaced with imputed data. Finally, proper missing data are inserted, and the dataset is examined by the cluster head for the optimization procedure.

4.2.3 Stage 3: Defuzzification Process

Defuzzification is the reversal of the fuzzification process. At this point, the input has already had its membership degree determined and will be in the form of a linguistic statement. To get a crisp numerical value as an output, the input must be decided to revert into defuzzification. Alternative ways for determining the defuzzification value are possible with the Mamdani type FIS. The user can define the defuzzification methods given below to obtain the desired output. Centroid, bisector, mean maximum, lowest of the maximum, and highest of the maximum are some defuzzification strategies. The centroid method returns a value from the fuzzy set's center of the area. Centroid is calculated by using the equation $F_{\text{Centroid}} = \frac{\int g(x_i).x_i}{\int g(x_i).x_i}$, where $g(x_i)$ is the membership value for a point. The bisector defuzzification method finds the line that divides the fuzzy set into two equal-sized subregions, which is usually the centroid line. Maximums in the middle, largest, and smallest are defuzzification procedures that take the greatest value of the plateau. Our proposed method is evaluated with available defuzzification methods.

5 Results on Evaluation

The performance of the proposed method is compared with the existing method, the conventional SOM, CSOM, MMFIS without optimized input structure portioning. The experiments related to the comparative study are performed by using MATLAB 2017 b. In existing work [18] anomaly detection is performed using Fuzzy C means input space structuring and Sugeno fuzzy inference-based classification model. The traditional SOM and CSOM have been evaluated with the dataset for anomaly detection. SOM with MMFIS and CSOM with MMFIS have been evaluated for the comparison study. The proposed methodology is evaluated using two real lab datasets and scalability is evaluated by using synthetic datasets. Table 3 describes the experimental setup of the two real datasets.

5.1 IBRL Dataset

The Intel Berkeley Research Lab dataset (IBRL) is based on currently accessible Intel Lab data and consists of actual measurements taken from 54 sensors placed at the IBRL Lab [34]. Mica2Dot sensors equipped with weather stations gathered environmental data such as temperature, humidity, light, and voltage regularly in every 31 s. Table 4 depicts the IBRL dataset outline structure.

5.2 ISSNIP Dataset

The intelligent sensors, sensor network, and information processing (ISSNIP) dataset contain actual sensor data gathered by motes in standard WSNs. Four sensor nodes are available indoor and outdoor deployment areas. The data comprise temperature and humidity readings taken at five-second intervals during 6 h. Probability-based corruptions were introduced randomly to generate inconsistent data [35]. Table 5 depicts the ISSNIP dataset outline structure. As a result, all ISSNIP anomalies are a series of erroneous readings induced by the fault occurrence in the dataset.

5.3 Performance Metrics

The proposed method is evaluated using the correlation measures like accuracy, false alarm rate, precision, detection rate, specificity, $F1$ score, and Matthews correlation coefficient (MCC) based on confusion matrix representing the number of True-Positive Indexes (TPI), True-Negative Indexes (TNI), False-Positive Indexes (FPI), and False-Negative Indexes (FNI) are the performance measures observed for evaluation. The performance assessment metrics are calculated using the equations listed below:

$$\text{Accuracy} = \frac{N(\text{TPI} + \text{TNI})}{N(\text{TPI} + \text{TNI} + \text{FPI} + \mathit{FNI})} \qquad \text{False Alarm Rate(FAR)} = \frac{\boldsymbol{N}(\text{FPI})}{\boldsymbol{N}(\text{FPI} + \text{TNI})}$$

$$\text{Precision} = \frac{\boldsymbol{N}(\text{TPI})}{\boldsymbol{N}(\text{TPI} + \text{FPI})} \qquad \text{Detection Rate(or)Sensitivit}y = \frac{N(\text{TPI})}{N(\text{TPI} + \text{FNI})}$$

$$\text{Specificity} = \frac{\boldsymbol{N}(\text{TNI})}{\boldsymbol{N}(\text{TNI} + \text{FPI})} \qquad F1\text{Score} = \frac{2 * N(\text{TPI})}{\text{Numberof}(2 * \text{TPI} + \text{FPI} + \mathit{FNI})}$$

$$\text{Mathhews correlation coefficient} = \frac{N(\text{TPI} * \text{TNI}) - N(\text{FPI} * \text{FNI})}{\sqrt{N(\text{TPI} + \text{FPI}) * N(\text{TPI} + \text{FNI}) * N(\text{TNI} + \text{FPI}) * N(\mathit{TNI} + \mathit{FNI})}}$$

The classification model predicts the class of each data instance, assigning anticipated label (positive or negative) to each sample: the confusion matrix $\text{CM} = \begin{pmatrix} \text{TPI} & \text{FNI} \\ \text{FPI} & \text{TNI} \end{pmatrix}$ expanded in Table 6. It represents the cataloging decision of the outcome. The real label forecasts are true-positive indexes and true-negative indexes, while the wrong predictions are false-negative indexes and false-positive indexes.

Table 5 ISSNIP dataset outline structure

Readings	Mote ID	Humidity	Temperature
9th May 2010 and 10th July 2010	1–4	0–100%	Degrees in Celsius

Table 6 Confusion matrix

	Anticipated Positive	Anticipated Negative
Real Positive	True Positive (True Index)	False Negative (False Index)
Real Negative	False Positive (False Index)	True Negative (True Index)

6 Results and Discussion

The proposed methodology is evaluated with the physical phenomena parameters with training dataset and test dataset of the IBRL dataset and the ISSNIP dataset is shown in Figs. 8 and 9, respectively. The corrupted data index in testing and the normalized data index in training is projected along with its correlation residual threshold. It shows the normal data index without spikes and the abnormal anomaly data index with red color marked spikes.

In the training phase, normalized data are considered for fixing threshold values from input space partitioning and fuzzy models. In CSOM, the nearest distance between the sensor nodes is calculated and the maximum distance is considered as threshold T_d and the multivariate correlation is used to fix threshold T_c. Minimum and maximum time difference frequency threshold values T_t are considered by analyzing the time interval of every sensed data. The thresholds for the IBRL are fixed as Humidity (28 to 46), Temperature (18 to 32), Light (26 to 34), Voltage (0.8 to 2.6), and ISSNIP dataset thresholds are fixed as Temperature (22 to 38), Humidity (36 to 47). The threshold for membership functions can be set by analyzing the minimum and maximum support value of the individual linguistics.

The proposed multivariate data analysis is used to discover anomalous data that denote the association or relationship, which identifies the relationship between the variables involved in the anomaly detection process. Anomaly detection in multivariate data analysis differs significantly from the regular study that includes the input variables in multidimensional space.

Figures 10 and 11 show a quick representation of attribute correlation, which was used in the assessment using IBRL dataset with no data contamination probability index. Temperature, humidity, light, and voltage are all

Table 7 Summary of the accuracy rate of the proposed method in the different datasets under different levels

Dataset	Total samples	Normal	Anomaly	Accuracy (%)	Contamination index
				10%	50%
IBRL	625,000	125,000	500,000	98	98.65
		375,000	250,000	98	98.12
		500,000	125,000	99	100
		562,500	62,500	99.12	100
		593,750	31,250	99.89	100
		609,375	15,625	100	100
		617,188	7812	100	100
	Average accuracy rate	99.34			
ISSNIP	4690	1550	3100	98.75	99.5
		2365	2325	99.58	99.8
		3528	1162	100	100
		4108	582	100	100
		4400	290	100	100
	Average accuracy rate	99.76			

Table 8 Comparison of performance metrics of the proposed method with IBRL dataset

Techniques	Accuracy	Sensitivity	FAR	Precision	Specificity	*F*1 Score	MCC
Proposed method	99.34	99.86	0.6	98.78	98.85	99.8	0.8
Existing method [18]	98.12	99	2.57	97.16	97.45	98.45	0.45
SOM with MMFIS	93.57	92.45	3.5	95.6	96.34	91.87	0.06
CSOM	89.67	87	3.5	90.7	83	91.39	0.24
DBSCAN algorithm	92.75	88	4.1	91.5	95.45	85.65	0.56
Subtractive clustering	96.76	91.78	1.4	95.87	95.92	90.34	0.64
ODCASC algorithm	85.56	88.67	4.87	94.6	86.87	88.05	0.34

Table 9 Comparison of performance metrics of the proposed method with ISSNIP dataset

Techniques	Accuracy	Sensitivity	FAR	Precision	Specificity	*F*1 Score	MCC
Proposed method	99.76	98.86	1.05	99.01	99.05	98.09	0.76
Existing method [18]	98	98.67	2.95	96.9	95.15	97.89	0.37
SOM with MMFIS	95.12	85.94	6.23	93.83	96.78	91	0.48
CSOM	87.67	90.63	8.85	80.82	83.98	87.91	0.56
DBSCAN algorithm	90.09	85.67	3.98	90.89	91.29	87.89	0.43
Subtractive clustering	94.23	91.23	1.98	92.83	92.67	92.07	0.39
ODCASC algorithm	84.89	85.78	5.87	91.91	84.84	83.23	0.21

Table 10 Complexity comparison of proposed method with possible methods

Techniques	Computational complexity	Communication complexity
Proposed method	$O(nc\rho + r)$	$O(n/2)$
Existing method [18]	$O(ndc)$	$O(nd)$
SOM with MMFIS	$O(nc + r^2)$	$O(n/2 + \alpha)$
CSOM	$O(nc\rho)$	$O(n/2 + \alpha)$

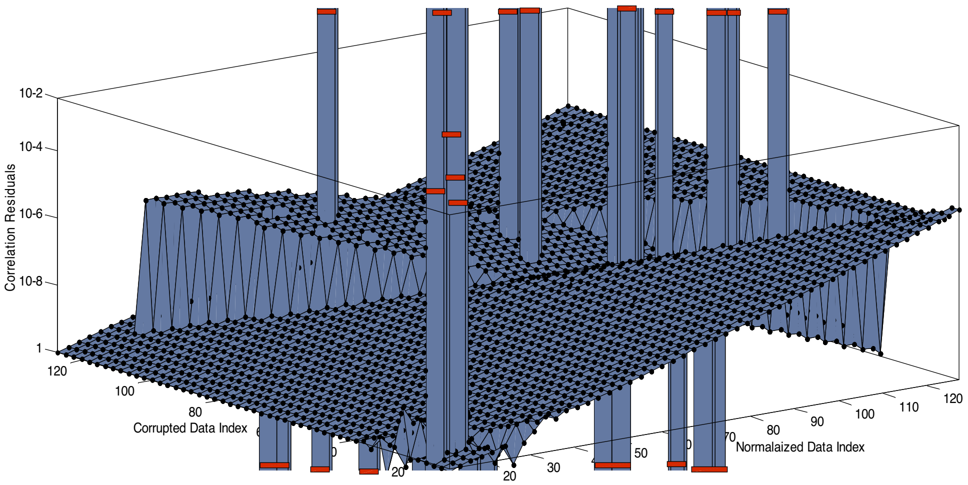

Fig. 8 Anomaly detection in IBRL dataset

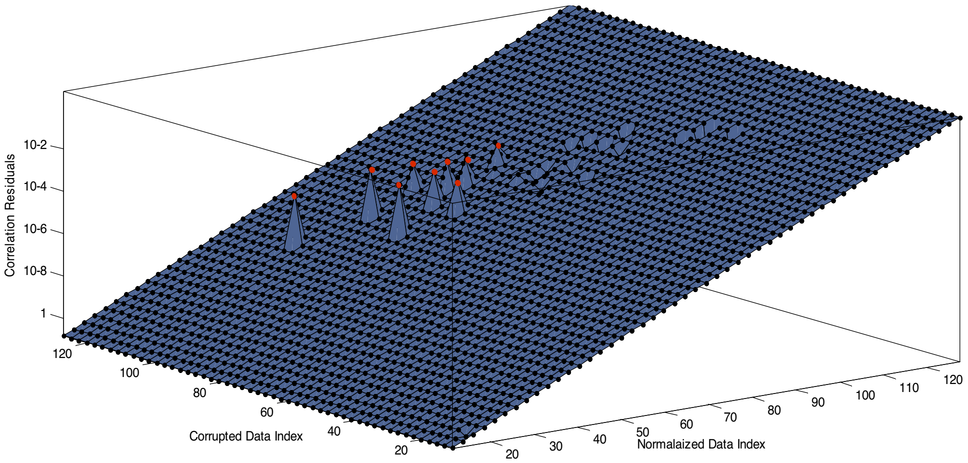

Fig. 9 Anomaly detection in ISSNIP dataset

multivariate data variables that have been standardized from the IBRL dataset. Attribute correlation threshold values are identified and fixed as minimum, maximum, and median correlation values.

The NN-based CSOM methodology is used to consider spatial correlation among real datasets. For computing spatial correlation with NNs in the clusters, dataset real values such as temperature and humidity values sensed by sensor nodes in the deployment region are evaluated with correlation-based Mahalanobis distance estimation. For examining the proposed methodology's performance, the cluster size could be changed. Figure 12 illustrates the threshold distance values of the attribute in the training phase. In Fig. 13, the NN distance with CSOM is analyzed for estimating the spatial correlation. A solid black line indicates the NN distance between the data indexes. At the end of 65 iterations, similar and dissimilar data indexes are separated with accurate spatial correlation with CSOM. For training SOM, a total of 625,000 samples and 4690 dataset samples are considered from the ISSNIP dataset. While increasing dataset size, our proposed system produces good performance in all aspects except computational complexity.

Figure 14 depicts the results, which indicate the detection rate of the various approaches with varying contamination percentages. The contamination probability goes from 5 to 70%. The proposed methodology performs well in terms of low contamination, whereas the existing method performs poor in terms of low contamination. The performance of the proposed method degrades marginally as the contamination rates increases because the contaminated data will be considered as real, while the contamination ratio crosses the limit of above 50% of its original value.

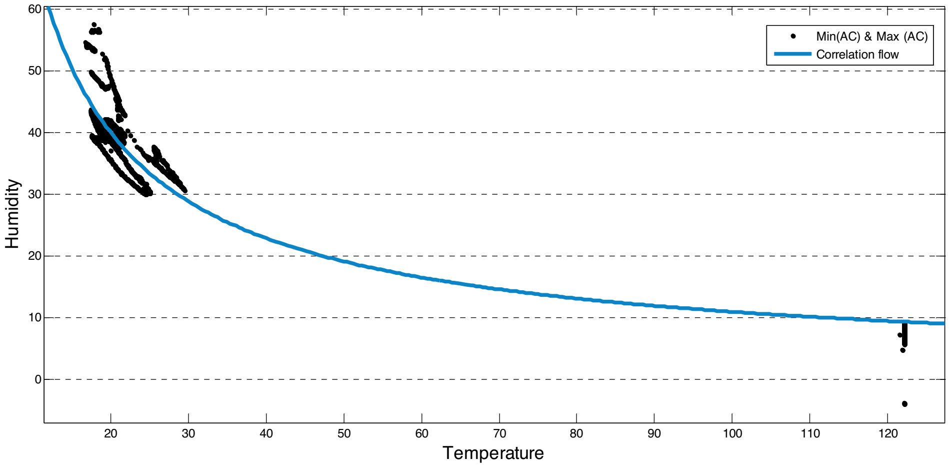

Fig. 10 Correlation flow between temperature and humidity

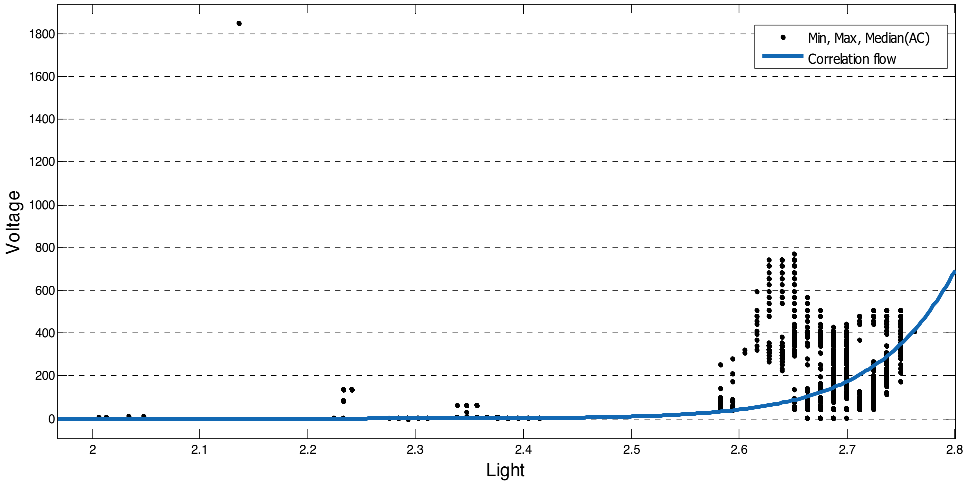

Fig. 11 Correlation flow between light and voltage

The tradeoffs between the resultant detection rate and false alarm rate for five tests are depicted in Fig. 15. Every point denotes a new contaminated training set, and the results show the contamination ratio index. As previously demonstrated, the existing method [18] can achieve acceptable detection rate for moderate contaminated training sets, but the false alar rate is also very high in comparison to the proposed method. For low-contaminated tests, the false alarm rate of CSOM maintains zero. Even though MMFIS without considering the spatial neighbor rule system of CSOM has a better performance for contamination which is equal to 5%, its detection rate, and high false alarm rate in comparison to the other methods. In summary, the existing method, CSOM, and MFIS without spatial rule inference have their average performances for contamination ratio of 5 to 6%. Based on these parameters, we may infer that CSOM–MMFIS is the more effective methodology. The primary cause is that the training of CSOM with the spatial distance-based weight adjustments along with well-defined rule inferences of MMFIS from the optimal data CSOM clusters will provide a high detection rate for all types of anomalies.

Table 7 depicts the overall performance of the proposed method using two datasets. CSOM with MMFIS achieves good accuracy in different contamination indexes while varying the number of anomalous data in the total samples. Tables 8 and 9 show the evaluation results related to the accuracy, sensitivity, FAR, precision, specificity, F1 score, and MCC of the proposed CSOM with MMFIS with other considerations and existing method that was discussed in related work. The metric MCC infers that the model produces good results only if it classifies both positive and negative elements.

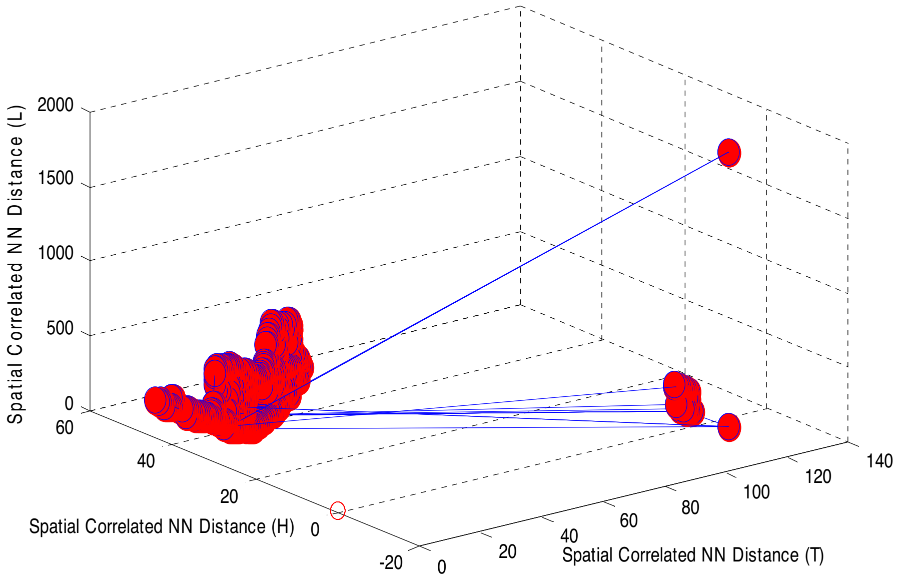

Fig. 12 Spatial nearest neighbor correlation distance threshold in the training phase

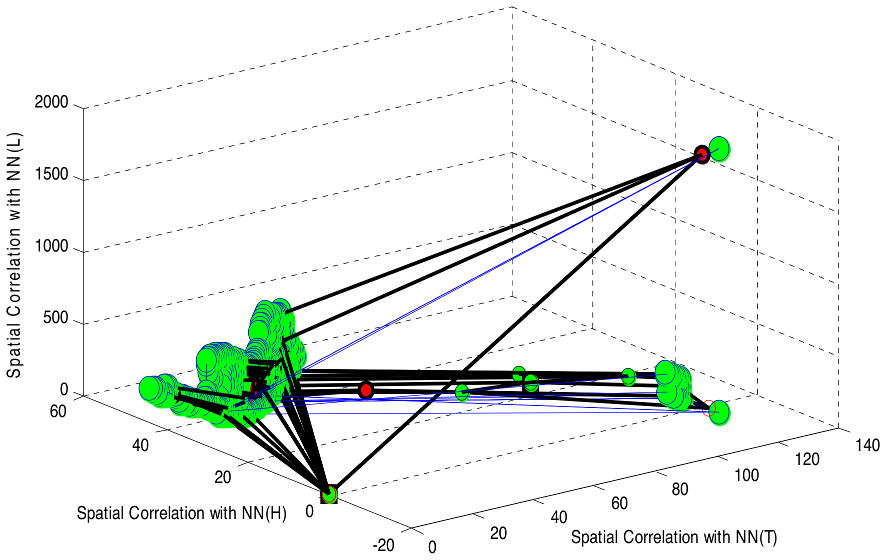

Fig. 13 Spatial nearest neighbor correlation distance threshold in the testing phase

The proposed model's detection performance is compared with existing work discussed in Sect. 2. From literature, the proposed CSOM with MMFIS model for anomaly detection are considered. Tables 7 and 8 show the comparison of proposed models with existing anomaly and fuzzy models with respect to detection accuracy, $F1$ score, MCC, etc., The overall performance of the proposed CSOM with MMFIS achieves high accuracy, high sensitivity, $F1$ score with a good MCC score compared to existing techniques in IBRL and ISSNIP datasets. The detection accuracy merely will not be a reliable indicator. The MCC is assessed to identify the proposed model's capabilities in terms of efficacy to eliminate the problem of unequal classification.

During the training phase, anomaly detection threshold values were determined for the MMFIS Inference system's decision-making process. During the testing phase, several rules are fired based on the condition of the premise, an inference mechanism is used, and an accurate anomaly prediction decision may be made. When the threshold

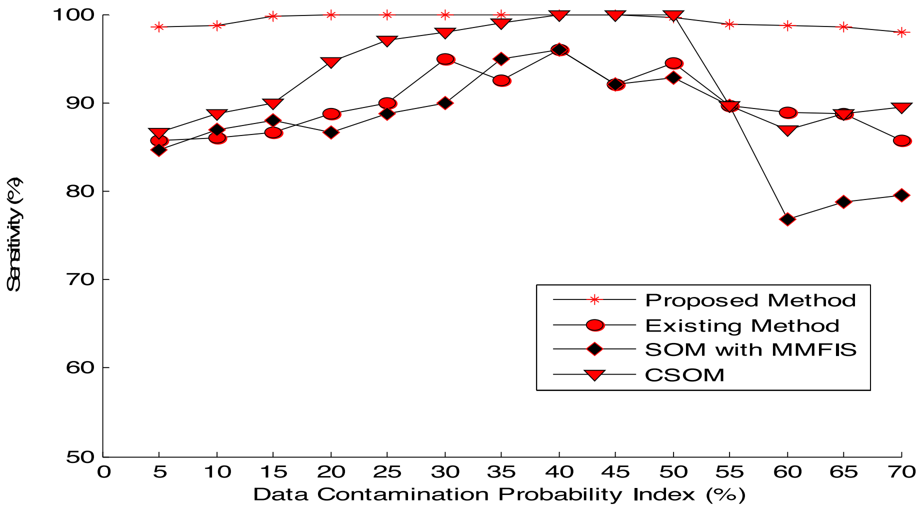

Fig. 14 Data contamination index versus sensitivity

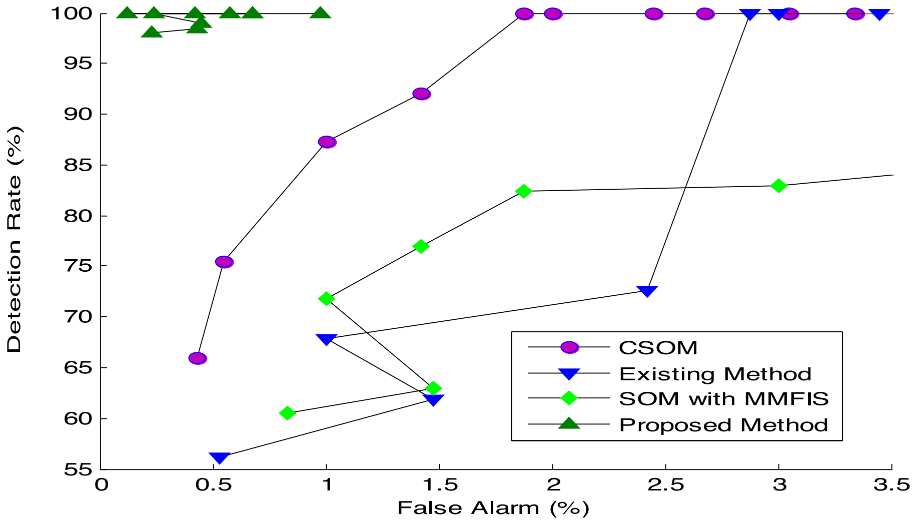

Fig. 15 Comparison of false alarm rate versus detection rate

cutoff is lower, inconsistent data begin to influence the fuzzy model's decision-making quality. If the number of rules is too minimal in this scenario, the model is too weak. When the number of rules is too excessive, the model fits against its training data indexes and takes additional computing time. Figure 16 depicts the accuracy of the suggested method in several corruption indexes versus several rules.

In Fig. 17, the proposed method scalability is compared with existing work by varying the dataset size from 10 to 100%. It shows the accuracy of the proposed method slightly decreases, while the dataset size reaches above 60% but the existing method scalability is poor for the same. Moreover, the computational complexity of the methodology might be increased while increasing the dataset size for analysis. Our proposed method running time is around 5.7 min on an average for the whole dataset, whereas the existing method crosses above 15 min on an average for computing tasks in the whole dataset.

The complexity of various approaches is explained in Table 10. It is necessary to compare the suggested strategy to existing anomaly detection approaches to fully comprehend its performance. The computational complexity, communication overhead, and energy are all taken into account while evaluating the efficiency of the CSOM with the MMFIS model. Our model has a computational complexity of $O(nc\rho+r)$, where n is the number of indexes in the dataset, c is the number of similar groups or clusters formed, ρ-correlation coefficients are the number of MFIS rules evaluated for decision making, and r is the number of

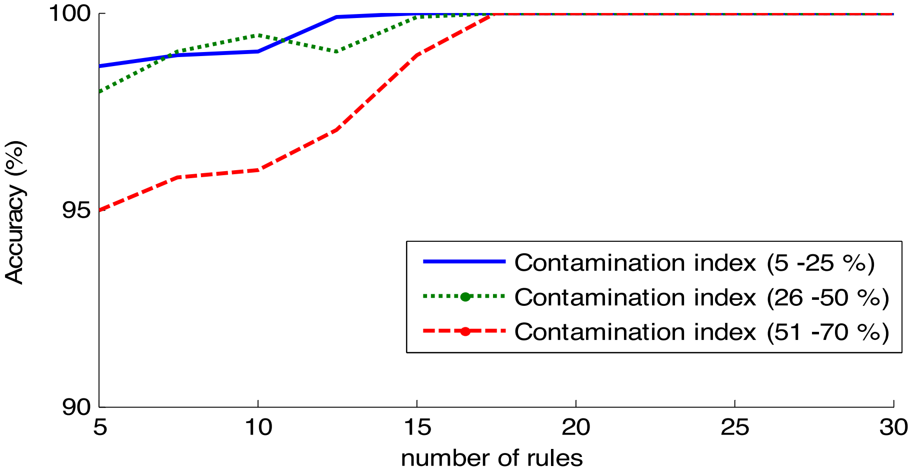

Fig. 16 Classification of accuracy with different contamination data indexes

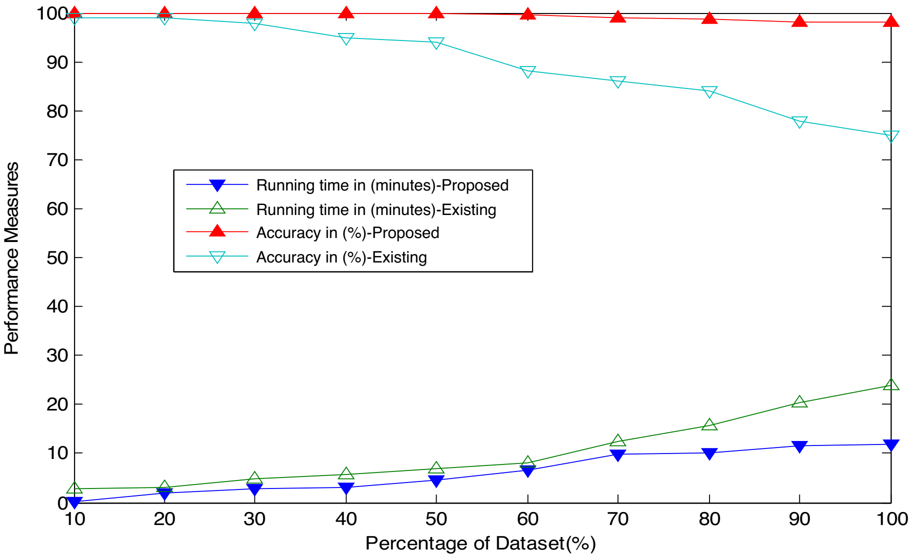

Fig. 17 Scalability of the proposed method

MFIS rules evaluated for decision making. The communication overhead is $O(n/2)$ in base stations that handle fused consistent dimensionality reduction data, and α reflects the number of mispredicted data transmissions that occur in existing approaches due to low detection rates. When compared to other peer methods, our method has a high level of accuracy and computational complexity, communication complexity, memory complexity are all marginally reduced.

7 Conclusion

We provide a customized self-organizing clustering method using a MMFIS in this study. Sensor networks have the property of correlating data between geographically close nodes. As a result, aggregating data in the network and summarizing data are critical in a WSN. Although grouping data decreases traffic and increases network lifetime, it may reduce data accuracy. The sensor network requires an efficient anomaly detection mechanism that does not impair the accuracy of data received by the base station. We employ correlation-based CSOM in conjunction with MMFIS to detect inconsistent data caused by physical phenomenon activity in the deployment region. In two real datasets, the evaluation results show that the proposed method surpasses the previous work in multiple aspects such as detection rate, accuracy, false alarm, specificity, $F1$ Score, and MCC.

References

1. Can, O., Sahingoz, O.K.: A survey of intrusion detection systems in wireless sensor networks. In: 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO), Istanbul, Turkey, 27–29 May 2015, pp. 1–6
2. Xie, M., Han, S., Tian, B., Parvin, S.: Anomaly detection in wireless sensor networks: a survey. J. Netw. Comput. Appl. **34** (4), 1302–1325 (2011)
3. Sheng, Z., Mahapatra, C., Zhu, C., Leung, V.: Recent advances in industrial wireless sensor networks towards efficient management in IoT. IEEE Access **3**, 622–637 (2015)
4. Wang, D., Xu, R., Hu, X., Su, W.: Energy-efficient distributed compressed sensing data aggregation for cluster-based underwater acoustic sensor networks. Int. J. Distrib. Sens. Netw. **2016**, 1–14 (2016)
5. Ramotsoela, D., Abu-Mahfouz, A., Hancke, G.: Survey of anomaly detection in industrial wireless sensor networks with critical water system infrastructure as a case study. Sensors **18**, 2491 (2018)
6. Alsheikh, M.A., Lin, S., Niyato, D., Tan, H.P.: Machine learning in wireless sensor network: algorithm, strategies & application. IEEE Commun. Surv. Tutor. **16**(4), 1996–2018 (2014)
7. Rajasegarar, S., Leckie, C., Palaniswami, M.: Anomaly detection in wireless sensor networks. IEEE Wirel. Commun. **15**, 34–40 (2008)
8. Aggarwal, C.C., Yu, P.S.: Outlier detection for high dimensional data. ACM SIGMOD Rec. **30**(2), 37–46 (2001)
9. Feng, H., Liang, L., Lei, H.: Distributed outlier detection algorithm based on credibility feedback in wireless sensor networks. IET Commun. **11**(8), 1291–1296 (2017)
10. Abid, A., Kachouri, A., Mahfoudhi, A.: Outlier detection for wireless sensor networks using density-based clustering approach. IET Wirel. Sens. Syst. **7**(4), 83–90 (2017)
11. Titouna, C., Aliouat, M., Gueroui, M.: Outlier detection approach using Bayes classifiers in wireless sensor networks. Wirel. Pers. Commun. **85**(3), 1009–1023 (2015)
12. Chatzigiannakis, V., Papavassiliou, S., Grammatikou, M., Maglaris, B.: Hierarchical anomaly detection in distributed large-scale sensor networks. In: Proceedings of the 11th IEEE Symposia Computer Communications (ISCC), June 2006, pp. 761–767
13. Xie, M., Hu, J., Han, S., Chen, H.-H.: Scalable hypergrid k-NN-based online anomaly detection in wireless sensor networks. IEEE Trans. Parallel Distrib. Syst. **24**(8), 1661–1670 (2013)
14. Chen, S.-L., Fang, Y., Wu, Y.-D.: A new hybrid fuzzy clustering approach to Takagi–Sugeno fuzzy modeling. Int. J. Digit. Content Technol. Appl. **6**(18), 341–348 (2012)
15. Barakkath Nisha, U., Uma Maheswari, N., Venkatesh, R., Yasir Abdullah, R.: Improving data accuracy using proactive correlated fuzzy system in wireless sensor networks. KSII Trans. Internet Inf. Syst. **9**(9), 3515–3537 (2015)
16. Kumaragea, H., Khalil, I., Tari, Z., Zomaya, A.: Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modeling. J. Parallel Distrib. Comput. **73**, 790–806 (2013)
17. Chen, J.-J., Fan, X.-P., Qu, Z.-H., Yang, X., Liu, S.-Q.: Subtractive clustering based clustering routing algorithm for wireless sensor networks. Inf. Control **37**(4), 201–219 (2008)
18. Barakkath Nisha, U., Uma Maheswari, N., Venkatesh, R., Yasir Abdullah, R.: Fuzzy based flat anomaly diagnosis and relief measures in distributed wireless sensor network. Int. J. Fuzzy Syst. **19**, 1528–1545 (2017)
19. Barakkath Nisha, U., Uma Maheswari, N., Venkatesh, R., Yasir Abdullah, R.: Robust estimation of incorrect data using relative correlation clustering technique in wireless sensor networks. In: IEEE International Conference on Communication and Network Technologies, 2014, Issue 1, pp. 314–318 (2014)
20. Vesanto, J., Alhoniemi, E.: Clustering of the self-organizing map. IEEE Trans. Neural Netw. **11**(3), 586–600 (2000)
21. Larabi-Marie-Sainte, S.: Outlier detection based feature selection exploiting bio-inspired optimization algorithms. J. Appl. Sci. **11**, 6769 (2021)
22. Erfani, S.M., Rajasegarar, S., Karunasekera, S., Leckie, C.: High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. Pattern Recognit. **58**, 121–134 (2016)
23. Yasir Abdullah, R., Mary Posonia, A, and Barakkath Nisha, U.: An Adaptive mountain clustering-based anomaly detection for distributed wireless sensor networks. In: International Conference on Communication, Control and Information Sciences (ICCISc), 2021, pp. 1–6 (2021)
24. Cannady, J.: An adaptive neural swarm approach for intrusion defense in ad hoc networks. In: SPIE Defense, Security, and Sensing, p. 80590P. International Society for Optics and Photonics, Washington, DC (2011)
25. Martins, H., Palma, L., Cardoso, A., Gil, P.: A support vector machine-based technique for online detection of outliers in transient time series. In: Proceedings of the 10th Asian Control Conference (ASCC), Kota Kinabalu, Malaysia, June 2015, pp. 1–6
26. Saeedi Emadi, H., Mazinani, S.M.: A novel anomaly detection algorithm using DBSCAN and SVM in wireless sensor networks. Wirel. Pers. Commun. **98**, 2025–2035 (2018)
27. Li, M., Sharma, A.: Abnormal data detection in sensor networks based on DNN algorithm and cluster analysis. J. Sens. **2022**, 1718436 (2022)
28. Samara, M.A., Bennis, I., Abouaissa, A., Lorenz, P.: A survey of outlier detection techniques in IoT: review and classification. J. Sens. Actuator Netw. **11**, 4 (2022). https://doi.org/10.3390/jsan11010004
29. Liu, F., Cheng, X., Chen, D.: Insider attacker detection in wireless sensor networks. In: Proceedings of the International Conference on Computer Communications; Honolulu, HI, USA, 13–16 August 2007, pp. 1937–1945 (2007)
30. Jang, J.-S.R., Sun, C.-T., Mizutani, E.: Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence, p. 26. Prentice-Hall, Upper Saddle River (1997)
31. Kaski, S., Honkela, T., Lagus, K., Kohonen, T.: Websom, “Self-organizing maps of document collections.” Neurocomputing **1998**(21), 101–117 (1998)
32. Chaudhary, V., Bhatia, R.S., Ahlawat, A.K.: A novel Self-Organizing Map (SOM) learning algorithm with nearest and farthest neurons. Alex. Eng. J. **53**(4), 827–831 (2014)
33. Shanmugam, B., Idris, N.B.: Improved intrusion detection system using fuzzy logic for detecting anomaly and misuse type of attacks. In: International Conference of Soft Computing and Pattern Recognition, 2009, pp. 212–217 (2009)
34. IBRL Dataset. http://db.csail.mit.edu/labdata/labdata.html. Accessed 12 Nov 2021

35. ISSNIP Dataset. https://home.uncg.edu/cmp/downloads/lwsndr.html. Accessed 23 Nov 2021

R. Yasir Abdullah has received his Bachelor of Engineering in Electronics and Communication Engineering from Anna University, Chennai. He has got his Master of Engineering in Computer Science with distinction from Anna University, Coimbatore, Tamilnadu, India in 2009. He is pursuing Ph.D. in the domain of data quality in sensor networks at Sathyabama Institute of Science and Technology, Chennai. Currently, he is working as an Assistant Professor at Sri Krishna College of Engineering and Technology, Coimbatore, India. His research interests lie in Wireless Networks, Information Security, Computer Networks, Sensor Networks, etc. He has a total of 16 years of teaching experience which includes 11 years of research experience. He has published various papers in SCI and SCOPUS indexed journals which include 20 papers in international journals and 5 papers in National journals, and has presented 22 papers at international conferences and 10 papers at National conferences. His research interests include sensor networks, data mining, data analytics, neural networks, soft computing, network security, and networks. He has published several book chapters in SCOPUS indexed Book series.

A. Mary Posonia has completed her B.E. (CSE) in 2002 from Manonmaniam Sundaranar University, M.E. (CSE) in 2004 from Sathyabama University and Ph.D. in 2018 from Sathyabama Institute of Science and Technology. She has 20 years of teaching experience and is currently working at Sathyabama Institute of Science and Technology as an Associate Professor in the Department of Computer Science and Engineering. Her area of interest includes Machine Learning, Data Mining, and Networking, and has done more than 40 publications in SCOPUS and Web of Science journals.

Usman Barakkath Nisha is a Senior Member of IEEE, she had contributed technically to various events conducted by IEEE. She has completed her Ph.D. in the area of Wireless Sensor Networks from Anna University, Chennai in March 2017. She has received her Bachelor of Engineering in Computer Science Engineering in 2008 with distinction from Anna University, Chennai. She has got her Master of Engineering in Computer Science Engineering from Anna University, Chennai. She has received Gold Medal for securing First Rank in her post-graduation. She has worked as Associate Professor in well-established engineering colleges for the past fourteen years. Currently, she is working as an Associate Professor in the Department of Information Technology at Sri Krishna College of Engineering and Technology (Autonomous), Coimbatore. Her area of Interest includes Ad hoc networks, Wireless sensor Networks, Data mining, and Natural Language Processing. She has published various papers in reputed SCI, SCOPUS indexed Journals, and has presented papers in various national and International Conferences. She is an active member of various professional societies. She has reviewed more than 20 papers as a reviewer for international journals and conferences.