Wireless Netw (2015) 21:1425–1442
DOI 10.1007/s11276-014-0858-z

# Anomaly detection and foresight response strategy for wireless sensor networks

**Mohammad GhasemiGol · Abbas Ghaemi-Bafghi · Mohammad Hossein Yaghmaee-Moghaddam · Hadi Sadoghi-Yazdi**

Published online: 28 November 2014


**Abstract** Anomaly detection is an important challenge in wireless sensor networks (WSNs) for fault diagnosis and intrusion detection applications. Sensor nodes are usually designed to be small and inexpensive, so they have limited capabilities, such as limited computational power, memory and energy. This paper presents novel light-weight distributed anomaly detection and a foresight response strategy based on support vector data description (SVDD) for wireless sensor network. SVDD could sometimes generate such a loose decision boundary, when some noisy samples (outliers) exist in the training set. In addition, it requires the solution of a computationally intensive quadratic programming approach which is not applicable in WSNs. Hence, we modified the standard version of SVDD, and proposed the Linear Programming-based Fuzzy-Constraint SVDD (LP-FCSVDD) method to detect the outliers with more accuracy in acceptable time. Then we present a foresight response strategy to resist the intentional, unintentional and false anomalies. The overall experiments show prominence of our proposed method to achieve high detection accuracies on a variety of real and synthetic wireless sensor network datasets.

**Keywords** Anomaly detection · Foresight response · Intrusion detection · Wireless sensor network · Fuzzy constraint support vector data description · Linear programming

## 1 Introduction

A wireless sensor network (WSN) is made up of a mass of distributed autonomous sensors, which monitor the environmental conditions, such as temperature, sound, vibration, pressure, motion and pollutants [1]. WSNs have been applied to many different domains, such as environmental monitoring, hospital-tracking systems, military applications, traffic control, intelligent buildings (or bridges) and other commercial applications. The characteristics of WSNs inevitably cause a sensor node be extremely restricted by resources, including energy, memory, computing, bandwidth, and transmission range. Also due to the nature of the sensor networks, they are vulnerable to security threats, both external and internal [1–3]. Most of the security techniques devised for traditional wired networks are not directly applicable to a WSN environment [4]. Hence, security in WSNs is a very big challenge especially in critical environments.

In a security system, intrusion prevention presents the first line of defense to reduce possible intrusions but it cannot eliminate them. On the other hand, intrusion detection and response system can be used as a second line of defense to detect any suspicious behavior in network traffic [4]. Intrusion detection is defined as the process of monitoring the events occurring in a computer system or

M. GhasemiGol (✉) · A. Ghaemi-Bafghi · M. H. Yaghmaee-Moghaddam · H. Sadoghi-Yazdi
Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran
e-mail: ghasemigol@wali.um.ac.ir

A. Ghaemi-Bafghi
e-mail: ghaemib@ferdowsi.um.ac.ir

M. H. Yaghmaee-Moghaddam
e-mail: hyaghmae@ferdowsi.um.ac.ir

H. Sadoghi-Yazdi
e-mail: h-sadoghi@ferdowsi.um.ac.ir

network, and analyzing them for any sign of possible incidents; which are violations (or imminent threats of violation) of computer policies, acceptable use policies, or standard practices [5]. Despite the fact that intrusion detection systems are well-implemented technologies in wired networks, there are still many open areas in intrusion detection of WSNs [6]. There are two general approaches to intrusion detection: *misuse intrusion detection* (MID) and *anomaly intrusion detection* (AID) [7–9]. Anomaly detection is best suited to WSN because its methodology is flexible and resource-friendly in general [1]. Anomaly detection is defined as the process of comparing definitions of normal activity against observed events in order to identify significant deviations. Anomaly may be caused by not only security threats, but also faulty sensor nodes in the network, or unusual phenomena in the monitoring zone [10].

Several approaches have been proposed for anomaly detection in wired/wireless networks, but they cannot be applied to WSN, as they are too computationally complex to be executed in sensor nodes. Anomaly detection techniques could be categorized as [1]: rule based methods, statistical techniques, machine learning and data mining approaches. Among them, machine learning and data mining schemes are characterized by effective detection of anomalies [11–13]. Classification methods are important systematic approaches in the data mining and machine learning community. They learn a classification model using a set of data instances (training) and classify a new incoming instance into one of the learned (normal/outlier) class (testing) [14].

The one-class classifier is a kind of classification approach that learns the boundary around the normal instances and declares any new instance falling outside this boundary as an outlier. In this kind of classification, we assume one class of data as the target class, and the remaining data is then classified as outliers. One-class classification is particularly significant in applications where only a single class of data-objects is applicable and easy to obtain. Objects from the other classes might be too difficult or expensive to be made available. Accordingly, we would only describe the target class to separate it from the outlier class. However, it is not appropriate to directly apply these kinds of anomaly detection methods which are usually used for wired networks into sensor networks, because of unique properties of sensor networks [15].

The SVDD is a kind of one-class classification approach based on support vector machine [16]. It tries to construct a boundary around the target data by enclosing the target data within a minimum hyper-sphere. Inspired by the support vector machines (SVMs), the SVDD decision-boundary is described by a few target objects, known as support vectors (SVs). A more flexible boundary can be obtained with the introduction of kernel functions, by which data is mapped into a high-dimensional feature-space. The most commonly used kernel function is Gaussian kernel [17]. The SVDD method has two major disadvantages: (1) it could sometimes generate loose decision boundaries when some noisy samples (outliers) exist in the training set, and (2) it requires the solution of a computationally-intensive quadratic programming approach, which is not applicable in WSNs.

In this paper, our goal is to propose a light-weight data mining approach for anomaly detection in WSNs and present a foresight response strategy to resist any kind of anomaly. The main contributions of this paper are as follows:

- We introduce a new one-class classification approach for anomaly detection in WSNs based on support vector data description (SVDD).
- We modify the standard version of SVDD, and propose a Linear Programming based Fuzzy-Constraint SVDD (LP-FCSVDD) method which is a linear optimization problem and can be solved by using the linear programming methods.
- By defining the fuzzy constraints, the LP-FCSVDD method can identify outliers in the training set and also tolerate the sensor failure.
- We present a foresight response strategy to resist the intentional, unintentional and false anomalies in WSNs.

The rest of the paper is organized as follows. In the next section, we summarize related work in this field. The proposed anomaly detection and foresight response strategy are explained in Sect. 3. Simulation results and performance evaluation of our approach are reported in Sect. 4. Finally, conclusions are given in Sect. 5.

## 2 Related work

In this section, we review related work using similar methods in the field of anomaly intrusion detection in wireless networks. In general, the anomaly detection techniques can be divided into four categories: rule based methods, statistical techniques, machine learning and data mining approaches.

In rule-based anomaly detection, the detector uses pre-defined rules to classify traffic as normal or anomalies. Silva et al. [18] have proposed a flexible rule-based detection scheme, in which a wide range of rules are available for a variety of application scenarios. Their approach is able to against attacks including message delay, repetition, wormhole, data alteration, jamming, message negligence, black-hole and selective forwarding. Another rule-based detection scheme is developed by

Ioannis et al. [19] that concerned with the packet dropping rate. They used a cooperative decision making approach to detect black-hole and selective forwarding attacks, requiring only small amounts of communication and computational resources. Karapistoli et al. [20] proposed a rule-based detection engine that accurately analyzes data packets to detect signs of sensor network anomalies. Their presented algorithm, named ADLU, has dedicated procedures for secure cluster formation, periodic re-clustering, and efficient cluster member monitoring. Since a lot of rules are available, the rule-based anomaly detection schemes are effective against many security issues if appropriate rules are running. But, without a security expert to suggest the appropriate use of rules, these schemes can be inefficient.

In statistical techniques used for anomaly detection in WSNs, the underlying assumption is that the density distribution of the data points being analyzed for anomalies is known a priori (e.g., a Gaussian distribution). The parameters of the distribution are first estimated, and then anomalies are flagged as those data points with low likelihood given that distribution [10]. Palpanas et al. [21] have proposed a statistical technique for distributed deviation detection in the environment of sensor networks. They tried to find those values that deviate significantly from the norm to identify faulty sensors, and to filter spurious reports from different sensors. Ngai et al. [22] have presented a multivariate statistical technique based on a Chi square test to detect sinkhole attacks in a wireless sensor network. Their algorithm consists of two steps. At first, it locates a list of suspected nodes by checking data consistency, and then identifies the intruder in the list through analyzing the network flow information. The nodes affected by the intrusion are detected by testing for anomalies in the data received at the base station using the proposed test statistic. Depending on the packet arrival process, another anomaly detection scheme is proposed by Onat and Miri [23]. In their approach the standard deviation of packet arrival intervals during a specified time period is trained as the normal profile for identifying anomaly. Each sensor node maintains the normal traffic profile on its one-hop neighbor nodes. Li et al. [24] proposed a statistical distribution-based scheme for intrusion detection in wireless sensor networks. They first partitioned the sensor nodes in a network into a number of groups such that the nodes in a group are physically close to each other and sense the similar observation. And then they adopted the Mahalanobis distance measurement and the OGK estimators in the intrusion detection algorithm to take into account the inter-attribute dependencies of multidimensional observed values and ensure a high breakdown with some missing data at a lower computational cost. The statistical techniques are just suitable when the underlying type of distribution of the data is well known [10].

Recently there has been much interest in applying machine learning and data mining approaches for anomaly detection problem in WSN [25–34]. Wang et al. [34] proposed a multi-agents-based detection scheme, by combination of self-organizing map (SOM) neural network algorithm and K-means clustering algorithm. Four kinds of agents including sentry, analysis, response, and management are attached to each node over the network. Each node executes different operations of detection due to its role. In this scheme, the cluster headers are responsible for monitoring all common member nodes in the cluster, while the common member nodes are responsible for monitoring the cluster headers. Ahmadi and Abadi [33] have presented a PCA-based centralized approach, called PCACID, for anomaly detection in WSNs. They partition a WSN into groups of sensor nodes. In each group, some nodes are selected as monitor nodes. In PCACID, every monitor node independently establishes a profile of its own normal network traffic using PCA and uses it to detect anomalous network traffic. Rajasegarar et al. [35] proposed a distributed, non-parametric anomaly detection algorithm that identifies anomalous measurements at nodes based on the data clustering. They use a hyper-spherical clustering algorithm and k-nearest neighbor scheme to collaboratively detect anomalies in wireless sensor network data. In another paper, Rajasegarar et al. [36, 37] proposed a distributed anomaly detection approach based on a one-class SVM, for wireless sensor networks. They have formulated a centered hyper-ellipsoidal SVM (CESVM) scheme to achieve high detection accuracy. The CESVM has limited scope for distributed implementation in sensor networks. Also they have proposed a quarter-sphere SVM (QSSVM), as a special case of the CESVM to perform distributed anomaly detection. In general, machine learning and data mining schemes can provide high detection accuracy through the other approaches. However, the high computational complexity and the lack of applying a proper response strategy to resist any kind of anomaly, are the main shortcomings of the mentioned methods.

Therefore, in this paper, we propose a light-weight data mining approach for anomaly detection in WSNs. For this purpose, we modify the SVDD method, as a well-known one-class classification approach, to increase its accuracy and reduce its computational complexity. After detecting the anomalies, the next goal is to eliminate the effects of anomalies as much as possible in order to increasing the lifetime of WSNs. So, we present a foresight response

strategy to resist the intentional, unintentional and false anomalies in WSNs.

## 3 Proposed methods

The support vector data description was presented by Tax and Duin [16, 38] and again in Tax and Duin [39] with extensions and a more thorough treatment. The SVDD is a one-class classification method that estimates the distributional support of a dataset. A flexible closed boundary function is used to separate trustworthy data on the inside from outliers on the outside. The basic idea of SVDD is to find a minimum hyper-sphere containing all the objective samples and none of the nonobjective samples. However, one of the major disadvantages of the SVDD method is that it cannot tolerate the outliers existed in the training set. For example, Fig. 1 shows a synthetic dataset with two noisy samples. Even with proper parameters, the SVDD cannot relinquish the outliers and they destroy the actual boundary of target object. In the following section, we propose a modified version of the SVDD to identify outliers in the training set.

### 3.1 Fuzzy constraint SVDD (FCSVDD)

The basic idea of FCSVDD is to find a minimum hyper-sphere with fuzzy constrains. The hyper-sphere is specified by its center $a$ and its radius $R$. The data description is achieved by minimizing the error function:

$$F(R,a) = R^2, \tag{1}$$

$$s.t. \quad \|x_i - a\|^2 \lesssim R^2, \quad \forall i. \tag{2}$$

The symbol $\lesssim$ means that we like to permit some violations in the satisfaction of the constraints. In order to the flexibility of the FCSVDD method, the distance of each training sample $x_i$ to the center of the sphere should not be strictly smaller than $R^2$. However, large distances should be penalized. Therefore, after introducing slack variables $\xi_i \geq 0$ the minimization problem becomes:

$$F(R,a) = R^2 + C\sum_i \xi_i, \tag{3}$$

$$s.t. \quad \|x_i - a\|^2 \lesssim R^2 + \xi_i, \quad \forall i. \tag{4}$$

The parameter C gives the tradeoff between the volume of the description and the errors. Note that the slack variables and C cannot tolerate noisy samples. They are tuned by system without any information about importance of samples. Therefore, we need a data description method which considers this aspect.

We can easily show that each fuzzy inequality can transform to a non-fuzzy inequality. According to Fig. 2, if A is fuzzy less than B then:

$$A \lesssim B \Rightarrow A \leq B + d(1-\alpha). \tag{5}$$

Now we transform the fuzzy constraints in Eq. (4) to a non-fuzzy inequality.

$$\left\|x_i - a^2\right\| \lesssim R^2 + \xi_i, \quad \forall i. \tag{6}$$

$$\left\|x_i - a^2\right\| \leq \left(R^2 + \xi_i\right) + d_i(1-\alpha), \quad \forall i. \tag{7}$$

We have two new concepts in Eq. (7). They are user defined parameters which indicate the importance of samples. The weight of each sample is defined by $d_i$ and the uncertainty of this weight is shown by $\alpha$. The value of $d_i$ can be defined by user or some automatic methods (such as inverse of distance of each sample to the center of training set). So we have the following error function:

$$F(R,a) = R^2 + C\sum_i \xi_i, \tag{8}$$

$$s.t. \quad \|x_{i-a}\|^2 \leq (R^2 + \xi_i) + d_i(1-\alpha), \quad \forall i. \tag{9}$$

In order to solve this problem the constraints should be incorporated into the error function by introducing Lagrange multipliers and constructing the Lagrangian function.

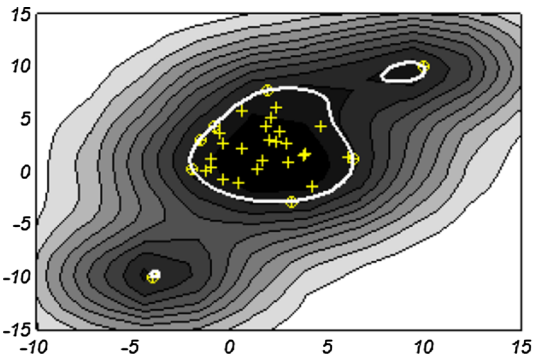


**Fig. 1** SVDD boundary for a synthetic dataset with two outliers in the training set

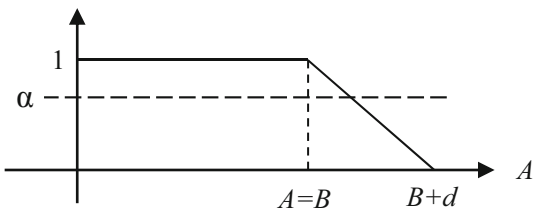


**Fig. 2** Presentation of a fuzzy inequality (A$\lesssim$B)

$$L(R, a, \alpha_i, \gamma_i, \xi_i) = R^2 + C\sum_i \xi_i - \sum_i \lambda_i\left\{\left(R^2 + \xi_i\right) + d_i(1-\alpha) - \left(\|x_i\|^2 - 2a.x_i + \|a\|^2\right)\right\} - \sum_i \gamma_i\xi_i, \tag{10}$$

with the Lagrange multipliers $\lambda_i \geq 0$ and $\gamma_i \geq 0$. Setting partial derivatives to 0 gives these constraints:

$$\frac{\partial L}{\partial R} = 0 : \sum_i \lambda_i = 1, \tag{11}$$

$$\frac{\partial L}{\partial a} = 0 : a = \frac{\sum_i \lambda_i x_i}{\sum_i \lambda_i} = \sum_i \lambda_i x_i, \tag{12}$$

$$\frac{\partial L}{\partial \xi_i} = 0 : C - \lambda_i - \gamma_i = 0. \tag{13}$$

From the above equations and the fact that the Lagrange multipliers are not all negative, when we add the condition $0 < \lambda_i < C$, Lagrange multipliers $\gamma_i$, can be safely removed. So the problem can be transformed into maximizing the following function L with respect to the Lagrange multipliers $\lambda_i$ :

$$\max \sum_i \lambda_i(x_i \cdot x_i) - \sum_{i,j} \lambda_i\lambda_j(x_i \cdot x_j) - \sum_i \lambda_i d_i(1-\alpha), \tag{14}$$

$$\text{s.t.} \quad 0 < \lambda_i < C. \tag{15}$$

Similar to the SVDD method, for more flexible boundaries, inner products of samples $(x_i \cdot x_j)$ can be replaced by a kernel function $K(x_i \cdot x_j)$, where $K(x_i \cdot x_j)$ satisfies Mercer's theorem [40]. This implicitly, maps samples into a nonlinear space to obtain a more tight and nonlinear boundary. In this context, the FCSVDD problem can be expressed as:

$$\max \sum_i \lambda_i K(x_i \cdot x_i) - \sum_{i,j} \lambda_i\lambda_j K(x_i \cdot x_j) - \sum_i \lambda_i d_i(1-\alpha), \tag{16}$$

$$\text{s.t} \quad 0 < \lambda_i < C. \tag{17}$$

Note that from Eq. (12), the center of the sphere is a linear combination of the training samples. Only those training samples $x_i$ which satisfy Eq. (7) by equality are needed to generate the description since their coefficients are not zero. Therefore these samples are called Support Vectors. The radius can be computed using any of the support vectors:

$$R^2 = (x_k \cdot x_k) - 2\sum_i \lambda_i(x_i \cdot x_k) + \sum_{i,j} \lambda_i\lambda_j(x_i \cdot x_j) - d_k(1-\alpha). \tag{18}$$

To judge whether a test sample $x_z$ is in the target class or not, its distance to the center of sphere is computed and compared with R, if satisfies Eq. (9), it will be accepted, and otherwise, rejected.

$$\|x_z - a^2\| = (x_z \cdot x_z) - 2\sum_i \lambda_i(x_z \cdot x_i) - \sum_{i,j} \lambda_i\lambda_j(x_i \cdot x_j) \leq R^2 + d_z(1-\alpha). \tag{19}$$

Several kernel functions have been proposed for the support vector classifiers. Not all kernel functions are equally useful for the FCSVDD. It has been demonstrated that using the Gaussian kernel results in tighter description. The results of FCSVDD for different value of $\alpha$ are shown in Fig. 3. We assume that the weights of outlier samples are half of the other samples. If $\alpha = 1$ or all of the samples have the same weights, the results of FCSVDD is the same as SVDD method.

### 3.2 Linear Programming based FCSVDD (LP-FCSVDD)

Here we address the computational challenge of FCSVDD for using in WSN. FCSVDD requires the solution of a computationally intensive quadratic programming approach which is not applicable in WSN. We solved this problem by formulating a centered hyper-spherical scheme, which enables us to use a linear programming approach. Consider the following minimization problem:

$$F(R, a) = R^2 + C\sum_i \xi_i, \tag{20}$$

$$s.t. \|x_i\|^2 \lesssim R^2 + \xi_i, \quad \forall i. \qquad \xi_i \geq 0, \quad \forall i. \tag{21}$$

As mentioned above, we can transform the fuzzy constraints in Eq. (21) to a non-fuzzy inequality. Hence the minimization problem becomes:

$$F(R, a) = R^2 + C\sum_i \xi_i, \tag{22}$$

$$s.t. \quad \|x_i\|^2 \leq \left(R^2 + \xi_i\right) + d_i(1-\alpha), \quad \forall i. \qquad \xi_i \geq 0, \quad \forall i. \tag{23}$$

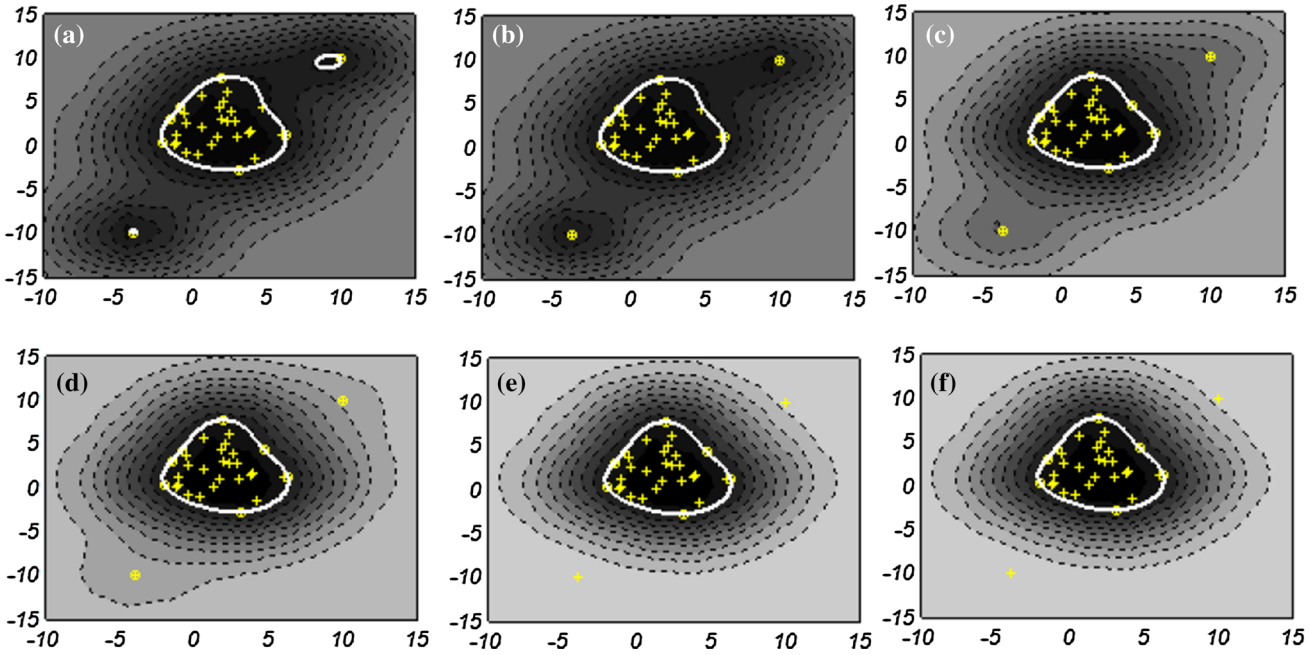


**Fig. 3** FCSVDD boundaries for a synthetic dataset with two outliers in the training set. The effect of outlier samples will be eliminated by decreasing the uncertainty value ($\alpha$). If $\alpha = 1$ the results of FCSVDD is the same as SVDD. **a** $\alpha = 1$, **b** $\alpha = 0.9$, **c** $\alpha = 0.7$, **d** $\alpha = 0.5$, **e** $\alpha = 0.3$, **f** $\alpha = 0.1$

The constraints can be incorporated into the error function by introducing Lagrange multipliers and constructing the Lagrangian.

$$\begin{aligned} L(R, \lambda_i, \gamma_i, \xi_i) = R^2 + C\sum_i \xi_i \\ - \sum_i \lambda_i \left\{ \left(R^2 + \xi_i\right) + d_i(1-\alpha) - \left\|x_i^2\right\| \right\} \\ - \sum_i \gamma_i \xi_i, \end{aligned} \tag{24}$$

With the Lagrange multipliers $\lambda_i \geq 0$ and $\gamma_i \geq 0$. Setting partial derivatives to 0 gives these constraints:

$$\frac{\partial L}{\partial R} = 0 : \sum_i \lambda_i = 1, \tag{25}$$

$$\frac{\partial L}{\partial \xi_i} = 0 : C - \lambda_i - \gamma_i = 0. \tag{26}$$

Therefore, the problem can be transformed into maximizing the following programming problem:

$$\max \sum_i \lambda_i (x_i \cdot x_i) - \sum_i \lambda_i d_i (1-\alpha), \tag{27}$$

$$\begin{aligned} \text{s.t} \sum_i \lambda_i = 1, \\ 0 < \lambda_i < C. \end{aligned} \tag{28}$$

Using the kernel trick, the inner product can be replaced by a kernel function $K(x_i \cdot x_i)$, and the following optimization problem is obtained:

$$\max \sum_i \lambda_i K(x_i \cdot x_i) - \sum_i \lambda_i d_i (1-\alpha) \tag{29}$$

$$\begin{aligned} \text{s.t.} \sum_i \lambda_i = 1, \\ 0 < \lambda_i < C. \end{aligned} \tag{30}$$

This dual problem is a linear optimization problem, so the $\lambda_i$ can be obtained using widely available linear optimization techniques. Compared to the SVDD and FCSVDD formulations, which require solving a quadratic optimization problem, this formulation with linear optimization is advantageous in terms of its computations.

Further, in Eq. (29) the solution is affected only by the norms of the non-linear mapping of data vectors using the kernel $K(x_i \cdot x_i)$. This creates a problem for the application of this approach with distance-based kernels such as the RBF kernel, as the norms of the kernels are now equal for all data vectors [36, 41]. In order to solve this problem, the centered image vectors can be computed by:

$$\tilde{\Phi}(x_i) = \Phi(x_i) - \frac{1}{n}\sum_{i=1}^{n}\Phi(x_i) \tag{31}$$

In other words, the mapped vectors are subtracted from the mean in the feature space. The dot product $\tilde{K} = \left(\tilde{\Phi}(x_i) \cdot \tilde{\Phi}(x_i)\right)$ of the centered image vectors can be obtained in terms of kernel $K = (\Phi(x_i) \cdot \Phi(x_i))$ as follows:

$$\tilde{K} = K - 1_n K - K 1_n + 1_n K 1_n, \tag{32}$$

where $1_n$ is an $n \times n$ matrix with all values equal to $1/n$. $\tilde{K}$ is called the centered kernel matrix [40]. Once the image vectors are centered, the norms of the kernels are no longer equal. Hence the dual problem might be solved now. Figure 4 shows the decision boundary which is generated by LP-FCSVDD method around the target class with cross markers. The LP-FCSVDD method can produce a very tight description for the target class similar to a quadratic one-class classifier. Moreover, it can detect the outliers and eliminate them from the target class boundaries.

### 3.3 Distributed anomaly detection and foresight response strategy in WSN

In this section, the LP-FCSVDD method is applied to resist anomalies in WSN. The sensor nodes can be organized as a flat network or a clustered network. In the flat architecture, all sensor nodes transmit their own data and relay data for other nodes to the sink. In the clustered architecture, adjacent nodes are organized as a cluster; a head is elected for each cluster. Sensor nodes that belong to the same cluster can only send or relay data to their cluster head. The cluster head then relays the data to the sink via a long-haul communication link [42].

Here we want to find local and global anomalies in the data measurements collected by each node in the network. Local anomalies are anomalous measurements in a sensor node's own (local) data measurements, where only the measurements at the same node are used as a basis for comparison. However, we are also interested in cases where the majority of measurements at a sensor node are anomalous in comparison to other nodes in the network. These global anomalies are anomalous measurements in the union of the measurements collected from multiple sensor nodes in the network. Local anomalies can be detected by considering local measurements of a sensor node without incurring any energy intensive communication overhead in the network. However, detecting global anomalies requires all the measurements from multiple nodes to be considered. Centralized schemes perform this by communicating all the sensor measurements to a central node to detect global anomalies. However, communicating all the measurements in the network is an energy-intensive operation, which affects the lifetime of the network. Hence, we require energy efficient distributed approaches to detect these anomalies in sensor networks. This motivates us to propose a distributed anomaly detection approach using LP-FCSVDD that can efficiently and effectively detect anomalies (local and global) in data measurements collected by sensor nodes in wireless sensor networks [37].

After detecting anomalies, a proper response strategy should be trigger to resist the anomaly effects. Here, we introduce a foresight response model for intentional, unintentional and false anomalies (see Fig. 5). If the anomaly is caused by a faulty sensor, we can estimate the

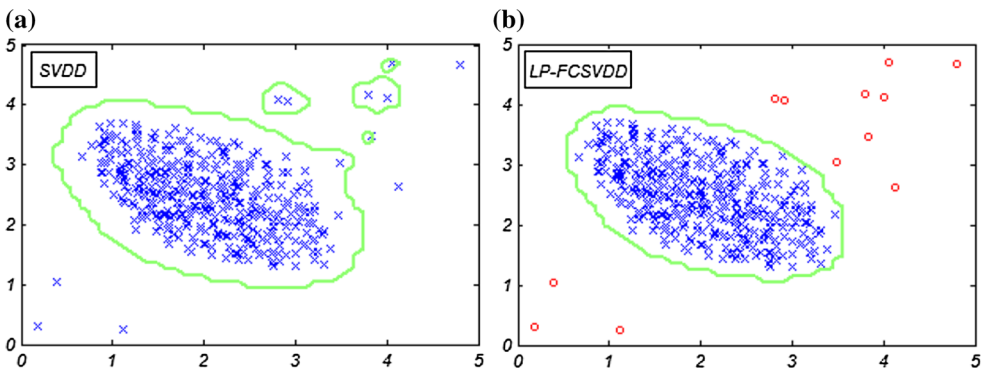


**Fig. 4** LP-FCSVDD in comparison with the standard SVDD. **a** SVDD could sometimes generate such a loose decision boundary, when some noisy samples exist in the training set. **b** The LP-FCSVDD method detects the outliers exist in the training set and generates much better decision boundaries

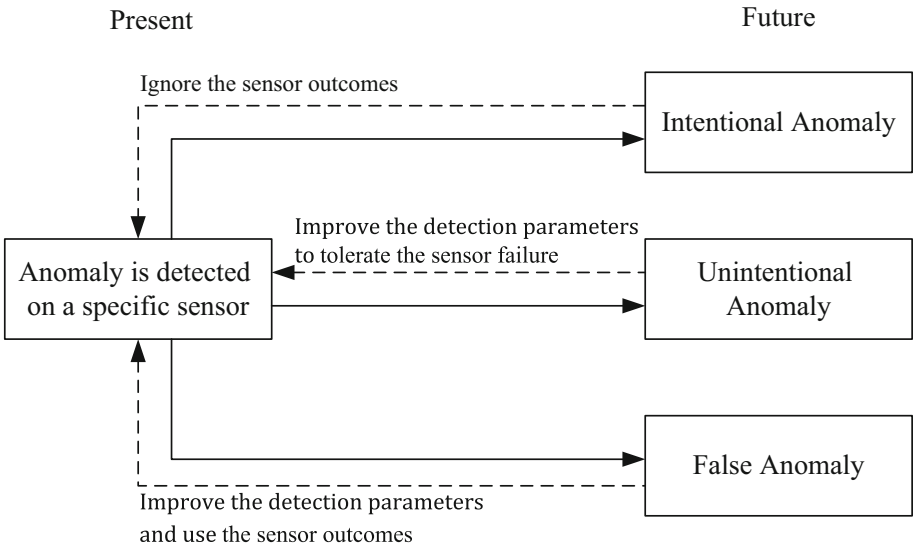


**Fig. 5** The possible future and foresight response strategy, when an anomaly is detected

sensor error to eliminate the negative impact in global radius computation. But when the error is intentional, ignoring the compromised sensor outcomes is the best solution for response. Also, when the detection process leads to false anomalies, we should improve the detection parameters and use the sensor outcomes. Hence, selecting the proper response in each situation minimizes the cost of response in future.

First, we consider a clustered based wireless sensor network, deployed to monitor an area of interest. In the distributed anomaly detection process, each sensor $(S_j)$ runs the LP-FCSVDD method on its own data to find the local radius $(R_j^{(t)})$ as a decision boundary in time-window (t). The cluster head $(S_0)$ collects these obtained local radii from all sensors in the cluster and computes the mean of them as the global radius $(R_g^{(t)})$. The cluster head sends the global radius to each sensor node in its cluster except the compromised nodes. After detecting the anomalies, we should trigger the foresight response strategy to mitigate the intentional, unintentional and false anomalies. For this purpose, we compute a trust value between 0 (untrusted) and 1 (fully trusted) for each sensor node.

According to the given trust value, we should apply one of the following response strategies:

- The sensor's trust value is very low (its value is close to zero): we should ignore the sensor outcomes.
- The sensor's trust value is very high (its value is close to one): we should improve the detection parameters and use the sensor outcomes.
- The sensor's trust value is medium (its value is around 0.5): we should estimate the sensor error to eliminate the negative impact in global radius computation.

Algorithm 1 presents the distributed anomaly detection and foresight response strategy using LP-FCSVDD to resist the local and global anomalies in a clustered based WSN. We use a similar method for a wireless sensor network with flat topology. Each sensor runs the LP-FCSVDD method on its own data to find the local radius and broadcast it to its neighbors. When a sensor receives the all of the neighbor's radii, computes the global radius, the trust value of its neighbors and its own trust value. The neighbors which have very low trust value are tagged as compromised nodes and ignored in global radius computation. Also, we can compute the weight of each sensor according to its own trust value. Then each sensor runs the LP-FCSVDD method again with the new samples weight. Algorithm 2 presents the distributed anomaly detection and foresight response strategy using LP-FCSVDD to resist the local and global anomalies in a flat WSN. Figure 6 shows the process of scheme according to the mentioned algorithms for distributed anomaly detection and foresight response strategy in a clustered and flat WSN.

**Algorithm 1: Distributed anomaly detection and foresight response strategy in a clustered WSN.**

**Initialization**

- $S = \{S_1, S_2, \ldots, S_n\}$ is the set of sensors in a cluster.
- In the first run $d_i = -\|x_i - \mu\|/\max(d_i)$ for each sensor data.
- The weight of each sensor is 0 in the first iteration ($W_j^{(1)} = 0$).
- The trust value of each sensor is 0.5 in the first iteration ($T_j^{(1)} = 0.5$).
- The global radius is not definable in the first iteration ($R_g^{(1)} = N/A$).
- There is no compromised sensor in the first iteration ($S_C = \emptyset$).

Do for each iteration (t=1, 2, ...):

**Distributed anomaly detection**

d1) Each sensor ($S_j$) runs LP-FCSVDD on its own data to find the local radius ($R_j^{(t)}$) and local outliers.

d2) Each sensor ($S_j$) sends its local radius ($R_j^{(t)}$) to the cluster head.

d3) The cluster head collects the radii information from its cluster nodes and computes the global radius $R_g^{(t)} = \frac{\sum_J R_J^{(t)}}{n}$ where $J = \{j \mid S_j \in S - S_C\}$.

d4) The cluster head sends the global radius ($R_g^{(t)}$) to each sensor node in its cluster except the compromised nodes.

d5) Each sensor ($S_j$) uses the global radius ($R_g^{(t)}$) to detect the global anomalies.

**Foresight response strategy**

r1) The cluster head computes the trust value of each sensor by $T_j^{(t+1)} = \propto T_j^{(t)} + (1-\propto)\tau_j^{(t)}$ where:
- $\tau_j^{(t)} = 2\,R_g^{(t)} / \left(R_g^{(t)} + R_j^{(t)}\right)$ if $R_j^{(t)} \geq R_g^{(t)}$
- $\tau_j^{(t)} = 2\,R_j^{(t)} / \left(R_g^{(t)} + R_j^{(t)}\right)$ if $R_j^{(t)} < R_g^{(t)}$
- $0 < \alpha < 1$

r2) According to the given trust value $T_j^{(t+1)}$, one of the following would be occurred:
- $T_j^{(t+1)}$ is very low (its value is close to zero): The cluster head introduces the sensor ($S_j$) as a compromised sensor and $S_C = S_C \cup \{S_i\}$. In this situation we should ignore the sensor outcomes.
- $T_j^{(t+1)}$ is medium (its value is around 0.5): The cluster head computes the difference between local radius and global ones to allocate a proper weight to each sensor node ($W_j^{(t)} = (R_g^{(t)} - R_j^{(t)})$). These weights are used in the LP-FCSVDD method to eliminate the effect of faulty sensors in global radius computation.
- $T_j^{(t+1)}$ is very high (its value is close to one): The cluster head sets $W_j^{(t)} = 0$ and applies the sensor outcomes in global radius computation.

r3) The cluster head sends the sensor weight ($W_j^{(t)}$) to each sensor node in its cluster except the compromised nodes.

r4) Each sensor runs the LP-FCSVDD method again with the new samples weight $d_i = (-\|x_i - \mu\|/\max(d_i)) + W_j^{(t)}$ to find the new local radius ($R_j^{(t+1)}$) and local anomalies with more precision.

**Algorithm 2: Distributed Anomaly Detection and foresight response strategy in a flat WSN.**

Do for each sensor ($S_k$) (k=1, 2, ..., n):

**Initialization**

- $S_N = \{S_1, S_2, \ldots, S_m\}$ is the set of sensors which are neighbors of ($S_k$).
- In the first run $d_i = -\|x_i - \mu\|/\max(d_i)$ for each sensor data.
- The weight of sensor ($S_k$) is 0 in the first iteration ($W_k^{(1)} = 0$).
- The trust value of each sensor ($S_l$) which is neighbor of ($S_k$) is 0.5 in the first iteration ($T_l^{(1)} = 0.5$).
- The global radius is not definable in the first iteration ($R_g^{(1)} = N/A$).
- There is no compromised sensor around ($S_k$) in the first iteration ($S_C = \emptyset$).

Do for each iteration (t=1, 2, ...):

**Distributed anomaly detection**

d1) Each sensor ($S_l$) which is neighbor of ($S_k$) runs LP-FCSVDD on its own data to find the local radius ($R_l^{(t)}$) and local outliers.

d2) Each sensor ($S_l$) which is neighbor of ($S_k$) broadcasts its local radius to its neighbors.

d3) Sensor ($S_k$) collects the radii information from its neighbors and computes the global radius $R_g^{(t)} = \frac{\sum_L R_L^{(t)}}{m}$ where $L = \{l \mid S_l \in S_N - S_C\}$.

d4) Sensor ($S_k$) uses the global radius ($R_g^{(t)}$) to detect the global anomalies.

**Foresight response strategy**

r1) Sensor ($S_k$) computes the trust value of each neighbor by $T_l^{(t+1)} = \propto T_l^{(t)} + (1-\propto)\tau_l^{(t)}$ where:
- $\tau_l^{(t)} = 2\,R_g^{(t)} / \left(R_g^{(t)} + R_l^{(t)}\right)$ if $R_l^{(t)} \geq R_g^{(t)}$
- $\tau_l^{(t)} = 2\,R_l^{(t)} / \left(R_g^{(t)} + R_l^{(t)}\right)$ if $R_l^{(t)} < R_g^{(t)}$
- $0 < \alpha < 1$

r2) If $T_l^{(t+1)}$ is very low (its value is close to zero) then the sensor ($S_l$) is tagged as a compromised neighbor and $S_C = S_C \cup \{S_l\}$. In this situation we should ignore the sensor outcomes.

r3) Sensor ($S_k$) computes the trust value of itself by $T_k^{(t+1)} = \propto T_k^{(t)} + (1-\propto)\tau_k^{(t)}$ where:
- $\tau_k^{(t)} = 2\,R_g^{(t)} / \left(R_g^{(t)} + R_k^{(t)}\right)$ *if* $R_k^{(t)} \geq R_g^{(t)}$
- $\tau_k^{(t)} = 2\,R_k^{(t)} / \left(R_g^{(t)} + R_k^{(t)}\right)$ *if* $R_k^{(t)} < R_g^{(t)}$
- $0 < \alpha < 1$

r4) According to the given trust value $T_k^{(t+1)}$, one of the following would be occurred:
I. $T_k^{(t+1)}$ is medium (its value is around 0.5): Sensor ($S_k$) computes the difference between its local radius and global ones to allocate a proper weight to itself ($W_k^{(t)} = (R_g^{(t)} - R_k^{(t)})$). This weight is used in the LP-FCSVDD method to eliminate the effect of error in global radius computation.
II. $T_k^{(t+1)}$ is very high (its value is close to one): Sensor ($S_k$) sets $W_k^{(t)} = 0$.

r5) Sensor ($S_k$) runs the LP-FCSVDD method again with the new samples weight $d_i = (-\|x_i - \mu\|/\max(d_i)) + W_k^{(t)}$ to find the new local radius ($R_k^{(t+1)}$) and local anomalies with more precision.

## 3.4 Complexity analysis

This section analyzes the complexity of the proposed anomaly detection and response strategy. Suppose that s is the number of sensor nodes in the network, n is the number of data vectors at a sensor node, and p is the number of dimensions in a data vector. The LP-FCSVDD method involves the computation of a kernel matrix K with a computational complexity of $O(n^2)$ and solving a linear optimization problem. Several approaches have been proposed for the linear programming in the literature. Khachiyan [43] was the first to show that the linear programming problem could be solved in time polynomial in the length of the binary encoding of the input. Karmarkar's original method requires a total complexity of $O(n^4L)$ arithmetic operations where n is the number of variables and L is the length of the input data. Subsequent results have reduced this to a total complexity of $O(n^3L)$ [44]. On the other hand, it is demonstrated that the linear programming problem with d variables and m constraints can be solved in $O(m)$ time when d is fixed [45]. According to Eq. (21), in our proposed linear programing model the

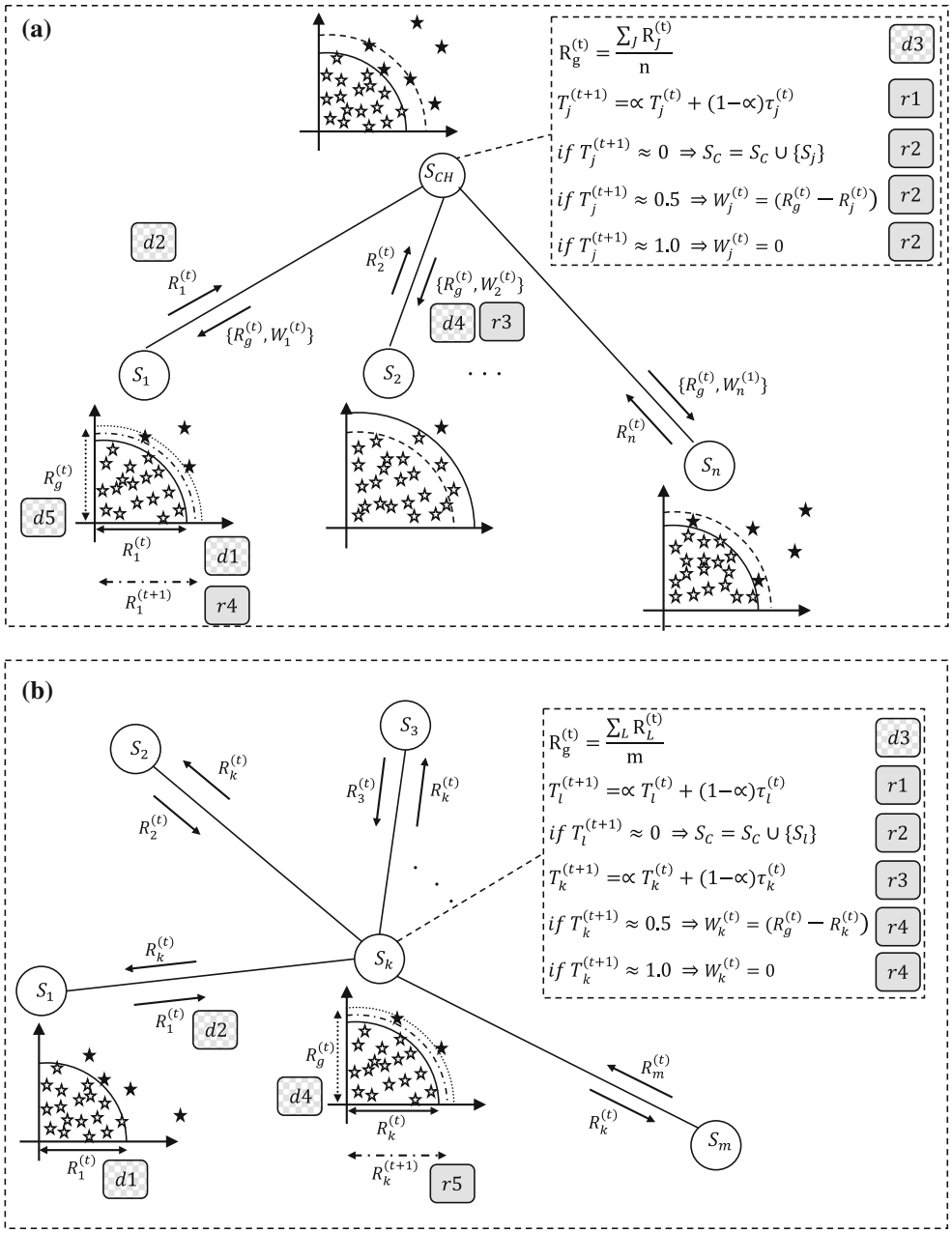


**Fig. 6** Distributed anomaly detection and response process using LP-FCSVDD method in **a** clustered WSN **b** flat WSN

number of constraints (m) is equal to the number of data vectors at a sensor node (n), so the total computational complexity of LP-FCSVDD is $O(n) + O(n^2)$.

In the proposed distributed scheme, each sensor node runs LP-FCSVDD on its own data with time complexity $O(n) + O(n^2)$ and the memory complexity of O(np). Furthermore, in the foresight response strategy, the trust value of each sensor is computed by time complexity O(n) and the memory complexity of O(n). So the total time complexity of proposed approach is $2O(n) + O(n^2)$ and the total memory complexity is $O(np) + O(n)$. It also requires communication of the radius information and the sensor weights with a communication complexity of O(1). In the centralized approach, each sensor node sends its local data into the central node (i.e., base station). So, the communication of the whole set of data measurements to a central node with a communication complexity of O(np) per link leads to a total complexity of O(snp). The central node runs LP-FCSVDD on the collected data with a maximum computational complexity of $O(s^2n^2)$, and also the trust value of each sensor is computed by time complexity O(sn). The memory complexity at the central node is $O(snp) + O(sn)$, which includes the memory complexity required to keep data vectors and the trust value of sensors.

**Table 1** Comparison of various complexities for the distributed and centralized anomaly detection schemes using LP-FCSVDD

| Scheme | Computational complexity | Memory complexity | Communication complexity (per link) |
|---|---|---|---|
| Distributed | $2O(n) + O(n^2)$ (per each node) | $O(np) + O(n)$ (per each node) | $O(1)$ |
| Centralized | $O(sn) + O(s^2n^2)$ (at central node) | $O(snp) + O(sn)$ (at central node) | $O(np)$ |

In this table, $s$ is the number of sensor nodes in the network, $n$ is the number of data vectors at a sensor node, and $p$ is the number of dimensions in a data vector

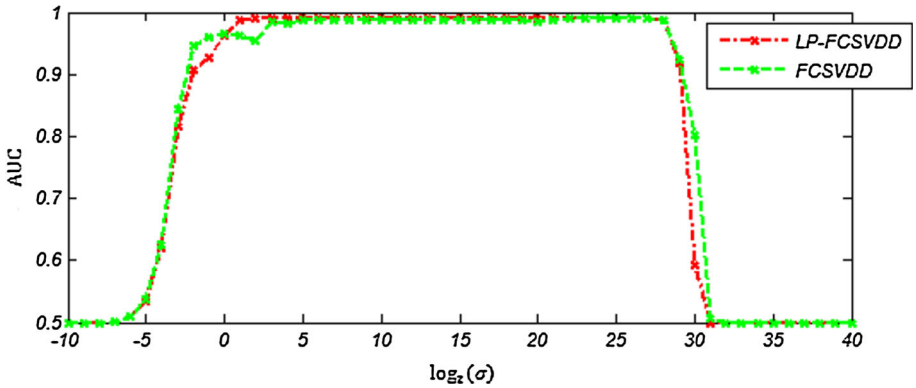


**Fig. 7** AUC for LP-FCSVDD and FCSVDD methods in the synthetic dataset by using the RBF kernel ($\sigma$ is the kernel parameter)

**Table 2** SVDD and LP-FCSVDD results for the synthetic dataset in centeralized scheme

| Scheme | Classification method | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|---|
| Centralized | LP-FCSVDD | 96.7391 | 6.5217 | 100.0000 | 94.0474 | 96.8900 |
| | SVDD | 93.4150 | 12.8842 | 99.7143 | 89.4701 | 94.0845 |

In Table 1 the complexity of proposed anomaly detection and foresight response strategy is summarized in both centralized and distributed schemes.

## 4 Experiments

In this section, we want to evaluate the performance of our proposed methods by applying them to real and synthetic datasets. The synthetic dataset is similar to that used by Rajasegarar et al. [37]. It has two features, each generated from a normal distribution with mean 1 and standard deviation 3. Noise samples are generated by uniformly distributed data around the normal samples. The synthetic dataset consists of 15 sensor nodes and comprises 1,575 data vectors of two features, including 75 outliers.

In the first experiment, we apply the proposed methods to the synthetic dataset. Figure 7 shows the area under the ROC curve (AUC) for LP-FCSVDD and FCSVDD, which have linear and quadratic complexities respectively. Here, we use the RBF kernel with different value for the $\sigma$ parameter in the range $2^{-10}$–$2^{40}$, in exponential intervals. This experiment shows that the proposed linear method has the same performance, compared to the quadratic one.

In the second experiment, we applied SVDD and LP-FCSVDD methods for the synthetic dataset in centralized scheme. The false positive rate (FPR) is then computed as the percentage ratio between the false positives and the actual normal measurements. Also, the true positive rate (TPR) is computed as the percentage ratio between the true positives and the actual anomalous measurements. Table 2 and Fig. 8 show the obtained results for SVDD and LP-FCSVDD methods in the synthetic dataset. In this experiment, we get the AUC value of 0.9964 and 0.9742 for LP-FCSVDD and SVDD methods respectively.

Now, we evaluate the proposed methods with two real WSN datasets namely the IBRL and GDI datasets. The IBRL dataset contains information about data collected from 54 sensors deployed in the Intel Berkeley Research Lab, between February 28th and April 5th, 2004. Mica2Dot sensors with weatherboards collected time-stamped topology information, along with humidity, temperature, light and voltage values once every 31 s. The data was collected using the TinyDB in-network query processing system, built on the TinyOS platform [46]. The sensors were arranged in the lab, according to the diagram shown in Fig. 9. We considered a part of this dataset formed from measurements collected by 5 sensor nodes, namely, the nodes 1, 2, 3, 33 and 35 which are closed to each other. A 24-h

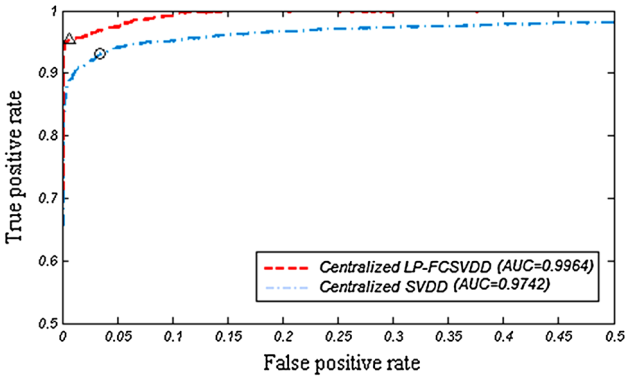


**Fig. 8** ROC curves for SVDD and LP-FCSVDD methods in the synthetic dataset in centralized scheme

period of data, recorded on March 6, 2004, was used in our evaluation. The sensor's data is divided into 41 time-windows and three attributes are used for each data vector including humidity, temperature, and light measurements.

The other real dataset that we used was the sensor measurements gathered from a deployment of wireless sensors in the Great Duck Island (GDI), Maine, USA [31, 47]. The network monitors the habitat of a sea bird called the Leach's Storm Petrel. This is an example of an outdoor environmental monitoring deployment. Each sensor recorded temperature, humidity, and barometric pressure at 5-min intervals. Five sensor nodes are selected for the evaluation, namely nodes 101, 103, 110, 111 and 129 that are physically close to each other and sense the similar observation. A 24-h period of data recorded on June 18, 2003, was used in our evaluation. Each data vector has three features: humidity, temperature and pressure.

We define two scenarios to evaluate the proposed anomaly detection and foresight response strategies. In the first scenario, some noisy samples are generated randomly and added to the normal data in each time-window. Then, the mentioned anomaly detection approach runs on each time-window. Tables 3 and 4 present the summary results

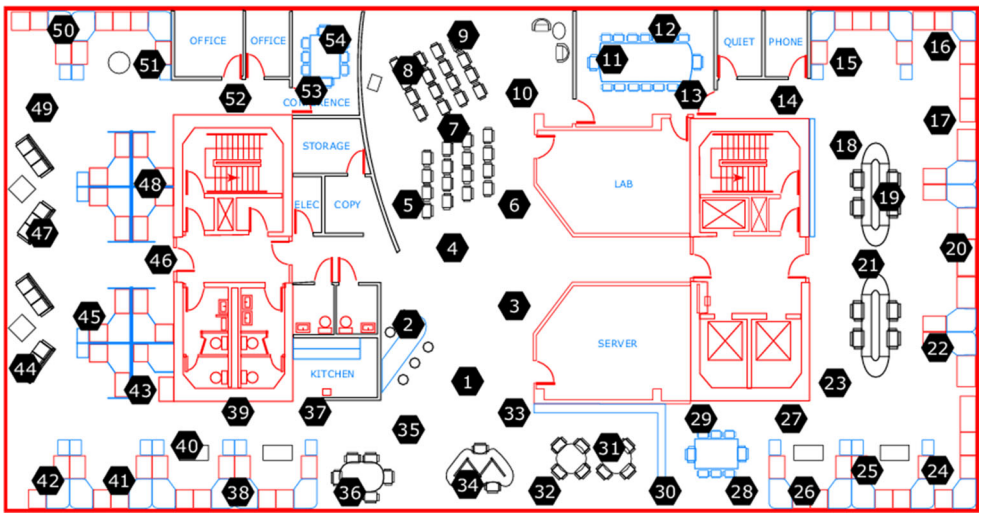


**Fig. 9** The sensors position in the IBRL dataset

**Table 3** SVDD and LP-FCSVDD results for the sensor nodes 1, 2, 3, 33 and 35in the IBRL dataset

| Sensor name | Classification method | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|---|
| 1 | LP-FCSVDD | 97.4549 | 5.0901 | 100.0000 | 95.4139 | 97.5888 |
| | SVDD | 89.2895 | 21.4210 | 100.0000 | 82.7942 | 90.4679 |
| 2 | LP-FCSVDD | 95.6301 | 8.7398 | 100.0000 | 92.2970 | 95.9083 |
| | SVDD | 99.0854 | 1.4228 | 99.5935 | 98.7617 | 99.1267 |
| 3 | LP-FCSVDD | 96.3415 | 7.3171 | 100.0000 | 93.5564 | 96.5755 |
| | SVDD | 90.8802 | 17.8155 | 99.5758 | 85.2332 | 91.7332 |
| 33 | LP-FCSVDD | 95.6446 | 8.7108 | 100.0000 | 93.2449 | 96.2479 |
| | SVDD | 92.3345 | 15.1568 | 99.8258 | 87.7165 | 93.1428 |
| 35 | LP-FCSVDD | 96.6624 | 6.6752 | 100.0000 | 94.1174 | 96.8750 |
| | SVDD | 97.2401 | 5.2632 | 99.7433 | 95.4011 | 97.4168 |
| Central | LP-FCSVDD | 96.2222 | 7.4761 | 99.9205 | 93.2464 | 96.4137 |
| | SVDD | 92.9878 | 3.9502 | 89.9258 | 95.8305 | 92.6388 |

**Table 4** SVDD and LP-FCSVDD results for the sensor nodes 101, 103, 110, 111 and 129 in the GDI dataset

| Sensor name | Classification method | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|---|
| 101 | LP-FCSVDD | 97.1922 | 5.6156 | 100.0000 | 95.0508 | 97.3703 |
| | SVDD | 71.2586 | 57.4828 | 100.0000 | 65.6230 | 78.5932 |
| 103 | LP-FCSVDD | 87.2235 | 21.9959 | 96.4429 | 83.7272 | 89.1236 |
| | SVDD | 81.0273 | 23.5843 | 85.6388 | 85.9068 | 82.4660 |
| 110 | LP-FCSVDD | 94.4632 | 11.0737 | 100.0000 | 91.5238 | 95.2182 |
| | SVDD | 91.4610 | 17.0780 | 100.0000 | 89.2719 | 93.4176 |
| 111 | LP-FCSVDD | 96.4344 | 7.1312 | 100.0000 | 94.4251 | 96.8755 |
| | SVDD | 91.9620 | 16.0759 | 100.0000 | 90.2838 | 93.9771 |
| 129 | LP-FCSVDD | 98.7140 | 2.5720 | 100.0000 | 97.6080 | 98.7607 |
| | SVDD | 68.0666 | 63.8667 | 100.0000 | 63.1568 | 76.7375 |
| Central | LP-FCSVDD | 93.3824 | 3.8235 | 90.5882 | 96.0691 | 93.0828 |
| | SVDD | 85.0980 | 26.1765 | 96.3725 | 79.3609 | 86.7500 |

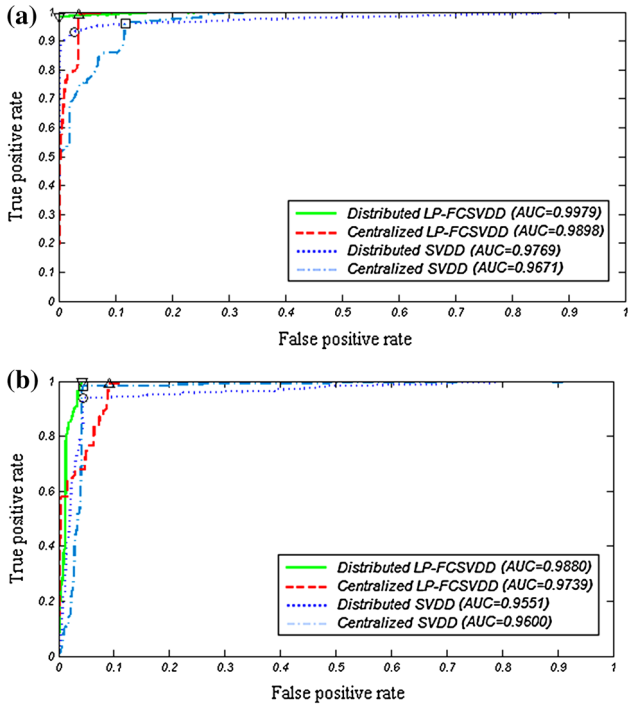


**Fig. 10** ROC curves for SVDD and LP-FCSVDD methods **a** in the IBRL dataset **b** in the GDI dataset

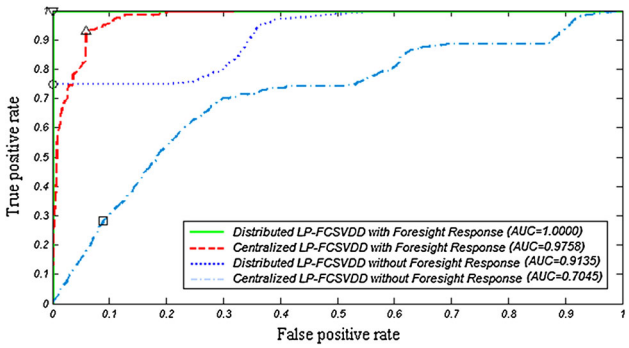


**Fig. 11** ROC curves for LP-FCSVDD with/without applying the foresight response strategy in the IBRL dataset. Suppose that node 1 is a compromised sensor and generates intentional anomalies

of simulation for each sensor node in the IBRL and GDI datasets respectively. The average of each measure for all time-windows is displayed in the results. The detailed results for applying LP-FCSVDD method on sensor node 1 in the IBRL dataset and sensor node 101 in the GDI dataset is described in Tables 7 and 8 of appendix. Figure 10 shows ROC curves for these experiments in both centralized and distributed schemes.

In the second scenario, in addition to injecting some noisy samples, we suppose that there is a compromised sensor which generates intentional anomalies. In this situation we

**Table 5** LP-FCSVDD results with/without applying the foresight response strategy in the IBRL dataset. Suppose that node 1 is a compromised sensor and generates intentional anomalies

| Scheme | Classification method | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|---|
| Distributed | With foresight response | 96.2066 | 7.5868 | 100.0000 | 93.7977 | 96.6185 |
| | Without foresight response | 90.0000 | 20.0000 | 100.0000 | 90.0000 | 93.3333 |
| Centralized | With foresight response | 91.9255 | 10.8696 | 94.0217 | 92.5763 | 93.0134 |
| | Without foresight response | 31.6848 | 9.2391 | 11.9928 | 79.9208 | 20.7454 |

**Table 6** LP-FCSVDD results with/without applying the foresight response strategy in the GDI dataset. Suppose that node 101 is a compromised sensor and generates intentional anomalies

| Scheme | Classification method | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|---|
| Distributed | With foresight response | 93.8216 | 10.1058 | 97.7489 | 91.9499 | 94.3946 |
| | Without foresight response | 87.6406 | 13.7063 | 88.9874 | 88.3351 | 87.9154 |
| Centralized | With foresight response | 93.4580 | 8.7948 | 95.1961 | 93.5617 | 94.2305 |
| | Without foresight response | 72.5490 | 11.5686 | 56.6667 | 83.0740 | 67.2078 |

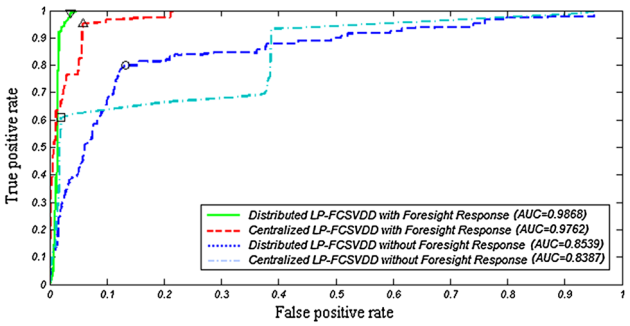


**Fig. 12** ROC curves for LP-FCSVDD with/without applying the foresight response strategy in the GDI dataset. Suppose that node 101 is a compromised sensor and generates intentional anomalies

should ignore the compromised sensor outcomes because it has negative impact in global radius computation. Suppose that node 1 is a compromised sensor and generates intentional anomalies in the IBRL dataset. Without applying the foresight response strategy, the accuracy of detection is severely decreased over the time. Table 5 and Fig. 11 show the LP-FCSVDD results with/without response strategy for the centralized and distributed schemes in the IBRL dataset. Similar results were obtained by repeating the experiment for the GDI dataset (see Table 6, Fig. 12).

The proposed anomaly detection and foresight response strategy has the following advantages: (1) it can detect the anomalies with high accuracy in polynomial time, (2) it can be used in a distributed scheme with minimal communication overhead, and (3) it can offer a proper response strategy to eliminate the effect of intentional anomalies. Accordingly, we can apply the LP-FCSVDD as a robust distributed anomaly detection method in WSNs which resist the intentional, unintentional and false anomalies.

## 5 Conclusion

In this paper, we have presented a new approach to address the problem of anomaly detection and response strategy in wireless sensor networks. The proposed approach is based on support vector data description (SVDD) as a popular one-class classifier. The SVDD method has two major disadvantages: (1) It could sometimes generate such a loose decision boundary when some noisy samples (outliers) exist in the training-set, and (2) It requires the solution of a computationally intensive quadratic programming approach, which is not applicable in WSN. We present the FCSVDD method to solve the first problem. The basic idea of FCSVDD method is to find a minimum hyper-sphere around the target class by using the fuzzy constrains. Unfortunately, the FCSVDD method requires a quadratic programming approach to find the decision boundaries. We solved this problem by formulating a centered hyper-spherical scheme, which enables us to use a linear programming approach and proposed the Linear-Programming based Fuzzy-Constraint SVDD (LP-FCSVDD) method. The using of fuzzy constraints leads to additional abilities that can be used in the response process to tolerate the sensor failure. Accordingly, we present a foresight response strategy to mitigate the intentional, unintentional and false anomalies. In order to evaluate our proposed approach, we use a synthetic dataset and two real WSN datasets namely the IBRL and GDI. The results show the prominence of the LP-FCSVDD method to detect local and global anomalies in WSNs.

## Appendix

See Tables 7 and 8.

**Table 7** LP-FCSVDD results for sensor node 1 in the IBRL dataset in each time window

| Time window | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|
| 1 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |
| 2 | 100 | 0 | 100 | 100 | 100 |
| 3 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 4 | 100 | 0 | 100 | 100 | 100 |
| 5 | 100 | 0 | 100 | 100 | 100 |
| 6 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 7 | 93.4783 | 13.0435 | 100 | 88.4615 | 93.8776 |
| 8 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |
| 9 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |

**Table 7** continued

| Time window | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|
| 10 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 11 | 100 | 0 | 100 | 100 | 100 |
| 12 | 100 | 0 | 100 | 100 | 100 |
| 13 | 100 | 0 | 100 | 100 | 100 |
| 14 | 100 | 0 | 100 | 100 | 100 |
| 15 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 16 | 93.4783 | 13.0435 | 100 | 88.4615 | 93.8776 |
| 17 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |
| 18 | 100 | 0 | 100 | 100 | 100 |
| 19 | 100 | 0 | 100 | 100 | 100 |
| 20 | 89.1304 | 21.7391 | 100 | 82.1429 | 90.1961 |
| 21 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |
| 22 | 93.4783 | 13.0435 | 100 | 88.4615 | 93.8776 |
| 23 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |
| 24 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 25 | 100 | 0 | 100 | 100 | 100 |
| 26 | 100 | 0 | 100 | 100 | 100 |
| 27 | 100 | 0 | 100 | 100 | 100 |
| 28 | 93.4783 | 13.0435 | 100 | 88.4615 | 93.8776 |
| 29 | 100 | 0 | 100 | 100 | 100 |
| 30 | 100 | 0 | 100 | 100 | 100 |
| 31 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 32 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 33 | 100 | 0 | 100 | 100 | 100 |
| 34 | 95.6522 | 8.6957 | 100 | 92 | 95.8333 |
| 35 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 36 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 37 | 93.4783 | 13.0435 | 100 | 88.4615 | 93.8776 |
| 38 | 97.8261 | 4.3478 | 100 | 95.8333 | 97.8723 |
| 39 | 91.3043 | 17.3913 | 100 | 85.1852 | 92 |
| 40 | 100 | 0 | 100 | 100 | 100 |
| 41 | 100 | 0 | 100 | 100 | 100 |
| Average | 97.4549 | 5.0901 | 100 | 95.4139 | 97.5888 |

**Table 8** LP-FCSVDD results for sensor node 101 in the GDI dataset in each time window

| Time window | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|
| 1 | 96.4286 | 7.1429 | 100 | 93.3333 | 96.5517 |
| 2 | 96.875 | 6.25 | 100 | 94.1176 | 96.9697 |
| 3 | 96.1538 | 7.6923 | 100 | 92.8571 | 96.2963 |
| 4 | 100 | 0 | 100 | 100 | 100 |
| 5 | 100 | 0 | 100 | 100 | 100 |
| 6 | 100 | 0 | 100 | 100 | 100 |
| 7 | 93.75 | 12.5 | 100 | 88.8889 | 94.1176 |
| 8 | 100 | 0 | 100 | 100 | 100 |

**Table 8** continued

| Time window | Classification accuracy (CA) | FPR | TPR | Precision | F-measure |
|---|---|---|---|---|---|
| 9 | 100 | 0 | 100 | 100 | 100 |
| 10 | 90 | 20 | 100 | 83.3333 | 90.9091 |
| 11 | 92.3077 | 15.3846 | 100 | 86.6667 | 92.8571 |
| 12 | 92.3077 | 15.3846 | 100 | 86.6667 | 92.8571 |
| 13 | 94.4444 | 11.1111 | 100 | 90 | 94.7368 |
| 14 | 100 | 0 | 100 | 100 | 100 |
| 15 | 100 | 0 | 100 | 100 | 100 |
| 16 | 100 | 0 | 100 | 100 | 100 |
| 17 | 100 | 0 | 100 | 100 | 100 |
| Average | 97.1922 | 5.6156 | 100 | 95.0508 | 97.3703 |

## References


1. Xie, M., Han, S., Tian, B., et al. (2011). Anomaly detection in wireless sensor networks: A survey. *Journal of Network and Computer Applications, 34*(4), 1302–1325.
2. Anwar, R. W., Bakhtiari, M., Zainal, A., et al. (2014). Security issues and attacks in wireless sensor network. *World Applied Sciences Journal, 30*(10), 1224–1227.
3. Alrajeh, N. A., Khan, S., & Shams, B. (2013). Intrusion detection systems in wireless sensor networks: A review. *International Journal of Distributed Sensor Networks, 2013*, 1–7.
4. Butun, I., & Sankar, R. (2011). A brief survey of access control in wireless sensor networks. In *Consumer communications and networking conference (CCNC)*, pp. 11181119.
5. Scarfone, K., & Mell, P. (2007). Guide to intrusion detection and prevention systems (IDPS). *NIST Special Publication, 800*(2007), 94.
6. Abduvaliyev, A., Pathan, A.-S. K., Zhou, J., et al. (2013). On the vital areas of intrusion detection systems in wireless sensor networks. *IEEE Communications Surveys and Tutorials, 15*(3), 1223–1237.
7. Khan, L., Awad, M., & Thuraisingham, B. (2007). A new intrusion detection system using support vector machines and hierarchical clustering. *The VLDB Journal—The International Journal on Very Large Data Bases, 16*(4), 507–521.
8. Zheng, J., & Hu, M.-Z. (2005). *Intrusion detection of DoS/DDoS and probing attacks for web services. Advances in Web-Age Information Management* (pp. 333–344). Berlin: Springer.
9. Ghosh, A. K. & Schwartzbard, A. (1999). A study in using neural networks for anomaly and misuse detection. *Proceedings of the 8th conference on USENIX Security Symposium*, Washington, DC.
10. Rajasegarar, S., Leckie, C., & Palaniswami, M. (2008). Anomaly detection in wireless sensor networks. *IEEE Wireless Communications, 15*(4), 34–40.
11. Zamani, M. (2013). *Machine learning techniques for intrusion detection*. arXiv preprint arXiv:1312.2177.
12. Dua, S., & Du, X. (2014). *Data mining and machine learning in cybersecurity*. Baco Racton: CRC Press.
13. Butun, I., Morgera, S., & Sankar, R. (2014). A survey of intrusion detection systems in wireless sensor networks. *IEEE Communication Surveys & Tutorials, 16*(1), 266–282.
14. Zhang, Y., Meratnia, N., & Havinga, P. (2010). Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys & Tutorials, 12*(2), 159–170.

15. Van Phuong, T., Hung, L. X., Cho, S. J., et al. (2006). An anomaly detection algorithm for detecting attacks in wireless sensor networks. *Intelligence and Security Informatics, 3975*, 735–736.
16. Tax, D. M., & Duin, R. P. (1999). Support vector domain description. *Pattern Recognition Letters, 20*(11), 1191–1199.
17. Guo, S.-M., Chen, L.-C., & Tsai, J. S. H. (2009). A boundary method for outlier detection based on support vector domain description. *Pattern Recognition, 42*(1), 77–83.
18. da Silva, A. P. R., Martins, M. H., Rocha, B. P. et al. (2005) Decentralized intrusion detection in wireless sensor networks In *Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks.* Montreal, Canada, pp. 16–23.
19. Ioannis, K., Dimitriou, T., & Freiling, F. C. (2007) Towards intrusion detection in wireless sensor networks. In *Proceeding of the 13th European Wireless Conference*, Paris, France.
20. Karapistoli, E., & Economides, A. A. (2014). ADLU: a novel anomaly detection and location-attribution algorithm for UWB wireless sensor networks. *EURASIP Journal on Information Security, 2014*(1), 1–12.
21. Palpanas, T., Papadopoulos, D., Kalogeraki, V., et al. (2003). Distributed deviation detection in sensor networks. *ACM SIGMOD Record, 32*(4), 77–82.
22. Ngai, E.-H., Liu, J., & Lyu, M. R. (2006). On the intruder detection for sinkhole attack in wireless sensor networks. In *Proceedings of the 2006 IEEE international conference on communications (ICC'06).* Istanbul, Turkey, pp. 3383–3389.
23. Onat, I., & Miri, A. (2005) A real-time node-based traffic anomaly detection algorithm for wireless sensor networks. In *Proceedings of systems communications*, Montreal, Canada, pp. 422–427.
24. Li, G., He, J., & Fu, Y. (2008). Group-based intrusion detection system in wireless sensor networks. *Computer Communications, 31*(18), 4324–4332.
25. Siripanadorn, S., Hattagam, W., & Teaumroong, N. (2010). Anomaly detection in wireless sensor networks using self-organizing map and wavelets. *International Journal of Communications, 4*(3), 74–83.
26. Branch, J. W., Giannella, C., Szymanski, B., et al. (2013). In-network outlier detection in wireless sensor networks. *Knowledge and Information Systems, 34*(1), 23–54.
27. OReilly, C., Gluhak, A., Imran, M., et al. (2014). Anomaly detection in wireless sensor networks in a non-stationary environment. *IEEE Communications Surveys and Tutorials, 16*(3), 1413–1432.
28. Moshtaghi, M., Leckie, C., Karunasekera, S., et al. (2014). An adaptive elliptical anomaly detection model for wireless sensor networks. *Computer Networks, 64*, 195–207.
29. Salem, O., Guerassimov, A., Mehaoua, A., et al. (2013). Anomaly detection scheme for medical wireless sensor networks. In B. Furht & A. Agarwal (Eds.), *Handbook of medical and healthcare technologies* (pp. 207–222). New York: Springer.
30. Zhang, Y., Meratnia, N., & Havinga, P. J. (2013). Distributed online outlier detection in wireless sensor networks using ellipsoidal support vector machine. *Ad Hoc Networks, 11*(3), 1062–1074.
31. Rajasegarar, S., Leckie, C., & Palaniswami, M. (2014). Hyperspherical cluster based distributed anomaly detection in wireless sensor networks. *Journal of Parallel and Distributed Computing, 74*(1), 1833–1847.
32. Salmon, H. M., de Farias, C. M., Loureiro, P., et al. (2013). Intrusion detection system for wireless sensor networks using danger theory immune-inspired techniques. *International Journal of Wireless Information Networks, 20*(1), 39–66.
33. Ahmadi Livani, M., & Abadi, M. (2011) A PCA-based distributed approach for intrusion detection in wireless sensor networks. In *Proceedings of the 2011 international symposium on computer networks and distributed systems (CNDS)*, Tehran, Iran, pp. 55–60.
34. Wang, H.-B, Yuan, Z., Wang, C.-D. (2009). Intrusion detection for wireless sensor networks based on multi-agent and refined clustering. In *International conference on communications and mobile computing*, Kunming, Yunnan, China, pp. 450–454.
35. Rajasegarar, S., Leckie, C., Palaniswami, M. et al. (2006). Distributed anomaly detection in wireless sensor networks. In *10th IEEE singapore international conference on communication systems*, Singapore, pp. 1–5.
36. S. Rajasegarar, C. Leckie, M. Palaniswami et al. (2007) Quarter sphere based distributed anomaly detection in wireless sensor networks. In: *IEEE International Conference on Communications (ICC'07)*, Glasgow, Scotland, pp. 3864–3869.
37. Rajasegarar, S., Leckie, C., Bezdek, J. C., et al. (2010). Centered hyperspherical and hyperellipsoidal one-class support vector machines for anomaly detection in sensor networks. *IEEE Transactions on Information Forensics and Security, 5*(3), 518–533.
38. Tax, D. M. & Duin R. P. (2000) Data description in subspaces. In *Proceedings of 15th international conference on pattern recognition,* Barcelona, Spain, pp. 672–675.
39. Tax, D. M., & Duin, R. P. (2004). Support vector data description. *Machine Learning, 54*(1), 45–66.
40. Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation, 10*(5), 1299–1319.
41. Laskov, P., Schäfer, C., Kotenko, I., et al. (2004). Intrusion detection in unlabeled data with quarter-sphere support vector machines. *Praxis der Informationsverarbeitung und Kommunikation, 27*(4), 228–236.
42. Song, M. & He, B. (2007). Capacity analysis for flat and clustered wireless sensor networks. In *International conference on wireless algorithms, systems and applications*, Chicago, Illinois, USA, pp. 249–253.
43. Khachiyan, L. G. (1980). Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics, 20*(1), 53–72.
44. Griva, I., Nash, S. G., & Sofer, A. (2009). *Linear and nonlinear optimization: Siam*.
45. Megiddo, N. (1984). Linear programming in linear time when the dimension is fixed. *Journal of the ACM (JACM), 31*(1), 114–127.
46. IBRL dataset. (2012). http://db.lcs.mit.edu/labdata/labdata.html
47. Szewczyk, R., Mainwaring, A., Polastre, J. et al. (2004) An analysis of a large scale habitat monitoring application. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, Baltimore, Maryland, USA, pp. 214–226.

**Mohammad GhasemiGol** was born in Birjand, Iran, in 1984. He received the B.S. degree in Computer Engineering from Payame Noor University (PNU), Birjand, Iran, in 2006. He also received the M.S. degree in Computer Engineering at Ferdowsi University of Mashhad (FUM), Iran, in 2009. Now, he is a Ph.D. candidate in computer engineering at FUM. He is a member of data and communication security lab at FUM. His research interests include network security, intrusion detection and response, alert management, machine learning and data mining, pattern recognition, and optimization problems.

**Abbas Ghaemi-Bafghi** was born on April 1973 in Bojnord, Iran. He received his B.S. degree in Applied Mathematics in Computer from Ferdowsi University of Mashhad, Iran in 1995. He received his M.S. and Ph.D. degrees in Computer engineering from Amirkabir (Tehran Polytechnique) University of Technology, Iran in 1997 and 2004 respectively. He is member of Computer Society of Iran (CSI) and Iranian Society of Cryptology (ISC). He is an associated professor in Department of Computer Engineering, Ferdowsi University of Mashhad, Iran. His research interests are in cryptology and security and he has published more than 70 conference and journal papers.

**Mohammad Hossein Yaghmaee-Moghaddam** received his B.S. degree in Communication Engineering from Sharif University of Technology, Tehran, Iran in 1993, and M.S. degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 1995. He received his Ph.D. degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 2000. He has been a computer network engineer with several networking projects in Iran Telecommunication Research Center (ITRC) since 1992. November 1998 to July 1999, he was with Network Technology Group (NTG), C&C Media research labs, NEC Corporation, Tokyo, Japan, as visiting research scholar. From September 2007 to August 2008, he was with the Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, USA as a visiting associate professor. He is the author of 5 books all in Farsi language. He has published more than 150 international conference and journal papers. His research interests are in Wireless Sensor Networks (WSNs), smart grid, traffic and congestion control, high speed networks including ATM and MPLS, Quality of Services (QoS) and fuzzy logic control. Currently he is full professor at Computer Engineering Department of Ferdowsi University of Mashhad. He is also head of IP-PBX type approval lab at this university.

**Hadi Sadoghi-Yazdi** is currently an Associate Professor of Computer Science and Engineering at Ferdowsi University of Mashhad (FUM). He received his B.S. degree in Electrical Engineering from FUM in 1994, and received his M.S. and Ph.D. degrees in Electrical Engineering from Tarbiat Modares University in 1996 and 2005, respectively. His research interests are in the areas of Pattern Recognition, Machine Learning, Machine Vision, Signal Processing, Data Mining and Optimization.