SPECTRAL ANOMALY DETECTION USING GRAPH-BASED FILTERING FOR WIRELESS SENSOR NETWORKS

Hilmi E. Egilmez and Antonio Ortega

Signal and Image Processing Institute, University of Southern California
hegilmez@usc.edu, ortega@sipi.usc.edu

ABSTRACT

This paper introduces a novel spectral anomaly detection method by developing a graph-based filtering framework. In particular, we consider the problem of unsupervised data anomaly detection over wireless sensor networks (WSNs) where sensor measurements are represented as signals on a graph. In our framework, graphs are chosen to capture useful proximity information about measured data. The associated graph-based filters are then employed to project the graph signals on normal and anomaly subspaces, and resulting projections are used in detection of data anomalies. The proposed approach has two main advantages over the standard spectral technique, principal component analysis (PCA). Firstly, graph-based filtering allows us to incorporate structural information known a priori (e.g., distance between sensors) in addition to data. Secondly, it provides localized transformations leading to effective distributed anomaly detection. Our experimental results show that our proposed solution outperforms PCA-based and distributed clustering-based anomaly detection methods in terms of receiver operating characteristics (ROCs).

Index Terms— Anomaly detection, graph signal processing, graph-based filtering, spectral methods, WSNs.

1. INTRODUCTION

Anomaly detection can be defined as the *identification of patterns that do not conform to the normal behavior*. Therefore, an anomaly detection approach requires (i) to define detection regions that represent the normal behavior, and then (ii) to declare observations which do not belong to the these regions as anomalies. Detecting anomalies can be very challenging depending on the application, the nature of the input data, the type of the anomaly and the availability/unavailability of data labels for supervised/unsupervised detection. Many techniques have been proposed for different anomaly detection applications such as intrusion detection in computer networks, fault detection and event detection in environmental monitoring [1]. In this work, we consider unsupervised anomaly detection in wireless sensor networks (WSNs), where the sensor nodes collect univariate data, and the anomalies of interest occur *locally in both time and space*. These are special cases of *collective anomalies* as defined in [1].

Anomaly detection in WSNs is further challenging due to the inherent limitations on energy resources and processing power. In WSNs the major source of energy consumption is communication rather than computation. Thus, it is crucial to design distributed anomaly detection techniques that minimize the communication cost using in-network processing to improve energy efficiency. Unsupervised methods are also useful in WSN applications, since they do not require labeled data, which is usually hard to obtain. Basically, unsupervised anomaly detection techniques can be classified as (i) nearest neighbor-based, (ii) clustering-based, and (iii) spectral methods [1–3]. Typically, nearest neighbor methods identify anomalies by thresholding anomaly scores defined based on distance between data points. On the other hand, clustering-based methods group similar data instances into clusters. Then, data points belonging to large and dense clusters are considered normal, and data instances in either small or sparse clusters are declared as anomalies. The main drawback in both nearest neighbor and clustering based techniques is that they show good detection performance only when normal data instances are densely clustered and anomalies appear in small and sparse data groups. In addition, their detection performance highly depends on the choices of features and distance measure, both of which are challenging. Conversely, spectral methods follow a different approach where the goal is finding subspaces for normal and anomalous regions in order to identify anomalous instances via projections. Although their detection performance strictly depends on the choice of lower-dimensional embeddings, spectral methods do not impose any assumptions on the density of data instances, and it has been shown that they achieve good detection performance when anomalies are collective [4]. This motivates us to introduce a new unsupervised spectral anomaly detection method.

In the literature, principal component analysis (PCA) is extensively used for spectral anomaly detection, where normal and anomaly subspaces are decomposed based on principal components. Lakhina *et al.* [5] originally propose PCA to identify volume anomalies for network intrusion detection. In [6], Huang *et al.* present a scalable extension for network intrusion detection where local filters are employed to aggregate network traffic data and PCA is used for centralized anomaly detection. Specifically for WSNs, Chatzigiannakis and Papavassiliou [7] apply PCA for supervised anomaly detection by performing an offline training to define subspaces using PCA. The principal components are then advertised to the sensors to allow distributed processing. In fact, most of the work on distributed anomaly detection focuses on statistical and clustering-based techniques [3, 8]. Most recently, Rajasegarar *et al.* [9] propose a distributed method where each sensor aggregates its data via clustering, sends clustered data to the base station, and a nearest neighbor procedure detects anomalous clusters. However, all of these methods are completely data-driven, and do not exploit any other information. The main motivation for our work is that for data anomaly detection over WSNs, in addition to raw sensor data, proximity information such as distance between sensors and any other prior knowledge about the environment can be useful to capture local anomalous behavior. Moreover, PCA-based techniques [4–7] are not suitable for distributed anomaly detection, since principal components of PCA are generated using complete data and have no locality information. In order to overcome these problems, this paper proposes a graph-based filtering [10] framework that (i) allows to jointly exploit data and proximity information between data instances, and (ii) enables effective distributed anomaly detection using localized transformations. In particular, we propose to support irregular data measurements as signals on nodes of a graph where weighted edges reflect the similarities between signals (i.e., data

This work has been partially supported by Intel under the VAWN project.

instances). Then, graph-based filters are employed to project graph signals onto normal and anomaly subspaces, and a thresholding mechanism is used to label anomalous instances. For distributed anomaly detection, the proposed framework allows us to define different localized transforms by designing both graph partitions and graph-based filters. In this work, we focus on localized transforms defined via graph partitioning.

There are few works on anomaly detection for graph-based data using spectral graph theory. Ide and Kashima [11] propose to use time-series graphs each of which is represented by an adjacency matrix capturing the dependencies between computer network services, and principal eigenvectors of adjacency matrices are used to detect anomalous services. Similarly, Miller *et al.* [12] introduce a method for detecting anomalous subgraphs by analyzing the eigenvectors of a graph modularity matrix. Although it is possible to combine data with proximity information to define edge weights on a graph, methods in [11, 12] only consider graph anomaly detection problem. Noble and Cook [13] detect graph anomalies based on the regularity of a graph without using spectral techniques. Most similar to our work, Crovella and Kolaczyk [14] apply wavelets on graphs for network traffic analysis. Yet, they only use vertex domain operations and do not provide any spectral interpretation. The proposed framework is more general than existing spectral graph and PCA-based techniques, since it has flexibility of choosing graphs and associated filtering to design various spectral decompositions.

The rest of the paper is organized as follows. Section 2 presents notations and basic concepts for graph-based filtering. The problem definition and spectral decomposition formulations are presented in Section 3. Section 4 discusses the proposed solution. We compare the performance of our method against both PCA-based and clustering-based [9] methods in Section 5. Section 6 draws conclusions based on experimental results.

2. PRELIMINARIES

2.1. Notation

Throughout the paper, we use capital bold letters for matrices and lower-case bold letters for column vectors. Scalar values are denoted as normal lower-case letters. The cardinality, Manhattan norm, Euclidean norm and Frobenius norm operators are $|\cdot|$, $\|\cdot\|_1$, $\|\cdot\|_2$ and $\|\cdot\|_F$, respectively. The set of nodes is $\mathcal{N} = \mathcal{N}_n \cup \mathcal{N}_a$ and the set of time instances is $\mathcal{T} = \mathcal{T}_n \cup \mathcal{T}_a$ where sets with subscripts n and a correspond to normal and anomaly partitions of each set, respectively. The data matrix $\mathbf{Y} = \begin{bmatrix}\mathbf{y}_1\, \mathbf{y}_2 \cdots \mathbf{y}_{|\mathcal{T}|}\end{bmatrix}^t$ is $|\mathcal{T}| \times |\mathcal{N}|$, where $\mathbf{y}_i$ is the column vector of $|\mathcal{N}|$ measurements at time $i \in \mathcal{T}$. Let $\mathbf{Y}_0$ be the mean removed data matrix, so the data covariance matrix is defined as $\mathbf{C} = \frac{1}{|\mathcal{T}|-1}\mathbf{Y}_0^t\mathbf{Y}_0$.

2.2. Graph-based Filtering

The signals of interest are defined on an undirected, weighted and connected graph $G(\mathcal{N}, \mathcal{E}, \mathbf{W})$ with no self-loops, where $\mathcal{N}$ is the set of nodes with $|\mathcal{N}|$ elements, $\mathcal{E}$ is the set of edges, and $\mathbf{W}$ is the weighted adjacency matrix with non-negative entries. If there is an edge between nodes i and j, the element at i-th row and j-th column of $\mathbf{W}$ is greater than zero ($w_{i,j} > 0$), otherwise $w_{i,j} = 0$. The signals are defined on the nodes of the graph and represented as a vector $\mathbf{y} \in \mathbb{R}^{|\mathcal{N}|}$ where the i-th element of vector $\mathbf{y}$ represents the signal at i-th node in $\mathcal{N}$.

The normalized graph Laplacian associated to a graph G is $\mathcal{L}_G = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$ where $\mathbf{D}$ is the diagonal degree matrix such that the i-th diagonal element $d_{i,i}$ is the sum of weights of edges incident to i-th node in $\mathcal{N}$. The normalized Laplacian $\mathcal{L}_G$ is a real symmetric matrix, and therefore it has a complete set of orthonormal eigenvectors, denoted as $\{\mathbf{u}_l\}_{l=1,...,|\mathcal{N}|}$, whose associated eigenvalues $\{\lambda_l\}_{l=1,...,|\mathcal{N}|}$ are real and non-negative. The eigenpairs $\{\lambda_l, \mathbf{u}_l\}_{l=1,...,|\mathcal{N}|}$ of $\mathcal{L}_G$ provide a Fourier-like frequency interpretation of signals defined on graphs [10], so that the spectrum on the graph is defined by the eigenvalues and the eigenvectors determine the harmonic modes for graph signals. In addition, the normalized graph Laplacian has the nice property that its spectrum is always contained within the range of $[0, 2]$ (i.e., $\lambda_l \in [0, 2]$ for $l = 1, ..., |\mathcal{N}|$). Since the normalized graph Laplacian is always diagonalizable, then $\mathcal{L}_G = \mathbf{U}_G\mathbf{\Lambda}_G\mathbf{U}_G^t$ where $\mathbf{U}_G$ is the eigenvector matrix composed of $\{\mathbf{u}_l\}_{l=1,...,|\mathcal{N}|}$ whose associated eigenvalues appear in the diagonal matrix $\mathbf{\Lambda}_G$. The Graph Fourier Transform (GFT) is defined by the matrix $\mathbf{U}_G^t$, and can be thought to be analogous to discrete Fourier transform (DFT) in traditional digital signal processing [15]. Graph-based filtering is defined in spectral domain by the normalized graph Laplacian that is, $h(\mathcal{L}_G) = \mathbf{U}_G(h(\mathbf{\Lambda}_G))\mathbf{U}_G^t$ where $h(\mathbf{\Lambda}_G) = diag\left(h(\lambda_1), ..., h(\lambda_{|\mathcal{N}|})\right)$. Hence, the input-output relation of graph based filtering on graph signals can be written as $\hat{\mathbf{y}} = h(\mathcal{L}_G)\mathbf{y} = \mathbf{U}_G(h(\mathbf{\Lambda}_G))\mathbf{U}_G^t\mathbf{y}$. For further details on graph-based filtering, we refer to [10, 15].

3. PROBLEM FORMULATION

We consider the problem of unsupervised data anomaly detection in WSNs where each sensor in $\mathcal{N}$ measures univariate data (single attribute) in a time window $\mathcal{T}$, and the anomalies of interest are special cases of collective anomalies [1] which appear locally in both time and space. For instance, a rapid variation in temperature caused by a fire in some region of a field is a collective data anomaly. Our main assumption is that the *normal data instances are more frequent than anomaly instances in time* i.e.,

$$|\mathcal{T}_n| \gg |\mathcal{T}_a|. \tag{1}$$

For space, we only assume that anomalies are localized. Centralized and distributed anomaly detection problems are defined as follows.

- In *centralized anomaly detection*, all sensor nodes send their data samples to the base station, and the base station identifies anomalous time instants.
- In *distributed anomaly detection*, the network is divided into communication clusters (i.e., graph partitions) denoted as $\mathcal{C}$, and a cluster head is chosen for each cluster. Then, all sensor nodes send their data samples to their dedicated cluster head which detects anomalous time instants. For example, clustering can be done using energy efficient protocols such as LEACH [16].

As discussed in Section 1, the spectral anomaly detection problem boils down to defining subspaces that represent the normal and anomalous regions. In the following subsections, we formulate the spectral decomposition using both graph-based filtering and PCA. In addition, we unify both formulations, and show that PCA is a special case of spectral decomposition with graph-based filtering.

In our formulations, $\mathbf{X} = \mathbf{Y}_0^t$ denotes the data matrix of size $|\mathcal{N}| \times |\mathcal{T}|$. We first assume that the anomalies in $\mathbf{X}$ are known (labeled) for each time instance, where $\mathbf{X}_n$ is the $|\mathcal{N}| \times |\mathcal{T}_n|$ matrix having normal data instances, and $\mathbf{X}_a$ is the $|\mathcal{N}| \times |\mathcal{T}_a|$ matrix whose each column has at least one anomalous data instance. In what follows we first present a supervised formulation. The unsupervised spectral decomposition only has access to the (unlabeled) matrix $\mathbf{X}$,

so based on the main assumption stated in (1) we propose a relaxed solution in Section 4.

3.1. Spectral decomposition using Graph-based Filtering

We formulate the general graph-filtering based spectral decomposition as finding an undirected, weighted and connected graph G, and two graph-based filters ($h(\lambda)$ and $\tilde{h}(\lambda)$), that solve the optimization problem,

$$\underset{G,h,\tilde{h}}{\text{minimize}} \ \|\mathbf{X}_n - h(\mathcal{L}_G)\mathbf{X}_n\|_F + \gamma\|\mathbf{X}_a - \tilde{h}(\mathcal{L}_G)\mathbf{X}_a\|_F \tag{2}$$

where the scalar γ is the weighting factor.

In this paper, we restrict our attention to ideal low-pass and high-pass filters, so the problem (2) reduces to finding G and $\lambda_c \in [0, 2]$ by solving the optimization problem,

$$\begin{aligned} &\underset{G,\lambda_c}{\text{minimize}} \ \|\mathbf{X}_n - h_n^{\lambda_c}(\mathcal{L}_G)\mathbf{X}_n\|_F + \gamma\|\mathbf{X}_a - h_a^{\lambda_c}(\mathcal{L}_G)\mathbf{X}_a\|_F \\ &\text{subject to } h_n^{\lambda_c}(\lambda) = \begin{cases} 1 & \text{if } \lambda \le \lambda_c \\ 0 & \text{if } \lambda > \lambda_c \end{cases}, \ h_a^{\lambda_c}(\lambda) = \begin{cases} 1 & \text{if } \lambda > \lambda_c \\ 0 & \text{if } \lambda \le \lambda_c \end{cases} \end{aligned} \tag{3}$$

where $h_n^{\lambda_c}(\mathcal{L}_G) = \mathbf{U}_G(h_n^{\lambda_c}(\mathbf{\Lambda}_G))\mathbf{U}_G^t$ is analogous to an ideal low-pass and $h_a^{\lambda_c}(\mathcal{L}_G) = \mathbf{U}_G(h_a^{\lambda_c}(\mathbf{\Lambda}_G))\mathbf{U}_G^t$ corresponds to an ideal high-pass filtering operation defined on the normalized graph Laplacian, $\mathcal{L}_G$, on graph G.

3.2. Spectral decomposition using PCA

The PCA-based spectral decomposition problem can be posed as finding an $r \in \{1, ..., (|\mathcal{N}| - 1)\}$ that solves the optimization problem,

$$\begin{aligned} &\underset{r}{\text{minimize}} \ \ \|\mathbf{X}_n - \mathbf{P}_n(r)\mathbf{X}_n\|_F + \gamma\,\|\mathbf{X}_a - \mathbf{P}_a(r)\mathbf{X}_a\|_F \\ &\text{subject to } \ \mathbf{P}_n(r) = \mathbf{V}_n(r)\mathbf{V}_n^t(r), \ \mathbf{P}_a(r) = \mathbf{V}_a(r)\mathbf{V}_a^t(r) \end{aligned} \tag{4}$$

where $\mathbf{V}_n(r) = [\mathbf{v}_1 \cdots \mathbf{v}_r]$ and $\mathbf{V}_a(r) = [\mathbf{v}_{r+1} \cdots \mathbf{v}_{|\mathcal{N}|}]$ are the matrices composed of principal components $\{\mathbf{v}_i\}_{i=1...|\mathcal{N}|}$ which also construct columns of $\mathbf{V}$ diagonalizing the covariance matrix $\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^t$. The projection matrices can be also written as

$$\mathbf{P}_n(r) = \sum_{j=1}^{r} \mathbf{v}_j\mathbf{v}_j^t \text{ and } \mathbf{P}_a(r) = \sum_{j=r+1}^{|\mathcal{N}|} \mathbf{v}_j\mathbf{v}_j^t. \tag{5}$$

The following proposition shows that PCA-based spectral decomposition problem can be considered as a special case of spectral decomposition using graph-based filtering.

Proposition 1 *The PCA-based decomposition problem in (4) is equivalent to graph-based decomposition problem stated in (3) if $\mathcal{L}_G$ is selected as $\mathbf{C}^{-1}$, that is inverse of the covariance matrix.*

Proof: Let any covariance matrix be $\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^t$ where the columns of $\mathbf{V}$ are the eigenvectors and $\mathbf{\Lambda} = diag(\lambda_1, ..., \lambda_{|\mathcal{N}|})$ has eigenvalues with $\lambda_1 \ge \lambda_2 \ge ... \ge \lambda_{|\mathcal{N}|}$. The inverse of $\mathbf{C}$, $\mathbf{C}^{-1} = \mathbf{V}\mathbf{\Lambda}^{-1}\mathbf{V}^t$ has same set of eigenvectors, but $\mathbf{\Lambda}^{-1} = diag(\lambda_1^{-1}, ..., \lambda_{|\mathcal{N}|}^{-1})$ where $\lambda_1^{-1} \le \lambda_2^{-1} \le ... \le \lambda_{|\mathcal{N}|}^{-1}$. Since $\forall r \ \exists\lambda_c$ (or $\forall\lambda_c \ \exists r$) such that $\mathbf{P}_n(r) = h_n^{\lambda_c}(\mathbf{C}^{-1})$ and $\mathbf{P}_a(r) = h_a^{\lambda_c}(\mathbf{C}^{-1})$ by their definitions, the statement is true.

In general, we cannot find a graph that satisfies $\mathcal{L}_G = \mathbf{C}^{-1}$. But, as a specific case, if data follows Gaussian Markov Random Field (GMRF) model, then there exists a graph such that $\mathcal{L}_G = \mathbf{C}^{-1}$ [17].

4. PROPOSED SOLUTION

The proposed solution for unsupervised anomaly detection using graph-based filtering involves three steps discussed below.

4.1. Graph construction

The goal is to design an undirected and connected graph $G(\mathcal{N}, \mathcal{E}, \mathbf{W})$ with non-negative weights as discussed above. In proposed graph construction, the nodes ($\mathcal{N}$) representing the sensors are fixed. Therefore, the graph construction becomes equivalent to finding an adjacency matrix $\mathbf{W}$ which also defines the edges $\mathcal{E}$. We present three graph designs by defining corresponding adjacent matrices $\mathbf{W}_d(\theta_d)$, $\mathbf{W}_c(\theta_c)$ and $\mathbf{W}_b(\Delta_c, \Delta_d)$. The entries of $\mathbf{W}_d(\theta_d)$ are determined based on the distance between sensor nodes as follows,

$$[w_{i,j}]_d = \begin{cases} \frac{1}{D(i,j)} & \text{if } D(i,j) \le \theta_d \text{ and } D(i,j) \ne 0 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $D(i, j)$ is the Euclidean distance between sensor nodes $i \in \mathcal{N}$ and $j \in \mathcal{N}$. θ_d is the threshold determining the graph connectivity. Similarly, we define $\mathbf{W}_c(\theta_c)$ whose entries are determined based on correlation coefficients obtained from the data,

$$[w_{i,j}]_c = \begin{cases} \|\rho(i,j)\|_1 & \text{if } \|\rho(i,j)\|_1 \ge \theta_c \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $\rho(i, j) = c_{i,j}/(\sqrt{c_{i,i}c_{j,j}})$ that is calculated using elements of the covariance matrix $\mathbf{C}$, and θ_c is the threshold. The entries of the third adjacency matrix $\mathbf{W}_b(\Delta_c, \Delta_d)$ are calculated based on both distance and correlation coefficients (ρ) as follows,

$$[w_{i,j}]_b = \exp\left(-\frac{(1 - \|\rho(i,j)\|_1)^2}{\Delta_c^2}\right) \cdot \exp\left(-\frac{\tilde{D}(i,j)^2}{\Delta_d^2}\right) \tag{8}$$

where $\tilde{D}(i, j) \in [0, 1]$ is the normalized distance between sensor nodes $i \in \mathcal{N}$ and $j \in \mathcal{N}$. Δ_c and Δ_d are the parameters determining exponential decay rate.

4.2. Finding cut-off frequency (λ_c) of graph-based filters

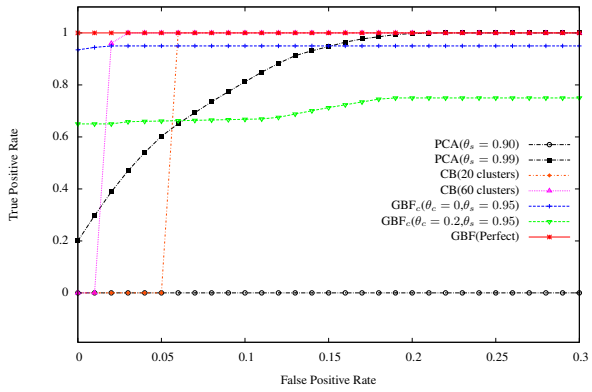
In this step, we separate normal and anomaly subspaces for unsupervised detection. Since the data is unlabeled, we approximate $\mathbf{X}_n \approx \mathbf{X}$ under the assumption stated in (1). The cut-off frequency λ_c is found based on the spreads of projected data instances, in $\mathbf{X}$, onto eigenvectors $\{\mathbf{u}_l\}_{l=1...|\mathcal{N}|}$ of $\mathcal{L}_G$. We define the spread of data projected on eigenvector $\mathbf{u}_l$ as,

$$\sigma_l^2 = s^2[\mathbf{u}_l^t\mathbf{X}] \tag{9}$$

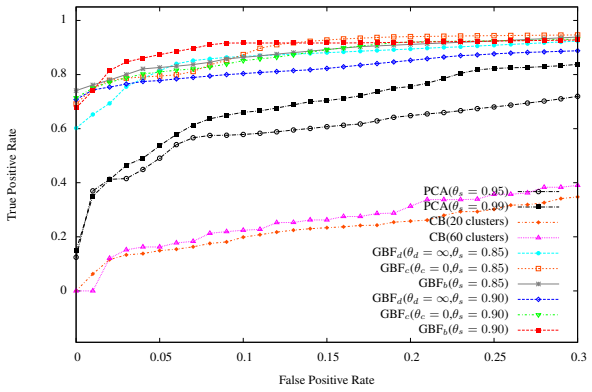
where $s^2[\mathbf{p}^t]$ denotes the sample variance of the elements of a row vector $\mathbf{p}^t$. Let us assume (without loss of generality) that $\sigma_1^2 \ge \sigma_2^2 \ge ... \ge \sigma_{|\mathcal{N}|}^2$ and define $\sigma_T^2 = \sum_{k=1}^{|\mathcal{N}|} \sigma_k^2$. The index c of λ_c is found as,

$$\arg\max_c \sum_{l=1}^{c} \frac{\sigma_l^2}{\sigma_T^2} \quad \text{subject to } \sum_{l=1}^{c} \frac{\sigma_l^2}{\sigma_T^2} \le \theta_s \tag{10}$$

where $\theta_s \in [0, 1]$ is to the target ratio of data spread in normal space to the total data spread in whole space. In practice, this parameter is selected depending on the application and expected frequency of anomalies by the WSN-operator, and it is analogous to *minimum*



(a)



(b)

Fig. 1. ROC results for (a) global and (b) distributed anomaly detection using graph-based filtering (GBF), PCA-based (PCA) and clustering-based (CB) methods and their associated parameters. The curves GBF$_d(\theta_d, \theta_s)$ and GBF$_c(\theta_c, \theta_s)$ correspond to graph construction with distance and correlation information, respectively. GBF$_b(\theta_s)$ corresponds to graph construction using (8) with $\Delta_c = \Delta_d = 1$. In (a), the curve with label GBF(Perfect) represents perfect detection results which are GBF$_b(\theta_s = \{0.90, 0.95\})$ GBF$_d(\theta_d = \{120, 240, \infty\}, \theta_s = \{0.90, 0.95\})$, GBF$_c(\theta_d = \{0, 0.2, 0.4\}, \theta_s = \{0.90\})$ and GBF$_c(\theta_d = \{0.4\}, \theta_s = \{0.95\})$. In (b), the selected graphs are fully connected.

Table 1. AUC comparison of proposed method (using fully connected graphs) against PCA and clustering-based (CB) methods

Method	**Global** (using score $\tilde{s}_i$)			**Global** (using score s_i)			**Distributed** (using score $\tilde{s}_i$)			**Distributed** (using score s_i)		
	θ_s=0.90	θ_s=0.95	θ_s=0.99	θ_s=0.90	θ_s=0.95	θ_s=0.99	θ_s=0.85	θ_s=0.90	θ_s=0.95	θ_s=0.85	θ_s=0.90	θ_s=0.95
PCA	0.1075	0.0241	0.9500	0.5645	0.5817	0.8949	0.4090	0.4268	0.5034	0.6987	0.7177	0.8170
GBF$_d$	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	0.8348	**0.9012**	**0.8795**	0.9329	0.9086	0.8613
GBF$_c$	**1.000**	0.9538	0.4826	**1.000**	0.9889	0.7896	**0.9283**	0.8753	0.7589	0.9455	**0.9424**	**0.9146**
GBF$_b$	**1.000**	**1.000**	0.5067	**1.000**	**1.000**	0.7958	0.9024	**0.9137**	0.8052	**0.9500**	**0.9413**	**0.9060**

CB	**Global** (20 clusters)	**Global** (60 clusters)	**Distributed** (20 clusters)	**Distributed** (60 clusters)
	0.9450	0.9846	0.5544	0.6109

cluster size chosen in clustering-based methods. For example, setting $\theta_s = 0.95$ means that λ_c will separate normal and anomaly spaces such that 95% of the data spread is in normal space and 5% is in anomaly space.

Note that the parameter r in spectral decomposition with PCA can be found by replacing $\mathbf{u}_l$ by $\mathbf{v}_l$ (eigenvectors of $\mathbf{C}$) for $l = 1, ..., |\mathcal{N}|$ in (9) and changing c to r in (10).

4.3. Thresholding anomaly scores

After spectral decomposition using graph-based filtering or PCA, we can project data instances onto normal and anomaly subspaces via orthogonal projection matrices $\mathbf{O}_n$ and $\mathbf{O}_a$ where $\mathbf{O}_n = h_n^{\lambda_c}(\mathcal{L}_G)$ and $\mathbf{O}_a = h_a^{\lambda_c}(\mathcal{L}_G)$ for graph-based filtering. For PCA, $\mathbf{O}_n = \mathbf{P}_n(r)$ and $\mathbf{O}_a = \mathbf{P}_a(r)$. Anomalies are detected by thresholding an anomaly score for each time instant $i \in \mathcal{T}$. The anomaly score introduced in [5] only uses the projections on anomaly space, that is, $s_i = \|\mathbf{O}_a \mathbf{x}_i\|_2^2$ where $\mathbf{x}_i$ is the i-th column of matrix $\mathbf{X}$. We define a different scoring using projections on both normal and anomaly spaces as $\tilde{s}_i = \|\mathbf{O}_n \mathbf{z}_i\|_2^2 - \|\mathbf{O}_a \mathbf{z}_i\|_2^2$ where $\mathbf{z}_i = \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|_2}$. In practice, a commonly used threshold is three times the score's standard deviation from its mean. In order to show overall detection performance of each method, different thresholds are chosen to generate ROC curves. The results are presented in the next section.

5. RESULTS

We show the performance of proposed graph-based filtering (GBF) approaches by benchmarking against PCA-based and clustering-based (CB) [9] methods in terms of ROCs and their area under curve (AUC). In our simulations, we generated a temperature map with 200 time snapshots ($\mathcal{T} = \{1, ..., 200\}$) over a 600×600 grid using an autoregressive (AR) model. The WSN has randomly positioned $|\mathcal{N}|$=100 nodes measuring temperature data. The collective data anomalies are generated using a highly varying AR model with different parameters at time instants $\mathcal{T}_a = \{50, ..., 59\}$ in $|\mathcal{N}_a|$=20 neighboring nodes' data. For the distributed detection, the LEACH protocol [16] divides the network into $|\mathcal{C}|$=6 clusters and determines cluster heads detecting anomalies.

We experiment the performance of each approach by varying its associated parameters (θ_s, θ_d or θ_c). ROC and AUC results are generated by thresholding over anomaly scores $\tilde{s}_i$ and s_i. In addition, the clustering-based method proposed in [9] is implemented, and the corresponding ROC results are obtained by varying number of clusters and k-nearest neighbor parameters. For each method, the simulation is repeated 20 times with different anomaly patterns, the average results are presented in Fig.1 and Table 1. As shown in ROC and AUC results, only GBF approaches can achieve perfect detection performance among all global detection methods. In the distributed case, proposed methods also provide better detection than PCA and CB methods. The best distributed detection performance (0.95 AUC) is achieved by GBF$_b$ which uses both distance and correlation in graph construction. Although the CB method shows reasonably good global detection performance, its distributed performance is worse than both GBF and PCA methods. Moreover, the detection performance of PCA method can seriously degrade depending on the choice of θ_s, but GBF methods are robust to changes in θ_s.

6. CONCLUSIONS

In this paper, we introduce a novel spectral anomaly detection framework using graph-based filtering. We also show that standard PCA-based methods are special cases of the proposed method. Inspection of experimental results lead us to following conclusions:

- The proposed approach significantly outperforms state-of-the-art methods in both global and distributed cases.
- Distance between sensors can be exploited to improve localized anomaly detection performance.
- The proposed method is robust to parameter perturbations and works well with different anomaly scoring metrics.

7. REFERENCES

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, July 2009.

[2] Y. Zhang, N. Meratnia, and P. Havinga, "Outlier detection techniques for wireless sensor networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 12, no. 2, pp. 159–170, 2010.

[3] M. Xie, S. Han, B. Tian, and S. Parvin, "Anomaly detection in wireless sensor networks: A survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1302 – 1325, 2011.

[4] B. Zhang, J. Yang, J. Wu, D. Qin, and L. Gao, "Pca-subspace method - is it good enough for network-wide anomaly detection," in *IEEE Network Operations and Management Symposium (NOMS 2012)*, 2012, pp. 359–367.

[5] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, pp. 219–230, Aug. 2004.

[6] L. Huang, M. I. Jordan, A. Joseph, M. Garofalakis, and N. Taft, "In-network pca and anomaly detection," in *NIPS 2006*, 2006, pp. 617–624.

[7] V. Chatzigiannakis and S. Papavassiliou, "Diagnosing anomalies and identifying faulty nodes in sensor networks," *IEEE Sensors Journal*, vol. 7, no. 5, pp. 637–645, 2007.

[8] S. Rajasegarar, C. Leckie, and M. Palaniswami, "Anomaly detection in wireless sensor networks," *IEEE Wireless Communications*, vol. 15, no. 4, pp. 34–40, 2008.

[9] S. Rajasegarar, C. Leckie, and M. Palaniswami, "Hyperspherical cluster based distributed anomaly detection in wireless sensor networks," *Journal of Parallel and Distributed Computing*, 2013.

[10] D.I. Shuman, S.K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[11] T. Idé and H. Kashima, "Eigenspace-based anomaly detection in computer systems," in *Proc KDD'04*, 2004, pp. 440–449.

[12] B. A. Miller, N. T. Bliss, and P. J. Wolfe, "Subgraph detection using eigenvector l1 norms," in *NIPS 2010*, 2010, pp. 1633–1641.

[13] C. C. Noble and D. J. Cook, "Graph-based anomaly detection," in *Proceedings of the ninth ACM SIGKDD*, 2003, KDD '03, pp. 631–636.

[14] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *INFOCOM 2003*, 2003, vol. 3, pp. 1848–1857 vol.3.

[15] A. Sandryhaila and J.M.F. Moura, "Discrete signal processing on graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1644–1656, 2013.

[16] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[17] C. Zhang and D. Florencio, "Analyzing the optimality of predictive transform coding using graph-based models," *IEEE Signal Process. Lett.*, vol. 20, no. 1, pp. 106–109, 2013.