 DOI 10.1109/TNSE.2022.3169220

Volterra Graph-Based Outlier Detection for Air Pollution Sensor Networks

Pau Ferrer-Cid, Jose M. Barcelo-Ordinas, and Jorge Garcia-Vidal

***Abstract*—Today's air pollution sensor networks pose new challenges given their heterogeneity of low-cost sensors and high-cost instrumentation. Recently, with the advent of graph signal processing, sensor network measurements have been successfully represented by graphs depicting the relationships between sensors. However, one of the main problems of these sensor networks is their reliability, especially due to the inclusion of low-cost sensors, so the detection and identification of outliers is extremely important for maintaining the quality of the network data. In order to better identify the outliers of the sensors composing a network, we propose the Volterra graph-based outlier detection (VGOD) mechanism, which uses a graph learned from data and a Volterra-like graph signal reconstruction model to detect and localize abnormal measurements in air pollution sensor networks. The proposed unsupervised decision process is compared with other outlier detection methods, state-of-the-art graph-based methods and non-graph-based methods, showing improvements in both detection and localization of anomalous measurements, so that anomalous measurements can be corrected and malfunctioning sensors can be replaced.**

***Index Terms*—Wireless Sensor Networks, Air Pollution Monitoring, Outlier Detection, Graph Signal Processing, Low-Cost Sensors.**

I. Introduction

AIR pollution is a growing problem that affects millions of people annually. In fact, air pollution is known to cause everything from respiratory problems to heart diseases [1]. Governments are therefore aware of the importance of monitoring air pollution to take measures to mitigate its effects on the population and the environment. Authorities currently deploy high-cost instrumentation capable of measuring the presence of pollutants in the air with high accuracy, but their high cost implies that the number of instruments available per area is limited. Given the growing field of the Internet of Things (IoT), low-cost sensors have provided an alternative solution that can coexist with precise instrumentation to improve the resolution obtained by the government's monitoring networks [2], [3]. Although these sensors have a low accuracy, their low-cost has led to the study of calibration techniques based on machine learning to improve the accuracy of these sensors; including both linear [4]–[9] and nonlinear models [4], [10]–[13].

Once low-cost sensors have been calibrated *in-situ* at reference stations, they are deployed to increase the spatial resolution of the monitoring network. Thus, one of the most important aspects of a sensor network is the quality of the data, as they can be used by institutions to carry out measures and raise public awareness. Therefore, the detection of outliers in this type of networks is essential to increase the reliability of the network [14]–[16]. Unsupervised methods are common in this type of network since there is no prior information on what measures may be outliers, and the sensors are assumed to work well during a training or calibration period. Moreover, if a network works cooperatively in correcting sensor measurements, i.e. using information from neighboring nodes, it is essential to be able to identify whether the network contains an outlier, and to identify which sensor is giving abnormal measurements. Once the sensor is identified, replacement actions or even the recalibration of the malfunctioning sensor can be carried out using the other sensors in the network.

There are many unsupervised outlier detection techniques, ranging from univariate z-score based techniques [17], to multivariate techniques based on machine learning models [14], [18]–[20]. Spectral decomposition of data using principal component analysis (PCA) is widely used [21]–[23], as well as residual-based techniques, where a spatial model is fitted and large sample residuals indicate the existence of outlierness [15]. However, in this specific paradigm of air pollution, different techniques have been used for modeling these networks, from spatial models [24] to graphs [25]. Indeed, the growing field of graph signal processing (GSP) has shown its flexibility in describing this type of network as well as providing classical signal processing techniques for their analysis [26], [27]. This field is based on the assumption of signal smoothness, assuming that similar sensors will be strongly connected while non-similar sensors will be weakly connected or disconnected. The interpretation of the measurements as a signal defined on a graph allows the calculation of the Fourier basis and the interpretation of the different frequency components through the Graph Discrete Fourier Transform (GDFT) [28]. Thus, the search for high frequencies to detect outliers has already been used for outlier detection, as the magnitude of the high frequencies are increased due to abrupt changes in similar nodes [29]. That is why the description of the topology by means of a graph and the subsequent application of filtering or anomalous frequency detection techniques is a good candidate for this type of sensor network. More recently, Xiao *et al.* [30] have developed a third order nonlinear polynomial graph filter (NPGF) to implement a residual-based outlier detector, with good results in the detection and localization of daily mean temperature outliers. These residual GSP-based techniques offer great outlier detection capabilities in the sensor network realm since they can locate which is the abnormal sensor measurement. However, heterogeneous air pollution

P. Ferrer-Cid (pau.ferrer.cid@upc.edu), J.M. Barcelo-Ordinas (jose.maria.barcelo@upc.edu) and J. Garcia-Vidal (jorge.garcia@upc.edu) are with the Department of Computer Architecture, Universitat Politecnica de Catalunya, Barcelona, Spain.

monitoring networks with reference stations and low-cost sensors have their own challenges, such as reporting data at the granularity of the reference station, e.g., hourly, which makes the amount of data to train an anomaly detection model small, or the fact that the signals may depend on emission sources such as vehicle traffic or industry. Most studies build graphs based on the geographical distance between nodes, although this approach performs well for some phenomena, networks that measure air pollution and other phenomena can be very complex. Therefore, as shown in [25], [31], the use of graphs learned from the data, resulting in a smooth structure with respect to the measured data, is a good candidate for these air pollution sensor networks, and the one we explore in this work. This approach is based on the fact of having a network of sensors where there are implicit relationships between the sensors that compose it so that the different sensors can benefit from the information of other sensors. This idea is in line with Heimann *et al.* [32] where it is explained that a dense network of sensors is needed to distinguish local air pollution emissions from regional emissions. Lately, it has been shown how sensors deployed in sparse areas without any information from nearby sensors, do not benefit from the network data nor from the graph modeling the network [31].

In this paper, we propose a decision process based on the use of a Volterra-based graph signal reconstruction (GSR) model [33] superimposed on a graph topology, which is learned from the network data using a signal smoothness criteria [34], to detect outliers in air pollution sensor networks. We call this algorithm *Volterra graph-based outlier detection* (VGOD). We perform several experiments on networks of reference stations in Spain measuring tropospheric ozone (O_3), as well as a sensor drift detection experiment using a small heterogeneous sensor network involving high-cost instrumentation and low-cost sensors, deployed in the Captor H2020 project [35]. Specifically, in this article we:

1) use a graph signal reconstruction Volterra-like model on top of a graph, whose edges are built based on graph signal smoothness criteria, as the principal components of the outlier detection process,
2) propose the VGOD; an outlier decision process that goes from graph learning and graph signal reconstruction, to the thresholding of the graph signals residuals,
3) show the model's ability to detect and identify signals and sensors with outliers in air pollution networks, and compare it with five state-of-the-art outlier detection algorithms,
4) show its application in the detection of sensor drift using a heterogeneous low-cost sensor network deployed by the H2020 Captor project.

The outline of this paper is as follows: section II shows the related work. Section III describes the proposed Volterra graph-based outlier detection process. Then, section IV introduces the data sets used in this paper, and section V shows the different experiments performed and their results. Finally, section VI presents the conclusions of the paper.

II. Related Work

Data quality in low-cost air pollution sensor networks: the enhancement of low-cost sensor technologies has enabled the study of their use for air pollution monitoring [2], [6]. In fact, despite being a less accurate solution than the government's instrumentation, they have proven to be useful in conjunction with the government's air pollution monitoring networks. Most of the literature focus on the use of machine learning techniques for the *in-situ* calibration of low-cost sensors [4], [5], [7]–[13], [36], with the aim of finding the best machine learning model and those features that improve the prediction of the measured pollutant. Nevertheless, these sensors are known to have aging and drifting problems as time progresses and environmental conditions change [16], [36], that is why the detection of sensor outliers is important for data correction, possible recalibration, or replacement of low-cost sensors.

Outlier detection in air pollution monitoring networks using univariate models: unsupervised models for detecting abnormal measurements in air quality monitoring networks is an important challenge [37], [38]. Most simple approaches use statistics such as the interquartile range or the z-score of samples from different sensors separately [14], [15], detecting whether the observed values correspond to extreme values with respect to the training set distribution. Nevertheless, this type of methods do not take into account the spatial distribution of the different sensors deployed together. In fact, in the field of spatial outlier detection, different studies use as statistic the difference between the value at one sensor with the mean or median [17] of the neighboring sensors at a given time instant without the need of a training stage. Beyond that, Shekhar *et al.* [39] describe spatial relationships with a graph where, instead of finding the nearest spatial sensors, the neighbors of a node are used. Kou *et al.* [40] do the same but using the weighted average of a node's neighbors measurements. A combination of these techniques is residual-based modeling, where a reconstruction model is fitted, and the observed value is compared to the value predicted by the model (potentially from nearby sensors) [15]. The benefit of all these techniques is that they compute a statistic per sensor, so the identification of the sensor that is causing the anomaly is implicit. Yet, most of these models are too simple to capture outliers that depend on other sensors jointly deployed (or other variables).

Outlier detection in air pollution monitoring networks using machine learning: there is also another type of outlier detection method used in environmental sensors, which are multivariate methods, where instead of looking sensor by sensor, the measurements of all sensors in the network are observed as a sample. In this way, machine learning methods such as local outlier factor (LOF) [41] or K-nearest neighbors (KNN) [16] have been used to detect whether an observation is anomalous. In addition, within the field of neural networks, many studies have been carried out using autoencoders to detect anomalies [19], [20]. Detection is performed by comparing the reconstructed vector with the observed values, as in the case of residual-based methods. This problem has also been tackled from another point of view, that of spectral decomposition, where using principal component analysis (PCA) it is assumed that normal information is contained in the components that explain more information, and anomalous changes affect the components with less information. Furthermore, Karkat *et al.*

[21] show how to identify which sensor is anomalous from the vector norm of the principal components that do not explain much information. However, most multivariate models do not naturally identify which sensor (or variable) is causing the anomaly, thereby limiting their use in this field.

Outlier detection using graph signal processing: recently, the field of graph signal processing has provided the possibility of using many classical signal processing and machine learning techniques on graphs [26], [27]. Thus, since sensor networks have already been modeled using graphs [25], outlier detection by studying signals over graphs is an important research topic, based on the fact that graphs learned from the data tend to be smooth with respect to the network measurements. In particular, Egilmez *et al.* [29] show how, in a similar way to the PCA, it is possible to perform data filtering with the graph, and through the Fourier transform detect an increase in high frequencies. Similarly, Gopalakrishnan *et al.* [42] directly use the signal smoothness, represented by total variation (TV), as a statistic to determine whether the joint signal from the sensors is anomalous, or at least different to the signals already observed in the training phase. However, the most important problem with approaching sensor outlier detection from a multivariate perspective (i.e. treating all measurements observed at a time step as one sample) is that it indicates that the entire sample is anomalous, but gives no clue as to which sensors are producing the anomalous values. Besides, graph neural networks have also been used for the detection of outliers for telemetry data [43]. However, neural networks are nonconvex models that present optimization difficulties when being fed by little training data [44], [45], requiring specific training methodologies. Thus, their use is limited in the field of low-cost air pollution sensor networks, where data for training are scarce. To overcome this problem, Xiao *et al.* [30] use a convex nonlinear polynomial graph filter (NPGF) to reconstruct the graph signals (temperature) and use a threshold on the differences of the reconstructed signal and the original signal to detect and locate the outliers. Thus, this residual-based method has proven to be a good alternative to other graph signal processing methods, as it is able to locate the abnormal sensors by inspecting the errors produced. Moreover, the NPGF has proven to be a convex alternative to neural networks with better outlier detection capabilities [46]. A shortcoming of these methods lies in the way the graph is constructed. Most of the proposed methods use as shifting matrix a matrix whose weights are calculated using a function that decays exponentially with the distance between nodes, and does not take into account the correlation of the data taken [28], [30].

Our proposal: In the area of air pollution monitoring networks, most of the work has focused on the calibration of low-cost sensors, and there is little research on how to detect outliers produced by air pollution sensors measuring signals such as O_3, NO_2 or $PM_{2.5}$. In order to benefit from the advantages of the field of graph signal processing and the intrinsic topology defined by a sensor network, we approach the problem of unsupervised sensor outlier detection from a graph-based perspective. Thus, in this paper, we propose an outlier detection process that first learns the graph encoded by the sensor network data, and then detects the outliers using a residual-based method based on a Volterra-like graph signal reconstruction model [33]. Indeed, this methodology poses three advances with respect to the literature:

1) Outlier detection methods for air pollution sensor networks are scarce, previously used methods include LOF, KNN and statistical methods [14]–[16]. Now, with the advent of graphs for air pollution monitoring networks [25], we propose a more complex graph-based outlier detection mechanism with localization capabilities, which allow the identification of outliers as in the case of drifting low-cost sensors in heterogeneous air pollution sensor networks.
2) While most previous work on graph-based outlier detection has used a graph based on distances between nodes [29], [30], we propose to use a graph learned from data. As discussed in [25], graphs learned from network data best describe complex networks than those using functions that decay exponentially with distance, such as low-cost heterogeneous sensor networks for air pollution monitoring. The choice of the shift matrix $\mathbf{S}$ defines the nodes' neighborhoods $\mathcal{N}(x_i)=\{x_j:\mathbf{S}_{ij}\neq 0\}$ and has impact on the signal reconstruction model as it participates in the shifting of the graph signals.
3) We apply a graph signal reconstruction model based on the classical Volterra series defined by Xiao et al. [33]. In fact, Volterra-like models have already been successfully applied to graphs [33], [47]. This model is similar to the NPGF model [30], which has proven a good performance in outlier detection, but requires fewer parameters to learn. This means a better computational response when reconstructing the signal.

These features give VGOD a better outlier detection capability compared to other algorithms such as LOF, KNN, NPGF, frequency-based GSP, or a linear graph filter (LGF). Finally, missing data are also a major problem in the use of low-cost sensors [25], [48], given the possible connectivity and data capture problems. Nevertheless, in this paper we focus on the outlier detection task, assuming that the data are complete when applying the graph signal reconstruction model. In fact, in the case of having missing data, these should first be imputed before going through the outlier detector.

III. Volterra Graph-Based Outlier Detection (VGOD) Process

The sensor network is described by means of a graph $\mathcal{G}$ defined as the triplet $\mathcal{G}=\{\mathbf{W},\mathcal{E},\mathcal{V}\}$, where $\mathbf{W}\in\mathbb{R}^{\mathbf{N}\times\mathbf{N}}$ is the weight matrix defining the relationship between nodes, $\mathcal{E}=\{e_{ij}: W_{ij}\geq 0\}$ is the set of edges, $\mathcal{V}=\{1,..,N\}$ is the set of nodes, and a signal defined over a graph is defined as the map $x:\mathcal{V}\rightarrow\mathbb{R}$ [26]. Table I summarizes the different symbols used throughout the paper. Bold lowercase symbols denote vectors, uppercase bold symbols denote matrices and lowercase symbols denote scalars. Throughout the following subsections we describe the three most important parts of the proposed *Volterra graph-based outlier detection* process; the graph learning, the graph signal reconstruction model, and the residual-based outlier detection.

TABLE I
NOTATION SUMMARY

SYMBOL	MEANING
N | P	Number of nodes | Number of observations
$\mathcal{G}$ | $\mathcal{V}$ | $\mathcal{E}$	Graph | Set of nodes | Set of edges
$\mathbf{S}$	Graph shift matrix
$\mathbf{A}$ | $\mathbf{W}$	Graph adjacency matrix | Graph weight matrix
$\mathbf{D}$ | $\mathbf{L}$	Graph degree matrix | Graph Laplacian matrix
x_i | $\mathcal{N}(x_i)$	ith vertex value | ith vertex neighborhood
$\mathbf{x}$ | $\odot$	Graph signal at a given time | Hadamard product
$\mathrm{tr}(\cdot)$ | $\|\cdot\|_2$	Trace of a matrix | l_2-norm of a vector
$\|\cdot\|_1$ | $\|\cdot\|_F$	l_1-norm of a matrix | Frobenius norm of a matrix
α | β	Graph learning hyperparameters
τ | N_{lof}	GSP hyperparameter | LOF hyperparameter
N_{knn} | W	KNN hyperparameter | Adaptive window
D | K	Graph signal reconstruction model degree | Depth
$\mathbf{0}$ | $\mathbf{1}$	Vector of zeros | Vector of ones

A. Graph learning: smoothness-based

The first step in our outlier detection process is discovering the underlying relationships between the sensors composing the network via a graph $\mathcal{G}$. So, given the training data $\mathbf{X}\in\mathbb{R}^{\mathbf{N}\times\mathbf{P}}$, the goal is to learn a graph shift matrix $\mathbf{S}$, which can be the weighting matrix $\mathbf{W}$ or the Laplacian matrix $\mathbf{L}$. The Laplacian matrix is object of study of the GSP [26], its combinatorial form is defined as $\mathbf{L}=\mathbf{D}-\mathbf{W}$, where $D_{ii}=\sum_j W_{ij}$ is the degree matrix. Moreover, the Laplacian eigendecomposition provides the Fourier basis for performing the graph discrete Fourier transform, and its quadratic form $\mathbf{x}^T\mathbf{L}\mathbf{x}$ provides the signal smoothness criterion. Many of the works related to environmental sensors such as temperature or air pollution construct the shift matrix using a distance exponential-decaying function between nodes [28], [30]. In this paper, we propose to construct the graph from the network data, thus creating edges between nodes that measure similar signals. Most of the optimization problems proposed in the literature to learn the graph connectivity from the data are based on the smoothness criterion and the imposition of a sparsity penalty on the resulting network. Some works focus on learning the weight matrix [49], and others focus on learning the Laplacian matrix directly [34]. For simplicity, we use the optimization problem defined by Dong *et al.* [34] since we adopt the Laplacian matrix as the graph shift operator $\mathbf{S}$ to detect outliers. Besides, this method has already been proved to efficiently describe air pollution sensor networks [25]. The Laplacian learning optimization problem solved in [34][1] is defined as:

$$\begin{aligned} \min_{L,Y} \quad & \|\mathbf{X}-\mathbf{Y}\|_{\mathbf{F}}^{\mathbf{2}} + \alpha\, \mathbf{tr}(\mathbf{Y^T L Y}) + \beta\|\mathbf{L}\|_{\mathbf{F}}^{\mathbf{2}} \\ \text{s.t.} \quad & tr(\mathbf{L}) = N, \\ & L_{ij} = L_{ji} \leq 0, \qquad i \neq j, \\ & \mathbf{L}\cdot\mathbf{1} = \mathbf{0}. \end{aligned} \tag{1}$$

Where $\mathbf{Y}$ is a filtered version of $\mathbf{X}$, and the hyperparameters $\alpha\in\mathbb{R}$ and $\beta\in\mathbb{R}$ control the smoothness and the sparsity of the resulting Laplacian $\mathbf{L}$. The network topology only needs to be learned once, with the training data, since it is assumed that the deployed sensors will at least work well for a certain period of time right after deployment. Thus, the temporal distribution of measurements may change, but the relationship between sensors, when working properly, is maintained over time. Once, the graph is learned from the data, the next step is to train a graph signal reconstruction model.

We would like to emphasize that one of the advantages of constructing the graph from measured sensor data rather than distances is that low-cost air pollution sensors have to be calibrated during deployment. This implies that a poorly calibrated sensor will actively participate in outlier detection in a distance-based method as the weights in the adjacency matrix $\mathbf{A}$ do not depend on how well calibrated the sensor is. In contrast, a poorly calibrated sensor will participate little if the Laplacian matrix $\mathbf{L}$ has been learned from the data, as the bad sensor data will be poorly correlated with the rest of the sensors.

B. Graph signal reconstruction: Volterra-based

As in all models based on the residual $\mathbf{R}(\mathbf{x})$ between a graph signal $\mathbf{x}$ and the reconstructed signal itself $\mathbf{f}(\mathbf{x})$, a signal reconstruction model is needed. In the VGOD mechanism, we apply a model similar to the Volterra series[2], recently defined by Xiao *et al.* [33]. For understanding purposes, we will now explain the relationship of the used model with the classical Volterra discrete model. The classical discrete Volterra model can be defined as:

$$y(t) = h_0 + \sum_{d=1}^{D}\sum_{\tau_1=a}^{b}\cdots\sum_{\tau_d=a}^{b} h_d(\tau_1,\ldots,\tau_d)\prod_{j}^{d} x(t-\tau_j) \tag{2}$$

Where $x(t)$ is a discrete signal defined at different time steps t, $h_d(\cdot)$ are the different learnable parameters of the model, D is the order of the Volterra series, and $x(t-\tau_j)$ can be seen as a signal shift by τ_j as in classical discrete signal processing. This model is known for being nonlinear and memory-based, since the output $y(t)$ depends on the inputs at previous times $x(t-\tau)$ in a nonlinear way.

Equivalently, the notion of signal shift [26] has been extended to the graph signal processing paradigm by applying a graph shift matrix $\mathbf{S}$ to a graph signal $\mathbf{x}$, $\mathbf{x}^{(1)}=\mathbf{S}\mathbf{x}$. The graph adjacency matrix $\mathbf{A}$ and the Laplacian matrix $\mathbf{L}$ have been widely used as the graph shift operator in the literature [26], [27]. In the specific case of a circular graph, the graph shift is equivalent to the signal shift in discrete signal processing. In this way, the Volterra-like model that describes the interactions between the signal at node x_i with the shifted versions of the signal at that node $(\mathbf{L}^j\mathbf{x})_i$ is defined in the following way:

$$\mathbf{f}(\mathbf{x}) = \mathbf{h_0} + \sum_{d=1}^{D}\sum_{k_d=0}^{K-1}\cdots\sum_{k_1=0}^{K-1} h_d(k_d,\ldots,k_1)\boldsymbol{\psi}_{d,k_d,\ldots,k_1}(\mathbf{x}) \tag{3}$$

[1]The authors of [34] provide the implementation of the graph learning problem.

[2]A python implementation of the proposed VGOD mechanism along with an implementation of the Volterra model [33] and the graph learning problem [34] using the CVXPY library are available at: http://sans.ac.upc.edu/?q=node/231.

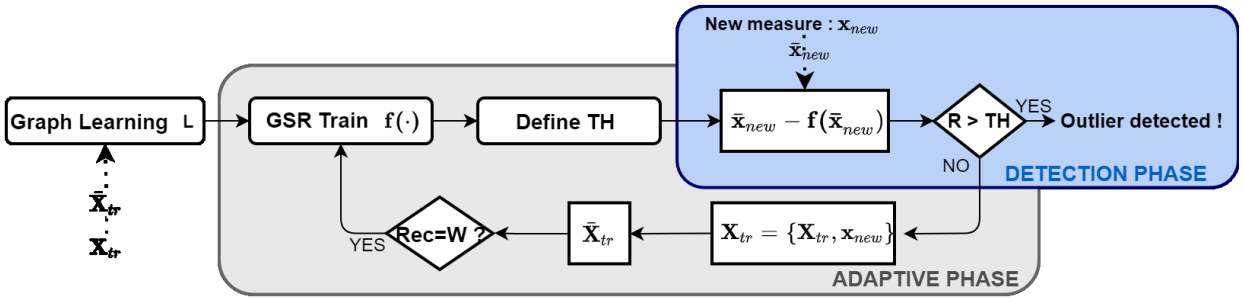


Fig. 1. General view of the VGOD process.

Where D is the order of the Volterra series, K is the maximum number of shifts to be applied (model depth), $\mathbf{h_0}\in\mathbb{R}^N$ and $h_d(k_d,\dots,k_1)\in\mathbb{R}^{K^d}$ are the parameters to be learned, and $\boldsymbol{\psi}_{d,k_1,\dots,k_D}(\mathbf{x})$ are the interactions defined as:

$$\begin{cases} \boldsymbol{\psi}_{1k_1}(\mathbf{x}) = \mathbf{L}^{k_1}\mathbf{x} \\ \boldsymbol{\psi}_{2k_2k_1}(\mathbf{x}) = (\mathbf{L}^{k_2}\mathbf{x}) \odot \boldsymbol{\psi}_{1k_1}(\mathbf{x}) \\ \dots \\ \boldsymbol{\psi}_{Dk_D\dots k_1}(\mathbf{x}) = (\mathbf{L}^{k_D}\mathbf{x}) \odot \boldsymbol{\psi}_{(D-1)k_{D-1}\dots k_1}(\mathbf{x}) \end{cases} \tag{4}$$

Where $\odot$ is the Hadamard product and k_i= $0,..,K-1$. For instance, the second order interactions take into account the interactions between the values at one node x_i (and its shifted versions) and the values at that node in its shifted versions $(\mathbf{L}^j\mathbf{x})_i$.

Now, the graph signal reconstruction model used for outlier detection is trained to recover the original signal $\mathbf{x}$ given a perturbed version of it $\tilde{\mathbf{x}}$=$\mathbf{x}$+ϵ, acting as a denoising model, and the following convex objective function is minimized to find the model's coefficients $\mathbf{h}$:

$$\min_{\mathbf{h}} \sum_i \|\mathbf{x_i} - \mathbf{f}(\tilde{\mathbf{x}}_\mathbf{i})\|_2^2 \tag{5}$$

When an unusual perturbation is present in a signal the model will incur in a larger error, being capable of identifying the anomalous node given the residuals $\mathbf{R}(\mathbf{x})$=$|\mathbf{x}-\mathbf{f}(\mathbf{x})|$. This problem constitutes a convex optimization problem since it is linear with respect to the coefficients of the model $\mathbf{h}$, so its optimization is easier than in nonconvex models such as neural networks. This is of special interest in cases such as the monitoring of air pollution sensor networks, where the training periods used to learn the models may be relatively small. We also expect the choice of the shift operator to have a very high impact on the model. In general, for many signals the distance between nodes does not have to be a good choice for generating the shift operator, as shown in [25].

Although the model can be trained correctly using a few weeks of data, air pollution data can suffer from a problem known as data set shift, commonly known as non-stationarity in the field of time series analysis. This is because the data present in the training set may not be representative of the testing set (or the posterior deployment conditions), e.g., mean concentrations may vary from month to month. Therefore, special care must be taken when applying the graph signal reconstruction model. A common approach to overcome this problem [50] consists of updating the detection model periodically with new data. For example, we can incorporate into the training set the samples predicted as normal during the test phase, and retrain the signal reconstruction model every time we have W normal samples. This increases the computational burden, but as we are solving convex optimization problems the increase remains bounded. However, it is important to keep the complexity of the models limited, i.e. their depth or number of learnable parameters, so that it is feasible to retrain them periodically.

C. Outlier detection: residual-based

Now, having learned the graph $\mathbf{L}$ and trained the graph signal reconstruction (GSR) model $\mathbf{f}(\cdot)$, the remaining stage is the identification of outliers through the inspection of the signal reconstruction residuals $\mathbf{R}(\mathbf{x})$, it is to say, the difference between the observed signal $\mathbf{x}$ and the reconstructed signal $\mathbf{f}(\mathbf{x})$:

$$\mathbf{R}(\mathbf{x}) = |\mathbf{x} - \mathbf{f}(\mathbf{x})| \tag{6}$$

Normal samples are supposed to have small residuals since the model has been trained with a similar pattern, while abnormal samples tend to have larger residuals, as they deviate from the normal pattern seen during the training. Then, using a thresholding value, TH, an indicator function can be implemented:

$$I_i(\mathbf{x}) = \begin{cases} 1 & , R_i(\mathbf{x}) > TH \\ 0 & , R_i(\mathbf{x}) \leq TH \end{cases} \tag{7}$$

where $I_i(\mathbf{x})$=1 indicates that the node x_i is the suspicious one. Other outlier scoring metrics can be used to detect outliers, for instance, if we are not interested in locating the error, but in indicating whether the whole sample is an outlier, using the norm of the residual $\|\mathbf{R}(\mathbf{x})\|_2$ can be useful to find abnormal graph signals. The threshold value TH can be defined depending on the application target performance, it is to say, the maximum false positive rate (FPR) or the minimum true positive rate (TPR) required by the application [29], [30]. This threshold TH can also be recomputed along with the signal reconstruction method to better adapt to the non-stationarity nature of the data.

Figure 1 summarizes the overall outlier detection process developed while algorithm 1 gives a precise description of the process. The outlier detection process parameters are; $\{\alpha,\beta\}$ hyperparameters to control the graph learning algorithm, the training data $\mathbf{X}_{tr}$, model depth K, model order D, acceptable

TABLE II
STATISTICS OF THE DATA SETS USED.

DATA SET LABEL	POLLUTANT	# NODES	# SAMPLES	PERIOD	RESOLUTION	MEAN ($\mu gr/m^3$)	POOLED STD. ($\mu gr/m^3$)
D.1	O_3	14	2798	2019/01/01 - 2019/05/31	1 h	45.32	31.78
D.2	O_3	43	1076	2021/09/01 - 2021/11/01	1 h	50.79	25.72
D.3	O_3	8	2368	2017/06/18 - 2017/09/01	30 min	68.82	35.14

Algorithm 1 Volterra Graph-Based Outlier Detection(VGOD).

Input: $\{\alpha, \beta, \mathbf{X}_{tr}, K, D, fpr, \epsilon, W\}$
$\bar{\mathbf{X}}_{tr} \leftarrow$ Standardization$(\mathbf{X}_{tr})$
$\mathbf{L} \leftarrow$ Graph_Learning$(\alpha, \beta, \bar{\mathbf{X}}_{tr})$
$\mathbf{f}(\cdot) \leftarrow$ Reconstruction_Model$(\bar{\mathbf{X}}_{tr}, K, D)$
$TH \leftarrow$ Define_Threshold$(\mathbf{f}(\cdot), \bar{\mathbf{X}}_{tr}, fpr, \epsilon)$
$rec \leftarrow 0$
while $\mathbf{x}_{new}$ **do** ◁ *Detection Phase*
  $\bar{\mathbf{x}}_{new} \leftarrow$ Standardization$(\mathbf{x}_{new})$
  $\mathbf{R}(\bar{\mathbf{x}}_{new}) \leftarrow \mid \bar{\mathbf{x}}_{new} - \mathbf{f}(\bar{\mathbf{x}}_{new}) \mid$
  if $R_i(\bar{\mathbf{x}}_{new}) > TH$ **then**
    $x_{new,i}$is outlier !
  else ◁ *Adaptive Phase*
    $\mathbf{X}_{tr} \leftarrow \{\mathbf{X}_{tr}$,$\mathbf{x}_{new}\}$
    $rec \leftarrow rec + 1$
    if $rec = W$ **then**
      $\bar{\mathbf{X}}_{tr} \leftarrow$ Standardization$(\mathbf{X}_{tr})$
      $\mathbf{f}(\cdot) \leftarrow$ Reconstruction_Model$(\bar{\mathbf{X}}_{tr}$,K,$D)$
      $TH \leftarrow$ Define_Threshold$(\mathbf{f}(\cdot), \bar{\mathbf{X}}_{tr}, fpr, \epsilon)$
      $rec \leftarrow 0$
    end if
  end if
end while

maximum false positive rate fpr to define the threshold, the perturbation ϵ to be introduced to define the threshold, and the model updating window size W. $\{\alpha, \beta, K, D\}$ are hyper-parameters that are obtained based on the training data $\mathbf{X}_{tr}$, while $\{fpr, \epsilon, W\}$ are user-defined parameters that depend on the specific data domain on which the algorithm is used.

IV. DATA SETS

To study the performance of the proposed outlier detection process for air pollution data sets, we use two different types of data. First, we use two data sets provided by the Spanish government consisting of forty-three nodes deployed in the Catalonia area, and fourteen nodes deployed in the Barcelona city area. These data are public and can be downloaded at the Catalonia open data web page[3]. In this way, we can simulate outliers and see how outlier detection works for tropospheric ozone sensor networks. Secondly, we use a data set collected by a heterogeneous network consisting of five low-cost sensors and three reference stations, deployed during summer 2017 for the H2020 Captor project, to detect drifting sensors. In summary, we experiment with the following three data sets:

1) *Spanish air pollution reference station network for O_3 for Barcelona city area*: this data set is formed by fourteen nodes, capturing hourly tropospheric ozone data between the months of January and May of 2019, with a total of 2798 samples.
2) *Spanish O_3 reference station network for Catalonia*: this data set is formed by forty-three nodes in the area of Catalonia capturing hourly tropospheric ozone data between the months of January and February of 2021, with a total of 1076 samples.
3) *H2020 Captor network [35]*: this data set is formed by eight nodes, five low-cost sensors and three reference stations, deployed in the area of Catalonia (Spain) during the summer of 2017 to capture half-hourly tropospheric ozone concentration levels. This data set has a total of 2368 samples.

These data sets are representative of air quality monitoring networks. The first two data sets correspond to governmental reference stations, while the third corresponds to a hybrid network of governmental reference stations and low-cost sensors. Table II shows the statistical characteristics of the three data sets. In addition, heterogeneous data from the Captor network allows us to explore one of the most important outlier detection applications in sensor networks, the detection of drifting or malfunctioning sensors. Large air pollution monitoring sensor networks can be reduced to smaller subnetworks using clustering techniques [31]. This reduces the computational cost, without losing the ultimate goal of the graph-based method, which is to detect anomalies using neighboring nodes selected with an algorithm that learns the connectivity of the graph based on a smoothness criterion.

V. RESULTS

This section shows the performance of the VGOD algorithm explained in section III applied to the real air quality monitoring data sets described in section IV. The proposed model is compared with other state-of-the-art outlier detection methods. In particular, we compare VGOD with i) outlier detection algorithms using global models which do not allow localization, such as the frequency-based GSP [29], the local outlier factor (LOF) [41], and the k-nearest neighbors (KNN) [16], and ii) with models based on reconstruction residuals and graph signal processing, such as the linear graph filter (LGF, $\mathbf{f}(\mathbf{x})$=$\mathbf{h_0}$+$\sum_{i=0}^{K-1} h_{1i}\mathbf{S}^i\mathbf{x}$) [28], and the third order NPGF model with a distance-based graph [30]. The data sets are divided into 60% of the data for training, and the remaining 40% for testing. Thus, the first 60% of the samples are used as training, and the remaining 40% as testing, mimicking the real case where the outlier detection model is trained just after the sensor network deployment and applied sequentially throughout the deployment lifetime. Four experiments are conducted, divided into the following five sections:

[3]https://analisi.transparenciacatalunya.cat/en/Medi-Ambient/Qualitat-de-l-aire-als-punts-de-mesurament-autom-t/tasf-thgu

(A) *Model training & selection*: the different models' parameters are described for both global and residual-based models, as well as the selection process of their hyperparameters.
(B) *Outlier detection over the training set*: outliers are simulated on the training of the data set *D.1*. The different models are applied non-adaptively on the training data set, i.e., they are trained using the training data set and the detection is also performed on the training data set. This simulates the best case, where the data distribution in the detection phase does not change. Such experiments allow us to analyze which parameters internal to the models will be used later in testing, e.g., the model depth K.
(C) *Outlier detection over the testing set*: outliers are simulated in the testing of the data sets *D.1* and *D.2*. The models are applied adaptively, as shown in the algorithm 1 since the data distribution changes over time.
(D) *Sensor drift detection*: a sensor drift is simulated in the testing set of data set *D.3*. As the data set *D.3* contains both sensors and reference stations, a malfunctioning sensor can be simulated. Again, the models are applied adaptively to detect and locate the drifting sensor.
(E) *VGOD scalability*: the scalability of the two best performing graph-based models, the graph signal reconstruction model using the Volterra-based model and the NPGF model, is compared.

The perturbations δ added to simulate the outliers have no units since these perturbations are introduced to the standardized data. Indeed, it is fairer to add perturbations proportional to the standard deviation of each of the sensor nodes.

A. Model training & selection

In this section, we explain what are the parameters of the different models, explaining from how to learn the graph and the signal reconstruction model, to how to define the thresholds. We assume that the sensors have no outliers during the graph learning and signal reconstruction training phases. The different resulting hyperparameters for the different models are summarized in Table III.

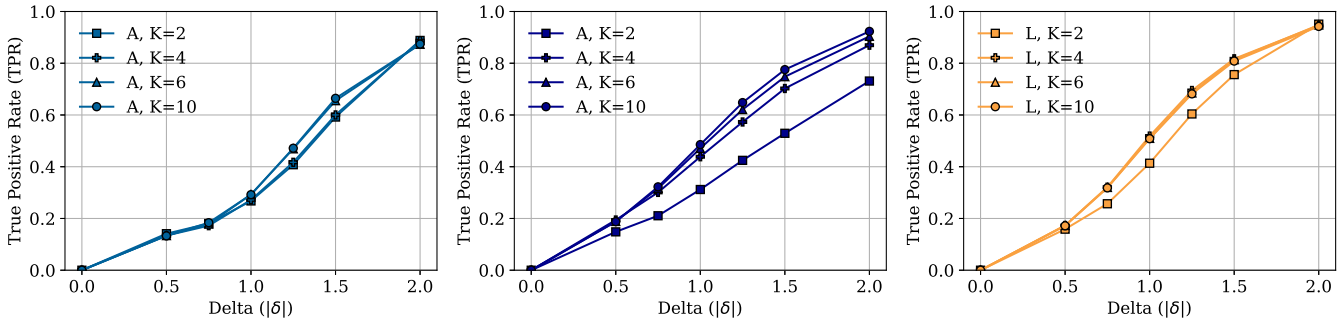
TABLE III
MODELS' HYPERPARAMETERS.

MODEL	INPUTS		HYPERPARAMETERS
	Shift Matrix S	Data	
Linear Graph Filter (LGF) [28]	$\mathbf{A}$	$\mathbf{X}_{tr}$	Depth (K)
Third Order NPGF [30]	$\mathbf{A}$	$\mathbf{X}_{tr}$	Depth (K)
VGOD	$\mathbf{L}$	$\mathbf{X}_{tr}$	$\{\alpha, \beta\}$, Depth (K), Order(D)
Frequency-based GSP [29]	$\mathbf{L}$	$\mathbf{X}_{tr}$	Variance kept (τ)
Local Outlier Factor (LOF) [41]		$\mathbf{X}_{tr}$	Neighbors (N_{lof})
K-nearest Neighbors (KNN) [16]		$\mathbf{X}_{tr}$	Neighbors (N_{knn})

1) Graph learning: residual-based models require a shift matrix $\mathbf{S}$ to perform the graph signal reconstruction, for example the adjacency matrix $\mathbf{A}$ or the Laplacian matrix $\mathbf{L}$. The state-of-the-art models LGF and NPGF use a distance-based adjacency matrix $\mathbf{A}$ as defined in [28], to define the relationships between the different network sensors. The VGOD process uses a Laplacian matrix learned from the network data using a signal smoothness criterion [34], i.e. based on the data collected during the training. As seen in [25], air pollution sensor networks encode highly complex relationships, which are best described by a data-driven graph, thus learning the Laplacian matrix implies learning more meaningful relationships. The Laplacian matrix $\mathbf{L}$ is learned from the data using the training set $\mathbf{X}_{tr}$ and the values of the hyperparameters $\{\alpha, \beta\}$, which control the sparsity of the graph in the optimization problem shown in section III-A. In this case, we choose a dense graph with a high number of neighbors per node, so that all nodes have enough neighboring information to detect the outliers. Further information on how to learn graphs for air pollution networks and the effect of different $\{\alpha, \beta\}$ values can be found in [25].

2) Graph signal reconstruction: once the shift matrix $\mathbf{S}$ has been learned, the second step is to train the reconstruction models to remove noise, as if it were a signal denoising task, by taking the training set $\mathbf{X}_{tr}$ and adding noise to a variable percentage of nodes for each signal, so using as input an artificially perturbed version $\tilde{\mathbf{X}}_{tr}$ of the training data. Regarding the hyperparameters of the signal reconstruction models, we have the filter depth K, which indicates the maximum number of shifted versions of the signal taken into account, and therefore controls the model complexity. As a general rule, the maximum value of K is set to the degree N_m of the minimal characteristic polynomial of the shift matrix $\mathbf{S}$, that is $K \leq N_m \leq N$. As the graph diameter in dense graphs is low, and we want the models to extrapolate to the test set, we explore simple models with small depths. The model based on the Volterra series [33] also includes the model order D, which indicates the maximum degree of interactions to take into account. This parameter drastically affects the complexity of the model as well as the number of parameters of the model. But in this experiment, we only take into account the third order model, D=3, so that we can fairly compare this model with the third order NPGF shown in [30]. The best filter depth K value is found by adding artificial outliers randomly in the training, and obtaining the K that corresponds to the best true positive rate and the least complex model so that it generalizes better, a common procedure used in the literature [29], [30]. Next section V-B shows the selection of the best K results. Once the graph signal reconstruction models are trained, we find the corresponding threshold TH for each model above which the residual $R_i(\mathbf{x})$ considers that sensor x_i is a possible outlier. The selection of the threshold can be done in different ways, but the most common choice depends on the false positive rate (FPR) or false alarm, and true positive rate (TPR) or probability of hit required by the application. Since the outlier detection process is used to maintain the network data quality, the TPR is maximized and the acceptable rate of false positives is set to 10%. In the paradigm of sensor data quality, it is important to have a high sensitivity (true positive rate) and the fact of having any false positive does not imply any high-cost action (e.g. sensor replacement). The decision on the value of W is shown in section V-C, where the adaptive application of the different models (adaptive phase, algorithm 1) is explained.

3) Global models: As for the global models, the frequency-based GSP needs the Laplacian $\mathbf{L}$, the training data $\mathbf{X}_{tr}$,



(a) Outlier detection using the LGF. (b) Outlier detection using the third order NPGF. (c) Outlier detection using the VGOD process with the third order Volterra-based graph model.

Fig. 2. Average true positive rates results for ten different repetitions and different perturbation magnitudes ($|\delta|$) using the residual-based models for data set *D.1*.

and the τ hyperparameter, which indicates the amount of variance retained by the selected frequencies as the normal components of the signal. The LOF uses the training data $\mathbf{X}_{tr}$, and the number of neighbors N_{lof} to take into account to compute the outlier score. Finally, the KNN uses the training set $\mathbf{X}_{tr}$ as a dictionary to compute distances, and the number of neighbors N_{knn} to consider when computing the distance. The hyperparameters and the thresholds for these models are selected in the same way as in the case of residual-based models.

B. Outlier detection over the training set

This section shows the results of applying the different models in a non-adaptive way in the training so that the models are trained and used on the same data. This process allows exploring the best case, where the data distribution does not change as the opposite of the adaptive case, and allows different hyperparameters to be examined and set as a baseline for testing. This is important because in the adaptive application case the model is retrained with an increasing training set, and the threshold is also recomputed based on the most recent samples of this same increasing training set.

Models are trained with different values for their hyperparameters, Table III. The only tested parameter for the VGOD is the filter depth K since the other have been set in the previous section. To this end, 30% of the training set is perturbed by adding different outlier perturbations δ (delta) at random, and ten repetitions are performed.

Figures 2a), b) and c) show the average TPR for the different residual-based models and a FPR of 10%. As for the depths of the models, it can be seen how from K=4 on wards the improvement for all three models is very little. Therefore, K=4 seems to be a good choice to keep the models' complexity bounded. In fact, for perturbations of one standard deviation ($|\delta|$=1) the LGF obtains a 27% TPR, the NPGF obtains around a 44% TPR, and in the case of the VGOD the TPR is 52%. Thus, using the combination of a graph learned from the data and the Volterra-based reconstruction outperforms the other two residual-based models using the distance-based adjacency matrix. Looking at the VGOD results we see that with a perturbation of $|\delta|$=1.0 standard deviation obtains a TPR of 52%, with perturbations of $|\delta|$=1.25 standard deviations the TPR reaches 69%, and with perturbations of two standard deviations it can nearly detect all the outliers with a TPR of 95%. Recall that in this case one standard deviation is approximately 31 $\mu gr/m^3$, this means that readings with deviations of around 39$\mu gr/m^3$ (1.25 standard deviations) are effectively detected with a 69% TPR. This value is actually good, since air pollution sensing nodes exhibit large variability, so stating that a measurement is an outlier can be difficult. Thus, this outlier detection process provides the necessary tools to detect outlying measures and sensors. Indeed, the proposed model can slightly improve the results of the NPGF even when this has larger depths (more complexity).

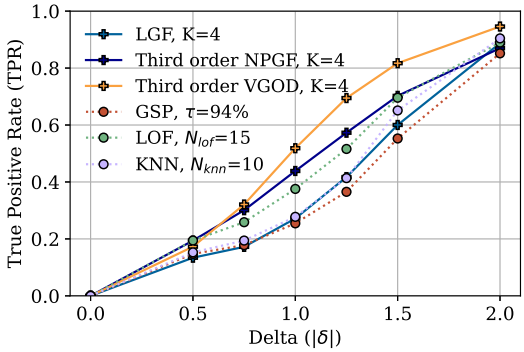


Fig. 3. Average true positive rate for the different models for data set *D.1*.

Now, Figure 3 compares the TPR for the residual-based models, with K=4, with the TPR for the three global models (frequency-based GSP, LOF, and KNN) with their best hyperparameters. It is clearly seen that the nonlinear residual-based models perform better than the global models. Indeed, VGOD is able to improve the detection rate by more than a 10% for perturbations greater than $|\delta|$=0.75 standard deviations. The NPGF improves the results of LOF by a 5% of TPR for perturbations in the range of $|\delta|$=0.5-1.5 standard deviations. The frequency-based GSP has a similar performance to the LGF, since they both use the high-frequency components of the signal to detect outliers, but the LGF performs slightly better than the frequency-based GSP for high magnitude

TABLE IV
AVERAGE OUTLIER DETECTION RESULTS OVER THE TEST SET WITH δ STANDARD DEVIATION PERTURBATIONS FOR DATA SET *D.1*.

MODEL	$\|\delta\|$=0.0		$\|\delta\|$=0.5		$\|\delta\|$=1.0		$\|\delta\|$=1.25		$\|\delta\|$=1.5		$\|\delta\|$=2.0	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
LGF [28]	0.0	0.19	0.20	0.18	0.31	0.15	0.41	0.14	0.56	0.13	0.80	0.12
Third order NPGF [30]	0.0	0.19	0.24	0.17	0.40	0.13	0.54	0.13	0.64	0.12	0.82	0.12
Third order VGOD	0.0	0.18	0.23	0.16	0.41	0.13	0.56	0.12	0.71	0.12	0.91	0.13
LOF [41]	0.0	0.15	0.20	0.15	0.32	0.15	0.45	0.14	0.61	0.15	0.84	0.16
KNN [16]	0.0	0.15	0.18	0.16	0.26	0.16	0.36	0.16	0.49	0.16	0.83	0.16
Frequency-based GSP [29]	0.0	0.15	0.19	0.15	0.34	0.15	0.48	0.15	0.61	0.15	0.78	0.16

perturbations. The KNN is observed to have a similar performance to GSP and LGF, with slightly higher detection capabilities for large perturbations. In addition to their lower detection capabilities, global models are not able to localize which one of the sensors in the network is producing the outlier, and this limits their application in real sensor network deployment scenarios. In the following section, we show the detection results using the adaptive algorithm to deal with unseen data distributions, as well as we show the localization abilities of the models.

C. Outlier detection over the testing set

Once we have seen how the different models work on the training, let's check how they work when applied adaptively, as for algorithm 1, on the test set. As already mentioned, in non-stationary environments is necessary to update the models by introducing samples with the new data distribution to adapt them. To this end, we use a time window of 10 samples (W=10), which is equivalent to recalculating the model once ten samples are considered normal. This approach is feasible for hourly measurements since in the best case the model would need to be recomputed every ten hours. Smaller time windows (e.g. W=1) could lead to problems depending on the data availability, the model's complexity, and the required training time. Thus, in the adaptive approach we add the new samples considered as non-outliers to the training set. This parameter is user-defined since its value will always depend on the specific data domain of the application and data resolution. In addition to recalculating the model, we recompute the threshold using the latest samples collected. We apply the same adaptive procedure for all models, global and residual-based. Perturbations of different magnitude δ are applied to the 30% of the test set, and five repetitions per perturbation magnitude are performed.

Table IV shows the average TPRs and FPRs for the selected models and perturbations of different magnitude δ. The same trend is observed as in the training results but with slightly lower TPRs in general. Firstly, in the case of the true positive rates, the VGOD process is the best method followed by the NPGF, in particular, the VGOD is able to improve the NPGF by about 2.5-11% TPR for outliers in the range between $|\delta|$=1.0-2.0 standard deviation. In general, VGOD and NPGF are better able to detect outliers with smaller perturbations, e.g. in the range of $|\delta|$=1.0, thus showing better sensitivity in these ranges than LGF, LOF, KNN and frequency-based GSP. For large perturbation values ($|\delta|$=2.0), all methods show a similar high ability, in the order of 78-84%, to detect outliers, except VGOD which goes up to 91%, outperforming all other methods.

On the other hand, Table IV also shows the FPR committed by the different models. In fact, given the non-stationarity of the test set, we can see FPRs around 15-19% for $|\delta|$=0, although in the case of the residual-based methods (LGF, NPGF, and VGOD) these FPRs are reduced to 12-14% as the magnitude of the outliers increases. That is, as outliers are larger in magnitude the residual-based models have better sensitivity and also produce fewer false positives. Depending on the characteristics of the data set and the computational capabilities, the adaptive window W can be reduced to improve the models.

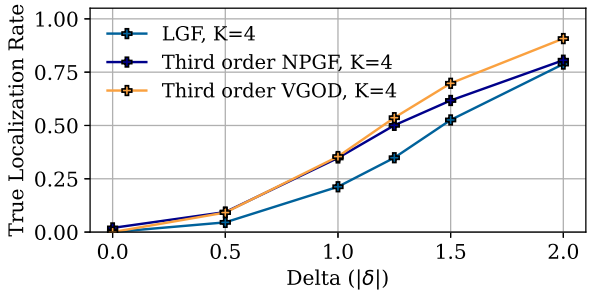


Fig. 4. Localization rate test set results for the three residual-based models for data set *D.1*.

Now, let's see how the residual-based models work with respect to the localization of the sensor that has the outlier. This step is very important in sensor networks in order to carry out actions to mitigate the effects of the outlier, actions such as the imputation of the sensor measurement, the removal of the measure, and even the replacement of the sensor if it malfunctions. Figure 4 shows the true localization rate results for the test set, this rate is defined as the precise detection of the outlying sensor, where the localization rates are slightly smaller than the detection rate, meaning that sometimes the models fails in locating the specific outlying sensor. However, the results show how VGOD outperforms the NPGF and the LGF. For perturbations of 1.0 standard deviation the two nonlinear models behave similarly with a location rate of about a 37%. Nevertheless, as the perturbation magnitude increases the performance gap between the three models also increases, leading to a localization rate of 70% with a 1.5 standard deviation perturbations with the VGOD process, more than

10% higher location rate than the others.

Figure 5, shows the outlier localization results for the data set comprising the Catalonia reference stations (*D.2*). The same trend as in the previous case is verified, where the nonlinear models have a better localization for outliers of magnitude in the range $|\delta|$=1.0-1.5 standard deviation. However, the gap between the localization performance of the VGOD and the NPGF is larger in this case for outliers of magnitude in the range $|\delta|$=1.0-2.0 standard deviation since the network is very heterogeneous, and a graph learned from the data captures better the relationships between nodes. The sensor network represented by the *D.2* data set includes sensors whose relationships are not well defined by distances, which is a common scenario in air pollution sensor networks whose nodes are deployed in specific locations with high concentrations of air pollution, and whose signals are highly dependent on ambient conditions and other pollutants. Therefore, outlier detection models using distance-based graphs do not work well in that case given that the graph signal reconstruction stage is distorted by erroneous sensor-to-sensor relationships. This problem does not happen with a graph learned from the data since uncorrelated sensors are weakly connected to other sensors or disconnected, proving to be a more robust alternative for air pollution monitoring networks.

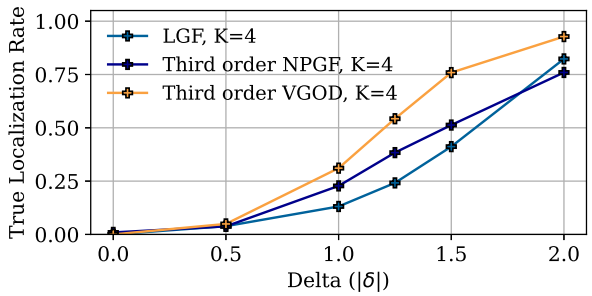


Fig. 5. Localization rate test set results for the three residual-based models for data set *D.2*.

D. Application: sensor drift detection

Let us now check the performance of the proposed outlier detection process for a special type of sensor error, sensor drift. Since an outlier detection model detects samples that have an unusual behavior, this technique can be further used to detect specific sensor errors by the inspection of the outlier detection results. Here, we use the data set of a real heterogeneous network deployment, described in section IV, composed by three reference stations (high-precision nodes) and five low-cost sensor nodes. Thus, forming an heterogeneous air pollution monitoring sensor network, to simulate a drifting sensor. To this end, we add an offset of increasing magnitude in one of the low-cost sensors, as $\epsilon_t \sim N(2.0/t, 0.1)$ and $t \in (0, 2.0]$. Figure 6 shows the result of applying the VGOD mechanism, where sensor 3 is the drifted sensor, and the model is able to detect the simulated drift after its magnitude nearly becomes half standard deviation. The obtained TPR is of 78% and a FPR of 13%, the linear graph filter obtained a TPR of 58% and a FPR of 10%, and finally, the NPGF obtained a TPR of 70% and a FPR of 13%. Again our proposed model outperforms the other two graph-based models. This example shows the importance of these graph-based techniques that enable the localization of the faulty sensor. In fact, given the localization capability of this model, this type of sensor malfunction can be detected, and the sensor can be replaced or can undergo a recalibration process.

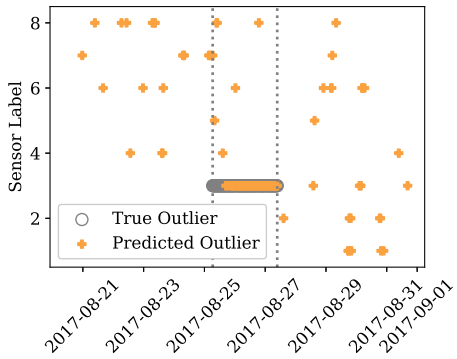


Fig. 6. Outlier detection results for a drifted sensor over the H2020 Captor data set.

E. VGOD scalability

VGOD has been the best performer in the different experiments, followed by the other state-of-the-art graph-based method, the third order NPGF [30]. It has been seen that these graph-based methods are a great option for outlier detection and localization for this type of network. In this section, we compare the scalability of the core element of the VGOD, the Volterra-based graph signal reconstruction model [33], with the third order NPGF [30]. There are two main differences between their outlier detector and ours: i) the authors in [30] use a shift operator built from a distance-based function between nodes, and we propose to use a shift operator that is built using the data measured by the nodes, and ii) the authors in [30] use the NPGF, whose structure is similar to the Volterra-based model but the higher order interactions differ. Indeed, the number of NPGF parameters for a degree of interaction D and depth K is $(N + K + K^2 + \sum_{i=1}^{D-2} K^2 N^i)$, while the number of the parameters for the Volterra-based model is $(N + \sum_{i=1}^{D} K^i)$. Table V compares the number of parameters for third order models (D=3) with models' depth four (K=4), for different network sizes N.

TABLE V
NUMBER OF PARAMETERS FOR THIRD ORDER NPGF AND THIRD ORDER VOLTERRA MODEL, WITH K=4.

MODEL	**N=10**	**N=50**	**N=100**
Third order NPGF	190	870	1720
Third order Volterra model	94	134	184

Now, to show how the number of model parameters of both, the graph signal reconstruction Volterra model and NPGF

model, affect the solving time of the convex problem in equation (5) we perform the following experiment: given a certain depth K=4, we simulate data sets with increasing number of nodes N and increasing number of samples P. To do this, we simulate the sensors coordinates in the unit square as $c_x, c_y \sim U(0,1)$, and define an adjacency matrix by the distance-based radial basis function $A_{ij}=e^{\frac{-d(i,j)^2}{2*0.5}}$, where $d(i,j)$ is the distance between sensor x_i and sensor x_j. The samples are generated as a zero-mean multivariate Gaussian with precision matrix equal to the Laplacian pseudoinverse with noise injected on the diagonal $\mathbf{x} \sim N(\mathbf{0}, L^{\dagger}+\sigma \mathbf{I}_N)$. Then, for each pair ($N$,$P$) we perform five repetitions to calculate the average solving time. Figure 7 shows the optimization solving times for both models. In fact, the Volterra-based model is invariant to the number of nodes in the network, resulting in much lower solving times as the number of network nodes increases. The opposite happens with the third order NPGF, where the solving time increases dramatically as the number of nodes increases. For instance, for 1000 samples and 46 nodes the third order NPGF takes almost 8 minutes, while the third order Volterra model takes just over 1 minute.

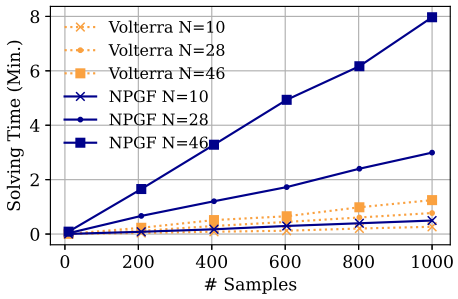


Fig. 7. Optimization problem solving times, using the Splitting Conic Solver (SCS) for the third order Volterra-based and the third order NPGF models.

VI. Conclusion

In this paper, we have proposed a novel outlier detection mechanism named *Volterra graph-based outlier detection* (VGOD) based on graph signal processing. The detection process consists of three stages: learning a graph based on the measured data, a graph signal reconstruction model based on the Volterra series, and the subsequent inspection of the residuals of the signal reconstruction task to identify and locate the outlying measurements. This process allows not only to detect an outlier in a sensor network sample, but also to localize the sensor that produces the outlier, which is of great importance in the air pollution sensor network realm so that replacement or recalibration actions can be done.

In summary, the VGOD method uses a shift matrix that is constructed using the measured data, unlike other graph-based methods that use shift matrices based on functions that decay exponentially with the distance between nodes. This aspect is key to the method as the shift matrix actively participates in two modules of the outlier detector, i) the selection of the edges of the graph and therefore of the neighbors of a node, and ii) in the graph signal reconstruction model. This feature improves the detection and localization of outliers. The second differential aspect is the use of Volterra series as a signal reconstruction method, which improves the computational performance by requiring fewer parameters than other nonlinear methods, such as NPGF.

The VGOD process has been compared to three state-of-the-art global outlier detection methods that detect but do not allow localization, the frequency-based GSP, the k-nearest neighbors (KNN), and the local outlier factor (LOF), and to two models based on reconstruction residuals and graph signal processing that detect and allow localization, the linear graph filter (LGF) and the third order NPGF model with a distance-based graph. The results show that VGOD increases the detection rate by at least 10% over the other models and has better localization of the sensors producing the outliers than the other two graph-based models. In addition, it is shown that the VGOD reconstruction model requires less training time than its closest graph-based competitor, the NPGF. Therefore, the proposed mechanism improves both outlier detection and model scalability with respect to NPGF.

Finally, the VGOD graph-based detection model has been applied to sensor drift detection in a low-cost heterogeneous sensor network. The results show the ability of the proposed method to detect the outlying samples and locate the drifting sensor, thus allowing the identification of the drifting sensor for a possible replacement or sensor recalibration. Regarding the method's weaknesses, it is worth mentioning that in the case of having a network with sensors deployed in sparse areas without significant relationships, the method may not be able to detect outliers in those sensors. In addition, the mechanism needs the graph to be learned from correct sensor values, so it is assumed that the sensors will function well during the training phase. As future work, it would be interesting to study the applicability of graph neural networks for air pollution low-cost sensor network outlier detection, with specific training techniques to deal with training on small data sets and the need for periodic retraining.

Acknowledgment

This work is supported by the National Spanish funding PID2019-107910RB-I00, by regional project 2017SGR-990, and with the support of Secretaria d'Universitats i Recerca de la Generalitat de Catalunya i del Fons Social Europeu.

References

[1] W.-J. Guan, X.-Y. Zheng, K. F. Chung, and N.-S. Zhong, "Impact of air pollution on the burden of chronic respiratory diseases in china: time for urgent action," *The Lancet*, vol. 388, no. 10054, pp. 1939–1951, 2016.

[2] D. E. Williams, "Low cost sensor networks: How do we know the data are reliable?" *ACS sensors*, vol. 4, no. 10, pp. 2558–2565, 2019.

[3] N. H. Motlagh, E. Lagerspetz, P. Nurmi, X. Li, S. Varjonen, J. Mineraud, M. Siekkinen, A. Rebeiro-Hargrave, T. Hussein, T. Petaja *et al.*, "Toward massive scale air quality monitoring," *IEEE Communications Magazine*, vol. 58, no. 2, pp. 54–59, 2020.

[4] P. Ferrer-Cid, J. M. Barcelo-Ordinas, J. Garcia-Vidal, A. Ripoll, and M. Viana, "A comparative study of calibration methods for low-cost ozone sensors in iot platforms," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9563–9571, Dec 2019.

[5] L. Spinelle, M. Gerboles, M. G. Villani, M. Aleixandre, and F. Bonavitacola, "Field calibration of a cluster of low-cost available sensors for air quality monitoring. part a: Ozone and nitrogen dioxide," *Sensors and Actuators B: Chemical*, vol. 215, pp. 249–257, 2015.

[6] A. Ripoll, M. Viana, M. Padrosa, X. Querol, A. Minutolo, K. M. Hou, J. M. Barcelo-Ordinas, and J. García-Vidal, "Testing the performance of sensors for ozone pollution monitoring in a citizen science approach," *Science of the Total Environment*, vol. 651, pp. 1166–1179, 2019.

[7] S. Munir, M. Mayfield, D. Coca, S. A. Jubb, and O. Osammor, "Analysing the performance of low-cost air quality sensors, their drivers, relative benefits and calibration in cities—a case study in sheffield," *Environmental monitoring and assessment*, vol. 191, no. 2, p. 94, 2019.

[8] M. A. Zaidan, N. H. Motlagh, P. L. Fung, D. Lu, H. Timonen, J. Kuula, J. V. Niemi, S. Tarkoma, T. Petäjä, M. Kulmala *et al.*, "Intelligent calibration and virtual sensing for integrated low-cost air quality sensors," *IEEE Sensors Journal*, vol. 20, no. 22, pp. 13 638–13 652, 2020.

[9] F. Kizel, Y. Etzion, R. Shafran-Nathan, I. Levy, B. Fishbain, A. Bartonova, and D. M. Broday, "Node-to-node field calibration of wireless distributed air pollution sensor network," *Environmental Pollution*, vol. 233, pp. 900–909, 2018.

[10] L. Spinelle, M. Gerboles, M. G. Villani, M. Aleixandre, and F. Bonavitacola, "Field calibration of a cluster of low-cost commercially available sensors for air quality monitoring. part b: NO, CO and CO2," *Sensors and Actuators B: Chemical*, vol. 238, pp. 706–715, 2017.

[11] E. Esposito, S. De Vito, M. Salvato, G. Fattoruso, and G. Di Francia, "Computational intelligence for smart air quality monitors calibration," in *International Conference on Computational Science and Its Applications*. Springer, 2017, pp. 443–454.

[12] P. Ferrer-Cid, J. M. Barcelo-Ordinas, J. Garcia-Vidal, A. Ripoll, and M. Viana, "Multi-sensor data fusion calibration in iot air pollution platforms," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3124–3132, 2020.

[13] D. Hagan, G. Isaacman-VanWertz, J. Franklin, L. Wallace, B. Kocar, C. Heald, and J. Kroll, "Calibration and assessment of electrochemical air quality sensors by colocation with regulatory-grade instruments," *Atmosph. Measurement Tech.*, vol. 11, no. 1, pp. 315–328, 2018.

[14] V. Van Zoest, A. Stein, and G. Hoek, "Outlier detection in urban air quality sensor networks," *Water, Air, & Soil Pollution*, vol. 229, no. 4, pp. 1–13, 2018.

[15] H. Wu, X. Tang, Z. Wang, L. Wu, M. Lu, L. Wei, and J. Zhu, "Probabilistic automatic outlier detection for surface air quality measurements from the china national environmental monitoring network," *Advances in Atmospheric Sciences*, vol. 35, no. 12, pp. 1522–1532, 2018.

[16] T.-B. Ottosen and P. Kumar, "Outlier detection and gap filling methodologies for low-cost air quality measurements," *Environmental Science: Processes & Impacts*, vol. 21, no. 4, pp. 701–713, 2019.

[17] D. Chen, C.-T. Lu, Y. Kou, and F. Chen, "On detecting spatial outliers," *Geoinformatica*, vol. 12, no. 4, pp. 455–475, 2008.

[18] Z. Wang, J. Feng, Q. Fu, S. Gao, X. Chen, and J. Cheng, "Quality control of online monitoring data of air pollutants using artificial neural networks," *Air Quality, Atmosphere & Health*, vol. 12, no. 10, pp. 1189–1196, 2019.

[19] D. Gong, L. Liu, V. Le, B. Saha, M. R. Mansour, S. Venkatesh, and A. v. d. Hengel, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1705–1714.

[20] S. Pidhorskyi, R. Almohsen, D. A. Adjeroh, and G. Doretto, "Generative probabilistic novelty detection with adversarial autoencoders," *arXiv preprint arXiv:1807.02588*, 2018.

[21] M.-F. Harkat, G. Mourot, and J. Ragot, "Sensor failure detection of air quality monitoring network," *IFAC Proceedings Volumes*, vol. 33, no. 11, pp. 529–534, 2000.

[22] M. F. Harkat, G. Mourot, and J. Ragot, "Sensor fault detection and isolation of an air quality monitoring network using non linear principal component analysis," in *16th IFAC World Congress*. Citeseer, 2005, pp. 4–8.

[23] T. Yu, X. Wang, and A. Shami, "Recursive principal component analysis-based data outlier detection and sensor data aggregation in iot systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2207–2216, 2017.

[24] N. Castell, F. R. Dauge, P. Schneider, M. Vogt, U. Lerner, B. Fishbain, D. Broday, and A. Bartonova, "Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates?" *Environment international*, vol. 99, pp. 293–302, 2017.

[25] P. Ferrer-Cid, J. M. Barcelo-Ordinas, and J. Garcia-Vidal, "Graph learning techniques using structured data for iot air pollution monitoring platforms," *IEEE Internet of Things Journal*, vol. 8, no. 17, pp. 13 652–13 663, 2021.

[26] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, 2014.

[27] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[28] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, 2014.

[29] H. E. Egilmez and A. Ortega, "Spectral anomaly detection using graph-based filtering for wireless sensor networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1085–1089.

[30] Z. Xiao, H. Fang, and X. Wang, "Nonlinear polynomial graph filter for anomalous iot sensor detection and localization," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4839–4848, 2020.

[31] P. Ferrer-Cid, J. M. Barcelo-Ordinas, and J. Garcia-Vidal, "Graph signal reconstruction techniques for iot air pollution monitoring platforms," *arXiv preprint arXiv:2201.00378*, 2022.

[32] I. Heimann, V. Bright, M. McLeod, M. Mead, O. Popoola, G. Stewart, and R. Jones, "Source attribution of air pollution by spatial scale separation using high spatial density networks of low cost air quality sensors," *Atmospheric Environment*, vol. 113, pp. 10–19, 2015.

[33] Z. Xiao, H. Fang, and X. Wang, "Distributed nonlinear polynomial graph filter and its output graph spectrum: Filter analysis and design," *IEEE Transactions on Signal Processing*, vol. 69, pp. 1–15, 2021.

[34] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning laplacian matrix in smooth graph signal representations," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, 2016.

[35] J. M. Barcelo-Ordinas, P. Ferrer-Cid, J. Garcia-Vidal, M. Viana, and A. Ripoll, "H2020 project captor dataset: Raw data collected by low-cost mox ozone sensors in a real air pollution monitoring network," *Data in Brief*, vol. 36, p. 107127, 2021.

[36] J. M. Barcelo-Ordinas, P. Ferrer-Cid, J. Garcia-Vidal, A. Ripoll, and M. Viana, "Distributed multi-scale calibration of low-cost ozone sensors in wireless sensor networks," *Sensors*, vol. 19, no. 11, 2019.

[37] A. Gaddam, T. Wilkin, M. Angelova, and J. Gaddam, "Detecting sensor faults, anomalies and outliers in the internet of things: A survey on the challenges and solutions," *Electronics*, vol. 9, no. 3, p. 511, 2020.

[38] Z. Zhang, A. Mehmood, L. Shu, Z. Huo, Y. Zhang, and M. Mukherjee, "A survey on fault diagnosis in wireless sensor networks," *IEEE Access*, vol. 6, pp. 11 349–11 364, 2018.

[39] S. Shekhar, C.-T. Lu, and P. Zhang, "Detecting graph-based spatial outliers: algorithms and applications (a summary of results)," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 2001, pp. 371–376.

[40] Y. Kou, C.-T. Lu, and D. Chen, "Spatial weighted outlier detection," in *Proceedings of the 2006 SIAM international conference on data mining*. SIAM, 2006, pp. 614–618.

[41] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.

[42] K. Gopalakrishnan, M. Z. Li, and H. Balakrishnan, "Identification of outliers in graph signals," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 4769–4776.

[43] L. Xie, D. Pi, X. Zhang, J. Chen, Y. Luo, and W. Yu, "Graph neural network approach for anomaly detection," *Measurement*, vol. 180, p. 109546, 2021.

[44] G.-J. Qi and J. Luo, "Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[45] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 843–852.

[46] Z. Xiao, H. Fang, and X. Wang, "Anomalous iot sensor data detection: An efficient approach enabled by nonlinear frequency-domain graph analysis," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3812–3821, 2020.

[47] G. Leus, M. Yang, M. Coutino, and E. Isufi, "Topological volterra filters," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5385–5399.

[48] Y. Zhou, S. De, W. Wang, R. Wang, and K. Moessner, "Missing data estimation in mobile sensing environments," *IEEE Access*, vol. 6, pp. 69 869–69 882, 2018.

[49] V. Kalofolias, "How to learn a graph from smooth signals," in *Artificial Intelligence and Statistics*, 2016, pp. 920–929.

[50] C. O'Reilly, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Anomaly detection in wireless sensor networks in a non-stationary environment," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1413–1432, 2014.

Pau Ferrer-Cid is a PhD student at the Statistical Analysis of Networks and Systems (SANS) research group, Universitat Politecnica de Catalunya (UPC). He holds a B.Sc in Computer Science and a M.Sc in Data Science by the UPC. His main research interests are the applications of novel data analysis methods to sensor data coming from IoT platforms and the analysis of other kinds of data from fields like biology and computer vision.

Jose M. Barcelo-Ordinas is an Associate Professor at Universitat Politecnica de Catalunya (UPC) from 1999. He holds a PhD and B.Sc+M.Sc in Telecommunication Engineering and a B.Sc+M.Sc in Mathematics. He has participated in many European projects such as WIDENS, EuroNGI, EuroNFI, EuroNF NoE and H2020 CAPTOR. His currently research areas are wireless sensor networks, mobility patterns, and the statistical analysis of sensor data.

Jorge Garcia-Vidal is since 2003, full professor at the Computer Architecture Department of UPC, and since 2012 responsible of the Smart Cities Initiative at Barcelona Supercomputing Center (BSC-CNS), coordinating the H2020 CAPTOR project or being the BSC-CNS responsible of the H2020 project ASGARD. His main current research interest is in problems related with the capture, processing and statistical analysis of sensor data.