 DOI 10.1109/JIOT.2021.3067717

Graph Learning Techniques Using Structured Data for IoT Air Pollution Monitoring Platforms

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

***Abstract*—Existing air pollution monitoring networks use reference stations as the main nodes. The addition of low-cost sensors calibrated in-situ with machine learning techniques allows the creation of heterogeneous air pollution monitoring networks. However, current monitoring networks or calibration techniques have limitations in estimating missing data, adding virtual sensors or recalibrating sensors. The use of graphs to represent structured data is an emerging area of research that allows the use of powerful techniques to process and analyze data for air pollution monitoring networks. In this paper, we compare two techniques that rely on structured data, one based on statistical methods and the other on signal smoothness, with a baseline technique based on the distance between nodes and that does not rely on the measured signal data. To compare these techniques, the sensor signal is reconstructed with a supervised method based on linear regression and a semi-supervised method based on Laplacian interpolation, which allows reconstruction even when data is missing. The results, on data sets measuring O_3, NO_2 and PM_{10}, show that the signal smoothness-based technique behaves better than the other two, and used together with the Laplacian interpolation is near-optimal with respect to the linear regression method. Moreover, in the case of heterogeneous networks, the results show a reconstruction accuracy similar to the in-situ calibrated sensors. Thus, the use of the network data increases the robustness of the network against possible sensor failures.**

***Index Terms*—IoT platform, Air pollution monitoring networks, Low-cost sensors, Graph Signal Processing, Signal reconstruction.**

I. Introduction

AIR pollution is becoming a major concern for most political leaders and people around the globe. Studies from well-known organizations such as the World Health Organization (WHO) inform about the increasing number of deaths per year due to the exposure to air pollution. Because of this problem, in recent years there has been a growing interest in deploying IoT platforms with applications in fields such as agriculture [1] and air quality monitoring [2]. In the latter case, the use of low-cost sensors to deploy IoT platforms is being studied and analyzed nowadays. The greatest benefit from using these low-cost sensors is their reduced cost compared to reference stations used by government organizations to monitor pollution levels. However, the major problem of these sensors is that they are not calibrated by the manufacturer or they have been calibrated in controlled chambers. Thus, they are not guaranteed to perform well or accurate enough at deployment locations.

During the last few years, the studies have focused on the process of in-situ calibration of low-cost sensors for air pollution [3], [4]. Most of them proceeded to place a sensor beside a reference station, which gives accurate information, and then calibrate the low-cost sensor with a supervised machine learning algorithm. Once calibrated, the low-cost sensors reconstruct the signal of the measured pollutant. The limitation of these techniques is that the error obtained in the estimation of the contaminant, e.g. root mean square error (RMSE), depends on the accuracy of each of the machine learning methods used in the calibration process [5], [6], [7]. In addition, these methods are not able to avoid the bias of pollutant measurements due to changing environmental conditions [6]. Moreover, when data is missing, which is often the case even for reference stations and low-cost sensors, the supervised methods cannot estimate the value of the signal at that time.

One way to reconstruct the signal when there is missing data, when a bias occurs in a sensor, or when a virtual sensor is introduced to increase the spatial resolution is to build a graph and use the knowledge of neighboring sensors that measure the same signal, and reconstruct the signal by adding that knowledge, including not only the values of neighboring low-cost sensors, but also the values of nearby reference stations. The following are examples of applications where these cases may occur: (i) a reference station or sensor fails to report data due to hardware malfunction, loss of power supply/batteries or maintenance situation, (ii) a virtual sensor is placed where a physical sensor is difficult to deploy, or (iii) to recalibrate a sensor that is drifting or aging. Hence, this network-based approach provides robustness to the monitoring network, being able to provide estimates in the cases mentioned above. For these reasons, it is interesting to investigate methods of reconstructing the signal produced by a heterogeneous sensor network that incorporates the knowledge of the neighbors when estimating the value of the pollutant concentration.

The construction of graphs for the representation, analysis, processing and visualization of signals on structured data such as those found in a heterogeneous sensor network has had a boost in recent years [8], [9], [10], [11], [12]. The classic method proposed in the literature [8], [13] for this kind of sensor networks is to create the edges that join the nodes of the graph following a function that depends on the distance between the nodes. This method has the disadvantage of not taking into account the similarity of the measurements taken

Pau Ferrer-Cid (pferrer@ac.upc.edu), Jose M. Barcelo-Ordinas (joseb@ac.upc.edu) and Jorge Garcia-Vidal (jorge@ac.upc.edu) are with the Universitat Politecnica de Catalunya, Barcelona, Spain.

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

by the sensors. Thus, one of the challenges and key points for the representation of a sensor network using graphs is to estimate the structure of the graph that underlies the measured data [8], [9], [13]. As Dong et al. [9] mention, there are several ways to build the graph from the data depending on whether you use i) statistical methods based on graphical probabilistic models such as Markov random fields (MRFs) or Bayesian networks, ii) methods based on graph signal processing (GSP), or iii) methods based on physically motivated models.

From the statistical point of view, there are several methods for finding the most similar nodes in a graph, but the most important one is Graphical Lasso [14]. Using this method, the connectivity of a graph can be found (i.e. which sensors are dependent), assuming that the data collected from the different sensors form a Gaussian Markov random field (GMRF). On the other hand, a field of growing interest called graph signal processing (GSP), translates the idea of a signal (i.e. sensor signal) to an irregular domain like graphs [8], [10]. A key problem tackled by the GSP community is finding the graph connectivity, by means of the Laplacian matrix $\mathbf{L}$, using a set of observed signals over the graph [11].

The purpose of this paper is to present some well-known and novel graphical tools for finding a good representation for sensor networks using structured data for air pollution monitoring, and consequently improving their robustness given the limitations of in-situ calibration. Since the relationships between nodes in a low-cost heterogeneous network may not be well described by the physical distances between nodes, we compare the statistical and GSP methods to air pollution data obtained from reference stations and real low-cost sensor data with the method based on distances between nodes. To make this comparison we apply two reconstruction techniques, based on the use of the measurements of the nodes neighboring the node where we want to reconstruct the signal, with different properties. The first reconstruction method is a supervised technique based on the use of a multiple linear regression in which the independent variables are the neighboring nodes. The second method of reconstruction is based on the use of a semi-supervised technique called Laplacian interpolation that has the advantage of not needing previous training, and that allows to reconstruct the signal even when there is missing data in the neighboring sensors. We call the resulting graph-based framework *graph sensing*. This inferred graph topology provides a baseline for applying a wide range of graph-based tools; community detection, clustering, filtering, etc. More specifically, in this paper we:

- introduce three well-known techniques such as Graphical Lasso, a distance-based graph and a smoothness-based graph to describe the topology of an air pollution monitoring network,
- compare the characteristics of the three topologies obtained with the real data obtained from reference stations in the Barcelona area, and with the real data obtained from the low-cost IoT platform H2020 CAPTOR deployed in Spain in 2017,
- compare the performance of signal reconstruction using a supervised method (multiple linear regression) and a semi-supervised method (Laplacian interpolation),
- show the estimation accuracy using neighboring information in homogeneous and heterogeneous air pollution monitoring networks, even when sensing nodes present missing data.

The paper is organized as follows: section II covers the related work. Section III outlines the different approaches for inferring the graph topology. Section IV describes the data sets used in the analysis. Section V explains the methodology for learning the graph topology and signal reconstruction. Section VI describes the results and discusses the performance of the models. Finally, section VII presents the conclusions.

II. Related work

The study of the use of low-cost sensors for the estimation of pollution concentrations is an important field of research [2], [5], [15], [16]. The main challenge lies in the calibration and accuracy of these low-cost sensors [17]. Since the low-cost sensors have been calibrated on specific chambers or have not been directly calibrated by the manufacturer, they have to be calibrated in-situ with reference stations in the deployment field [3], [4]. In the process of calibrating low-cost sensors in uncontrolled environments, sensors are calibrated by positioning the sensor in a reference station and comparing both data using machine learning techniques such as multiple linear regression (MLR) [5], [18], [19], [20], k-nearest neighbors (KNN) [6], [19], support vector regression (SVR) [6], [15], [21], random forest (RF) [6], [15] or artificial neural networks (ANN) [5].

De Vito et al. [22] discuss the problems of robustness in multi-sensor air quality deployments by pointing out the importance of changes in operating conditions over time and space such as the relocation of calibrated multi-sensor platforms and sensor drift. For example, De Vito et al. point out that if the learning algorithm is fed with a low number of samples, the model will be incomplete and will not survive seasonal changes or changing human activities. Miskell et al. [23] show that reference stations can correct the drift of nearby sensor nodes. Miskell et al. propose a hierarchical network of sensor nodes and reference stations in which the reference stations provide regulatory quality data at selected sites, establish appropriate criteria for the choice of proxies, and provide proxy data that verify the reliability of the low-cost network that aims to expand the spatial scale. Fishbain et al. [24] propose wireless distributed environmental sensory networks (WDESN), an efficient method for aggregating measurements acquired by an uncalibrated WDESN, and producing accurate estimates of the observed environmental variable's true levels. Other distributed calibration methods in wireless sensor networks are discussed in the survey [4], including multi-hop calibration or the use of consensus algorithms and gossip algorithms.

Thus, the next step in the study of pollutant concentration measurements is the estimation, representation and analysis of the pollutant concentrations using the data measured by a sensor network. In this sense, graph-based techniques that adjust their topology to the measured data seem to be a good candidate to improve the estimates of the values measured by

the sensors [9]. The emerging field of graph signal processing (GSP) places emphasis on signal analysis on graphs. Many of the surveys [8], [10] in GSP applied to sensors propose simple methods such as using the distances between the nodes to build the graph, that is, to decide which nodes are connected in a undirected graph and to obtain the weights assigned to the edges. Once the network is created, the researchers [8], [10], [11] focus on the Fourier analysis of the graph from the Laplacian obtained from the weight matrix, such as the representation of the vertex domain and the graph spectral domain, the study of generalized operators for signals on graphs such as filtering, convolution, modulation, or studies related to graph coarsening or downsampling.

Ribeiro et al. [13] analyze, using GSP, data coming from taxis in Manhattan and pluviometry in Brazilian cities. Jablonski [25] analyzes a network of one hundred reference stations monitoring tropospheric ozone (O_3) in Poland. In this first work applying GSP to air pollution, Jablonski uses a smoothness-based model to construct the topology of the graph and obtain the weights of the edges and thus the Laplacian matrix. The application that Jablonski shows is the identification of clustering patterns between nodes.

There is a rich literature on obtaining an optimal Laplacian from structured data. Looking for the application to the content of this paper, [8], [10], [11], [12], [13] give an overview of how Laplacian is used in the calculation of various properties in GSP and [9] gives an overview of how various techniques and algorithms are used to obtain a weight matrix or an optimal Laplacian from the data. Among the most widely used methods, in addition to using a distance-based kernel, is the use of statistical methods and the use of methods based on signal smoothness.

In the case of using statistical models and specifically GMRF, the problem is to find the joint distribution of the graph vertex-indexed variables as a zero-mean Gaussian with precision matrix $\mathbf{\Theta}$ (inverse covariance matrix) of dimension the number of nodes in the graph. The problem, then, is to find such a precision matrix that guarantees conditional independence between the nodes. Meinshausen et al. [26] propose a neighborhood selection algorithm to estimate $\mathbf{\Theta}$. However, the most popular approach is to define an optimization problem called Graphical Lasso [14] where the precision matrix is forced to be sparse, and encodes the network topology as well as the conditional independence of the variables. GMRF have also been compared with Gaussian geostatistical models to model $PM_{2.5}$ concentrations [27].

In the case of tackling the problem from a GSP perspective, emphasis is placed on criteria where the Laplacian assumes smoothness in the data and that the Laplacian matrix is sparse. The property of a signal being smooth is that it varies slowly in the graph. Dong et al. [28] and V. Kalofolias [29] propose optimization frameworks to obtain an optimal Laplacian that fits to the data measured with smooth signals and in which the Laplacian matrix is sparse.

In this paper, we propose to construct the graph topology of air pollution monitoring networks using the data provided by the nodes instead of using distance-based functions. In the following, we will compare the classical distance-kernel method [8], [10], [11] with two graph inference methods, the Graphical Lasso [14] that obtains the adjacency matrix by estimating the precision matrix, and a graph signal processing technique that obtains the Laplacian optimally using the smoothness of the signal [28]. The latter two methods use the signal measurements to construct the graph topology in a more efficient way than the distance methods. The graph adjacency will be used to obtain the neighbors that will participate in the signal reconstruction, adding robustness and resilience to air pollution monitoring networks.

TABLE I
LIST OF SYMBOLS

Symbol	Meaning
N | P	Number of nodes| Number of observations
G | V | E	Graph | Set of nodes | Set of edges
$\mathbf{A}$ | $\mathbf{W}$	Graph adjacency matrix | Graph weight matrix
$\mathbf{D}$ | $\mathbf{L}$	Graph degree matrix | Graph Laplacian matrix
x_i | N(i)	ith vertex | ith vertex neighborhood
$\mathbf{\Sigma}$ | $\mathbf{\Theta}$	Covariance matrix | Precision matrix
tr($\cdot$) | det($\cdot$)	Trace of a matrix | Determinant of a matrix
$\|\cdot\|_1$ | $\|\cdot\|_F$	l_1-norm of a matrix | Frobenius norm of a matrix
λ	Graphical Lasso hyperparameter
τ | TH	Distance-based method hyperparameters
α | β	Smoothness-based method hyperparameters
S_K	Set of observed nodes at time step K
$\mathbf{P}$	Matrix of Partial Least Squares loadings
$\mathbf{0}$ | $\mathbf{1}$	Vector of zeros | Vector of ones

III. NETWORK TOPOLOGY INFERENCE

This section provides a self-contained introduction to the approaches used in the paper for estimating the structure of the graph given a set of observations, $\mathbf{X}\in\mathbb{R}^{P\times N}$, where N corresponds to the number of nodes and P to the number of measurements per node. Each node $i\in N$, then, is represented as x_i and the data at node i as $\mathbf{x}_i\in\mathbb{R}^P$. We denote matrices and vectors by bold letters. The notation used throughout the paper is listed in Table I. An undirected graph G=(V, E, $\mathbf{W}$) is a triplet consisting of a collection of nodes V=$\{1,2,\ldots,\mathrm{N}\}$ connected by a set of edges E, where edge e_{ij} connects node i with node j, and a symmetric matrix of non-negative weights $\mathbf{W}$ where weight W_{ij} is assigned to edge e_{ij}.

A. Statistical-based approach

We consider a Gaussian Markov random field (GMRF) where each node, x_i, in the network can be seen as a random variable over the graph G, $\{x_i{:}v_i\in V\}$ which satisfies the Markov property, and where the joint distribution of the vertex-indexed variables follows a zero-mean Gaussian distribution with precision matrix $\mathbf{\Theta}$ (the inverse covariance matrix $\mathbf{\Sigma^{-1}}$). The pairwise Markov property states that two vertices that are not connected in the graph ($e_{ij}\notin E$ or W_{ij}=0) are conditionally independent given all other vertices in the graph. The main focus of the statistical inference of the topology is the estimation of the precision matrix $\mathbf{\Theta}$ whose entries encode the conditional independencies of each vertex. In fact, there is a direct relationship between the entries of the precision matrix and the partial correlations of each variable.

The first attempt to estimate the entries of the precision matrix was named covariance selection, Dempster [30]. It solves a recursive likelihood-based thresholding procedure using the empirical covariance matrix. However, problems may appear when dealing with larger scale graphs. Friedman et al. [14] introduced an algorithm to estimate a sparse precision matrix, the Graphical Lasso. The proposed optimization problem solves a l_1-regularized maximum likelihood where the regularization term promotes the sparsity of the precision matrix:

$$\arg\max_{\mathbf{\Theta}} \quad \underbrace{\log\,\det(\mathbf{\Theta}) - \text{tr}(\hat{\mathbf{\Sigma}}\mathbf{\Theta})}_{\text{log-likelihood GMRF}} - \underbrace{\lambda \left\|\mathbf{\Theta}\right\|_1}_{\text{sparsity}} \tag{1}$$

Specifically, (1) shows the two main terms in the optimization problem; the log-likelihood of the data under the GMRF assumption and the term that promotes sparsity. The matrix $\hat{\mathbf{\Sigma}}$ is the empirical covariance matrix and tr$(\cdot)$ is the trace of a matrix. The λ hyperparameter controls the l_1-norm of the precision matrix, the higher the λ the more entries of the precision matrix will be set to zero. Once the precision matrix $\mathbf{\Theta}$ is obtained, an adjacency matrix $\mathbf{A}$ can be built. Entries with values greater than zero in the precision matrix indicate an adjacency between nodes with weight equal to one, while those entries with values equal to zero in the precision matrix indicate that two nodes are not connected.

B. Graph signal processing-based approach

The emerging field of graph signal processing (GSP) extrapolates the idea behind the classical signal processing to signals defined over graphs. A signal [8] $\mathbf{x}$:$V\rightarrow\mathbb{R}^N$ defined on the vertices of the graph may be represented as a vector $\mathbf{x}\in\mathbb{R}^N$, where the ith component of the vector $\mathbf{x}$ represents the function value at the ith vertex in V. The Laplacian matrix of the graph, $\mathbf{L}$, is the object of study of the GSP. Indeed, its eigendecomposition provides a Fourier-like interpretation of the signals defined over a graph. The most used Laplacian matrix is the combinatorial Laplacian which has properties like symmetry and positive-semidefiniteness and can be obtained as:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \tag{2}$$

where $\mathbf{D}$ is the degree matrix ($D_{ii} = \sum_j W_{ij}$). There are several methods of finding the Laplacian matrix. Some are based on defining a matrix of weights $\mathbf{W}$ using a distance or similarity function that adjusts to the physical phenomenon to then obtain the Laplacian matrix, while others try to define an optimization problem that directly obtains the optimal values of the Laplacian matrix, and that fit some criteria of the problem.

1) Distance-based approach: A way to define the weight matrix $\mathbf{W}$ is to use prior information. The most commonly employed information in these cases is the physical distance between nodes. The concentration of some pollutants exhibits spatial variability, i.e. concentrations vary in space and it is likely that the closest sensors will have similar measurements. Therefore, one way to construct the weight matrix, $\mathbf{W}$, specifying the connectivity of the graph G is to use a similarity function and a threshold (TH) to set the weight proportionally to the distance and set some entries to zero. A common choice of the weight of an edge e_{ij} is via a thresholded Gaussian kernel weighting function [8], [13]:

$$W_{ij} = \begin{cases} e^{-\frac{d_{ij}}{2\tau}} & if \;\; d_{ij} \leq TH \\ 0 & if \;\; d_{ij} > TH \end{cases} \tag{3}$$

for some τ and TH parameters, and where d_{ij} is the Haversine distance between node i and node j. The rationale behind this assumption is that the value of the signal measured at a vertex i is similar to the value of the signal measured at the nodes in its neighborhood.

2) Smoothness-based approach: A graph signal $\mathbf{x}$ is smooth if the signal values associated with the two end vertices of edges with large weights in the graph tend to be similar, or in other words, the signal changes smoothly between connected nodes. It is easy to quantify the smoothness on an undirected weighted graph for the nodes that are connected [8], [13], $\text{W}_{ij}>0$, using the following term:

$$\frac{1}{2}\sum_{ij} W_{ij} \left\|\mathbf{x_i} - \mathbf{x_j}\right\|^2 = \text{tr}(\mathbf{X^T}\,\mathbf{L}\,\mathbf{X}) \tag{4}$$

Then, a signal is smooth if the signal values for strong connected vertices are similar, and therefore tr($\mathbf{X}^T\mathbf{L}\mathbf{X}$) is small. In GSP, the Laplacian matrix is used in a variety of applications, including clustering, outlier detection and signal reconstruction among others. In fact, the entries of the Laplacian describe the connectivity of the graph and the strength of the connections. Because of that, a major challenge has been to find the best Laplacian given the set of observations $\mathbf{X}\in\mathbb{R}^{N\times P}$. Dong et al. [28] propose a not jointly convex optimization problem to find the best Laplacian given that the observations $\mathbf{X}$ are smooth:

$$\begin{aligned} \min_{L,Y} \quad & \underbrace{\|\mathbf{X}-\mathbf{Y}\|_{\mathbf{F}}^{\mathbf{2}}}_{data\;\; fidelity} + \underbrace{\underbrace{\alpha\, tr(\mathbf{Y^T L Y})}_{smoothness} + \beta\|\mathbf{L}\|_{\mathbf{F}}^{\mathbf{2}}}_{sparsity} \\ \text{s.t.} \quad & tr(\mathbf{L}) = n, \\ & L_{ij} = L_{ji} \leq 0, \qquad i \neq j, \\ & \mathbf{L}\cdot\mathbf{1} = \mathbf{0}. \end{aligned} \tag{5}$$

Where the first term of the objective function penalizes the differences between the observations $\mathbf{X}$ and their filtered version $\mathbf{Y}$. Moreover, the tr$(\cdot)$ and $\|\cdot\|_F$ denote the trace and Frobenius norm of a matrix, respectively. The two hyperparameters, α and β are positive scalars that control the smoothness of the learned representation, $\mathbf{Y}$, over the Laplacian and the Laplacian sparsity. The first constraint forces the trace of the learned Laplacian to be some scalar n in order to avoid trivial solutions. The second and third constraints ensure that the resulting Laplacian is symmetric and with non-positive off-diagonal elements, leading to a positive-semidefinite matrix. This optimization problem is not jointly convex in $\mathbf{L}$ and $\mathbf{Y}$. This is why the problem is split into an alternating minimization procedure where the minimization of $\mathbf{L}$ and $\mathbf{Y}$ alternates iteratively.

IV. Data sets

To perform the experiments, accurate data on the pollution levels at different points in the measured area are required.

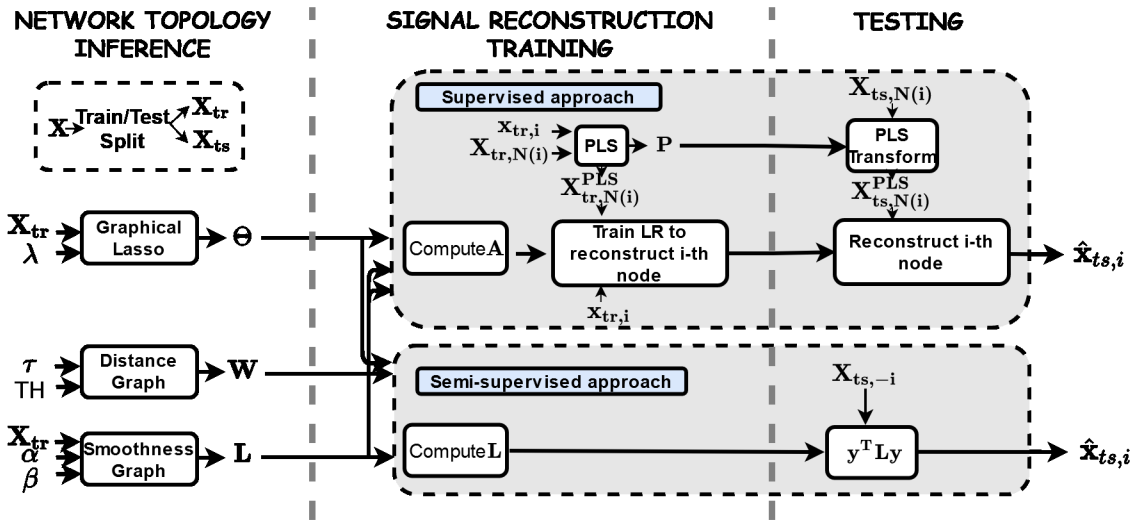


Fig. 1. Data flow showing the graph learning and signal reconstruction process for the different methods.

Therefore, data from reference stations in the metropolitan area of Barcelona, Spain, and data taken from a real deployment of low-cost sensors in Spain are used. The data obtained by the reference stations are very precise and these data are available on-line by a government agency[1]. The measurements used for the research are hourly and are taken between January 1, 2019 and May 30, 2019, and these include three different pollutants: tropospheric ozone (O_3), nitrogen dioxide (NO_2) and particulate matter 10 micrometers (PM_{10}). These data sets are labeled as data sets 1, 2 and 3, and are listed in Table II.

TABLE II
DATA SETS 1, 2 AND 3 USE REFERENCE STATIONS IN BARCELONA, SPAIN, AND DATA SET 4 USES THREE REFERENCE STATIONS AND FIVE LOW-COST SENSORS DEPLOYED IN THE H2020 CAPTOR PROJECT IN SPAIN.

Data set ID	Pollutant	# Nodes	# Samples	Period
1	O_3	15	2775	01/01/2019 - 30/05/2019
2	NO_2	20	2526	01/01/2019 - 30/05/2019
3	PM_{10}	13	2595	01/01/2019 - 30/05/2019
4	O_3	8	2831	18/06/2017 - 16/09/2017

The fourth data set is made up of data collected from a network of low-cost sensors, deployed in the H2020 CAPTOR project in the summer of 2017. The data set consists of three reference stations and five SGX Sensortech MICS 2614 O_3 metal oxide sensors distributed in a rural area of Spain. The different low-cost sensors were mounted in devices called Captor's that contained an Arduino Yun for data collection and data transmission through 3G modem to a central server, a power supply, four ozone sensors, a relative humidity sensor and a temperature sensor [2]. The sensors of the Captor nodes were calibrated during few weeks using a multiple linear regression placing the nodes in close reference stations (Tona, Vic and Manlleu). Once calibrated, the Captor nodes were deployed to volunteer houses during the summer of 2017.

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

V. METHODOLOGY

In this section, we apply the graph learning techniques, explained in section III, to obtain the network topology, and then we apply two different signal reconstruction techniques; one based on training a linear model (supervised method) and another one that minimizes the smoothness of the graph signal (semi-supervised method), provided that we have learned a Laplacian matrix.

A. Graph inference

The first step is to split the data $\mathbf{X}$ into training and testing, $\mathbf{X_{tr}}$ and $\mathbf{X_{ts}}$, then the training data along with a graph learning model and its hyperparameters are used to learn the graph. Figure 1 summarizes the different graph estimation processes evaluated and Table III shows the hyperparameters of the different methods along with their role on the graph learning process and their range of values. First, we learn the network topology with one of the different algorithms, the training set $\mathbf{X_{tr}}$ and its corresponding hyperparameters. A 5-fold cross-validation (CV) procedure is applied to the training to find the optimal hyperparameters leading to a graph which together with the reconstruction method obtain a minimum average cross-validation RMSE. The result is a precision matrix $\mathbf{\Theta}$ for the graphical Lasso method, a weight matrix $\mathbf{W}$ for the distance-based method and a Laplacian $\mathbf{L}$ for the smoothness-based method. Since these three matrices are related, we can obtain for each method a matrix of adjacencies, weights and a Laplacian that can be used in the signal reconstruction methods.

B. Signal reconstruction methods

We now apply a signal reconstruction method to compare how the three graph inference models behave in an air pollution monitoring network. We consider a supervised and semi-supervised method to reconstruct the signal. The supervised method is based on a linear regression and is optimal, but has

TABLE III
DIFFERENT METHOD'S HYPERPARAMETERS, THEIR ROLE AND RANGE OF VALUES TRIED IN THE CROSS-VALIDATION TO LEARN THE NETWORK TOPOLOGY.

Method	Parameters	Role	Range of values
Graphical Lasso	λ	Controls the precision matrix sparsity	[0.0001, 1.0]
Distance-based graph	τ	Gaussian kernel parameter	[0.01, 15000]
	Threshold (TH)	Radius (in meters) in which edges are taken into account, controls sparsity	[0, max(distance)]
Smoothness-based graph	α	Smoothness penalization constant. Controls sparsity of the Laplacian and smoothness.	[0.0001, 0.0025]
	β	L-1 norm Laplacian penalization constant. Controls sparsity of the Laplacian	[0.001, 1.0]

the disadvantage that it needs all the neighboring measurements at each time step. The semi-supervised method is based on Laplacian interpolation and even without being optimal has the advantage of being used with missing data.

1) Signal reconstruction using linear regression with partial least squares: We use the weight or adjacency matrix, and call the neighborhood of the ith node $N(i)$, that is, the nodes that are connected to the ith node. Now, the signal reconstructed in the i vertex, $\hat{x}_i$, is a linear combination of the signal measured in the $N(i)$ neighborhood:

$$\hat{x}_i = \sum_{k \in N(i)} a_k x_k \tag{6}$$

Since the signal measured at the nodes is highly correlated, the problem of multicollinearity may arise. To avoid multicollinearity, we use the partial least squares (PLS) method to obtain components that are orthogonal to each other and make a dimensional reduction to avoid ill-posed conditioned matrix problems.

In a first phase, the training data of the neighbors of the ith vertex, $\mathbf{X_{tr,N(i)}}$, are used together with the data of vertex i to obtain the PLS components and the loading matrix $\mathbf{P} \in \mathbb{R}^{\mathbf{N} \times \mathbf{N}}$. The loading matrix $\mathbf{P}$ is now used to project the training regressors, $\mathbf{X_{tr,N(i)}}$, onto the PLS components $\mathbf{X}^{PLS}_{\mathbf{tr,N(i)}}$. The goal is to keep only a few components until the condition number of the moment matrix $\mathbf{X}^T\mathbf{X}$ is small enough. To obtain the coefficients a_k of the linear regression, we train the linear model with the projections of $\mathbf{X_{tr,N(i)}}$ as regressors ($\mathbf{X}^{PLS}_{\mathbf{tr,N(i)}}$) and with $\mathbf{x_{tr,i}}$ as the dependent variable. Finally, by taking new data from the test data set and projecting it to the PLS components, we can reconstruct the $\hat{\mathbf{x}}_{\mathbf{ts,i}}$ signal from neighboring node signals. The disadvantage of this method is that to reconstruct the signal in the i vertex, we need all the values of the signals in its neighborhood. If there is missing data from any neighboring node for a given instant, then it is not possible to reconstruct the signal at that instant. So, if we are interested in reconstructing M nodes of the network, using the same constructed graph we will create M linear regression models to be able to reconstruct the signal in those nodes. Another clear disadvantage is that for the calculation of the loading matrix $\mathbf{P}$ the values of the neighboring nodes $\mathbf{X_{tr,N(i)}}$ and the training values of the target node $\mathbf{X_{tr,i}}$ are needed, that's why as with the regression model we have to obtain a loadings matrix for each different node signal reconstruction model.

2) Signal reconstruction using Laplacian interpolation: This method can be applied when data are missing from one or more nodes and can be addressed from the graph semi-supervised learning framework, where the data are assumed to be in or near a manifold represented by a graph. Then, the global quantity represented by the smoothness is optimized to find the remaining values.

We assume that we know the structure of the graph through the Laplacian, we know the value of the signal in k<N vertices, and we have gaps, missing or corrupted values in the other N-k vertices. Let's call $\mathbf{y}$ the signal to be estimated since we know $\mathbf{x}$ and the set of nodes S_K where the values are known. The aim of the interpolation process is to estimate $\mathbf{y}$ such that y_i=x_i for those vertices where x_i is known ($i \in S_K$). This method is a semi-supervised transductive method, which means that at each time step given the observed values, the unobserved ones are calculated. This method does not need a training phase and can be applied directly to reconstruct the signal when values are missing in time step K, given the set of observed nodes S_K, or when we want to include a virtual sensor, a case that represents a sensor where all values are missing. This problem, in which the observed values remain unaltered, is called graph interpolated regularization and can be formulated as [31][2].

$$\begin{aligned} \min_{y} \quad & \mathbf{y^T L y} \\ \text{s.t.} \quad & y_i = x_i, \qquad \forall i \in S_K, \end{aligned} \tag{7}$$

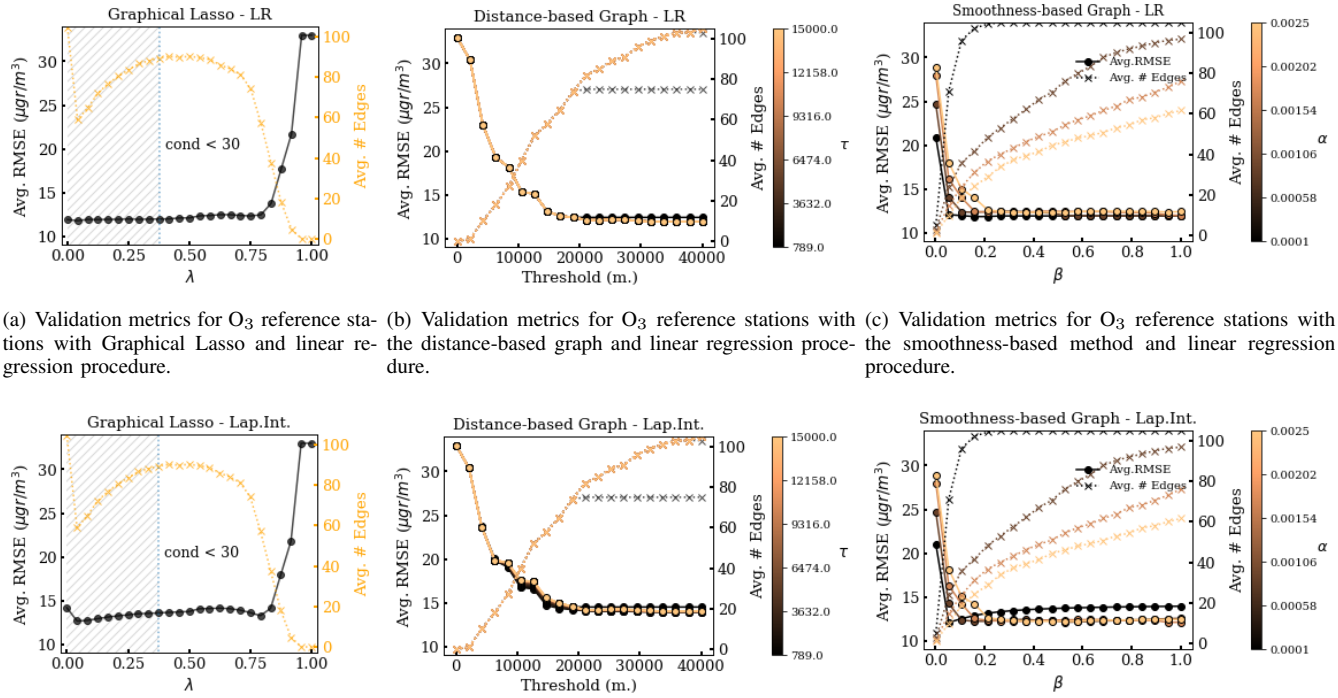
VI. RESULTS AND DISCUSSION

This section presents the results of the different methods for topology inference and signal reconstruction applied to the different data sets presented in section IV.

A. Reference station testbeds

Firstly, we apply the different methodologies to the data sets of the reference stations in the Barcelona area (data sets 1, 2 and 3), which correspond to the concentrations of O_3, NO_2 and PM_{10}. As an illustration, we select the results from the O_3 (data set 1). Figure 2 shows at the same time the performance of the three methods in obtaining the network topology, and the performance of the two signal reconstruction methods. Figures 2.a) and .d) show the optimal parameters for choosing the best topology using graphical Lasso and reconstructing the signal using linear regression and Laplacian interpolation

[2]This problem had also been formulated within the GSP community where the matrix used for regularization is the graph shift or a modified version of the graph shift [32].



(a) Validation metrics for O_3 reference stations with Graphical Lasso and linear regression procedure.

(b) Validation metrics for O_3 reference stations with the distance-based graph and linear regression procedure.

(c) Validation metrics for O_3 reference stations with the smoothness-based method and linear regression procedure.

(d) Validation metrics for O_3 reference stations with Graphical Lasso and Laplacian interpolation procedure.

(e) Validation metrics for O_3 reference stations with the distance-based graph and Laplacian interpolation procedure.

(f) Validation metrics for O_3 reference stations with the smoothness-based method and Laplacian interpolation procedure.

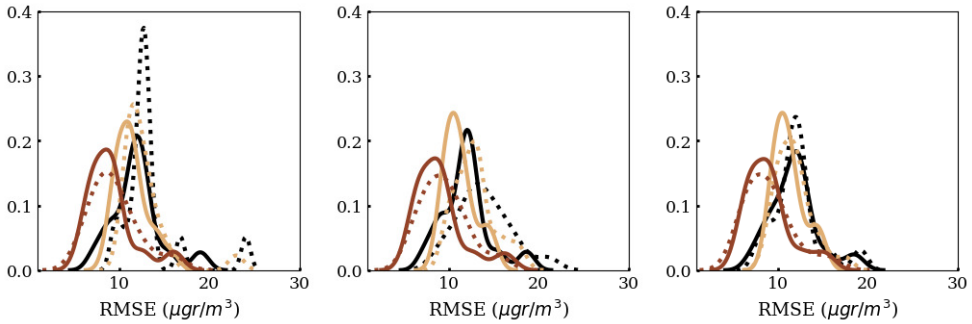
Fig. 2. Average cross-validation RMSE and average number of edges of the different techniques applied to data set 1. Shaded area correspond to hyperparameter values that produce ill-posed problems (condition number of the initial guess greater than 30).

respectively. Figures 2.b) and .e) and 2.c) and .f) show the optimal parameters for the choice of the best topology using the distance method and the smoothness method with both linear regression and Laplacian interpolation respectively. First, we will analyze the relationship between the hyperparameters of each model with the final number of edges in the resulting graph and with the root mean square error (RMSE) value obtained by each signal reconstruction method. Then, we will compare the performance of the models, discussing their advantages and disadvantages. The RMSE used to select the best hyperparameters is the average of the cross-validation RMSE of all the nodes participating in the network.

1) Graphical Lasso with signal reconstruction: Figures 2.a) and d), orange curve, show the lambda hyperparameter as a function of the number of edges for the graphical Lasso method. It can be seen that as λ approaches one the number of edges decreases, becoming the precision matrix more and more sparse while if λ approaches 0.5 the number of edges increases. We can also observe that λ values between 10^{-4} and 0.35 produce an increase in the number of edges (hump effect) and then in the $10^{-}4$ value (there is almost no regularization of the matrix) produce a totally connected graph. This effect of having λ between 10^{-4} and 0.35 is produced by a poor conditioning of the covariance matrix when solving the graphical Lasso method and has already been reported in other studies [33]. This is due to the presence of multicollinearity, indeed, Heinvaara et al. [33] explain how this problem can lead to a bad conditioning of the covariance matrix and to the instability of the results. A condition number above thirty indicates that the regression may have severe multicollinearity [34]. For all this, only values of λ for which the initial guess of the covariance matrix has a condition number less than thirty are taken into account, thus obtaining a stable solution. This, together with the PLS procedure, reduces the effects of multicollinearity. The minimum average RMSE value with graphical Lasso and the linear regression method is 11.957 μgr/m^3, and is produced with a λ value of 0.375 and an average number of edges in the graph of 89.4 edges, while with Laplacian interpolation, the minimum RMSE is produced with λ of 0.792 and RMSE of 13.264 μgr/m^3 and 57.2 edges in average, Table IV. We can see that the signal reconstruction method with linear regression produces less RMSE but with a higher number of edges than the Laplacian interpolation method. The same behavior is observed for the other two pollutants (NO_2 and PM_{10}), Table IV. Regarding the R^2, O_3 gives better values than NO_2, which in turn is better than the PM_{10} values. Figure 3.a) shows the empirical distribution of the RMSEs of the reference stations for the best Graphical Lasso configurations. As can be seen from the three contaminants and the two reconstruction methods, the distributions are positive skewed with some reference stations with a large RMSE compared to the mean. These ones are the less correlated which produce worst estimations. The Graphical Lasso method is less accurate than the smoothness graph method and more accurate than the distance-based method.

TABLE IV
VALIDATION METRICS FOR DATA SETS 1, 2, AND 3 RESPECTIVELY.

		O_3			NO_2			PM_{10}		
		RMSE	**# Edges**	**R^2**	**RMSE**	**# Edges**	**R^2**	**RMSE**	**# Edges**	**R^2**
Graphical Lasso	**LR**	11.957	89.4	0.701	11.387	109.0	0.659	9.114	61.2	0.459
	Lap.Int	13.264	57.2	0.541	12.747	53.2	0.571	9.503	60.2	0.448
Distance Graph	**LR**	11.902	101.0	0.705	11.225	190.0	0.673	9.019	76.0	0.479
	Lap.Int	13.937	103.0	0.623	12.893	140.0	0.556	9.495	77.0	0.446
Smoothness Graph	**LR**	11.835	102.2	0.709	11.225	190.0	0.673	8.996	40.4	0.489
	Lap.Int	12.082	71.2	0.708	11.805	56.0	0.632	9.040	70.2	0.500



(a) RMSE distribution obtained with the Graphical Lasso.

(b) RMSE distribution obtained with the distance-based graph.

(c) RMSE distribution obtained with the smoothness-based graph.

Fig. 3. Empirical RMSE distributions for the best configurations for the different graph learning and signal reconstruction methods. The continuous lines draw the linear regression while the dotted lines draw the Laplacian interpolation. The black, orange and red lines represent O_3, NO_2 and PM_{10} respectively.

2) Distance-based method with signal reconstruction: The distance method uses two hyperparameters, Table III: τ is a Gaussian kernel normalization parameter, and the threshold of the distance TH that controls the sparsity of the resulting weight matrix. The units of τ and TH are in meters. Figure 2.b) and e), orange and gray curves with crosses, show the growth of the number of edges as the TH parameter grows. The orange curve is for a high τ while the gray curve is for a smaller τ. The first thing is to indicate that considering an increasing TH distance threshold we are increasing the number of neighbors a node has, and therefore the number of edges of a node increases. Secondly, for those neighboring nodes which are within the radius TH meters, a large τ value will assign a small weight, while a small τ value will assign a large weight. This makes large values of τ produce more connected networks with more edges than smaller values of τ. The lowest RMSE value with linear regression is 11.902 μgr/m^3 with 101 edges, and TH of 33710 meters and τ of 1580 meters. The lowest RMSE value with Laplacian interpolation is 13.937 μgr/m^3 with 103 edges, and TH of 35817 meters and τ of 2360 meters. We can see that the signal reconstruction method with linear regression produces less RMSE than the Laplacian interpolation method with similar values in the number of edges for O_3 and PM_{10}. For NO_2, the RMSE value is also lower for linear regression, but in this case with a higher number of edges. Again, in general, R^2 gives better values for O_3 than NO_2, which in turn is better than the PM_{10} values. The empirical distributions in Figure 3.b) show the same trend as with the Graphical Lasso, some stations cannot be predicted too well with the others, showing low efficiency in capturing the correlations between nodes, since there are nodes that are far away from others or nodes whose correlation is not proportional to their distance.

3) Smoothness-based method with signal reconstruction: The method based on the smoothness of the signal has two hyperparameters. The parameter α favors the smoothness of the signal and also controls the sparsity of the Laplacian matrix, while the parameter β is a penalty constant that only governs the sparsity of the Laplacian matrix. Low values of β promote a low number of edges, and large values of α, orange curves with crosses in 2.c) and f), produce a low number of edges and the smoothness of the signal. This is because the Frobenius norm of $\mathbf{L}$ tends to be small when β increases, and decreasing β has the opposite effect. Moreover, when α increases the trace of the quadratic term $\mathbf{Y}^T\mathbf{L}\mathbf{Y}$ is small, which promotes both sparsity and smoothness. The lowest RMSE value with linear regression is 11.835 μgr/m^3 with 102 edges, and α of 10^{-4} and β of 0.15. The lowest RMSE value with Laplacian interpolation is 12.082 μgr/m^3 with 71 edges, and α of 10^{-4} and β of 0.05, being the most accurate of the three graph infering techniques. Again, we can see that the signal reconstruction method with linear regression produces less RMSE than the Laplacian interpolation method with lower values in the number of edges for O_3 and NO_2. For PM_{10}, the RMSE value is also lower for linear regression, but in this case with a higher number of edges. And again, in general, R^2 gives better values for O_3 than NO_2, which in turn is better than the PM_{10} values. Finally, Figure 3.c) shows the distributions of the RMSEs obtained with the smoothness-based graph. The same trend is observed for all methods and

contaminants, a positive skewed distribution, showing some stations that cannot be reconstructed with less error.

4) Comparison and discussion of the signal reconstruction methods with graph topology inference: The first point to comment is that the amount of edges and therefore the connectivity produced by each method depends on the hyperparameters of each model. The choice of optimal hyperparameters depends then on the target application. In our case, we want the average cross-validation RMSE to be the minimum possible, but there may be other applications where it is interesting to achieve a maximum number of edges. For example, Figure 4.a) shows the connectivity map with an 57-edge target, with graphical Lasso using data set 1.

As mentioned above, graphical Lasso suffers from instability in the case of multicollinearity between the data captured by the network nodes, so we have to obtain the condition number to find out in which region the method is unstable. Also in the example shown, there is a sharp edge drop for a small interval, [0.75, 0.85] of λ, which makes the sparsity in graphical Lasso very sensitive to this parameter. In contrast, the distance-based method is very sensitive to the TH parameter, the radius below which we accept a neighbor. This method does not take into account the correlation of the neighboring data, and we can find a node whose data is very little correlated with a neighbor, but its proximity to the neighbor makes the weight assigned to the edge very large. Finally, the smoothness-based method is the one that achieves best results when reconstructing the signal with Laplacian interpolation, although it is quite sensitive to variations in its hyperparameters. In Figure 2.c) and f), it can be seen that the optimal RMSE is produced with a low α parameter value, when for higher values of α the model gets a more sparse matrix and higher smoothness.

To see the potential reduction of the number of edges of each model, we obtain the number of edges if we allow an increase of 0.5 in the RMSE. The graphical Lasso model with linear regression goes from 11.957 with 89 edges to 12.451 with 57 edges, and with Laplacian interpolation it is not able to lower the number of edges. The distance-based model with linear regression goes from 11.902 μgr/m^3 with 101 edges to 12.422 μgr/m^3 with 74 edges, and with Laplacian interpolation goes from 13.937 μgr/m^3 with 103 edges to 14.358 μgr/m^3 with 65 edges. Finally, the smoothness-based model with linear regression goes from 11.835 μgr/m^3 with 102 edges to 12.440 μgr/m^3 with 31 edges, and with Laplacian interpolation goes from 12.082 μgr/m^3 with 71 edges to 12.428 μgr/m^3 with 33 edges. It can be seen how the smoothness-based model obtains very close to optimal RMSE values with very high edge number reductions. Therefore, by relaxing the restriction of achieving the optimal RMSE and aiming at reducing the number of edges, a good trade-off can be achieved between having a low RMSE and a sparse connectivity matrix that implies a lower complex network.

Let's now compare the two methods of signal reconstruction combined with the three methods of network construction. Signal reconstruction using linear regression, as can be seen in Figure 2 and Table IV, is always more effective than the method based on Laplacian interpolation. This is because the conditional expectation of the predicted value in the ith node given the neighbors represents the minimum mean square error (MMSE) of the prediction x_i using the random variables $\mathbf{x}$ [35]. In other words, once we have fixed the graph structure, linear regression is the best method using a linear combination of the neighbors defined by that structure. What we can see is that we have found that a large number of edges minimizes the RMSE if we use a linear regression. However, if we want a more sparse graph, it will be at the cost of increasing the RMSE. This can be seen in Figures 2.a), b) and c) where decreasing the number of edges, with respect to the optimum, increases the RMSE. On the other hand, the method of reconstruction with Laplacian interpolation, in almost all cases, has a higher RMSE than the reconstruction with linear regression but with a much lower number of edges or similar.

Although it seems that the signal reconstruction method with linear regression is better than the Laplacian interpolation, it has a disadvantage from the engineering point of view. To reconstruct the signal at a given instant with the linear regression method, it is necessary that all the neighbors have a sample, that is, there are no gaps in the data. This fact reduces the robustness of supervised methods compared to semi-supervised methods. This is not a problem for Laplacian interpolation, which can predict values even if there are gaps in a node's neighbors' data, providing robustness and resilience to the network.

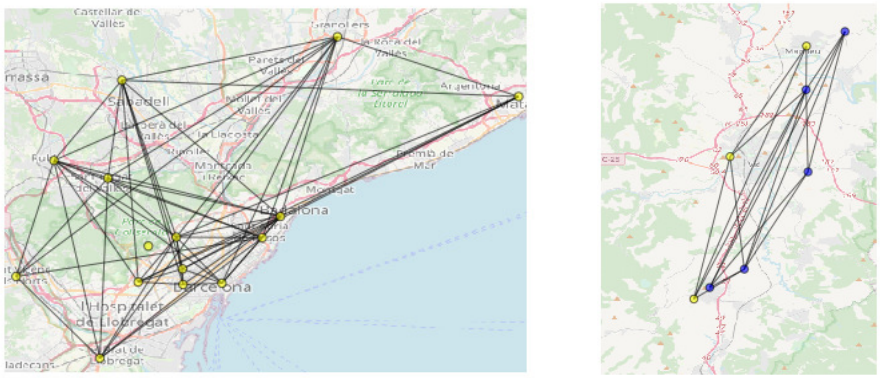
TABLE V
REFERENCE STATIONS WITH INCOMPLETE OBSERVATIONS

	Avg. % missings per station	**% incomplete samples graph with ∼50% edges**	**% incomplete samples**
Data set 1	1.66	10.73	21.88
Data set 2	1.90	15.87	28.87
Data set 3	2.60	12.00	26.94

TABLE VI
RMSE WITH 5% OF MISSING DATA AT EACH NODE USING SMOOTHNESS-BASED METHOD AND LAPLACIAN INTERPOLATION.

Mean RMSE	**Std RMSE**	**Max RMSE**	**Min RMSE**
12.154	2.702	19.277	8.069

To prove the importance of this assessment, in Table V we show the percentage of samples that do not appear ("missings") in the data obtained from the reference stations, something that is normal both in official data and in data obtained with low-cost sensors. Although the number of missing samples per station is small, approximately 2% on average, to analyze the network or reconstruct the signal with a machine learning method, it is necessary that there are no missing samples in any of the neighboring nodes at any given time. That is, if a node is missing a sample at a given instant, it is not possible to reconstruct the signal in the network with a supervised machine learning method. Table V also shows the percentage of samples with at least one missing. About 21-28% of the samples contain at least one node missing in the period, so the entire row of data should be eliminated if a linear regression is used. In the case of a more intelligent data cleansing where only the data is deleted in



(a) Network topology learned with the graphical lasso with 57 edges and λ = 0.5, area of Barcelona.

(b) Network topology learned with the smoothness method with 18 edges, α = 0.0017 and β = 0.6319, on the low-cost CAPTOR network.

Fig. 4. Two areas studied in the research. On the left the metropolitan area of Barcelona with several ozone reference stations. On the right the area of Manlleu, Vic and Tona, where the CAPTOR network was deployed. Reference stations depicted in yellow and sensors in blue.

TABLE VII
RESULTS FOR THE CAPTOR PLATFORM, INCLUDING THE CORRESPONDING RESULTS AT THE REFERENCE STATIONS LOCATIONS OF MANLLEU, VIC AND TONA.

		Manlleu RMSE	Vic RMSE	Tona RMSE	Mean RMSE	Std RMSE	Max RMSE	Min RMSE	# Edges
Graphical Lasso	**LR**	11.724	10.082	11.411	11.794	1.634	15.104	9.923	16
	Lap.Int.	12.018	10.208	12.087	11.795	0.852	12.941	10.208	14
Distance Graph	**LR**	12.785	9.772	12.224	11.430	1.146	12.785	9.772	25
	Lap.Int.	20.596	20.102	18.552	15.943	3.502	20.596	12.044	28
Smoothness Graph	**LR**	12.057	9.772	11.706	11.458	1.056	12.722	9.772	28
	Lap.Int.	11.817	10.108	11.588	11.649	0.853	12.718	10.108	18
in-situ low-cost sensor		10.850	11.299	12.212					

case one of the neighbors is missing, the percentage is lower. For example, assuming a topology with 50% of edges with relative to a full mesh graph, the percentage of deleted data is about 10-15%. This shows that having a reduced neighborhood for each node, i.e., a method that promotes matrix sparsity, reduces the impact of having gaps in the samples. In view of this, the smoothness method promotes sparsity and the Laplacian interpolation method naturally fits the situation of simultaneously estimating the missing nodes and the target node.

Just as an example of how Laplacian interpolation can handle missings, we performed an experiment with the smoothness-based graph and data set 1 (O_3) where a 5% random loss occurs in all nodes in the test phase. The result is shown in Table VI, and as you can see, the average RMSE has increased slightly from 12.082 (see Table IV) to 12.154 μgr/m^3, but the reconstruction has been possible in all nodes with a small increase of the RMSE, showing the resilience provided by using graph inference based on the smoothness of the signal with semi-supervised signal reconstruction.

B. Low-cost sensor testbed

The data sets used in the previous section came from reference nodes, where the data are very accurate. In this section we will assume a heterogeneous network with three reference stations, called Vic, Tona and Manlleu, and five low-cost sensors monitoring O_3. The sensors were deployed in Spain in 2017 during the H2020 CAPTOR project and only measured O_3. We want to verify two aspects: i) how a network with a mix of reference stations with accurate values and low-cost nodes with less accurate measurements behaves when using a network with a topology created using structured data, and ii) if the estimations made by a sensor network improves or approaches the measurement of a single (in-situ) low-cost sensor deployed at one point, which will allow applications such as recalibration or multi-hop calibration [4].

1) Signal reconstruction methods with topology graph inference in an heterogeneous network: Table VII shows the results of this network with heterogeneous nodes, where the estimations are made on the points where the reference stations have been deployed. When an estimate is made using the neighborhood of a reference station, the reference station does not participate in the estimate, and that estimate is compared with the value of the reference station to obtain the RMSE. The other columns show the mean, the standard deviation, the minimum and maximum RMSE considering all the nodes in the network. The first thing we notice is that the models behave in a similar way when we have a heterogeneous network of nodes than when we had only reference stations giving accurate values. Signal reconstruction using linear regression gives better results than Laplacian interpolation, and in general the smooth method with reconstruction based on Laplacian interpolation gives good results with few edges. It is important

to note the disadvantage of the distance method. It can be observed first that the number of edges is the highest among the three methods, and that with Laplacian interpolation the RMSE is very high, in the order of 20 μgr/m^3. This is because when using distances, it uses low-cost sensors close by, instead of reference stations with more correlated values. In this sense, both the graphical Lasso and the smoothness-based methods are effective in using the correlation of the data captured by the nodes regardless of the proximity of the nodes. Figure 4.b) shows the connectivity map with an 18-edge target, with the smoothness-based model.

2) Comparison of the network prediction with respect to an in-situ low-cost sensor: We want to test the network's ability to predict O_3 concentrations at points where it has not been possible to deploy a sensor or where a recalibration can be performed due to drift or aging of the sensors. To do this, we place a low-cost sensor at a point, e.g., at the reference stations, and compare the value given by the sensor with the predicted value using the different models at that point. This sensor is identified in Table VII with the label in-situ low-cost sensor. We can see in Table VII that except for the distance-based model with Laplacian interpolation, the rest of the models are capable of making an estimation with similar accuracy as having a low-cost sensor. This is due to the effect of the reference stations in the neighborhood of the target node, which shows the potential of this methodology to recalibrate the sensors without relocating the node in a reference station or increase the spatial resolution by including virtual sensors in the network.

VII. Conclusions

In this paper, we have introduced a graph sensing framework, which consists in describing the relationships between spatially distributed air pollution monitoring nodes through a graph that represents structured data. This way, we have described the fundamentals of graph topology inference and use the inferred topologies to perform signal reconstruction in the different nodes of the network using a supervised (multiple linear regression) and a semi-supervised method (Laplacian interpolation). It has been proven that the value of the concentration of a pollutant at a given point can be approximated from the concentrations of the pollutant at nearby points in space (neighborhood in the network).

The different graph topology inference methods have presented several limitations and advantages. The graphical lasso has been seen to suffer from multicollinearity, so that for denser graphs (large number of edges and low precision matrix sparsity) the results are not reliable. However, it is possible to tune the hyperparameter λ that controls the sparsity of the precision matrix to avoid ill-conditioning problems. The distance-based method has not been seen to capture informative relationships between the nodes. Even though air pollutants are known to vary in space, a node which is more distant than another may have a larger correlation. Finally, the smoothness-based method has been seen the most accurate method, as it does not suffer from the multicollinearity issues seen in graphical lasso. Indeed, its hyperparameters α and β, that control smoothness and sparsity of the Laplacian, can produce a large variety of resulting graphs, with several degrees of connectivity and smoothness properties.

The signal reconstruction process has been useful to show the ability of predicting concentrations at a node given the neighboring nodes concentrations. The linear regression applied to the reference stations data sets and the Captor low-cost network has obtained the lowest possible error in the three topological models, showing thus the best accuracy with respect the semi-supervised method. On the other hand, the Laplacian interpolation has only produced almost near-optimal results in conjunction with the smoothness topology inference method. However, the linear regression has one disadvantage with respect to the Laplacian interpolation, in that it is less robust and offers less resilience, which means that linear regression fails when it is required to estimate concentrations at different nodes as some samples may be missing. This setting naturally fits with the Laplacian interpolation method, which is a graph-based semi-supervised learning method that is able to handle simultaneous signal reconstruction at the different missing nodes adding robustness and resilience to the network.

The different methods have been tested with data sets of different nature. The results indicate that the O_3 and the NO_2 can be fairly well approximated given concentrations at different locations. Moreover, the errors obtained with the heterogeneous Captor low-cost network indicate that the models are able to produce concentrations with an accuracy similar to a low-cost sensor deployed in-situ which proves to be useful if recalibration is required. As future work, further study of signal reconstruction techniques may reduce the lower limit of error through more sophisticated methods and the study of virtual sensors and data missing in air pollution monitoring techniques are open fields.

References

[1] J. M. Barcelo-Ordinas, J.-P. Chanet, K.-M. Hou, and J. García-Vidal, "A survey of wireless sensor technologies applied to precision agriculture," in *Precision agriculture13*. Springer, 2013, pp. 801–808.

[2] 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.

[3] B. Maag, Z. Zhou, and L. Thiele, "A survey on sensor calibration in air pollution monitoring deployments," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4857–4870, Dec 2018.

[4] J. M. Barcelo-Ordinas, M. Doudou, J. Garcia-Vidal, and N. Badache, "Self-calibration methods for uncontrolled environments in sensor networks: A reference survey," *Ad Hoc Networks*, vol. 88, pp. 142–159, 2019.

[5] 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.

[6] 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.

[7] ——, "Multisensor data fusion calibration in iot air pollution platforms," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3124–3132, 2020.

[8] 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.

[9] X. Dong, D. Thanou, M. Rabbat, and P. Frossard, "Learning graphs from data: A signal representation perspective," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 44–63, 2019.

[10] 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.

[11] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 16–43, 2019.

[12] 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.

[13] G. B. Ribeiro and J. B. Lima, "Graph signal processing in a nutshell," *Journal of Communication and Information Systems*, vol. 33, no. 1, 2018.

[14] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[15] A. Bigi, M. Mueller, S. K. Grange, G. Ghermandi, and C. Hueglin, "Performance of no, no_2 low cost sensors and three calibration approaches within a real world application," *Atmospheric Measurement Techniques*, vol. 11, no. 6, pp. 3717–3735, 2018.

[16] E. G. Snyder, T. H. Watkins, P. A. Solomon, E. Thoma, R. W. Williams, G. S. Hagler, D. Shelow, D. A. Hindin, V. J. Kilaru, and P. Preuss, "The changing paradigm of air pollution monitoring," *Environmental science & technology*, vol. 47, no. 20, pp. 11 369–11 377, 2013.

[17] 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.

[18] J. M. Barcelo-Ordinas, J. Garcia-Vidal, M. Doudou, S. Rodrigo-Muñoz, and A. Cerezo-Llavero, "Calibrating low-cost air quality sensors using multiple arrays of sensors," in *Wireless Communications and Networking Conference (WCNC)*. IEEE, 2018, pp. 1–6.

[19] 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.

[20] 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.

[21] S. De Vito, E. Esposito, M. Salvato, O. Popoola, F. Formisano, R. Jones, and G. Di Francia, "Calibrating chemical multisensory devices for real world applications: An in-depth comparison of quantitative machine learning approaches," *Sensors and Actuators B: Chemical*, vol. 255, pp. 1191–1210, 2018.

[22] S. De Vito, E. Esposito, N. Castell, P. Schneider, and A. Bartonova, "On the robustness of field calibration for smart air quality monitors," *Sensors and Actuators B: Chemical*, vol. 310, p. 127869, 2020.

[23] G. Miskell, K. Alberti, B. Feenstra, G. S. Henshaw, V. Papapostolou, H. Patel, A. Polidori, J. A. Salmond, L. Weissert, and D. E. Williams, "Reliable data from low cost ozone sensors in a hierarchical network," *Atmospheric Environment*, vol. 214, p. 116870, 2019.

[24] B. Fishbain and E. Moreno-Centeno, "Self calibrated wireless distributed environmental sensory networks," *Scientific reports*, vol. 6, p. 24382, 2016.

[25] I. Jabłoński, "Graph signal processing in applications to sensor networks, smart grids, and smart cities," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7659–7666, 2017.

[26] N. Meinshausen, P. Bühlmann *et al.*, "High-dimensional graphs and variable selection with the lasso," *The annals of statistics*, vol. 34, no. 3, pp. 1436–1462, 2006.

[27] H.-R. Song, M. Fuentes, and S. Ghosh, "A comparative study of gaussian geostatistical models and gaussian markov random field models," *Journal of Multivariate analysis*, vol. 99, no. 8, pp. 1681–1697, 2008.

[28] 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.

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

[30] A. P. Dempster, "Covariance selection," *Biometrics*, pp. 157–175, 1972.

[31] M. Belkin, I. Matveeva, and P. Niyogi, "Regularization and semi-supervised learning on large graphs," in *International Conference on Computational Learning Theory*. Springer, 2004, pp. 624–638.

[32] S. Chen, A. Sandryhaila, G. Lederman, Z. Wang, J. M. Moura, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovačević, "Signal inpainting on graphs via total variation minimization," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 8267–8271.

[33] O. Heinävaara, J. Leppä-Aho, J. Corander, and A. Honkela, "On the inconsistency of l 1-penalised sparse precision matrix estimation," *BMC bioinformatics*, vol. 17, no. 16, p. 448, 2016.

[34] A. Lazaridis, "A note regarding the condition number: the case of spurious and latent multicollinearity," *Quality & Quantity*, vol. 41, no. 1, pp. 123–135, 2007.

[35] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under laplacian and structural constraints," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 6, pp. 825–841, 2017.

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