



**HAL**  
open science

# On the Optimization of WSN Deployment for Sensing Physical Phenomena: Applications to Urban Air Pollution Monitoring

Ahmed Boubrima, Walid Bechkit, Hervé Rivano

► **To cite this version:**

Ahmed Boubrima, Walid Bechkit, Hervé Rivano. On the Optimization of WSN Deployment for Sensing Physical Phenomena: Applications to Urban Air Pollution Monitoring. Mission-Oriented Sensor Networks and Systems: Art and Science, 163, Springer, pp.99-145, 2019, Studies in Systems, Decision and Control, 978-3-319-91145-8. 10.1007/978-3-319-91146-5\_4. hal-02159770

**HAL Id: hal-02159770**

**<https://inria.hal.science/hal-02159770v1>**

Submitted on 19 Jun 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# On the Optimization of WSN Deployment for Sensing Physical Phenomena: Applications to Urban Air Pollution Monitoring

Ahmed Boubrima, Walid Bechkit, Hervé Rivano

June 19, 2019



# Contents

<b>1 On the Optimization of WSN Deployment for Sensing Physical Phenomena: Applications to Urban Air Pollution Monitoring</b>	<b>1</b>
1.1 Introduction	2
1.2 WSN deployment	3
1.2.1 Event-aware deployment methods	3
1.2.2 Correlation-aware deployment methods	4
1.2.3 Discussion	5
1.3 Air pollution prediction	6
1.3.1 Atmospheric dispersion-based methods	6
1.3.2 Interpolation-based prediction methods	8
1.3.3 Regression-based prediction methods	8
1.4 Model 1: WSN deployment for air pollution mapping based on predicted data	8
1.4.1 Problem formulation	8
Overview	8
Objective function	9
Coverage	10
Connectivity	13
ILP model	14
1.4.2 Simulation results	14
Simulation dataset	14
Proof of concept	15
Evaluation of the coverage results	16
Evaluation of the network connectivity	17
1.5 Model 2: WSN deployment for air pollution detection based on predicted data	18
1.5.1 Problem formulation	18
Overview	18
Identification of pollution zones	20
ILP deployment model	21
1.5.2 Simulation results	23
Simulation dataset	23
Proof of concept	23
Evaluation of the coverage results	24

	Evaluation of the network connectivity . . . . .	24
1.6	Model 3: WSN deployment for air pollution detection based on emission inventory . . . . .	27
1.6.1	Problem formulation . . . . .	27
	Overview . . . . .	27
	Objective function . . . . .	27
	Coverage . . . . .	28
	Connectivity . . . . .	30
	ILP Model . . . . .	31
1.6.2	Simulation results . . . . .	32
	Simulation dataset . . . . .	32
	Proof of concept . . . . .	33
	Evaluation of the network connectivity . . . . .	34
	Evaluation of the coverage results . . . . .	37
1.7	Discussion and future directions . . . . .	40
1.8	Conclusion . . . . .	40

## Chapter 1

# On the Optimization of WSN Deployment for Sensing Physical Phenomena: Applications to Urban Air Pollution Monitoring

Wireless sensor networks (WSN) are widely used in environmental applications where the aim is to sense a physical phenomenon. Air pollution is one of the main physical phenomena that still need to be studied and characterized. Using WSN for air pollution monitoring usually targets two main applications: regular mapping and the detection of high pollution concentrations. Both of these applications need a careful deployment of sensors in order to get better knowledge of air pollution while ensuring a minimal deployment cost. In this chapter, we present three formulations of the deployment issue of sensor and sink nodes based on integer linear programming modeling (ILP) while tackling the two main applications of air pollution monitoring. In the first ILP model, we target the regular mapping of air pollution based on predicted pollution maps. In the second and third ILP model, we target the detection of pollution threshold crossings. The second model is based on predicted pollution maps as in the first one whereas the third one is based on pollution emission inventory which describes pollution sources and their emission rates. In addition to the constraints of pollution coverage, we also ensure that the deployed networks are connected and their financial deployment cost is minimized. We perform extensive simulations in order to analyze the performance of the proposed models in terms of coverage and connectivity results.

Wireless sensor networks (WSN), Deployment, Coverage, Connectivity, Air pollution monitoring, Air pollution mapping, Air pollution detection, Air pollution prediction, Atmospheric dispersion modeling.

## 1.1 Introduction

Wireless sensor networks (WSN) are widely used in environmental applications where the aim is to sense a physical phenomenon such as temperature, humidity, air pollution, etc. In this context of application, the use of WSN allows to understand the variations of the phenomenon over the monitoring region and therefore be able to take adequate decisions regarding the impact of the phenomenon. Air pollution is one of the main physical phenomena that still need to be studied and characterized because it highly depends on other phenomena such as temperature and wind variations. Moreover, air pollution affects human health dramatically according to the World Health Organization (WHO), which states that exposure to air pollution is accountable to seven million casualties in 2012 [1].

Most of current air pollution monitoring is performed using measuring stations which are accurate but massive, inflexible and expensive [2]. An alternative – or complementary – solution would be to use electrochemical sensor networks, which are smaller and cheaper and have a reasonable measurement quality [3][4]. The main advantages of a WSN infrastructure for air pollution monitoring is to obtain a finer spatiotemporal granularity of measurements, thanks to the resulting lighter installation and operational costs [5].

Using WSN for air pollution monitoring usually targets two main applications: i) the regular mapping where the objective is to construct pollution maps based on the data gathered by sensors; and ii) the detection of high pollution concentrations where the objective is to ensure that data gathered by sensors allow to detect the threshold crossings of air pollution [6]. Both of these applications need a careful deployment of sensors in order to get better knowledge of air pollution while ensuring a minimal deployment cost.

In this chapter, we present three formulations of the deployment issue of sensor and sink nodes based on integer linear programming modeling (ILP) while tackling the two main applications of air pollution monitoring. In the first ILP model, we target the regular mapping of air pollution while constraining the positions of sensors by the quality of pollution maps which will be constructed by data gathered by sensors once deployed. Since gathering data cannot be done before deployment, we base our first model on predicted pollution maps. In the second and third ILP models, we target the detection of pollution threshold crossings while ensuring that sensors will be deployed at positions where high concentrations may occur. The second model is based on predicted pollution maps as in the first one whereas the third one is based on pollution emission inventory which describes pollution sources and their emission rates. In addition to the constraints of pollution coverage, we also ensure that the deployed networks are connected so that every sensor is capable of sending its data to a sink node directly or through relay nodes. In all the three models, the objective is to minimize the financial cost of WSN deployment which is usually equivalent to minimizing the number of nodes to deploy. For each ILP model, we show a proof-of-concept performed on a real world data set. We also analyze the performance of the proposed models in terms of coverage and connectivity results.

The remainder of this chapter is organized as follows. We first review the related works on the WSN deployment issue in section 1.2. Then, we present and analyze the

literature methods used for air pollution prediction in section 1.3. Next, we present the formulation and the simulation results of our three ILP models in sections 1.4, 1.5 and 1.6. Finally, we provide some future directions and conclude the chapter in sections 1.7 and 1.8.

## 1.2 WSN deployment

The deployment issue of wireless sensor networks has been addressed extensively in the literature where several mathematical models, optimal algorithms and near-optimal heuristics have been proposed [7]. The problem consists in determining the optimal positions of sensors and sinks so as to cover the environment and ensure the network connectivity while optimizing an objective function such as the deployment cost or the coverage quality [8]. The network is said connected if each sensor can communicate information to at least one sink. Coverage requires that each point of interest is monitored with one or more sensor nodes. In addition to coverage and connectivity, the main issues targeted in the literature are energy consumption, network lifetime and the network deployment cost.

This section identifies what lacks in the literature and motivates the need of application-aware deployment models for air pollution monitoring where the aim is to place sensors in order to optimize their number and the quality of pollution coverage that results from the data gathered by sensors once deployed. We present the related works based on their coverage definition while identifying their formulation of connectivity and network lifetime.

Existing deployment approaches can be considered as either event-aware like the works presented in [9][10][11][12][13][14][15][16] or correlation-aware like the works presented in [17][18][19][20][21][22][23]. In the first case, a sensor is assumed to have a detection range, usually circular, within which the sensor is capable of detecting any event that may happen. The second class of deployment approaches is based on the correlation that sensor measurements may present in order to select the minimum number of sensor nodes.

### 1.2.1 Event-aware deployment methods

Chakrabarty et al. [9] represent the deployment region as a grid of points and propose a nonlinear formulation for minimizing the deployment cost of sensors while ensuring complete coverage of the deployment region. Then, they apply some transformations to linearize the first model and obtain an ILP formulation. The authors formulate coverage based on the distance between the different points of the deployment field. Each sensor has a circular detection area, which defines the points that the sensor can cover. Unfortunately, this measure of coverage is inadequate to the air pollution monitoring using electrochemical sensors since a sensor positioned at a point A cannot cover a neighboring point B if there is a difference between pollution concentrations at the two points.

Altinel et al. [10] proposed another formulation based on the Set Cover Problem, which is equivalent to the aforesaid model but less complex. They also extend their for-



mulation to take into account the probabilistic sensing of sensor nodes while assuming that a node is able to cover a given point with a certain predefined probability. Despite that, this new formulation is still generic since the dependency between the errors of the deployed sensors is not considered. However, this has to be taken into account when doing air pollution estimation.

Chang et al. [11] proposed to use data fusion in the definition of coverage in order to take into account the collaborative detection of targets. They based in their work on a probabilistic sensing model to define the probability of target detection and the false alarm rate. Then, they formulated a nonconvex optimization problem minimizing the number of nodes under coverage constraints. They presented resolution algorithms and showed that the obtained solutions are near-optimal and hence very close to the optimal ones. Still, this work considers the existence of a detection range.

In addition to coverage formulation, the authors of [12] formulate connectivity based on the flow problem while assuming that sensors generate flow units in the network and verify if sinks are able to recover them. Another connectivity formulation has been introduced in [13] where authors base on an assignment approach. They introduce in their ILP formulation new variables to define the communication paths between sensors and sinks. However, this model involves more variables than the one based on the flow problem and is therefore more complex.

In [14], authors study the trade-off between coverage, connectivity and energy consumption. They formulate the problem as an ILP model and then propose a multi-objective approach to optimize coverage, the network lifetime and the deployment cost while maintaining the network connectivity.

In some other works, authors suppose that a set of connected sensors that ensure coverage are already deployed, and propose integer programming formulations to find optimal sinks locations and sensors-to-sinks routes. Authors in [15] evaluate firstly the shortest path cost between each sensor and potential sink location using the Dijkstra algorithm. Two main metrics were proposed to compute shortest paths: energy cost and financial cost. In the second case, the proposed ILP model aims to find the optimal sinks positions while minimizing the financial cost of the sinks deployment and the sensors-to-sinks routes. Two other formulations based on flows were proposed in [16] where authors present a single commodity flow and a multi-commodity flow formulations. However, they show that the integer programming model presented in [15] is better. Moreover, they propose and test good heuristics for this latter.

## 1.2.2 Correlation-aware deployment methods

In [17], Roy et al. tackled the problem of finding the most informative locations of sensors for monitoring environmental applications. They assume the existence of a set of data snapshots characterizing the phenomenon to monitor. Then, they formulate the problem to find the best locations of sensors in order to reconstruct the data of the whole phenomenon with a required precision. Two optimization models are proposed to handle both stationary and non stationary-fields. An iterative resolution algorithm is proposed to solve the two deployment problems. Unfortunately, this work is based on a strong assumption; that is sensor measurements are perfect and do not present any

drift, which is not the case of pollution sensors where sensed values may be different than pollution ground truth values.

In [18], Krause et al. tackle the same problem based on the assumption that the variations of the phenomenon are Gaussian. They also assume a pre-deployment phase allowing to gather data that can be used to characterize the phenomenon. In order to select the best positions of sensors, they use the concept of mutual information in order to define the quality of a given topology. After the formulation of the problem, they use the sub-modularity of mutual information to define a polynomial algorithm. This work considers only coverage and is extended in [19] to take into account the cost of connectivity where the links qualities are assumed to be Gaussian. Since air pollution is not necessarily Gaussian, this work does not fit our application case.

In [20], authors focus on the specific application of soil moisture sensing while assuming the Gaussian distribution of the phenomenon. They first show that their application case presents some particular characteristics that can be used to design application-aware deployment schemes. Based on these characteristics, they propose a clustering approach that allows them to divide the deployment problem into a set of disjoint sub-problems. They perform extensive simulations while considering different correlation-aware deployment algorithms in the resolution of the sub-problems. They conclude based on the obtained results that their clustering approach allows them to get solutions that are as good as the global solution of the deployment problem.

In [21], authors consider a different context where some sensors are already deployed for the monitoring of the Columbia River. They perform field estimation based on sensor measurements in addition to a simulation model, which is commonly known as data assimilation. Their work is mainly designed for the determination of sensor nodes whose information is redundant and therefore can be turned off to optimize the network lifetime. Their approach can be also used to determine the optimal positions of sensors that should be added to the network in order to improve coverage quality. However, the proposed approach cannot be used to perform a first deployment of sensors without considering existing nodes or a trial deployment phase.

As in [21], authors of [22] consider an already deployed sensor network and propose an algorithm to define a sensing topology to select active sensors and turn off the others. They estimate the variations of the phenomenon in an online way to decide whether a sensor is to keep active or not. In contrary to this work, in our case, the sensing locations have to be chosen in an offline way since the selection of sensing points is performed before the network deployment.

The mathematical characteristics of the correlation-aware deployment problem has been studied by Ranieri et al. in [23] while considering a generic form. A greedy heuristic is proposed to solve the problem. They perform extensive simulations to show that their algorithm is capable of solving the problem in a short time compared to the existing heuristics while providing a near optimal solution.

### **1.2.3 Discussion**

Even if the recent works take into account network constraints like connectivity and energy consumption, all coverage formulations either assume that sensors have a given detection range, which is the case of event-aware methods, or the assumption is instead

made on the distribution of sensor measurements, which is the case of correlation-aware methods. Novel application-aware deployment methods have been recently proposed to consider the characteristics of the application case in the design of the deployment approach; examples include the work of [24] on wind monitoring and the work of [25] for water pollution monitoring. Following the same direction, we propose in the remainder of this chapter to consider the context of air pollution monitoring and present appropriate formulations of coverage in addition to network connectivity.

### 1.3 Air pollution prediction

The three realistic formulations presented in this chapter are based on the nature of the phenomenon. This is ensured by integrating in the ILP models the variations of pollutants over the deployment region. The variations can be estimated using the so-called air pollution prediction. In this section, we introduce the main methods used in this latter.

Air pollution prediction allows to estimate pollution concentrations at a given point in the environment based on some measurements performed in the neighborhood of the point in question, weather conditions, the urban characteristics of the environment and the characteristics of the pollution sources located within the deployment region, also known as the emission inventory [26]. Three major methods are used in air pollution prediction: atmospheric dispersion, interpolation and land-use regression [27].

#### 1.3.1 Atmospheric dispersion-based methods

Atmospheric dispersion models take as input locations of pollution sources, the pollutant emission rate of each pollution source and meteorological data in order to measure the pollutant concentration at a given location [27]. The obtained concentrations can then be calibrated using the measurements of sensors. The theoretical study of pollution atmospheric dispersion is mainly based on fluid mechanics theory [28]. For the sake of clarity, we present in this section only steady state dispersion, in particular Gaussian dispersion. The basic Gaussian model estimates the concentrations of a pollutant gas released by a pointwise pollution source in a free space environment [29]. The estimated value  $C$  ( $g/m^3$ ) at a measurement location  $(x, y, z)$  is given by Formula (1.1). Table 1.1 details the parameters of the model.

$$C(x, y, z) = \frac{Q}{2\pi V_w \sigma_y \sigma_z} e^{-\frac{y^2}{2\sigma_y^2}} \left( e^{-\frac{(z-H)^2}{2\sigma_z^2}} + e^{-\frac{(z+H)^2}{2\sigma_z^2}} \right) \quad (1.1)$$

$$\Delta h = \frac{1,6 \cdot F^{1/3} \cdot x^{2/3}}{V_w} \quad (1.2)$$

The pollution source is located at the point  $(0, 0, h_s)$  and the measurement point location is given according to a 3D coordinate system where the x-axis is oriented in the wind direction  $D_w$ . Parameters  $\sigma_y$  and  $\sigma_z$  describe the stability of the atmosphere and can be approximated using Briggs formulas:  $\sigma_y = a_y \cdot |x|^{b_y}$  and  $\sigma_z = a_z \cdot |x|^{b_z}$ . The parameter  $H$ , which represents the pollutant effective release height, is equal to the sum

<b>Measurement location</b>	
$x$	Downwind distance from the pollution source (m)
$y$	Crosswind distance from the pollution source (m)
$z$	Hight (m)
<b>Pollution emission parameters</b>	
$h_s$	Pollutant source height (m)
$Q$	Mass flow rate at the emission point (g/s)
$V$	Volumetric flow rate at the emission point ( $m^3/s$ )
$T_s$	Pollutant temperature at the emission point (K)
<b>Weather</b>	
$T$	Ambient air temperature (K)
$V_w$	Wind velocity (m/s)
$D_w$	Wind direction (degree)
<b>Constants</b>	
$a_y, b_y$	Horizontal dispersion coefficients
$a_z, b_z$	Vertical dispersion coefficients
$g$	Gravity constant ( $9.8m/s^2$ )

Table 1.1: Parameters of the Gaussian dispersion model

of the pollutant source height  $h_s$  and the plume rise  $\Delta h$ . The pollution plume is located above the pollution source and  $\Delta h$  is the vertical distance between the source and the center of the pollution plume. Briggs formulas are commonly used for the calculation of the  $\Delta h$  parameter. To simplify the analysis, we only consider the case where the temperature of the pollutant  $T_s$  is greater than the ambient air temperature  $T$ , which is usually the case. In this case, the value of  $\Delta h$  is given by Formula (1.2) where  $F$ , which denotes the pollutant gas buoyancy, is computed using Formula (1.3).

$$F = \frac{g}{\pi} \cdot V \cdot \left( \frac{T_s - T}{T_s} \right) \quad (1.3)$$

The Gaussian model considers only one scenario of weather conditions at a time to compute pollution concentrations: wind direction affects the direction of the pollution plume and variations of the ambient temperature and the wind velocity affect the concentrations of the pollution plume. Moreover, formula (1.1) takes into account only pointwise pollution sources, and thus cannot be applied to area sources like crossroads and line sources like highways. Multiple extensions have been proposed in the literature to deal with these kinds of pollution sources. In addition, many enhanced systems have been developed based on the Gaussian model to take into account complex meteorological data, the effect of buildings on pollution dispersion, etc.

### 1.3.2 Interpolation-based prediction methods

Interpolation methods formulate the estimated concentration  $\widehat{\mathcal{L}}_p$  at a given location  $p \in \mathcal{P}$  as a weighted combination of the measured concentrations  $\mathcal{L}_q, q \in \mathcal{P} - \{p\}$  [30]. The weights of the measured concentrations  $\mathcal{W}_{pq}$  can be evaluated in a deterministic way based on the distance between the location of the measured concentration and the location of the estimated concentration. In this case, which is called the Inverse Distance Weighting interpolation,  $\widehat{\mathcal{L}}_p$  is evaluated using formula 1.4. The concentration weights can also be evaluated in a stochastic way, the most used method doing so is called kriging.

$$\widehat{\mathcal{L}}_p = \frac{\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * \mathcal{L}_q}{\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq}} \quad (1.4)$$

### 1.3.3 Regression-based prediction methods

Land-use regression models are a kind of stochastic regression models [31]. The idea behind these models is to evaluate the pollution concentration at a given location based on the concentrations of locations that are similar in terms of land-use parameters such as the elevation and the distance to the closest busy road.

## 1.4 Model 1: WSN deployment for air pollution mapping based on predicted data

In this section, we present an integer programming model of WSN deployment ensuring air pollution mapping. We formulate the constraint of air pollution coverage based on interpolation methods in order to determine the optimal positions of sensors allowing to better estimate pollution concentrations at positions where no sensor is deployed. Our coverage formulation takes into account the sensing drift of sensor nodes and the impact of weather conditions on air pollution dispersion. We base on the flow problem to formulate the connectivity constraint that ensures that the deployed sensors are able to send pollution data to at least one sink. We provide an evaluation of the ILP model on a dataset of the Lyon city while analyzing the coverage and connectivity results.

### 1.4.1 Problem formulation

#### Overview

We consider as input of our model the map of a given urban area that we call the deployment region. We start by discretizing the deployment region in order to get a set of points  $\mathcal{P}$  approximating the urban area at a high-scale ( $|\mathcal{P}| = \mathcal{N}$ ). Our goal is to be able to determine with a high precision the concentration value at each point  $p \in P$ . We ensure that for each point  $p \in P$ , either a sensor is deployed or the pollution concentration can be estimated with a high precision based on data gathered by the neighboring deployed sensors.

In general case, the set  $\mathcal{P}$  is thus considered as the set of potential positions of WSN nodes. However, in smart cities applications, some restrictions on node positions may apply because of authorization or practical issues. For instance, in order to alleviate the energy constraints, we may place sensors on only lampposts and traffic lights as experimented in [32]. When this is the case, we do not consider as potential positions the points  $p \in P$  where sensors cannot be deployed.

We use decision variables  $x_p$  (respectively  $y_p$ ) to specify if a sensor (respectively a sink) is deployed at point  $p$  or not. Sensors and sinks may have different costs, thus we denote by  $c_p^{sensor}$  (respectively  $c_p^{sink}$ ) the sensor (respectively the sink) deployment cost at position  $p$ . We summarize in Table 1.2 the notations used in the formulations.

<b>Parameters</b>	
$\mathcal{P}$	Set of points approximating the deployment region
$\mathcal{N}$	Number of points approximating the deployment region
$\mathcal{L}_p$	Reference pollution concentration at point $p$
$\mathcal{W}_{pq}$	Correlation coefficient between points $p$ and $q$
$\mathcal{D}$	The correlation distance function
$d$	Maximum correlation distance
$\alpha$	Attenuation coefficient of the correlation distance
$\Gamma(p)$	Communication neighborhood of a node deployed at point $p$
$\mathcal{R}$	Communication range of sensor nodes
$\mathcal{E}_p$	The tolerated estimation error at point $p$
$\mathcal{M}$	The maximum number of sinks
$c_p^{sensor}$	The cost of deploying a sensor at point $p$
$c_p^{sink}$	The cost of deploying a sink at point $p$
<b>Variables</b>	
$x_p$	Define whether a sensor is deployed at point $p$ or not $x_p \in \{0, 1\}, p \in \mathcal{P}$
$y_p$	Define whether a sink is deployed at point $p$ or not $y_p \in \{0, 1\}, p \in \mathcal{P}$
$g_{pq}$	Flow quantity transmitted from node $p$ to node $q$ $g_{pq} \in \{0, 1, \dots\}, p \in \mathcal{P}, q \in \Gamma(p)$

Table 1.2: Summary of the model notations.

### Objective function

The objective of the ILP model is to minimize the overall financial cost of the resulting network. The cost function to minimize is thus given as follows:

$$\mathcal{F} = \sum_{p \in \mathcal{P}} c_p^{sensor} * x_p + \sum_{p \in \mathcal{P}} c_p^{sink} * y_p \quad (1.5)$$

## Coverage

**Basic formulation** As claimed before, our idea is to base on interpolation methods in order to ensure that the deployed sensors allow to estimate with a high precision the pollution concentrations at locations where no sensor is deployed. This means that we need to have an idea on the dispersion of pollution concentrations in the deployment region in order to be able to formulate the coverage constraint. More exactly, we need to know the variability of pollution concentrations among the set of points  $P$  in order to use the formulation of interpolation methods. Fortunately, using numerical atmospheric dispersion models, we can obtain simulated pollution concentrations that may be considered as reference pollution concentrations [33]. This does not mean that these reference concentrations are real but they reflect the best today's pollution knowledge.

Let  $\mathcal{Z}_p$  denote the reference concentration value at point  $p$ . Given the set of selected points where sensors will be deployed  $\{p \text{ where } x_p = 1\}$ , we evaluate the estimated pollution concentrations  $\widehat{\mathcal{Z}}_p$  at points  $\{p \text{ where } x_p = 0\}$  based on reference values corresponding to the selected points, i.e. based on  $\mathcal{Z}_p$  where  $p \in \{p \text{ where } x_p = 1\}$ , as follows:

$$\left\{ \begin{array}{l} \widehat{\mathcal{Z}}_p = \frac{\sum_{q \in \mathcal{D} - \{p\}} \mathcal{W}_{pq} * \mathcal{Z}_q * x_q}{\sum_{q \in \mathcal{D} - \{p\}} \mathcal{W}_{pq} * x_q}, p \in \mathcal{D} \ \& \ x_p = 0 \\ \sum_{q \in \mathcal{D} - \{p\}} \mathcal{W}_{pq} * x_q > 0, p \in \mathcal{D} \ \& \ x_p = 0 \end{array} \right. \quad (1.6)$$

The  $\widehat{\mathcal{Z}}_p$  expression is formulated based on formula 1.4 given in section 1.3.2. We have chosen this formula because the weights  $\mathcal{W}_{pq}$  are given in a deterministic way, which allows to integrate them to the ILP deployment model. We ensure that the denominator of  $\widehat{\mathcal{Z}}_p$  is never equal to zero using the second part of formula 1.6. The  $\mathcal{W}_{pq}$  parameter is the correlation coefficient between points  $p$  and  $q$  and is calculated using formula 1.7 based on the distance between the two points.  $\mathcal{D}(p, q)$  is the distance function.  $\alpha$  is the attenuation coefficient of the correlation distance, this means that for greater values of  $\alpha$ , very low correlation coefficients are assigned to far points. The last parameter of formula 1.7 is the maximum correlation distance, which defines the range of correlated neighboring points of a given point.

In order to take into account the impact of the urban topography on the dispersion of pollutants, let  $\mathcal{D}$  be the shortest distance along the roads network. This allows to assign small correlation values to points that are separated by buildings, even if they are close [34].

$$\mathcal{W}_{pq} = \begin{cases} \frac{1}{\mathcal{D}(p, q)^\alpha} & \text{if } q \in \text{Disc}(p, d) - \{p\} \\ 0 & \text{if } q \notin \text{Disc}(p, d) \end{cases} \quad (1.7)$$

In order to ensure that the concentration is estimated with high precision at points where no sensor is deployed, we define constraint 1.8. The  $\mathcal{E}_p$  parameter corresponds to the estimation error that is tolerated at point  $p$ . The choice of different values of  $\mathcal{E}_p$

in function of  $p$  allows to assign low tolerated estimation errors to locations that are sensitive to air quality such as hospitals, primary schools, etc.

$$\left| \widehat{\mathcal{L}}_p - \mathcal{L}_p \right| \leq \mathcal{E}_p, \quad p \in \mathcal{P} \ \& \ x_p = 0 \quad (1.8)$$

By replacing  $\widehat{\mathcal{L}}_p$  by its expression given in formula 1.6, we obtain the coverage constraints 1.9 and 1.10. These two constraints should be linearized in order to get an ILP formulation.

$$\left| \frac{\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * \mathcal{L}_q * x_q}{\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q} - \mathcal{L}_p \right| \leq \mathcal{E}_p, \quad p \in \mathcal{P} \ \& \ x_p = 0 \quad (1.9)$$

$$\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q > 0, \quad p \in \mathcal{P} \ \& \ x_p = 0 \quad (1.10)$$

**Linearization of constraint 1.9** The first step is to linearize the fraction part; this allows to get constraint 1.11. Then, we have to ensure that the constraint is relaxed when  $x_p = 1$ . To do so, notice that the left member of constraint 1.11 can be bounded as presented in formula 1.12. Based on this, we add  $x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |\mathcal{L}_q - \mathcal{L}_p|$  to the right member of constraint 1.11 to relax it when  $x_p = 1$ . Hence, we obtain constraint 1.13. Finally, we have to linearize the absolute-value function. Hence, we get the linear form of constraint 1.9 in constraints 1.14 and 1.15.

$$\left| \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q * (\mathcal{L}_q - \mathcal{L}_p) \right| \leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q, \quad p \in \mathcal{P}, x_p = 0 \quad (1.11)$$

$$\left| \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q * (\mathcal{L}_q - \mathcal{L}_p) \right| \leq \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |\mathcal{L}_q - \mathcal{L}_p| \quad (1.12)$$

$$\left| \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q * (\mathcal{L}_q - \mathcal{L}_p) \right| \leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q + x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |\mathcal{L}_q - \mathcal{L}_p|, \quad p \in \mathcal{P} \quad (1.13)$$

$$\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q * (\mathcal{L}_q - \mathcal{L}_p) \leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q + x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |\mathcal{L}_q - \mathcal{L}_p|, \quad p \in \mathcal{P} \quad (1.14)$$

$$\sum_{q \in \mathcal{P} - \{p\}} -\mathcal{W}_{pq} * x_q * (\mathcal{L}_q - \mathcal{L}_p) \leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q + x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |\mathcal{L}_q - \mathcal{L}_p|, \quad p \in \mathcal{P} \quad (1.15)$$



**Linearization of constraint 1.10** The only thing to do to linearize constraint 1.10 is to relax the constraint when  $x_p = 1$ . This can be obtained by replacing the right member of the constraint by  $-x_p$ , which allows to get the constraint 1.16.

$$\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q > -x_p, p \in \mathcal{P} \quad (1.16)$$

**Taking into account sensing drift** Usually, the pollution concentration measured at point  $q$  is not equal to the ground truth value  $\mathcal{L}_q$  and depends on the sensing technology and the quality of sensors. This involves a certain drift in pollution measurements. The sensing drift is usually given by two parameters  $a_q$  and  $b_q$ , which define the measured concentration that is equal to  $a_q * \mathcal{L}_q + b_q$ . By introducing parameters  $a_q$  and  $b_q$  in formula 1.6, we get in formula 1.17 the new definition of the estimated pollution concentration at a given point  $p$  depending on the deployed sensors. Using this new definition, we transform the coverage constraints 1.14 and 1.15 into constraints 1.18 and 1.19, which allows us to include the sensing drift in our coverage model. In this formulation, parameters  $a_q$  and  $b_q$  are assumed to be constants. When it is not the case, stochastic programming should be used instead of integer programming.

$$\widehat{\mathcal{L}}_p = \frac{\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * (a_q * \mathcal{L}_q + b_q) * x_q}{\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q} \quad (1.17)$$

$$\begin{aligned} \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q * (a_q \mathcal{L}_q + b_q - \mathcal{L}_p) \leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q \\ + x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |a_q \mathcal{L}_q + b_q - \mathcal{L}_p|, p \in \mathcal{P} \end{aligned} \quad (1.18)$$

$$\begin{aligned} \sum_{q \in \mathcal{P} - \{p\}} -\mathcal{W}_{pq} * x_q * (a_q \mathcal{L}_q + b_q - \mathcal{L}_p) \leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q \\ + x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |a_q \mathcal{L}_q + b_q - \mathcal{L}_p|, p \in \mathcal{P} \end{aligned} \quad (1.19)$$

**Taking into account weather conditions** Air pollution dispersion highly depends on weather conditions such as wind and temperature. For instance, reference pollution concentrations  $\mathcal{L}_p$  can be totally different if there is a change in wind direction. In order to cope with that, we consider multiple snapshots of reference concentrations. The resolution of snapshots may be yearly, monthly or daily depending on the needed deployment accuracy and the data availability. Let  $\mathcal{T}$  be the set of snapshots and  $\mathcal{L}_p^t$  be the reference pollution concentration at point  $p$  in snapshot  $t$ . We propose to ensure that constraints 1.18 and 1.19 are verified for each snapshot  $t \in \mathcal{T}$ , hence we get constraints 1.20 and 1.21. This allows us to place sensor nodes while taking into account the different weather scenarios corresponding to each pollution snapshot.

$$\begin{aligned}
\sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q * (a_q \mathcal{L}_q^t + b_q - \mathcal{L}_p^t) &\leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q \\
+x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |a_q \mathcal{L}_q^t + b_q - \mathcal{L}_p^t|, &p \in \mathcal{P}, t \in \mathcal{T}
\end{aligned} \tag{1.20}$$

$$\begin{aligned}
\sum_{q \in \mathcal{P} - \{p\}} -\mathcal{W}_{pq} * x_q * (a_q \mathcal{L}_q^t + b_q - \mathcal{L}_p^t) &\leq \mathcal{E}_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * x_q \\
+x_p * \sum_{q \in \mathcal{P} - \{p\}} \mathcal{W}_{pq} * |a_q \mathcal{L}_q^t + b_q - \mathcal{L}_p^t|, &p \in \mathcal{P}, t \in \mathcal{T}
\end{aligned} \tag{1.21}$$

### Connectivity

We formulate the connectivity constraint as a network flow problem. We consider the same potential positions set  $\mathcal{P}$  for sensors and sinks. We first denote by  $\Gamma(p)$ ,  $p \in \mathcal{P}$ , the set of neighbors of a node deployed at the potential position  $p$ . This set can be determined using sophisticated path loss models. It can also be determined using the binary disc model, in which case  $\Gamma(p) = \{q \in \mathcal{P} \text{ where } q \in \text{Disc}(p, R)\}$  where  $R$  is the communication range of sensors. Then, we define the decision variables  $g_{pq}$  as the flow quantity transmitted from a node located at potential position  $p$  to another node located at potential position  $q$ . We suppose that each sensor of the resulting WSN generates a flow unit in the network, and verify if these units can be recovered by sinks. The following constraints ensure that the deployed sensors and sinks form a connected wireless sensor network; i.e. each sensor can communicate with at least one sink.

$$\sum_{q \in \Gamma(p)} g_{pq} - \sum_{q \in \Gamma(p)} g_{qp} \geq x_p - (\mathcal{N} + 1) * y_p, p \in \mathcal{P} \tag{1.22}$$

$$\sum_{q \in \Gamma(p)} g_{pq} - \sum_{q \in \Gamma(p)} g_{qp} \leq x_p, p \in \mathcal{P} \tag{1.23}$$

$$\sum_{q \in \Gamma(p)} g_{pq} \leq \mathcal{N} * x_p, p \in \mathcal{P} \tag{1.24}$$

$$\sum_{p \in \mathcal{P}} \sum_{q \in \Gamma(p)} g_{pq} = \sum_{p \in \mathcal{P}} \sum_{q \in \Gamma(p)} g_{qp} \tag{1.25}$$

$$\sum_{p \in \mathcal{P}} y_p \leq \mathcal{M} \tag{1.26}$$

Constraints 1.22 and 1.23 are designed to ensure that each deployed sensor, i.e. such that  $x_p = 1$ , generates a flow unit in the network. These constraints are equivalent to the following.

$$\sum_{q \in \Gamma(p)} g_{pq} - \sum_{q \in \Gamma(p)} g_{qp} \begin{cases} = 1 & \text{if } x_p = 1, y_p = 0 \\ = 0 & \text{if } x_p = y_p = 0 \\ \leq 0, \geq -\mathcal{N} & \text{if } x_p = 1, y_p = 1 \end{cases}$$

The first case corresponds to deployed sensors that should generate, each one of them, a flow unit. The second case, combined with constraint 1.24, ensures that absent nodes, i.e.  $x_p = y_p = 0$ , do not participate in the communication. The third case concerns deployed sinks, and ensures that each sink cannot receive more than  $\mathcal{N}$  units. Constraint 1.25 means that the overall flow is conservative. The flow sent by deployed sensors has to be received by deployed sinks. Finally, constraint 1.26 allows to fix the maximum number of sinks  $\mathcal{M}$  of the resulting network.

### ILP model

The ILP model allowing for air pollution mapping based on predicted pollution maps is as follows:

$$\begin{aligned}
 & [ILP\ 1] \\
 & \textit{Minimize} \quad (1.5) \\
 & \textit{Subject to.} \quad (1.16), (1.20), (1.21), (1.22), (1.23), (1.24), (1.25) \\
 & \quad \textit{and} \quad (1.26)
 \end{aligned}$$

### 1.4.2 Simulation results

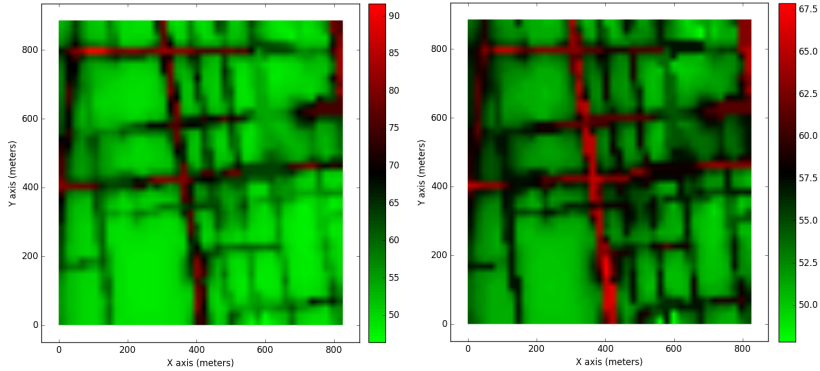
In this section, we present the simulations that we have performed in order to validate our model and analyze its performances. We first present the data set that we have used and the common simulation parameters. Then, we give a proof-of-concept in order to show how error-bounded deployment is done. Finally, we analyze the results of pollution coverage and network connectivity while studying the compromise between the estimation precision and the deployment cost under different configurations of the correlation distance.

#### Simulation dataset

In order to consider the real dispersion of air pollutants in the reference pollution concentrations  $\mathcal{Z}_p^t$ , we perform our simulations on 2 pollution snapshots generated by an enhanced atmospheric dispersion simulator called SIRANE [33]. This simulator is designed for urban areas and takes into account the impact of street canyons on pollution concentrations. The dataset has been provided by Air-Rhone-Alpes, which is an observatory for air pollution monitoring within the Lyon region of France [35].

We depict the 2 reference pollution maps in Figure 1.1. We focus on nitrogen dioxide ( $NO_2$ ) monitoring since this pollutant is mainly due to road traffic. We evaluate our ILP model on the La-Part-Dieu district, which is the heart of the Lyon City. Pollution map granularity is around 5 meters and concentrations correspond to the years 2012 and 2013.

We have implemented the ILP formulation using IBM ILOG CPLEX Optimization Studio and used a computer with Intel Xeon E5649 processor under Linux. Simulation parameters are summarized in Table 1.6. We discretize the deployment region which is of around  $0.7km^2$  using a resolution of 50 meters, thus we get 306 discrete points.



(a) NO<sub>2</sub> concentrations ( $\mu\text{g}/\text{m}^3$ ) of 2012 (b) NO<sub>2</sub> concentrations ( $\mu\text{g}/\text{m}^3$ ) of 2013

Figure 1.1: Reference NO<sub>2</sub> concentrations in Lyon La-Part-Dieu district, average over years 2012 and 2013 (Ref:Air-Rhone-Alpes).

We consider all these points as potential positions of nodes. We use the same tolerated estimation error  $\mathcal{E}_p = \mathcal{E}$  for all the points  $p \in \mathcal{P}$ . By default, we use the distance along roads for the evaluation of the correlation coefficients and we suppose that sensing is perfect. In addition, we fix the maximum number of sinks to 1 in order to get mono-sink networks.

Parameter	Notation	Value
Number of discrete points	$\mathcal{N}$	306
Maximum correlation distance	$d$	100m
Attenuation coefficient of correlation distance	$\alpha$	2
Communication range of sensor nodes	$\mathcal{R}$	150m
The tolerated estimation error at point $p$	$\mathcal{E}_p$	$10\mu\text{g}/\text{m}^3$
The maximum number of sinks	$\mathcal{M}$	1
The cost of deploying a sensor at point $p$	$c_p^{\text{sensor}}$	1 <i>unity</i>
The cost of deploying a sink at point $p$	$c_p^{\text{sink}}$	10 <i>unities</i>

Table 1.3: Default values of simulation parameters.

### Proof of concept

In order to validate our formulation of error-bounded WSN deployment, we ran the ILP model using the 2 reference pollution maps while considering 3 values for the tolerated estimation error: 2, 5 and 8  $\mu\text{g}/\text{m}^3$ . We depict in table 1.4 the resulting optimal cost corresponding to the snapshot of 2012 alone, the snapshot of 2013 alone and the two snapshots together (using formulations given in section 1.4.1). We notice that snapshot 2012 needs more sensors than snapshot 2013. This is because the range of pollution concentrations is larger in snapshot 2012 as shown in figure 1.1, which involves higher

pollution variability and thus more interpolation errors. In addition, when trying to satisfy both of the two snapshots, we place at least the sensors that are required by snapshot 2012 since this snapshot is more complicated than the other one.

Tolerated estimation error	$2\mu\text{g}/\text{m}^3$	$5\mu\text{g}/\text{m}^3$	$8\mu\text{g}/\text{m}^3$
Snapshot 2012 alone	221	146	105
Snapshot 2013 alone	171	67	48
Snapshots 2012 and 2013 together	237	148	105

Table 1.4: Deployment cost (monetary units) depending on snapshots and the tolerated estimation error.

We now depict in Figure 1.2 the obtained positions of sensors and sinks when using only snapshot 2012. We evaluate at each point of the map the estimated concentration and then we calculate the resulting estimation error, i.e. the difference between the reference concentrations and the estimated concentrations. The obtained errors are also depicted in Figure 1.2.

We notice that more sensors are used when the tolerated estimation error decreases. This is expected since better deployment precision needs more sensor nodes. In addition, Figure 1.2 shows that the maximum error value is bounded by the tolerated estimation error, which fits with our coverage formulation. Moreover, the obtained nodes form a connected network as formulated in our connectivity constraint. In the next simulation cases, we execute the model only on the snapshot of year 2012.

### Evaluation of the coverage results

In this simulation scenario, we study the dependency between the deployment precision and the needed number of sensors under different configurations of the correlation distance. Since we are studying the cost of the monitoring precision, we execute only the coverage constraint. We depict in Figure 1.3 the optimal deployment cost depending on the tolerated estimation error while considering two different functions of the correlation distance: the Euclidean distance and the distance along roads. We notice in the two curves that fewer sensors are needed when the tolerated estimation error decreases. This is because less tolerated estimation error involves high-precision deployment and thus more nodes. In addition, the Euclidean distance gives less number of sensors, which is explained by the fact that the distance along roads is more realistic and hence involves more nodes to better estimate pollution concentrations.

We now investigate the impact of the correlation distance on the number of sensors that are needed to cover a point where no sensor is deployed. We consider different configurations of the maximum correlation distance  $d$  and the attenuation coefficient of the correlation distance  $\alpha$ . We depict in Figure 1.4 the average number of sensors that are deployed within the maximum correlation distance  $d$  of each point where no sensor is deployed. We notice that fewer sensors are used when considering greater values of the attenuation coefficient  $\alpha$  and less values of the maximum correlation distance  $d$ . To explain this, we recall that smaller values of the  $d$  parameter allow to consider

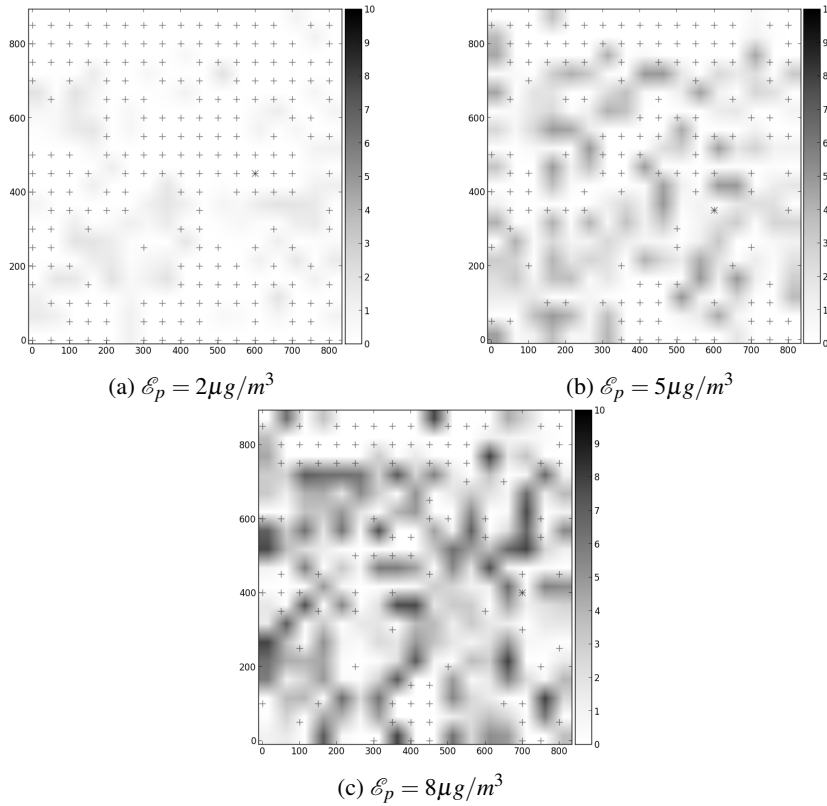


Figure 1.2: Deployments with increasing estimation errors ( $\mu\text{g}/\text{m}^3$ ) for snapshot 2012. Sensors (respectively sinks) are depicted using "plus signs" (respectively stars).

less points in the interpolation formula, and higher values of the  $\alpha$  parameter allow to assign smaller values of correlation coefficients to the far sensors. This means that with smaller values of  $d$  and higher values of  $\alpha$ , the interpolation is done with fewer sensors, which fits with the results depicted in Figure 1.4.

### Evaluation of the network connectivity

In this simulation case, we analyze the deployment cost that is due to the connectivity constraint, which involves the deployment of sink and relay nodes. We consider 3 possible values of the communication range of sensor nodes: 80m for short range communications like ZigBee, 150m for medium range communications like WiFi and 1000m for long-range communications. We evaluate the number of nodes depending on the tolerated estimation error and depict the results in Fig. 1.5. Obviously, the longer the communication range, the less the number of sensors is. However, the tolerated estimation error has a considerable impact on the connectivity of the network. On the one hand, the medium and long range communications involve nearly the same

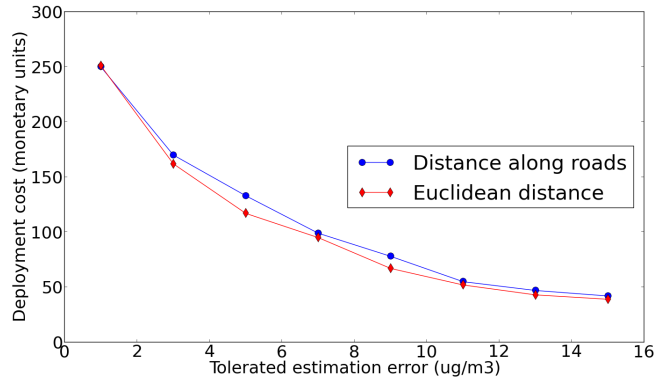


Figure 1.3: Optimal coverage cost vs. tolerated estimation error.

number of nodes. For instance, when estimation errors are less than  $10\mu\text{g}/\text{m}^3$ , the medium range communications need at most only two more nodes than the high range communications. This is explained by the fact that small tolerated errors imply a very high density of the network so that sensors are placed almost in all positions. On the other hand, short range communications are very costly and need almost 70% more nodes than the long range communications when the estimation errors reach  $15\mu\text{g}/\text{m}^3$ . This is because the distance between sensor nodes that are responsible for coverage is very important when high estimation errors are tolerated, which causes the need of too much relay nodes if the communication range is very short.

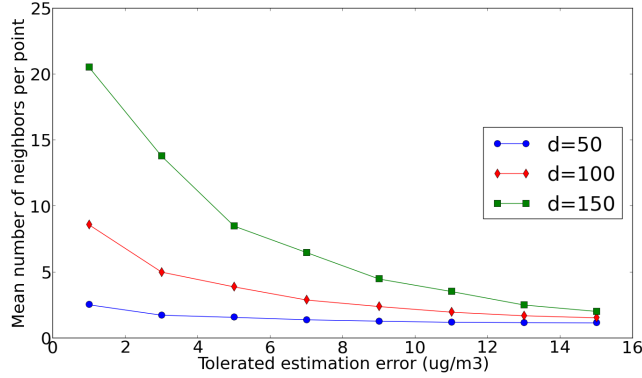
## 1.5 Model 2: WSN deployment for air pollution detection based on predicted data

In this section, we present an integer programming model of WSN deployment ensuring air pollution detection based on spatial analysis of air pollution predicted maps. We consider as input of the model a set of air pollution zones where sensors have to be deployed. This set of zones is determined using ZSCAN, an extension of DBSCAN which is a well-known and widely used spatial data clustering algorithm [36]. The ILP model that we present in this section allows to perform a minimum-cost deployment of sensors while ensuring pollution coverage and network connectivity in a joint way. We evaluate the ILP model on a data set of the Paris city and analyze the impact of some parameters on the deployment results.

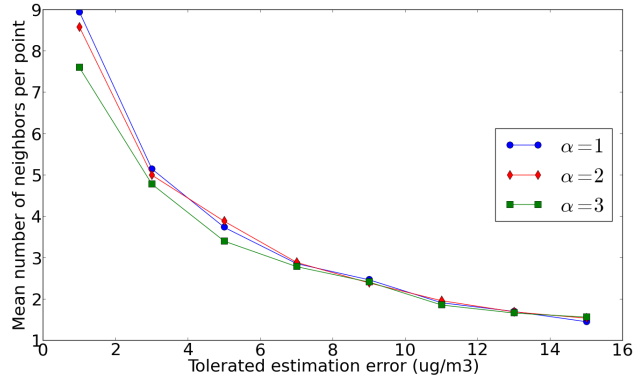
### 1.5.1 Problem formulation

#### Overview

In order to deploy sensors efficiently in a given city, the ILP model requires as a primary input, air pollution predicted maps. Given a pollutant to monitor, this consists of estimated values of the pollutant concentrations in the whole city for different time



(a) Impact of the maximum correlation distance



(b) Impact of the attenuation coefficient of correlation distance

Figure 1.4: Impact of correlation distance parameters on the mean number of neighbors per point.

instants. As explained in section 1.3, these concentrations can be estimated using atmospheric dispersion modeling based on the locations of pollution sources and meteorological data. They can be also obtained using interpolation algorithms based on some measurements established by a set of monitoring stations.

In the following, we denote by  $\mathcal{T}$  the set of time instants when pollution is estimated, and  $\mathcal{S}$  the set of spatial points representing the city. The second set is defined by applying a high resolution discretization process. For each time instant  $t \in \mathcal{T}$  and spatial point  $i \in \mathcal{S}$ , let  $C_{t,i}$  denote the estimated or measured pollution concentration. In addition to air pollution estimated concentrations, the model also requires data on sensor potential positions, this corresponds to positions where sensors can be deployed. We denote in what follows this set by  $\mathcal{P}$ .

Before the execution of the ILP model, a spatial clustering algorithm is applied to the air pollution maps in order to determine pollution zones that are due to the same pollutant sources. The ILP model allows to define a mono-sink connected wireless



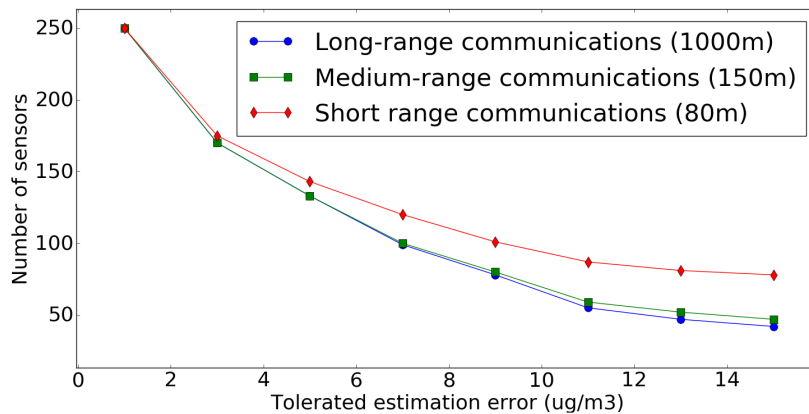


Figure 1.5: Optimal number of sensors depending on the communication range.

sensor network that covers all the pollution zones  $z \in \mathcal{Z}$  while minimizing the deployment cost. Pollution monitoring is ensured by deploying at least one sensor in each pollution zone to ensure the coverage constraint. Once the deployment positions of the connected sensors are determined, the sink location can be defined among sensor positions in order to optimize energy consumption, end-to-end transmission delay, etc. The sets used in the ILP model are summarized in table 1.5.

Notation	Description
$\mathcal{T}$	Set of time instants of pollution estimations
$\mathcal{I}$	Set of discrete points of pollution estimations
$\mathcal{P}$	Set of potential positions of sensors and sinks
$\mathcal{Z}$	Set of pollution zones

Table 1.5: Summary of the model sets.

### Identification of pollution zones

We present hereafter ZSCAN, a clustering algorithm which is applied to the estimated concentrations in order to group points that belong to the same pollution zone.

As presented in Algorithm 1, ZSCAN identifies all the pollution zones occurring in each time instant. To this end, pollution peaks, i.e. points having the highest pollution concentration, are first identified. A pollution zone is created every time a peak is identified using the *construct* function, which starts by adding all the neighbors of the pollution peak  $i$  to the zone under construction. The neighborhood of a point in the map is defined as the set of closer and unvisited points whose pollution concentration estimated in  $t$  is less. The neighbors of each chosen point are then added to the current zone. This process stops when it arrives at a point whose neighborhood set is empty,

---

**Algorithm 1** ZSCAN

---

**Inputs:**  $\mathcal{T}, \mathcal{S}, \{C_{t,i}; t \in \mathcal{T}, i \in \mathcal{S}\}$   
**Output:**  $\mathcal{Z}$   
 $\mathcal{Z} \leftarrow \emptyset$   
**for**  $t \in \mathcal{T}$  **do**  
    Mark all the points in  $\mathcal{S}$  as unvisited  
    **repeat**  
        Let  $i$  be the unvisited point having the highest  
        concentration  $C_{t,i}$   
         $z \leftarrow \text{construct}(i, t)$   
        Mark all the points in  $z$  as visited  
         $z \leftarrow \text{filter}(z, \Delta C)$   
         $z \leftarrow \text{points\_to\_polygon}(z)$   
         $\mathcal{Z} \leftarrow \mathcal{Z} \cup \{z\}$   
    **until** all the points in  $\mathcal{S}$  are visited

---

meaning that its neighbors have higher pollution concentration values. Once the current pollution zone is completely identified, a filtering function is applied to keep only points where pollution concentration is sufficiently closer the peak value, i.e. points where pollution concentration difference with the peak value is less than a threshold that we denote by  $\Delta C$ . This increases the chance that a sensor deployed in the detected zone is able to monitor the corresponding pollution sources. At the end, a geometrical form is given to the found zone by applying the function `points_to_polygon`.

The main difference between DBSCAN and ZSCAN is that this latter visits points in an ordered way, starting by pollution peaks, and then expands the peak neighborhood to detect pollution zones.

### ILP deployment model

Let  $G = \{V, A\}$  be a flow oriented graph where vertices set  $V$  corresponds to the pollution zones and the sensors potential positions, i.e.  $\mathcal{Z} \cup \mathcal{P}$ . Note that each zone is considered as a single vertex. Let  $A(i)$  denote the neighborhood of  $i \in V$ . We define a first set of arcs from each pollution zone  $z \in \mathcal{Z}$  to sensor potential positions  $p$  that are within its region, i.e.  $p \in z$ . A second set of arcs is defined from each sensor potential position  $p \in \mathcal{P}$  to positions which are in its communication range that we denote by  $\Gamma(p)$ .

The idea behind the formulation is that each pollution zone inserts one flow unit in the network through the first set of arcs. For a given selected positions of sensor nodes, the latter ensures network coverage and connectivity if and only if the received units from pollution zones can be forwarded by these nodes through the second set of arcs so that a chosen pollution zone can recover all the generated units (which ensures the connectivity of the defined graph). This particular zone does not generate units but recovers all of them from a sensor that is placed within its region. With these considerations, the selected sensor positions ensure jointly coverage and connectivity

if the recovering pollution zone gets all the flow units generated by the other pollution zones and forwarded by the selected sensors. Indeed, Flow passing in the first set of arcs guaranties coverage, and connectivity is verified due to flow forwarded by only selected positions through the second set of arcs. In what follows, we choose the first pollution zone  $\mathcal{Z}^0$  to be the one that recovers flow units, meaning that the other zones generate, each one, a unique flow unit.

We use binary decision variables  $x_p$  to specify if a sensor should be placed at a position  $p$  or not. Sensors cost may depend on their positions, thus we denote by  $cost_p$  the cost of deploying a sensor at a position  $p$ . We also define the positive integer decision variables  $f_{ij}$  as the flow quantity transmitted from  $i$  to  $j$ . The flow domain is set to  $\{0, |\mathcal{Z}| - 1\}$  for  $j = \mathcal{Z}^0$  in order to ensure that  $\mathcal{Z}^0$  recovers the units from only one sensor.

The proposed model **ILP 2** minimizes the overall deployment cost as formulated in the objective function. Constraints 1.28 ensure that each pollution zone except the first one generates exactly a flow unit. Sensors are flow conservative thanks to constraints 1.30. Constraints 1.31 ensure that sensors that are not selected ( $x_p = 0$ ) do not participate in communication. This means that generated flow units will be transmitted by only present sensors. Finally, the first pollution zone receives all the generated units, thanks to constraint 1.29.

$$[\text{ILP 2}] \text{Min } \sum_{p \in \mathcal{P}} cost_p * x_p \quad (1.27)$$

S.T.

$$\sum_{p \in \mathcal{Z}} f_{zp} = 1, \quad z \in \mathcal{Z} - \{\mathcal{Z}^0\} \quad (1.28)$$

$$\sum_{p \in \mathcal{Z}^0} f_{p\mathcal{Z}^0} = |\mathcal{Z}| - 1 \quad (1.29)$$

$$\sum_{q \in A(p)} f_{pq} - \sum_{q \in A(p)} f_{qp} = 0, \quad p \in \mathcal{P} \quad (1.30)$$

$$\sum_{q \in A(p)} f_{pq} \leq (|\mathcal{Z}| - 1) * x_p, \quad p \in \mathcal{P} \quad (1.31)$$

After the execution of the ILP model, one of the selected sensor positions can be designated as the sink position since the resulting sensor positions form a connected network. As we are investigating the use of urban facilities in this work, i.e. energy constraints are alleviated, we will consider in the simulation part the sensor-to-sink delay as the main metric for the location of the sink position. One way to do that is to first determine shortest paths between all the pairs of the sensor positions using the delay metric. Then, an optimization model similar to the works presented in [15, 16] can be used in order to find an optimal sink location that minimizes the maximum delay in the network.

## 1.5.2 Simulation results

In this section, we present the simulations that we have performed in order to validate our model and analyze its performances. We first present the data set that we have used and the common simulation parameters. Then, we give a proof-of-concept showing the use of the ZSCAN algorithm and validating the ILP deployment model. Finally, we analyze the impact of some parameters on the coverage and connectivity results.

### Simulation dataset

We evaluate the ILP model on a pollution dataset of the Paris city while we focus on monitoring  $NO_2$  pollutant particles (Nitrogen dioxide). The pollution dataset was provided by AirParif, a French air quality monitoring association. We based on pollution data measured by 22 monitoring stations to estimate pollutant concentrations in the whole city. Because pollution achieved maximum values in Paris in March 2014 [37], we decided to base on pollution estimation for some periods of this month where pollution concentrations were high. Overall, we constructed 10 snapshots of pollution concentrations. We used the kriging interpolation method to estimate the pollution values with a map resolution of 100 meters. A set of 21201 spatial data records was then obtained for each snapshot. In addition to pollution data, we used lamppost locations as sensors potential positions  $\mathcal{P}$ . The lampposts dataset was provided by the open data service of the Paris city. We summarize in table 1.6 the common values of simulation parameters.

Parameter	value
Map discretization resolution (for set $\mathcal{S}$ )	100m
Number of time instants (set $\mathcal{T}$ )	10
$\Delta C$ (used in ZSCAN)	$5\mu\text{g}/\text{m}^3$
Sensors communication range (used for $\Gamma(p), p \in \mathcal{P}$ )	100m
Sensors cost ( $\text{cost}_p, p \in \mathcal{P}$ )	1 unity (constant)

Table 1.6: Default values of simulation parameters

### Proof of concept

First, we execute the ZSCAN algorithm to identify pollution zones where sensors will be deployed. Pollution peaks are detected and then expanded to form these zones. Points in each obtained pollution zone have as maximum concentration difference with the pollution peak of the zone  $5\mu\text{g}/\text{m}^3$ , this corresponds to the value of the parameter  $\Delta C$  as mentioned in table 1.6. A number of pollution zones is extracted from each time snapshot. The execution of the spatial clustering algorithm identified 29 pollution zones that occurred in March 2014, these zones are depicted in Figure 1.6. We notice that some zones occur in different snapshots with a little bit different shape. This is because these zones correspond usually to the same pollutant sources, the different shapes are due to the evolution of weather conditions.

In order to alleviate the execution of the ILP model, we propose to group pollution zones that share intersections into a region site where a mono sink sensor network will be deployed. In this simulation case, four sites were identified and are illustrated in Figure 1.6.

We now execute the **ILP 2** optimization model in order to find sensor optimal positions based on the generated pollution zones. The ILP model is executed on each site to locate sensors ensuring coverage and connectivity. The results are depicted in Figure 1.6. The latter shows that sensors are placed in intersections in order to minimize the financial cost. In addition to sensors placed to ensure the coverage of pollution zones, Figure 1.6 shows that some sensors are deployed to serve as relay nodes and hence ensure the network connectivity.

As mentioned in the previous section, we base on the works of [15, 16] in order to find the optimal positions of sinks while minimizing the maximum sensor-to-sink delay. We consider in our tests that the delay is linearly proportional to the hop count to the sink. Results are depicted in the same figure 1.6. We notice that sinks are placed in the center of each sub-network, which minimizes the maximum hop count and thus the maximum sensor-to-sink communication delay.

### **Evaluation of the coverage results**

In this section, we assess the coverage of the model ILP 2 while showing a comparison to the literature generic formulation presented in section 1.2 (i.e. coverage and connectivity are modeled independently, coverage is formulated by analogy to the Set Covering Problem using a basic detection model and connectivity formulation is based on the flow concept).

The coverage formulation of ILP 2 takes into account the nature of the phenomenon based on spatial analysis of pollution data. This is not the case of the generic formulation given in the literature, which assumes that a sensor is able to detect pollutants within a detection range, i.e. pollution is homogeneous within the detection range of a sensor. Even though this assumption is unrealistic since sensors can only detect pollutants that come into their contact, we compare in this simulation scenario the coverage cost given by our model and the generic formulation.

We depict in Fig. 1.7 the coverage deployment cost of sites 2 and 3 obtained using the generic formulation while considering different values of the detection range compared to our model. Fig. 1.7 shows that our model is at least 5 times better than the generic approach when the detection range is less than or equal to 500m. The coverage cost computed by the generic approach decreases when the detection range increases. However, our model remains better since pollution cannot be homogeneous, as assumed in the generic approach, especially within very large detection ranges.

### **Evaluation of the network connectivity**

We now analyze the deployment cost given by the model ILP 2 for sites 2 and 3 while considering different values of the communication range of nodes. We plot in Fig. 1.8 the obtained results. We recall that the optimum value of deployment cost when connectivity constraint is not taken into account is equal to 2 as shown in Fig. 1.8.

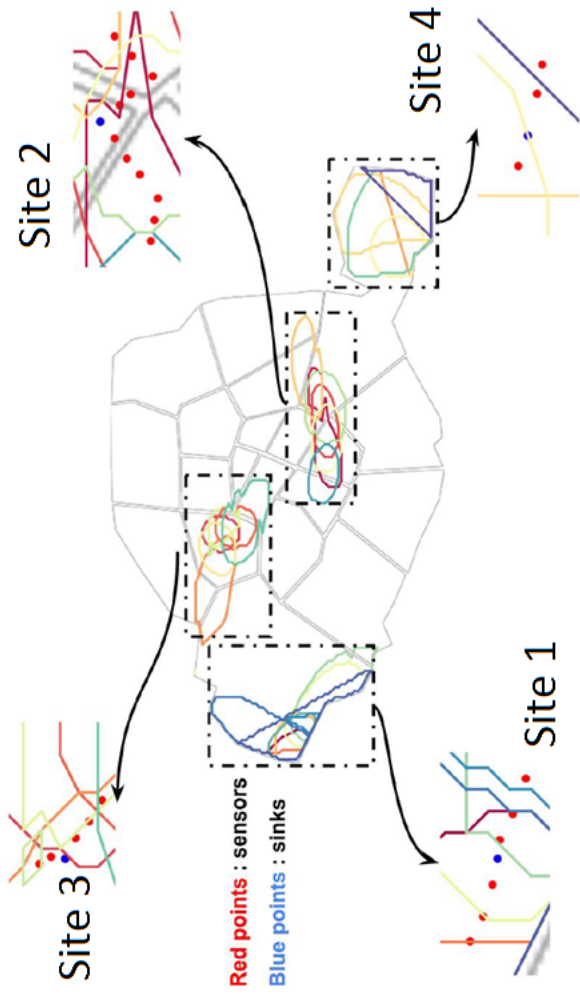


Figure 1.6: Proof-of-concept

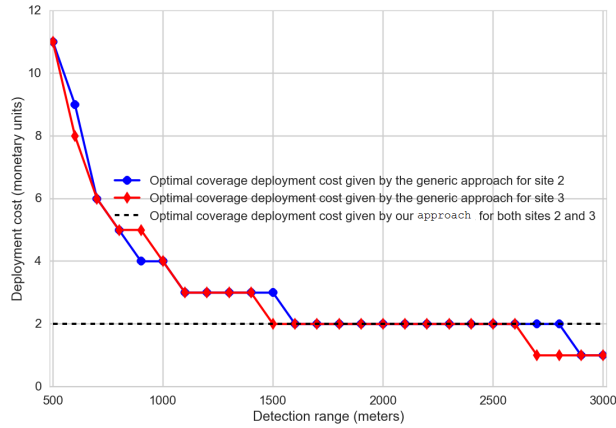


Figure 1.7: Optimal coverage deployment cost given by the generic formulation depending on the detection range of sensors.

We notice that the larger the communication range, the smaller the deployment cost is. This is expected since using a larger communication range allows to minimize the number of nodes used to connect sensors which are positioned within pollution zones. Fig. 1.8 also shows that when the communication range increases significantly, the overall deployment cost tends to the coverage deployment cost.

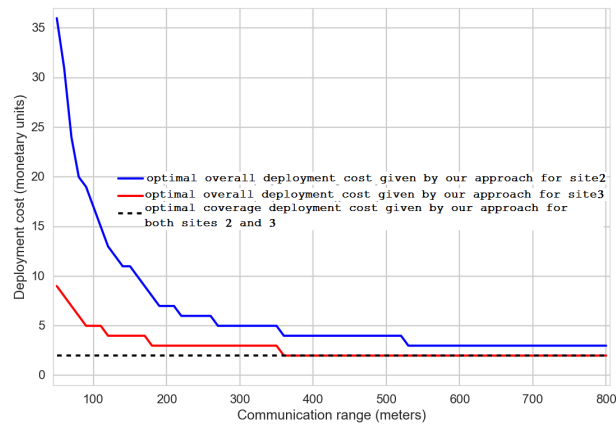


Figure 1.8: Optimal deployment cost given by our model depending on the communication range of nodes.

## 1.6 Model 3: WSN deployment for air pollution detection based on emission inventory

In this section, we present an ILP optimization model for the deployment of WSN for air pollution detection. Based on the pollution dispersion modeling applied on pollution emission inventory and the ILP related works, we first present pollution coverage and network connectivity independently. Then, we show how both coverage and connectivity can be formulated in a joint way using the flow concept. The ILP model takes into account the probabilistic sensing of pollution sensors and is designed to handle multiple scenarios of weather conditions. We apply the model on a data set of the London city in order to assess the impact of the model inputs on the deployment results.

### 1.6.1 Problem formulation

#### Overview

We base our coverage formulation on a pollution dispersion model. For the sake of clarity, we use the Gaussian dispersion model presented in section 1.3.1 in order to define the inputs of the ILP deployment model. However, these inputs can be also provided by other dispersion models, which may take into account the impact of buildings and urban structures. Pollution sources include industrial sources as well as traffic sources such as highways and crossroads. Table 1.7 depicts the main notations used in the integer programming model.

In the following, we consider a set of a predefined potential positions, denoted  $\mathcal{P}$ , which is obtained by discretizing the deployment field while considering only the allowed positions. In free space environments without deployment restrictions, that would be a regular grid. We denote  $\mathcal{N} = |\mathcal{P}|$  the number of potential positions. The locations of pollution sources, e.g. factories, sewage treatment plants, crossroads, highways..., is denoted  $\mathcal{S}$ .  $\mathcal{M}$  denotes the number of pollution sources. Let the binary decision variables  $x_p$ , resp.  $y_p$ , define if a sensor, resp. a sink, is placed at position  $p$ .

We consider that sinks are equipped with pollution sensors. They are also connected to a backbone network. Deploying a sink is therefore more expensive than a regular sensor node. The cost of deploying a sensor, resp. a sink, at position  $p$  is denoted  $c_p^{sensor}$ , resp.  $c_p^{sink}$ .

#### Objective function

The optimization model minimizes the sensors and sinks overall deployment cost. Thus, the objective function is defined as follows:

$$\mathcal{F} = \sum_{p \in \mathcal{P}} c_p^{sensor} \cdot x_p + \sum_{p \in \mathcal{P}} c_p^{sink} \cdot y_p \quad (1.32)$$

Since a sink embeds sensing capabilities, a sink and a sensor cannot be deployed at the same potential position  $p$  as formulated in constraint 1.33.

$$x_p + y_p \leq 1, \quad p \in \mathcal{P} \quad (1.33)$$



<b>Sets</b>	
$\mathcal{P}$	Set of potential positions of sensors and sinks
$\mathcal{N}$	Number of sensors and sinks potential positions
$\mathcal{I}$	Set of pollution sources
$\mathcal{M}$	Number of pollution sources
$\mathcal{S}$	Set of weather scenarios
<b>Parameters</b>	
$\mathcal{Z}_i^s$	The pollution zone formed by source $i$ under scenario $s$
$\mathcal{B}_{ip}^s$	Define whether the position $p$ belongs to the zone $\mathcal{Z}_i^s$ or not
$\Gamma(p)$	The neighborhood of the potential position $p$
$c_p^{sensor}$	The cost of deploying a sensor at position $p$
$c_p^{sink}$	The cost of deploying a sink at position $p$
$\beta$	Minimum coverage probability to ensure for each zone
$\mathcal{W}_{ip}^s$	The probability of detecting the zone $\mathcal{Z}_i^s$ at position $p$
$\delta$	Percentage of scenarios that have to be taken into account
$\alpha_s$	Probability that scenario $s$ is realized
$C_0$	Pollutant concentration threshold
<b>Variables</b>	
$x_p$	Define whether a sensor is deployed at position $p$ or not $x_p \in \{0, 1\}, p \in \mathcal{P}$
$y_p$	Define whether a sink is deployed at position $p$ or not $y_p \in \{0, 1\}, p \in \mathcal{P}$
$t_i^s$	Define whether the zone $\mathcal{Z}_i^s$ is covered or not $t_i^s \in \{0, 1\}, i \in \mathcal{I}, s \in \mathcal{S}$
$g_{pq}$	Flow quantity transmitted from node $p$ to node $q$ $g_{pq} \in \{0, 1, \dots\}, p \in \mathcal{P}, q \in \Gamma(p)$
$f_{ip}^s$	Flow quantity transmitted from zone $\mathcal{Z}_i^s$ to node $p$ $f_{ip}^s \in \{0, 1\}, i \in \mathcal{I}, s \in \mathcal{S}, p \in \mathcal{Z}_i^s$

Table 1.7: Main notations used in the deployment model.

### Coverage

The coverage constraints rely on the modeling of the atmospheric dispersion. We assume that pollution sources release pollutants independently and may have simultaneous release. Our formulation ensures the coverage of threshold crossings in all cases. As explained in section 1.3.1, pollution concentrations vary depending on weather conditions. Hence, we consider a set of possible weather scenarios  $\mathcal{S}$  that can be obtained based on statistical data or weather forecast. A scenario corresponds to a tuple of ambient temperature  $T$ , wind velocity  $V_w$  and wind direction  $D_w$ :  $s = (T^s, V_w^s, D_w^s)$ . Each scenario  $s$  has probability  $\alpha_s$  to happen. We assume that  $\mathcal{S}$  is a partition of the space of weather conditions, i.e.  $\sum_{s \in \mathcal{S}} \alpha_s = 1$  and  $s1 \cap s2 = \emptyset, \forall s1, s2 \in \mathcal{S}$ . Here  $s1 \cap s2 = \emptyset$  if the weather scenario  $s1$  occurs independently from  $s2$ .

Using an atmospheric dispersion model, we determine the set of generated pollution zones. Each zone  $\mathcal{Z}_i^s$  corresponds to the geographical area, i.e. set of positions, where the pollution threshold is crossed when the pollution source  $i$  is releasing pollutants under the weather scenario  $s$ .

Let the binary parameter  $\mathcal{B}_{ip}^s$  denote whether a position  $p$  belongs to  $\mathcal{Z}_i^s$  or not. A pollution zone  $\mathcal{Z}_i^s$  is therefore the set  $\{p \in \mathcal{P} \text{ where } \mathcal{B}_{ip}^s = 1\}$ . When using the pointwise Gaussian dispersion model, the value of  $\mathcal{B}_{ip}^s$  is calculated using Formula (1.34) where  $\sigma_y$ ,  $\sigma_z$ ,  $Q$  and  $H$  are the parameters presented in Section 1.3,  $p = (x, y, z)$  and  $C_0$  is the threshold of pollutant concentration above which a point is considered as polluted.

$$\mathcal{B}_{ip}^s = \begin{cases} 1 & \text{if } \frac{Q}{2\pi V_w^s \sigma_y \sigma_z} e^{-\frac{y^2}{2\sigma_y^2}} (e^{-\frac{(z-H)^2}{2\sigma_y^2}} + e^{-\frac{(z+H)^2}{2\sigma_y^2}}) \geq C_0 \\ 0 & \text{otherwise} \end{cases} \quad (1.34)$$

A sensor exposed to a given pollutant will detect its concentration with a probability depending on the sensing accuracy. We denote  $\mathcal{W}_{ip}^s \in ]0, 1[$  the probability of detecting the pollution source  $i$  under the weather scenario  $s$  at position  $p$ ,  $p \in \mathcal{Z}_i^s$ . The  $\mathcal{W}_{ip}^s$  parameters are mainly due to the technical characteristics of pollution sensors and are not related to the dispersion model.

Once the pollution zones  $\mathcal{Z}_i^s$  are identified and the probability parameters  $\mathcal{W}_{ip}^s$  are computed, we formulate the coverage of each pollution source  $i$  under each weather scenario  $s$  with a probability  $\beta$  in constraint 1.35.

$$\prod_{p \in \mathcal{Z}_i^s} (1 - \mathcal{W}_{ip}^s \cdot (x_p + y_p)) \leq (1 - \beta), \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (1.35)$$

When a sensor or a sink is placed at position  $p$ , i.e.  $x_p + y_p = 1$ ,  $1 - \mathcal{W}_{ip}^s \cdot (x_p + y_p)$  is then equal to  $1 - \mathcal{W}_{ip}^s$ , the probability that the node deployed at  $p$  do not cover the pollution zone  $\mathcal{Z}_i^s$  at position  $p$ . Assuming that the detection events are independent among all potential positions, constraint 1.35 ensures therefore that each zone  $\mathcal{Z}_i^s$  is covered with a probability  $\beta \in ]0, 1[$ .

Constraint 1.35 ensures that each pollution source is covered with a probability  $\beta$  under each scenario  $s$ . One could want to relax this constraint and ask only for the coverage of each pollution source under  $\delta$  percent of weather scenarios, with a  $\beta$  probability for each scenario. For that, we introduce the binary variable  $t_i^s$  that define whether source  $i$  is covered during weather scenario  $s$ . Therefore  $t_i^s = 1$  if a sufficient number of sensors is placed in the pollution zone  $\mathcal{Z}_i^s$ . The percentage of weather conditions where  $i$  can be detected is the sum of the probabilities that a scenario in which  $i$  is detected happens. As a result, the coverage formulation of the partial coverage case is given by the constraints 1.36 and 1.37.

$$\prod_{p \in \mathcal{Z}_i^s} (1 - \mathcal{W}_{ip}^s \cdot (x_p + y_p)) \leq (1 - \beta \cdot t_i^s), \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (1.36)$$

$$\sum_{s \in \mathcal{S}} t_i^s \cdot \alpha_s \geq \delta, \quad i \in \mathcal{I} \quad (1.37)$$

Constraint 1.36 should be linearized in order to get an ILP formulation. We first apply the log function.

$$\log \prod_{p \in \mathcal{Z}_i^s} (1 - \mathcal{W}_{ip}^s \cdot (x_p + y_p)) \leq \log(1 - \beta \cdot t_i^s), \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (1.38)$$

$$\sum_{p \in \mathcal{Z}_i^s} \log(1 - \mathcal{W}_{ip}^s \cdot (x_p + y_p)) \leq \log(1 - \beta \cdot t_i^s), \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (1.39)$$

Since  $x_p + y_p$  and  $t_i^s$  are binary, the log can be rewritten as follows and constraint 1.40 is linear.

$$\sum_{p \in \mathcal{Z}_i^s} (x_p + y_p) \cdot \log(1 - \mathcal{W}_{ip}^s) \leq t_i^s \cdot \log(1 - \beta), \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (1.40)$$

### Connectivity

As in [38, 12] and [39], we formulate in this first model the connectivity constraint as a network flow problem. In contrast to these works, we consider the same potential positions set  $\mathcal{P}$  for sensors and sinks and we do not assume that potential positions of sinks are known or different from those of sensors. We first denote by  $\Gamma(p)$ ,  $p \in \mathcal{P}$  the set of neighbors of a node deployed at the potential position  $p$ . This set can be computed using any adequate propagation models. Then, we define the decision variables  $g_{pq}$  as the flow quantity transmitted from a node located at potential position  $p$  to another node located at potential position  $q$ . We suppose that each sensor of the resulting WSN generates a flow unit in the network, and verify if these units can be recovered by sinks. The following constraints ensure that deployed sensors and sinks form a connected wireless sensor network; i.e. each sensor can communicate with at least one sink.

$$\sum_{q \in \Gamma(p)} g_{pq} - \sum_{q \in \Gamma(p)} g_{qp} \geq x_p - \mathcal{N} \cdot y_p, \quad p \in \mathcal{P} \quad (1.41)$$

$$\sum_{q \in \Gamma(p)} g_{pq} - \sum_{q \in \Gamma(p)} g_{qp} \leq x_p, \quad p \in \mathcal{P} \quad (1.42)$$

$$\sum_{q \in \Gamma(p)} g_{pq} \leq \mathcal{N} \cdot x_p, \quad p \in \mathcal{P} \quad (1.43)$$

$$\sum_{p \in \mathcal{P}} \sum_{q \in \Gamma(p)} g_{pq} = \sum_{p \in \mathcal{P}} \sum_{q \in \Gamma(p)} g_{qp} \quad (1.44)$$

Constraints 1.41 and 1.42 are designed to ensure that each deployed sensor, i.e. such that  $x_p = 1$ , generates a flow unit in the network. These constraints are equivalent to the following.

$$\sum_{q \in \Gamma(p)} g_{pq} - \sum_{q \in \Gamma(p)} g_{qp} \begin{cases} = 1 & \text{if } x_p = 1, y_p = 0 \\ = 0 & \text{if } x_p = y_p = 0 \\ \leq 0, \geq -\mathcal{N} & \text{if } x_p = 0, y_p = 1 \end{cases}$$

The first case corresponds to deployed sensors that should generate, each one of them, a flow unit. The second case, combined with constraint 1.43, ensures that absent nodes, i.e.  $x_p = y_p = 0$ , do not participate in the communication. The third case concerns deployed sinks, and ensures that each sink cannot receive more than  $\mathcal{N}$  units. The case  $x_p = y_p = 1$  is not possible because of constraint 1.33. Constraint 1.43 ensures also that deployed sinks cannot transmit flow units, and only act as receivers. Constraint 1.44 means that the overall flow is conservative. The flow sent by deployed sensors has to be received by deployed sinks.

### ILP Model

Now, we propose to combine coverage and connectivity constraints defined in the two previous sections using only the flow concept. By considering pollution sources as a part of the network, we obtain a homogeneous coverage/connectivity formulation as a network flow problem. Each pollution source  $i$  should transmit some flow units to potential nodes  $p$  which are located within the pollution zone corresponding to each weather scenario  $s$  i.e.  $p \in \mathcal{Z}_i^s$ . In addition, sensors are flow conservative and the sinks receive the flow units generated by pollution sources. Therefore, the definition of the joint coverage/connectivity is to ensure that sinks will be informed each time that a threshold crossing occurs. In this regard, a sensor has to receive at most one unit from a given pollution zone. We hence define the binary decision variable  $f_{ip}^s$  as the flow quantity from the pollution source  $i$  to the potential node  $p$  in the case of weather scenario  $s$ . The following constraints ensure coverage and connectivity for air pollution detection in a joint way.

$$\sum_{s \in \mathcal{S}} t_i^s \cdot \alpha_s \geq \delta, \quad i \in \mathcal{I} \quad (1.45)$$

$$\sum_{p \in \mathcal{Z}_i^s} f_{ip}^s \cdot \log(1 - \mathcal{W}_{ip}^s) \leq t_i^s \cdot \log(1 - \beta), \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (1.46)$$

$$\sum_{i \in \mathcal{I}, s \in \mathcal{S}: p \in \mathcal{Z}_i^s} f_{ip}^s + \sum_{q \in \Gamma(p)} g_{qp} - g_{pq} \leq \mathcal{N} \cdot \mathcal{M} | \mathcal{S} | y_p, \quad p \in \mathcal{P} \quad (1.47)$$

$$\sum_{i \in \mathcal{I}, s \in \mathcal{S}: p \in \mathcal{Z}_i^s} f_{ip}^s + \sum_{q \in \Gamma(p)} g_{qp} - g_{pq} \geq 0, \quad p \in \mathcal{P} \quad (1.48)$$

$$\sum_{q \in \Gamma(p)} g_{pq} \leq \mathcal{N} \cdot \mathcal{M} | \mathcal{S} | x_p, \quad p \in \mathcal{P} \quad (1.49)$$

$$f_{ip}^s \leq x_p + y_p, \quad p \in \mathcal{P}, i \in \mathcal{I}, s \in \mathcal{S} \quad (1.50)$$

Coverage is formulated in constraints 1.45 and 1.46. Constraint 1.45 ensures coverage of each pollution source under a  $\delta$  percentage of weather scenarios. Constraint 1.46 is derived from constraint 1.40 and ensures that each pollution source  $i$  generates a sufficient number of flow units in the network. Constraint 1.50 enforces that all the flow units are received by deployed nodes. Thanks to constraints 1.47 and 1.48, when a sensor is deployed on point  $p$  (case  $y_p = 0$  and  $x_p = 1$ ), we ensure that the inflow of sensor  $p$  is equal to its outflow; i.e. The flow is conservative on deployed sensors. In

addition, constraints 1.47 and 1.48 also ensure that the sinks are allowed to gather all the flow units that are generated in the network (case  $y_p = 1$  and  $x_p=0$ ). Constraints 1.49 and 1.50 combined with constraints 1.47 and 1.48 ensure that absent nodes do not participate in the communication. As a results, the deployed sensors have to send the flow units gathered from pollution sources to the sinks in order to get the connectivity constraints verified.

Finally, the ILP optimization model can be written as follows:

$$\begin{aligned}
 & [ILP\ 3] \\
 & \text{Minimize} \quad (1.32) \\
 & \text{Subject to.} \quad (1.33), (1.45), (1.46), (1.47), (1.48), \\
 & \quad \quad \quad (1.49) \text{ and } (1.50)
 \end{aligned}$$

## 1.6.2 Simulation results

In this section, we present the simulations that we have performed to evaluate the ILP 3 deployment model. We first present the data set of Greater London that we used in simulation. Then, as a proof of concept we apply the model to the London Borough of Camden. Finally, we study the impact of the sink/sensor cost ratio, nodes height, pollution sources density, the probabilistic sensing of sensors and weather conditions. This study allows us to derive engineering insights for effective deployments of air pollution sensors in an urban environment.

### Simulation dataset

We evaluate the deployment model on a data set provided by the Greater London community [40]. London is one of the most polluted cities in Europe [41]. The data set corresponds to the locations of urban pollution sources. In this data set, mostly urban facilities have the potential to affect the air quality such as petrol stations, waste oil burners, cement works, etc. Fig. 1.9 depicts the set of pollution sources, spread over the 32 boroughs of Greater London. Overall, 1090 pollution sources are considered. Pollution sources distribution per borough depends on the surface of the borough and ranges from 6 sources to 161 sources.

In addition to pollution sources locations, we compute the weather scenarios leveraging statistical data gathered by a weather station of London. The location of this station is depicted in Fig. 1.9. We consider weather conditions of each month of the year averaged over the last past 10 years [42]. The set of weather conditions is depicted in table 1.8, each scenario corresponds to values of ambient temperature, wind direction and wind velocity. As proof of concept and without loss of generality, we assume that weather scenarios provided by the considered weather station are homogeneous in all the area of Greater London.

ILP formulations are implemented using the IBM ILOG CPLEX Optimization Studio and executed on a PC with Intel Xeon E5649 processor under Linux. The ILP solver

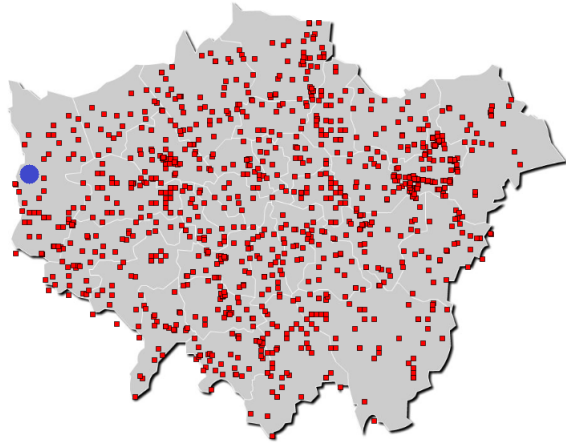


Figure 1.9: Pollution sources (squares) and the considered weather station (disk).

Month of year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Ambient temperature ( $^{\circ}C$ )	7	7	9	13	15	19	21	20	18	14	9	7
Wind velocity ( $m/s$ )	5	5	5	5	5	5	5	5	5	5	5	5
Wind direction (degree)	225	247	270	270	270	225	225	225	225	202	225	247

Table 1.8: Weather statistics of London.

is executed with a time limit of 30 minutes. The default values of simulation parameters are summarized in Table 1.9. We generate the pollution inputs of our deployment model using the Gaussian model presented in section 1.3 while considering the pollutant characteristics depicted in table 1.10. We define the nodes neighboring  $\Gamma$  based on a given transmission range. Moreover, we assume that the cost of nodes is independent of the position of the node, i.e.  $c_p^{sensor} = c^{sensor}$  and  $c_p^{sink} = c^{sink}$ . Furthermore, we investigate the coverage of pollution sources with respect to all the considered scenarios, i.e.  $\delta = 1.0$ . In addition, we consider that the probabilistic sensing value  $\mathcal{W}_{ip}^s$  is constant and equal to 90%.

### Proof of concept

As a proof of concept, we first execute the deployment model on the London Borough of Camden. We use streetlights as potential positions of sensors in order to alleviate the energy constraints. The streetlights data set was provided by the Camden DataStore [43]. Camden is spread over an area of around  $8km \times 6km$  and contains 19 pollution sources. Fig. 1.10 depicts the pollution zones obtained by running of the Gaussian dispersion model while taking into account weather conditions of the month of January. Fig. 1.10 also shows the obtained positions of wireless sensor network nodes computed by the deployment model. We notice that sensors are placed at the intersections of the different pollution zones in order to minimize the coverage deployment cost. Moreover,

Parameter	Value
Nodes transmission range	100m
Nodes height	10m
Sensors cost ( $c_p^{sensor}$ )	1 monetary unit
Sinks cost ( $c_p^{sink}$ )	10 monetary units
Coverage requirements of pollution zones ( $\beta$ )	0.98
Coverage requirements of weather conditions ( $\delta$ )	1.0
Detection sensitivity of sensors ( $\mathcal{H}_{ip}^s$ )	0.9
Ambient Temperature ( $T$ )	7°C
Wind velocity ( $V_w$ )	5m/s
Wind direction ( $D_w$ )	225°
Pollution threshold ( $C_0$ )	20 $\mu$ g/m <sup>3</sup>

Table 1.9: Summary of default simulation parameters.

Parameter	value
$h_s$	25m
$Q$	5g/s
$V$	1.9mm <sup>3</sup> /s
$T_s$	30°C
$a_y$	1.36
$b_y$	0.82
$a_z$	0.275
$b_z$	0.69

Table 1.10: Common values used for simulation of the Gaussian model

the resulting network consists of 7 sub-networks, a sink is deployed in each one and some sensors are added to ensure connectivity.

The following results have been obtained by running the deployment model on a hundred of 1200m  $\times$  1200m blocks extracted from the Greater London map. The density of pollution sources varies between 3 and 18 sources per block. We discretize each block with a resolution of 100m to get a 2D grid of points that we consider as potential positions of WSN nodes.

### Evaluation of the network connectivity

**Evaluation of the number of nodes** In this simulation case, we analyze the number of sinks and sensors in the resulting networks while varying the ratio between sink cost and sensor cost. We plot in Fig. 1.11 the impact of the cost ratio on the optimal number of sensors and sinks. The cost ratio ranges from 1 to 12 and the results are averaged over all the London blocks defined in the previous section. On one hand, we notice that sensors are less used when their cost is close to the sinks cost. For instance, only

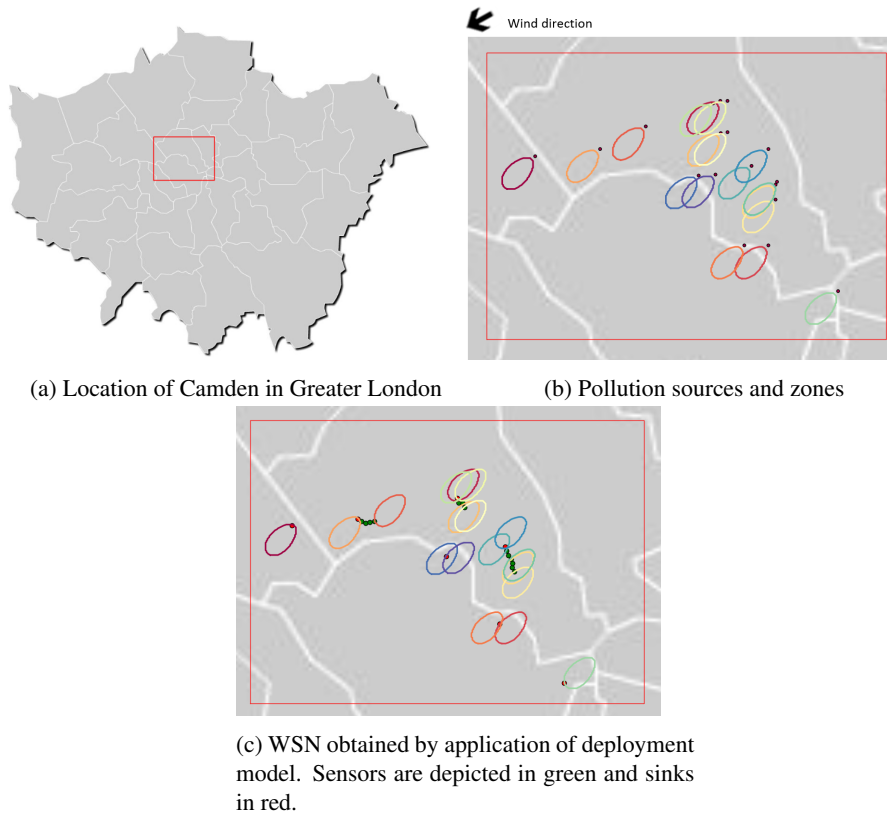


Figure 1.10: Application of our deployment model to the London Borough of Camden

sinks are used when the cost ratio is equal to 1. On the other hand, when the cost ratio increases, more sensors are used and the number of required sinks tends to one. As a result, the network is usually formed by only one sink when the cost ratio is greater than 10. This is explained by the fact that adding some relay sensor nodes to ensure connectivity has a less cost than using a lot of sinks that are equipped with pollution sensors.

In the following simulations, we execute the deployment model with a default value of sink/sensor cost ratio equal to 10 as shown in table 1.9. Thus, we use formulations corresponding to the mono-sink case.

**Evaluation of the number of hops to sink nodes** In this simulation case, we evaluate the obtained networks in terms of the number of hops to sink nodes, which is a measure of the network lifetime and communication delay [44]. As formulated in our connectivity constraints, the positioning of sink nodes does not take into account the length of sensor-to-sink paths. However, sinks can be relocated on the obtained network when the network is monosink, which is the case as shown in the previous



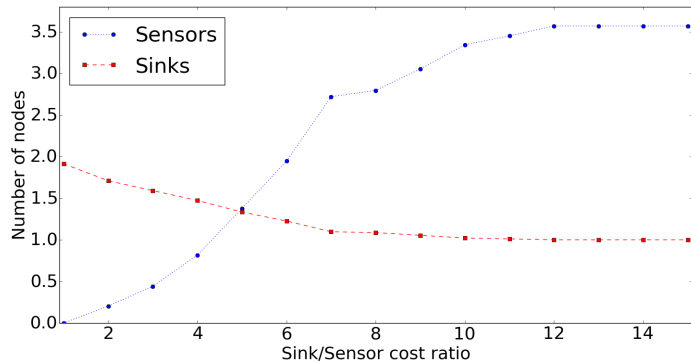


Figure 1.11: Average number of sinks and sensors depending on their cost ratio.

subsection. The sink node can be relocated in such a way that the maximum distance to the sink in terms of hops is minimized. This distance is called the radius of the network and describes how much the network is well connected when considering the best position of the sink node. If the sink position is given randomly, the maximum distance to the sink is bounded by the diameter of the network, which is the distance to the sink node when choosing the worst position of the sink.

We depict in Fig. 1.12 the cumulative distribution functions of the network radius and diameter. Results show that the network radius is as at most equal to 5 for more than 96% of the instances. This means that the number of hops to the sink node after relocation is at most equal to 5 in almost all instances. Moreover, we notice that the network radius is nearly equal to the diameter of the network for the rest of the instances, which means that the sink relocation does not improve much the network connectivity in this case.

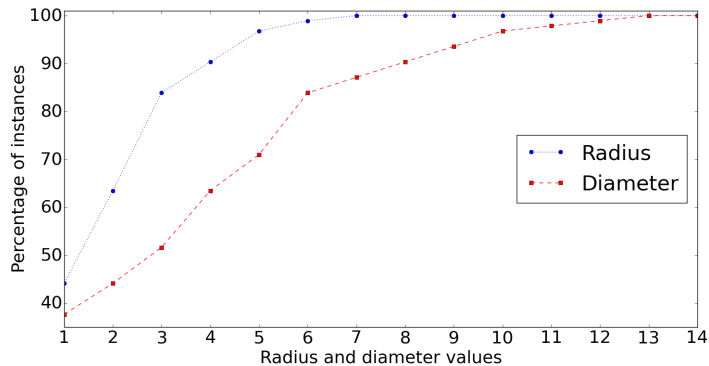


Figure 1.12: Cumulative distribution functions of the network radius and diameter.

## Evaluation of the coverage results

**Impact of nodes height** We now study the impact of the height at which are placed sensors and sinks on the deployment cost. We assume that the height of pollution sources is equal to  $25m$ , and all the sensors and sinks are deployed at the same height, which is considered in the range from 5 to 40 meters. We plot in Fig. 1.13 the sensors and sinks overall deployment cost depending on their height when applying two different weather scenarios, those corresponding to January and December. The results are averaged over all the London blocks. On one hand, we notice that the deployment cost is minimal when the nodes height is close to the effective release height of pollution sources  $H$ , which is nearly equal to  $25.1$  in our case. This is explained by the fact that pollution concentration gets the highest values when being near to the pollutant effective release height  $H$ . On the other hand, pollutants are more likely to drop than to increase, which is due to gravitation. Indeed, the deployment cost at  $40m$  is much greater than the deployment cost at  $5m$ . Fig. 1.13 also shows that when using different weather scenarios, the deployment cost is not the same. Indeed, weather conditions impacts the disposition of pollution zones allowing for more or less intersections. As a result, the obtained WSN topology depends on the weather conditions taken into account.

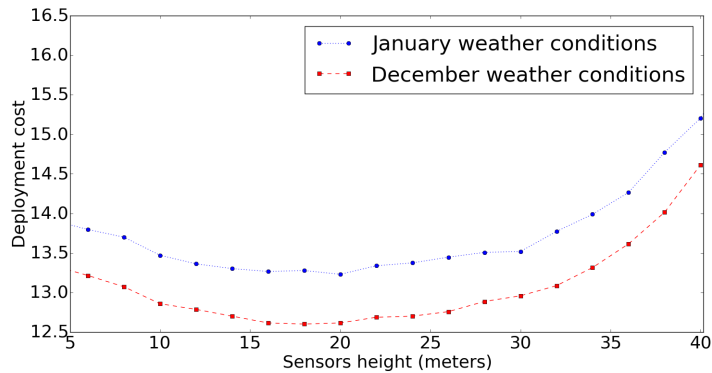


Figure 1.13: Deployment cost average depending on nodes height with different weather conditions.

**Impact of pollution sources density** In this scenario, we study the impact of pollution sources density on the deployment cost. For this purpose, we take the results of the previous scenario corresponding to January weather conditions and averaged with respect to the number of pollution sources of each instance, i.e. the number of pollution sources within each block instance. We plot in Fig. 1.14 the deployment cost variations depending on the nodes height while considering three different densities: 4, 5 and 6 pollution sources per instance. Fig. 1.14 shows that more there are pollution sources in the environment, more there are sensors required and thus higher is the deployment cost. This can be explained by the number of pollution zones that increases with the number of pollution sources, and thus requires much sensors to ensure the coverage

requirements. In addition, the increasing in the deployment cost from 5 sources density to 6 sources density is less than the increasing from 4 sources density to 5 sources density. This is because when the number of pollution sources increases, more intersections between pollution zones appear and affect the increasing of the deployment cost.

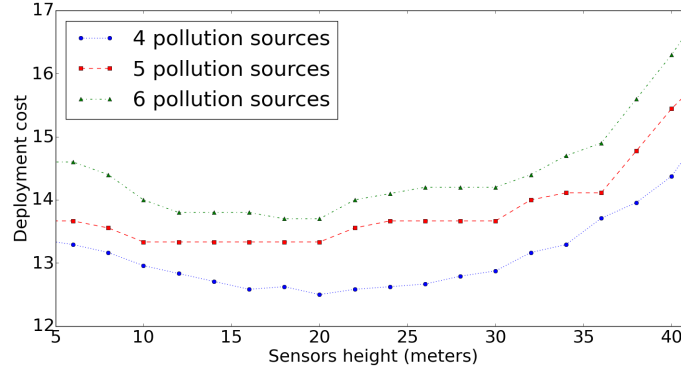


Figure 1.14: Deployment cost average depending on nodes height and pollution sources density.

**Impact of probabilistic sensing** The probabilistic sensing of pollution sensors is one of the most important factors that affect the topology of sensor networks used for pollution monitoring. Fig. 1.15 depicts the average cost of the resulting deployments of the block instances while considering two values of the detection sensitivity of sensors:  $\mathcal{W}_{ip}^s = 0.9$  and  $\mathcal{W}_{ip}^s = 0.8$ . As expected, using sensors with better detection sensitivity yields less deployment cost. We notice that the ratio between the two curves is around 1.1. This is explained by the intersection existence between the different polluted zones, which means that in some cases a sensor can monitor more than one pollution source.

**Impact of the number of weather conditions** In this simulation case, we study the impact of using a small number of weather scenarios on the deployment results. It is clear that when considering all the possible weather scenarios, the resulting WSN ensures better pollution monitoring. However, when there is a huge number of weather scenarios, considering a less number of these scenarios alleviates the deployment model allowing their application on large-scale instances.

We recall that  $\mathcal{S}$  is the set of the monthly weather scenarios presented in table 1.8. Given a subset  $\mathcal{S}'$  of  $\mathcal{S}$ , we define the missed pollution zones percentage as the percentage of pollution zones that cannot be covered by the WSN resulting from executing the model under only weather scenarios  $\mathcal{S}'$ . Fig. 1.16 illustrates the variations of the missed pollution zones percentage depending on the number of weather scenarios taken into account starting from January weather scenario in the first curve and starting from December weather scenario is the second one. Fig. 1.16 shows that the percentage of

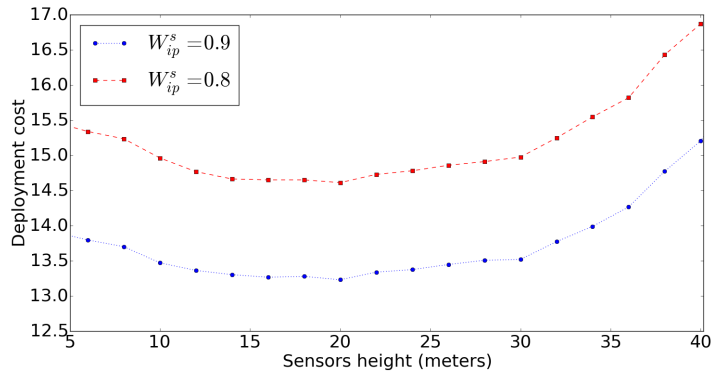


Figure 1.15: Deployment cost average depending on nodes height and probabilistic sensing values.

missed pollution zones usually decreases when considering more weather scenarios, this is expected since the number of pollution zones depends on the weather scenarios set. However, in some cases, the missed pollution zones percentage remains the same when considering additional scenarios of weather conditions. Indeed, additional weather scenarios involve new pollution zones that may be, in some cases, already included in the set of pollution zones formed without taking into account the additional scenarios. This may happen for instance when additional weather scenarios are slightly different from those already taken into account.

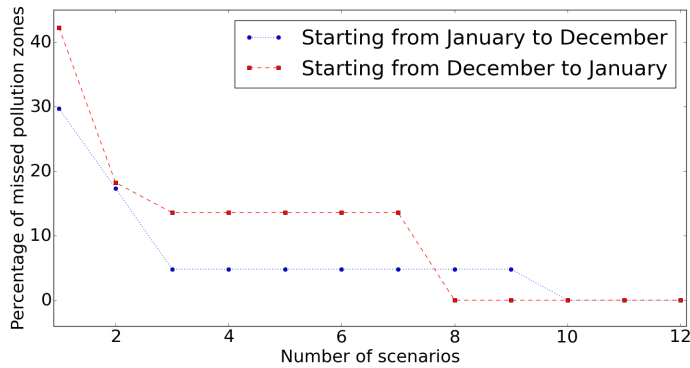


Figure 1.16: Average percentage of missed pollution zones depending on weather scenarios.

In addition to the impact of the number of weather scenarios, their similarity has also to be taken into account. As shown in Fig. 1.16, considering only weather scenarios from December to May meaning only 8 scenarios allows to cover the whole set of pollution zones in contrary to the scenarios set from January to October that requires 10 scenarios.

## 1.7 Discussion and future directions

In this chapter, we focused on what the literature lacks in WSN deployment for air pollution monitoring, which is an appropriate coverage formulation of both regular mapping and high concentrations detection. We have also shown how connectivity and coverage can be merged into one constraint in order to make the deployment models less complex. More details about the three presented models can be found respectively in [45], [46], [47] and [48].

Since the proposed models are integer linear programs, their complexity mainly depends on the number of binary variables which represent the potential positions of sensor nodes. Therefore, the execution time of solvers increases exponentially with the area of the deployment region (if we keep the same spatial resolution of maps). As for the number of pollution snapshots which characterize weather scenarios, it mainly impacts the number of constraints. Hence, the execution time increases linearly with the number of timeframes.

We believe that our models can be extended in order to take into account the network lifetime and mobile deployment. We provide hereafter some details about how the extension of our models can be achieved.

**Optimization of the network lifetime** One possible solution is to base on the work of [12] where authors define the energy constraints based on the flow concept. This solution is viable since our models are also based on the flow concept. The idea is to split the lifetime of the network into a sequence of timeframes (a sequence of minutes for instance) and enforce each sensor node to send a flow unit in each timeframe. Then, a new constraint should be added to the models in order to ensure that for each sensor, the sum of the energy that is consumed in the set of timeframes is less than the energy of the battery of the sensor. The lifetime objective function corresponds to the maximum number of timeframes where the network is operating.

**Mobile deployment** Indeed, our models are designed for static networks, which is argued by the fact that pollution sensors operate well when they are static [49]. However, sinks can be considered as mobile nodes and this can be integrated in our models based on existing mobile-sinks formulations such as the work of [12]. The idea is to consider a set of timeframes as in the extension of network lifetime. The mobile sink changes its positions in each timeframe. Then, the flow constraints should be formulated in order to ensure that the flow is conservative in the network in each timeframe. This means that in each timeframe, each sensor generates a flow unit and the sink node receives all the generated units.

## 1.8 Conclusion

Air pollution is becoming a major problem of smart cities due to the increasing industrialization and the massive urbanization. In this chapter, we focused on the use of wireless sensor networks for the two main applications of air pollution monitoring:

regular mapping and the detection of threshold crossings. We addressed the deployment issue and presented three optimization models ensuring pollution coverage and network connectivity with the minimum cost. Unlike the inadequate related works, which are either generic or rely on a detection model, we based on the nature of the phenomenon in order to provide realistic models.

We evaluated the impact of the parameters of the different models on the deployment results while using real data sets. Among our conclusions, the fact that sensors should be placed at a height close to the one of pollution sources and the fact that the size of the resulting network depends on the degree of the variations of pollution concentrations within the deployment region. We also concluded that the desired monitoring precision impacts well the density of sensors and hence the connectivity of the network.

As future work, we are trying to consider the impact of the nature of pollutants and the urban topography on the coverage results. We are also working on the validation of our deployment models.

**Acknowledgement** This work has been supported by the "LABEX IMU" (ANR-10-LABX-0088) and the "Programme Avenir Lyon Saint-Etienne" of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).



# Bibliography

- [1] World Health Organization, “Burden of disease from household air pollution for 2012,” on [http://www.who.int/phe/health\\_topics/outdoorair/databases/FINAL\\_HAP\\_AAP\\_BoD\\_24March2014.pdf](http://www.who.int/phe/health_topics/outdoorair/databases/FINAL_HAP_AAP_BoD_24March2014.pdf) [2016-02-27].
- [2] AirParif, “The air quality monitoring organization of the paris agglomeration,” <http://www.airparif.fr/en/index/index>.
- [3] A. Kumar, H. Kim, and G. P. Hancke, “Environmental monitoring systems: a review,” *Sensors Journal, IEEE*, vol. 13, no. 4, pp. 1329–1339, 2013.
- [4] M. Mead, O. Popoola, G. Stewart, P. Landshoff, M. Calleja, M. Hayes, J. Baldovi, M. McLeod, T. Hodgson, J. Dicks *et al.*, “The use of electrochemical sensors for monitoring urban air quality in low-cost, high-density networks,” *Atmospheric Environment*, vol. 70, pp. 186–203, 2013.
- [5] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [6] ETSI, “Electromagnetic compatibility and radio spectrum matters (erm); system reference document (srdoc): Spectrum requirements for short range device, metropolitan mesh machine networks (m3n) and smart metering (sm) applications,” available on [http://www.etsi.org/deliver/etsi\\_tr/103000\\_103099/103055/01.01.01\\_60/tr\\_103055v010101p.pdf](http://www.etsi.org/deliver/etsi_tr/103000_103099/103055/01.01.01_60/tr_103055v010101p.pdf), last access: 2016-01-20.
- [7] B. Liu, O. Dousse, P. Nain, and D. Towsley, “Dynamic coverage of mobile sensor networks,” *Parallel and Distributed Systems, IEEE Transactions on*, vol. 24, no. 2, pp. 301–311, 2013.
- [8] C. Zhu, C. Zheng, L. Shu, and G. Han, “A survey on coverage and connectivity issues in wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 35, no. 2, pp. 619–632, 2012.
- [9] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, “Grid coverage for surveillance and target location in distributed sensor networks,” *Computers, IEEE Transactions on*, vol. 51, no. 12, pp. 1448–1453, 2002.
- [10] İ. K. Altinel, N. Aras, E. Güney, and C. Ersoy, “Binary integer programming formulation and heuristics for differentiated coverage in heterogeneous sensor networks,” *Computer Networks*, vol. 52, no. 12, pp. 2419–2431, 2008.



- [11] X. Chang, R. Tan, G. Xing, Z. Yuan, C. Lu, Y. Chen, and Y. Yang, "Sensor placement algorithms for fusion-based surveillance networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 8, pp. 1407–1414, 2011.
- [12] M. E. Keskin, İ. K. Altinel, N. Aras, and C. Ersoy, "Wireless sensor network lifetime maximization by optimal sensor deployment, activity scheduling, data routing and sink mobility," *Ad Hoc Networks*, vol. 17, pp. 18–36, 2014.
- [13] M. Rebai, H. Snoussi, F. Hnaïen, L. Khoukhi *et al.*, "Sensor deployment optimization methods to achieve both coverage and connectivity in wireless sensor networks," *Computers & Operations Research*, vol. 59, pp. 11–21, 2015.
- [14] S. Sengupta, S. Das, M. Nasir, and B. K. Panigrahi, "Multi-objective node deployment in wsns: In search of an optimal trade-off among coverage, lifetime, energy consumption, and connectivity," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 1, pp. 405–416, 2013.
- [15] E. Güney, İ. K. Altinel, N. Aras, and C. Ersoy, "Efficient integer programming formulations for optimum sink location and routing in wireless sensor networks," in *ISCIS'08. 23rd International Symposium on*. IEEE, 2008, pp. 1–6.
- [16] E. Güney, N. Aras, İ. K. Altinel, and C. Ersoy, "Efficient integer programming formulations for optimum sink location and routing in heterogeneous wireless sensor networks," *Computer Networks*, vol. 54, no. 11, pp. 1805–1822, 2010.
- [17] V. Roy, A. Simonetto, and G. Leus, "Spatio-temporal sensor management for environmental field estimation," *Signal Processing*, vol. 128, pp. 369–381, 2016.
- [18] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research*, vol. 9, no. Feb, pp. 235–284, 2008.
- [19] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg, "Robust sensor placements at informative and communication-efficient locations," *ACM Transactions on Sensor Networks (TOSN)*, vol. 7, no. 4, p. 31, 2011.
- [20] X. Wu, M. Liu, and Y. Wu, "In-situ soil moisture sensing: Optimal sensor placement and field estimation," *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, no. 4, p. 33, 2012.
- [21] T. Dang, S. Frolov, N. Bulusu, W.-c. Feng, and A. Baptista, "Near optimal sensor selection in the columbia river (corie) observation network for data assimilation using genetic algorithms," in *International Conference on Distributed Computing in Sensor Systems (DCOSS)*. Springer, 2007, pp. 253–266.
- [22] P. G. Liaskovitis and C. Schurgers, "Leveraging redundancy in sampling-interpolation applications for sensor networks: A spectral approach," *ACM Transactions on Sensor Networks (TOSN)*, vol. 7, no. 2, p. 12, 2010.

- [23] J. Ranieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Transactions on signal processing*, vol. 62, no. 5, pp. 1135–1146, 2014.
- [24] W. Du, Z. Xing, M. Li, B. He, L. H. C. Chua, and H. Miao, "Sensor placement and measurement of wind for water quality studies in urban reservoirs," *ACM Transactions on Sensor Networks (TOSN)*, vol. 11, no. 3, p. 41, 2015.
- [25] Z. Khalfallah, I. Fajjariy, N. Aitsaadiz, R. Langar, and G. Pujolle, "A new wsn deployment algorithm for water pollution monitoring in amazon rainforest rivers," in *Global Communications Conference (GLOBECOM), 2013 IEEE*. IEEE, 2013, pp. 267–273.
- [26] J. D. Marshall, E. Nethery, and M. Brauer, "Within-urban variability in ambient air pollution: comparison of estimation methods," *Atmospheric Environment*, vol. 42, no. 6, pp. 1359–1369, 2008.
- [27] M. Jerrett, A. Arain, P. Kanaroglou, B. Beckerman, D. Potoglou, T. Sahuvaroglu, J. Morrison, and C. Giovis, "A review and evaluation of intraurban air pollution exposure models," *Journal of Exposure Science and Environmental Epidemiology*, vol. 15, no. 2, pp. 185–204, 2005.
- [28] A. Daly and P. Zannetti, "Air pollution modeling—an overview," *Ambient air pollution*, 2007.
- [29] J. M. Stockie, "The mathematics of atmospheric dispersion modeling," *Siam Review*, vol. 53, no. 2, pp. 349–372, 2011.
- [30] D. W. Wong, L. Yuan, and S. A. Perlin, "Comparison of spatial interpolation methods for the estimation of air quality data," *Journal of Exposure Science and Environmental Epidemiology*, vol. 14, no. 5, pp. 404–415, 2004.
- [31] G. Hoek, R. Beelen, K. De Hoogh, D. Vienneau, J. Gulliver, P. Fischer, and D. Briggs, "A review of land-use regression models to assess spatial variation of outdoor air pollution," *Atmospheric environment*, vol. 42, no. 33, pp. 7561–7578, 2008.
- [32] V. Gallart, S. Felici-Castell, M. Delamo, A. Foster, and J. J. Perez, "Evaluation of a real, low cost, urban wsn deployment for accurate environmental monitoring," in *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*. IEEE, 2011, pp. 634–639.
- [33] L. Soulhac, P. Salizzoni, P. Mejean, D. Didier, and I. Rios, "The model sirane for atmospheric urban pollutant dispersion; part ii, validation of the model on a real case study," *Atmospheric Environment*, vol. 49, pp. 320–337, 2012.
- [34] A. Tilloy, V. Mallet, D. Poulet, C. Pesin, and F. Brocheton, "Blue-based no2 data assimilation at urban scale," *Journal of Geophysical Research: Atmospheres*, vol. 118, no. 4, pp. 2031–2040, 2013.

- [35] Air Rhône-Alpes, “The air quality monitoring organization of the lyon agglomeration,” <http://www.air-rhonealpes.fr> [2016-01-27].
- [36] S. Shekhar, J. Kang, and V. Gandhi, “Spatial data mining,” in *Encyclopedia of Database Systems*, L. LIU and M. ÖZSU, Eds. Springer US, 2009, pp. 2695–2698.
- [37] A. A. L. Kama, H. Hammadou, B. Meurisse, and C. Papaix, “Le pic de pollution a paris du 12 au 17 mars 2014,” Tech. Rep.
- [38] M. Cardei, M. O. Pervaiz, and I. Cardei, “Energy-efficient range assignment in heterogeneous wireless sensor networks,” in *ICWMC’06. International Conference on*. IEEE, 2006, pp. 11–11.
- [39] M. Patel, R. Chandrasekaran, and S. Venkatesan, “Energy efficient sensor, relay and base station placements for coverage, connectivity and routing,” in *IPCCC 2005. 24th IEEE International*. IEEE, 2005, pp. 581–586.
- [40] London DataStore, “London atmospheric emissions inventory 2010,” on <http://data.london.gov.uk/dataset/london-atmospheric-emissions-inventory-2010> [2016-01-15].
- [41] The guardian, “London ranks among worst european cities for air pollution,” on <http://www.theguardian.com/environment/2011/sep/07/london-worst-european-cities-air-pollution> [2016-02-15].
- [42] Wind Finder, “Wind and weather statistics london-heathrow,” on <http://www.windfinder.com/windstatistics/london-heathrow> [2016-01-15].
- [43] Camden Data, “Camden lighting point,” on <http://www.camdendata.info/Pages/Home.aspx> [2015-07-15].
- [44] S. R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, “Energy efficient schemes for wireless sensor networks with multiple mobile base stations,” in *Global telecommunications conference, 2003. GLOBECOM’03. IEEE*, vol. 1. IEEE, 2003, pp. 377–381.
- [45] A. Boubrima, W. Bechkit, and H. Rivano, “Error-bounded air quality mapping using wireless sensor networks,” in *Local Computer Networks (LCN), 2016 IEEE 41st Conference on*. IEEE, 2016, pp. 380–388.
- [46] —, “A new wsn deployment approach for air pollution monitoring,” in *Consumer Communications and Networking (CCNC), 2017 IEEE 14th Conference on*. IEEE.
- [47] A. Boubrima, F. Matigot, W. Bechkit, H. Rivano, and A. Ruas, “Optimal deployment of wireless sensor networks for air pollution monitoring,” in *ICCCN’15. 24th International Conference on Computer Communications and Networks*. IEEE, 2015, pp. 1–7.

- [48] A. Boubrima, W. Bechkit, and H. Rivano, "Optimal wsn deployment models for air pollution monitoring," *IEEE Transactions on Wireless Communications*, To appear.
- [49] A. Velasco, R. Ferrero, F. Gandino, B. Montrucchio, and M. Rebaudengo, "A mobile and low-cost system for environmental monitoring: A case study," *Sensors*, vol. 16, no. 5, p. 710, 2016.