AMERICAN UNIVERSITY OF BEIRUT

DEVELOPMENT OF A SMART ALGORITHM FOR AIR POLLUTION SOURCES IDENTIFICATION USING PHYSICAL DISPERSION MODELING AND BAYESIAN INFERENCE

by
ELISSAR TAREK AL AAWAR

A thesis
submitted in fulfillment of the requirements
for the degree of Master of Engineering
to the Department of Mechanical Engineering
of Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
May 2021

AMERICAN UNIVERSITY OF BEIRUT

DEVELOPMENT OF A SMART ALGORITHM FOR AIR POLLUTION SOURCES IDENTIFICATION USING PHYSICAL DISPERSION MODELING AND BAYESIAN INFERENCE

by
ELISSAR TAREK AL AAWAR

Approved by:

Dr. Issam Lakkis, Professor — Advisor
Mechanical Engineering

aoude

Dr. Dany Abou Jaoude, Assistant Professor — Member of Committee
Mechanical Engineering

Dr. Ibrahim Hoteit, Professor — Member of Committee
Earth Sciences and Engineering
Applied Mathematics and Computational Science, KAUST

Date of thesis defense: May 31, 2021

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name:	Al Aawar	Elissar	Tarek
	Last	First	Middle

☑ Master's Thesis ○ Master's Project ○ Doctoral Dissertation

☑ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

□ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes after: **One ___ year from the date of submission of my thesis, dissertation or project.**
Two ___ years from the date of submission of my thesis , dissertation or project.
Three ___ years from the date of submission of my thesis , dissertation or project.

Elissar

June 14, 2021

Signature Date

This form is signed when submitting the thesis, dissertation, or project to the University Libraries

An Abstract of the Thesis of

Elissar Tarek Al Aawar for Master of Engineering
Major: Mechanical Engineering

Title: Development of a Smart Algorithm for Air Pollution Sources Identification using Physical Dispersion Modeling and Bayesian Inference

Air pollution plumes are commonly observed in the atmosphere above many cities and residential areas. These plumes may be the result of either a normal operation or an accidental release from certain sources. In both cases, it is of great importance to identify and characterize these sources for the assessment of the harmful effects of their resulting pollution fields and for the proper construction of an emergency response plan in case of accidental releases. This involves the inverse problem, from destination of pollution back to its source, and the inference of the different parameters characterizing this source given certain known or measured sets of observations.

The aim of this thesis work is to introduce and develop a smart algorithm that is able to identify and characterize an air pollution source that is responsible for an observed concentration field of pollutants in a specific urban location. As an application, we will infer several parameters of an active source that is releasing air contaminants into the atmosphere of a selected domain around KAUST (King Abdullah University for Science and Technology) in the region of Thuwal, KSA. These parameters include the source geographic location, emission strength and emission duration. A stochastic approach using Bayesian inference and Monte Carlo sampling will be implemented to solve the ill-posed inverse problem and characterize the emitting source. In this scope, the forward Lagrangian model will be adopted to study the atmospheric dispersion of pollutants and resolve the urban characteristics of the domain. The implementation of this model will be done while considering the prevailing wind field as the main driving source and based on the well known urban configuration of buildings (serving as obstacles) and the natural topographic features of the location.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Air pollution is one of the main issues posing threats to the health, environment and global climate. Environmental monitoring is excessively being oriented towards examining and studying the different atmospheric dispersion patterns of the different air contaminants including any chemical, biological or radiological compounds.These contaminants can be the result of either normal operations or accidental events. Normal operations include conventional human activities that can deliberately release pollutants into the atmosphere. On the other hand, accidental releases can be attributed to leaks that may occur due to low maintenance, fugitive emissions, human error or ignorance, terrorist attacks, system failures, super natural events, fires, explosions... In both cases, there is an insisting need to find the sources of such dispersion patterns for proper mitigation strategies, risk assessment and emergency responses during accidents. This need comes as environmental monitoring is being a main field for extensive research and remarkable advancements. This includes progress in both sensing networks and remote techniques coupled with significant progress in modeling and computations.These two growing domains convey the basic fundamentals that can be used to simulate atmospheric dispersion and to reinforce the simulated results with the proper measurements and observations.

However, in the backward approach of estimating the source parameters responsible for an observed set of concentration values, the challenge lies in the ill-posed nature of the question at hand. The main issue is in the non-uniqueness of the obtained solution as there may be a set of various possible source parameters that would fit the observation. Another issue rises at the level of the observation itself where an observation revealing the concentration values of a pollutant or set of pollutants in a certain spatial or temporal reference can frequently be non-representative of the real case. This is mainly referred to the errors in the acquisition phase due to the noise introduced by the sensing network or the remote measurement instrument and in the data treatment phase. Coming into the dispersion modeling itself, an additional key factor is the complexity of the physical mechanism relating the source parameters to the observations and its high dependence on meteorological, topographic and urban features at the mesoscale and mircoscale. Variations in these inputs introduce an additional source of uncertainty that may affect the output of the model and hence the source term estimation.

Several approaches were used to implement methods that are able to solve the problem of source term estimation. These are mainly divided into deterministic and probabilistic approaches. To overcome the problem of uniqueness of solution, the former searches for an optimal best-fit solution that minimizes a certain cost function. In this scope, the most intuitive approach is the inverse transport where the dispersion model is integrated backward from the observation time to the emission time [1]. This approach was implemented in a study that used the solution of the adjoint tracer transport to characterize the air pollution sources responsible for a certain contamination [2]. That is by which a backward integration in time of the adjoint of a linear dispersion model was done using the Lagrange duality relation. This particular study predicted the source of a large-scale radioactive tropospheric contamination of nuclear tracers based on atmospheric measurements. Other studies used the evolutionary computations of the genetic optimization algorithm to combine the forward dispersion model with a backward receptor model. Both the puff and the Gaussian plume models were tested using synthetic data and resulted in successful solutions [3, 4]. In addition to this method, different optimization approaches are implemented while relying on regularization to buildup a unique well-posed solution that minimizes the cost function obtained from the original ill-posed inverse problem [5, 6, 7].

On the contrary, instead of searching for one optimal combination of source parameters, the probabilistic approach characterizes an ensemble of source parameters configurations. Such an approach provides purely probabilistic indications about the solution and hence allows for the quantification of uncertainty which is not achieved by the first approach. In this scope, the inverse problem is regarded as a Bayesian problem to be dealt with stochastically [8]. This is being favored over the optimization approach due to the various advantages provided by the Bayesian framework [9, 10, 11, 12]. This framework is then able to find the probability distribution of each of the source parameters under inspection and hence the most probable configuration of source parameters leading to the observation at hand. To do so, the Bayesian inference is usually coupled with a stochastic sampling tool to fully study the possible combinations of source parameters and quantify the relative uncertainties. The most frequently used sampling tool is the family of Markov Chain Monte Carlo (MCMC) algorithms [13]. This combination between Bayesian inference and MCMC sampling was used to reconstruct an emission source from synthetic data [10] or real measurements [11, 9] and to optimally define characteristics of a newly developed air pollution sensing network or to estimate the costs and benefits of an existing network [14]. Another study applied this approach to real accidental nuclear release at a continental scale based on some measurements from an existing sensing network [15].

In this research work, we will solve the ill-posed air pollution source determination problem using a Bayesian inference framework coupled with the of MCMC sampling and a Lagrangian atmospheric dispersion model in an urban environment. Several natures of observations including concentration fields, contours and point-wise values, will be tested. Section 2 represents the scientific background of the different theories used to fulfill the aim of our research. Section 3 provides detailed description of our methodology including an overview about the used dispersion model, the studied domain, the Bayesian settings and the algorithm. Section 4 is dedicated to define the numerical experiments and the settings of their conduction based on synthetic observations of different natures. Finally, section 5 will represent the obtained results of these experiments along with their interpretaions.

Chapter 2

Literature Review

The objective of this literature survey is to provide an overview of the different concepts, theories and methodologies that are used as a basis for the current study and to present previous work related to source reconstruction problem. The main concepts to be covered are related to pollution dispersion modeling, Bayesian inference, Monte Carlo sampling techniques, optimal transport theory and environmental monitoring. These concepts will guide the work into fulfilling our main aim that is the inverse approach of identifying and characterizing the sources that are responsible for an air contamination field in a certain region.

2.1 Problem Statement

Atmospheric dispersion modeling stands for the set of mathematical computations that aim to understand how pollutants or contaminants are traveling in the atmosphere. In this scope, there are two directions of in-time approaches for dispersion modeling. The first approach is by computing the concentrations of a certain pollutant which is transported downwind of a known source. This is known as the forward approach and it requires basic knowledge of the characteristics of the source and the parameters governing its emissions. However, a backward approach also exists and it is done from destination back to source. That is trying to reconstruct the locations of sources contributing to a field of observed or measured concentrations in a certain region. This approach has little or no information about the source and is directed towards computing its different parameters with a certain level of uncertainty. These approaches are very useful in the different research domains including air pollution analysis [16, 17, 18], marine transport [19, 20], oil and gas studies [21, 22] and many other domains.

The problem of interest in this thesis work is the source inversion problem commonly referred to as source term determination. This approach consists of characterizing a source or a group of sources along with their corresponding parameters based on a set of known observations or measured data. The parameters of interest studied in this work are the source location (x and y coordinates or latitude and longitude), the source height, the emission rate and the emission duration. It is worth noting that the inverse approach of source characterization is getting more and more important. This is basically due to the fact that obtaining knowledge about the sources is a basic and fundamental requirement for risk assessment, risk mitigation and emergency response plans.

2.2 Air Pollution Dispersion Modeling

As air pollution imposes serious threats at different levels, it is of great interest to monitor the levels of pollutants in the atmosphere. The most effective way of such a task is by installing a well designed network of sensors that could give real time measurements of the concentrations of the different pollutants present in a certain region. These networks can measure the concentrations of different pollutants by using different techniques that serve one goal: air quality monitoring [23, 24, 25, 26]. However, a major problem could make the usage of sensing networks limited; that is their high costs and continuous need for maintenance. Therefore, many governments, agencies or research centers are using dispersion modeling as an alternative.

Dispersion modeling involves the tracking of pollutants in the atmosphere that are driven by the prevailing wind conditions and redirected by the different surface features. These features include the surface topographic geometry and the different man made structures [27]. So, dispersion models usually require a certain set of inputs including wind characteristics, atmospheric stability classes, ambient temperatures, emission patterns and characteristics, terrain features and elevations, obstructing man made structures characteristics...

There are several types of dispersion models including box, Gaussian, Lagrangian and Eulerian models [28, 29]. The box model is a very simple model that assumes that the domain under investigation is in the form of a box. It also considers the air pollutants to be homogeneously distributed in this box and performs computations accordingly [30]. These assumptions disregard many essential factors affecting the pollution pattern; thus, making the box model impractical in realistic settings. On the other hand, the Gaussian model is the oldest and most common model used for studying continuous and non-continuous puff emissions. It usually assumes that the plume follow a Gaussian distribution in the vertical and crosswind directions. It also considers the effect of ground reflection of the plume and performs computations accordingly [31]. As for Lagrangian models, these are used to track the parcels as they move in the atmosphere through a moving reference frame. They compute the trajectories along which these parcels move based on the prevailing meteorological conditions and domain features. Similarly, the Eulerian models handles the computations as Lagrangian models do, but in a fixed grid as a reference [32, 33].

Based on the previous, as the characteristics of the landscape get more heterogeneous, it becomes more challenging to explicitly execute dispersion models [34]. That is the case of urban environment where the abundance of buildings and on-surface structures should be resolved by the model; however, not all models are able to handle such environments. For instance, the Gaussian plume model can not handle the street canyons and surface features in urban environments [35]. On the contrary, Lagrangian models could handle three dimensional domains that can properly resolve the urban environments and their complex features [36].

Despite the differences in assumptions and conditions used in the different dispersion models, the air pollution sources are similarly classified. This classification can be on the basis of the shapes of the sources i.e. point, line or area sources. For instance, an emitting stack is considered a point source, cars along a road are considered line sources and a fire is referred to as an area source. These sources can also be classified based on their emission duration to puff sources or continuous sources. Puff sources refer to short term and accidental emissions unlike normal activities having continuous sources.

These dispersion models are highly used in researches and studies by many governmental agencies and environmental organizations. They are used as tools for air pollution control, assessing human activities impact on air pollution, epidemiological studies, managing air quality, compliance testing [37, 38]...

2.3 Solutions for the Inverse Problems of Atmospheric Contamination

Several approaches and algorithms were investigated by researchers in order to reconstruct the air pollution sources responsible for a certain contamination. These approaches are divided into two main categories: deterministic optimization methods and stochastic Bayesian approaches.

2.3.1 Deterministic Optimization

2.3.1.1 Definition

Deterministic solutions are those unique solutions that are found in mathematical modeling for a well-posed problem; that is a problem that have three basic conditions: existence, uniqueness and stability of the solution(s) [39, 40]. However, the backward inverse problems are usually ill-posed problems where most often the condition of stability is violated. In such problems, a small variation in the initial data can cause a larger variation in the obtained solution(s).

Based on that, deterministic optimization methods are used in the inverse problems. These methods aim to select the optimal solution to the problem at hand, out of an infinite set of solutions. The selected solution is the one that best fits the data and that minimizes the residuals between the obtained and observed data. These residuals are obtained from an objective function that we aim to minimize by the selection of the best fit solution without any quantification of the resulting uncertainty.

These methods usually use the different principles of linear algebra as they use gradients of the objective function to solve the minimization problem. The objective function to be minimized is the sum of residuals between modeled and real data. A residual R can be represented by Equation 2.1:

$$R_i = y_i - f(x_i) \tag{2.1}$$

where y_i is the i^{th} observed value of the dependant variable under inspection and $f(x_i)$ is the model function that is predicting the observed value based on the independent variable x.

2.3.1.2 Deterministic Optimization Methods

There are several methodologies that are used in this scope. These include the least square method, genetic algorithm, maximum entropy...

a. Least Square Method

This is the most basic approach to reduce residuals. It minimizes an objective function that can be stated as a sum S of residuals R for n data points:

$$S = \sum_{i=1}^{n} R_i^2 \tag{2.2}$$

In this case, we are minimizing the sum of the squares of the residuals that is expressed in Equation 2.3.

$$\|R_i\|^2 = \|y_i - f(x_i)\|^2 \tag{2.3}$$

where $\|.\|$ is the Euclidean norm.

The simplicity and reliability of this method depends on the nature of problem at hand. If the problem is linear, then the solution converges in a straightforward manner. However, many problems are non linear and require further approximations and assumptions. The approximations depend on using Jacobians and Taylor's series expansion up to a certain order. This results in a reordering of the residual expression. Also, certain iterations must take place to specify a suitable initial point. The inadequate choice of the initial point can negatively affect the solution where it will converge to local minima rather than finding the global solution.

b. Regularization of Least Square Method

In ill posed problems, as it is the case of inverse problems, the ordinary least square method results cannot always obtain a unique solution where the system of equations is either over determined or under determined. That is why this method aims to regularize the previous method by constraining the resulting solution. This regularization process adds a cost term to the objective function in order to find an optimal solution and reduce the probability of overfitting. This regularization term holds some prior knowledge about the problem. The objective function CF can the be stated as shown in Equation 2.4:

$$CF(x) = \|y - f(x)\|^2 + \alpha\gamma(x) \tag{2.4}$$

where CF is the cost function, y is the observed value of the dependant variable, f(x) is the model function that is predicting the observed value based on the independent variable x, α is a parameter reflecting a certain weight between the regularization function γ representing the constraint and the original residuals.

The different regularization methods vary based on the choice of the regularization function used. Two main popular regularization methods are Tikhonov and LASSO (least absolute shrinkage and selection operator) methods. The regularization function used in Tikhonov method is an L_2 norm whereas that of LASSO method is an L_1 norm. Another difference lies in the feature selection that results from each process, where some methods can select all features of a problem; while others discard some and consider the rest [41, 42, 43].

c. Genetic Algorithm

This is one global search approach that is classified within the evolutionary algorithms which are highly used in the artificial intelligence domain. These algorithms are named evolutionary due to the fact that they are inspired from natural biological evolution processes [44]. The genetic algorithm (GA) is mainly inspired from genetic processes including chromosomal mutations, crossover and selection.

This algorithm first requires a genetic representation of the space of candidate solutions over which genetic processes take place. These representation are usually encoded in binary format i.e. a combination of 0's and 1's. Then, successive processes are done and aim to evolve the whole genetic combination at hand into a better one. In this scope, deciding whether a new population is better or not depends on the fitness function of the problem. These processes include initialization of the population to obtain a first generation from which a certain combination having the best value of fitness is selected. Then, these selected "parent" solutions undergo breeding through mutations and crossovers to produce "child" solutions forming the second generation with better fitness. This process is repeated iteratively until a certain number of iterations is reached or until convergence is achieved. An illustration of the iterative process of the GA algorithm is represented in Figure 2.1.

Many challenges are encountered in the GA including the formulation of a fitness function, the population size, the choice of computational parameters and the selection criteria of the new population, computational time [45]... In this scope, many other optimization algorithms exist and are also inspired from natural behaviors or evolutionary processes. These methods include Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Differential Evolution (DE)...

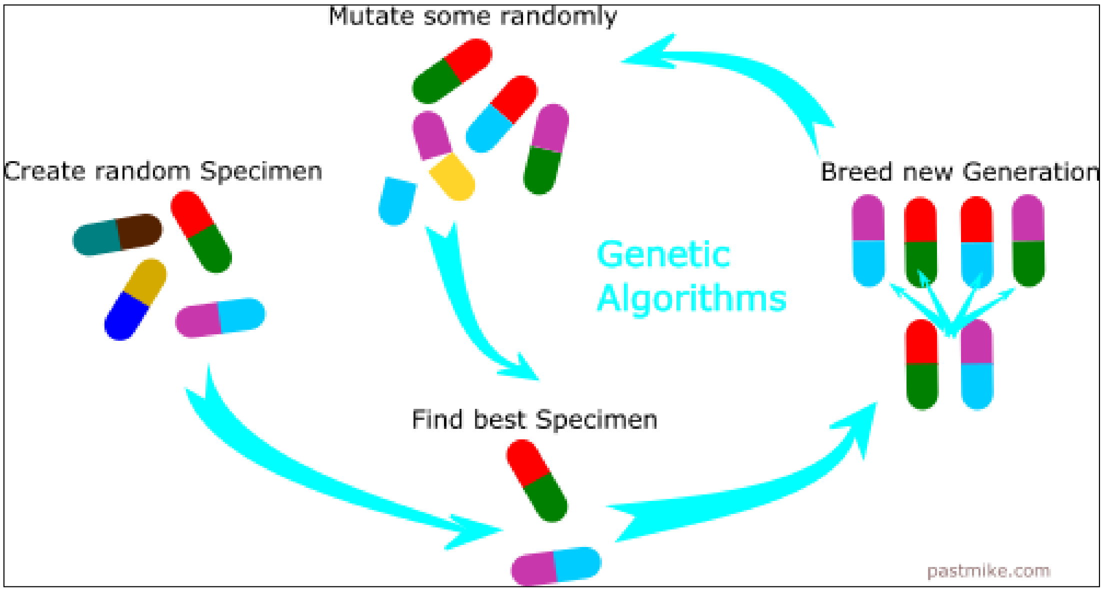


Figure 2.1: Illustration of genetic algorithm stages.

2.3.2 Stochastic Bayesian Methods

The second class of methods to solve the inverse problem is the stochastic probabilistic approach. This approach is expressed within a Bayesian reference that represents the final solution as a posterior probability and that can deal with the uncertainty of the input data, model and resulting solution.

On the other hand, unlike deterministic approaches, this approach overcomes the limitation of convergence to a local minima; especially when equipped with an efficient sampling tool. Such a tool will allow full exploration of the solution space and will find the global solution of the inverse problem.

2.3.2.1 Bayesian Inference

Bayesian inference is an algorithm used in statistical inference. It was established based on Bayes' theorem named after Thomas Bayes who was the first to employ conditional probability coupled with prior knowledge of conditions leading to specific events; that is to obtain probabilities of certain aspects or parameters of those events [46]. Bayes' theorem is mathematically expressed in Equation 2.5:

$$P(M|D) = \frac{P(M)P(D|M)}{P(D)} \tag{2.5}$$

The left hand term of the equality $P(M|D)$ is called the posterior. It is the probability of a certain event M knowing the set of data related to the event D. Usually, M refers to the model parameters and D refers to measured or calculated quantities. Here, we can notice the inverse approach included in Bayes' theorem where M contains the parameters of a model whose results would be represented in D. So, at the level of the model, we are calculating the probability of the input knowing the output. This probability is our desired probability that contains the solution to our inverse problem.

As for the right hand term, P(M) is known as the prior that is our knowledge of the parameters of the model M before having the data of event D. Explicitly, we have no prior knowledge about M reflecting our ignorance about the proposals they convey. However, in most cases, the prior is assumed to follow a uniform distribution for Cartesian variables.

The second term of the numerator $P(D|M)$ is the likelihood probability. It represents the probability of having a certain set of data D given a certain combination of model parameters as defined by M. The likelihood is usually used to measure the discrepancies between the synthetic obtained data and the original data.

The denominator term P(D) is the evidence or the marginal likelihood which in other terms means the probability of obtaining the specific set of data D for all possible combinations of parameters of the model. So, it is obtained by integrating the likelihood over all the space of combinations of model parameters represented in M (Equation 2.6).

$$P(D) = \int_{allM} P(M)P(D|M)dM \tag{2.6}$$

. When the number of sources is known and finite, this probability is usually a constant that is disregarded in the calculations. Having the denominator as a constant, the posterior is then proportional to the product of the prior and the likelihood. This leaves us with the reduced form of Bayes' theorem in Equation 2.7:

$$P(M|D) \propto P(M)P(D|M) \tag{2.7}$$

The above theorem is then used iteratively in mathematical inference to dynamically update the probability of a certain hypothesis and gain more and more knowledge reaching our target posterior. Hence, Bayesian inference constitutes a logical framework that converges into the determination of parameters involved in a certain model M and leading to a set of well known data D. This framework is very popular and has been applied to a wide range of domains include earth sciences, engineering, medicine and philosophy [47, 48, 12].

2.3.2.2 Markov Chain Monte Carlo Sampling

Monte Carlo sampling is a stochastic sampling technique that is used in combination with Bayesian inference. This sampling method is very useful in high dimensional parameter space where obtaining analytical solutions is not practical. The technique consists of sampling from a certain probability distribution and creating a Markov chain of sampled states in an iterative manner until the chain converges to the target distribution of parameters. This target distribution usually represents the posterior distribution of the inferred parameters. For proper updating of the chains, an acceptance criteria should be established on the basis of given proposal and likelihood distributions.

One popular algorithm of sampling used in this scope is Metropolis Hastings algorithm. It obtains random samples of parameters from a certain probability distribution and then creates a Markov chain based on a well known acceptance criteria. The algorithm works as follows:

1. Initialize a sample x_0 out from the parameters' space, where x_i is a vector with the same dimension as the parameters' space.

2. Suggest a proposal distribution f(x), that is a distribution that describes the choice of the new candidate point. Usually, this distribution is selected to be a normal distribution centered at the current sample.

3. Start the iterative process. That's for a certain number of iterations, do the following:

i- Sample a new candidate point x_i based on f(x). Usually, sample x_i follows a normal distribution having x_{i-1} as its mean.

ii- Sample a value of u from a uniform distribution between 0 and 1.

iii- Calculate the acceptance ratio r that is $\mathrm{g}(_{new})/\mathrm{g}(\mathrm{x}_{old})$, where g is a distribution that is proportional to the posterior's distribution.

iv- Compare u to r to see whether to accept or to reject the candidate sample.
a) if $\mathrm{u} \leq \mathrm{r}$, accept.
b) if $\mathrm{u} > \mathrm{r}$, reject.

v- Update the Markov chain accordingly. If the sample is accepted, the new chain value is x_{new}. Otherwise, if the sample is rejected, the new chain value remains x_{old}.

2.4 Optimal Transport Theory

In this section, an overview about the optimal transport theory will be discussed. This theory has several applications and uses different metrics for its computations. Among these metrics is the Wasserstein distance which will also be introduced.

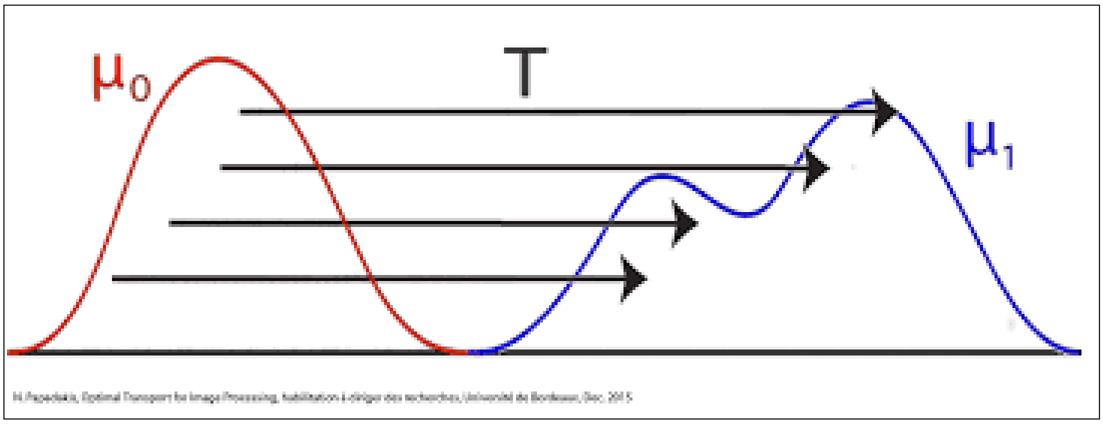
2.4.1 Definition and Purpose

The optimal transport theory is a mathematical theory that tackles the problem of transportation between different fields or functions. The history of this theory started with Gaspard Monge in 1781 [49] for military uses. Monge was trying to figure out the optimal and most economical way to transform a terrain with an initial landscape into a new target landscape [50]. This should be done by moving a certain amount of sand from a quarry to a construction location while minimizing the required amount of work or workers. Monge's problem is stated in Equation 2.8.

$$\min_{T:X \longrightarrow X} \int_{x} c(x, T(x))u(x)dx \tag{2.8}$$

where T is a translation function or map (if it exists) that transforms u into v, c is a convex distance (which is the Euclidean distance in Monge's case) and u,v are positive functions describing the problem at hand. This minimization process is subjected to certain constraints that are mainly referred to conservation of quantities [51]. For instance, Monge's problem has the conservation of mass of land to be moved as a constraint.

Generally, the optimal transport theory is a mathematical theory in the scope of convex optimization that includes several schemes or sets of metrics that relate different distributions. It handles minimization problems and can be used to make distance measurements between two functions or two fields [52]. This is applied while enforcing a certain conservation property as a constraint. A set of transformations can be found, yet, the main aim is to find the optimal transformation T i.e. the one that gives the minimal distance between the two functions (Figure 2.2).



N. Papadakis, Optimal Transport for Image Processing, habilitation à diriger des recherches, Université de Bordeaux, Dec. 2015

Figure 2.2: Representation of an optimal transport problem. In this example, the optimal transformation T is found between two functions μ_0 and μ_1.

The modern optimal transport theory involves using Kantorovich duality formulation that overcomes the difficulties in the ill-posed Monge's formulation. These difficulties can be witnessed at the level of the conservation constraint and uniqueness of the solution [53]. This new formulation is a convex relaxation of the initial Monge's problem that introduces probability measures of u and v and minimizes the cost function over the set of transport plans between u and v. This made the optimal transport theory applicable in different domains including machine learning, fluid dynamics, statistics, optics [54, 55, 56]...

2.4.2 Wasserstein Metric

There are many metrics that we use to measure the distance between two distributions. These distances include total variation, Hellinger, L_2 and χ^2 norms.

Despite being useful, such distances have certain drawbacks. One drawback relies in the fact that we cannot use them to measure the distance between two fields of different natures i.e. continuous and discrete fields. It also disregards the geometry of the distributions over the multidimensional space. That is by which such distances represent only a number disregarding any qualitative aspects of the two distributions. This can be illustrated by considering one distribution that we intend to interpolate into a new distribution as shown if Figure 2.3. The path between the two distributions is computed using the Wasserstein distance in the upper row and using the Euclidean L_2 norm in the lower one. We notice that the Euclidean path did not preserve the structure of the initial distribution [57].

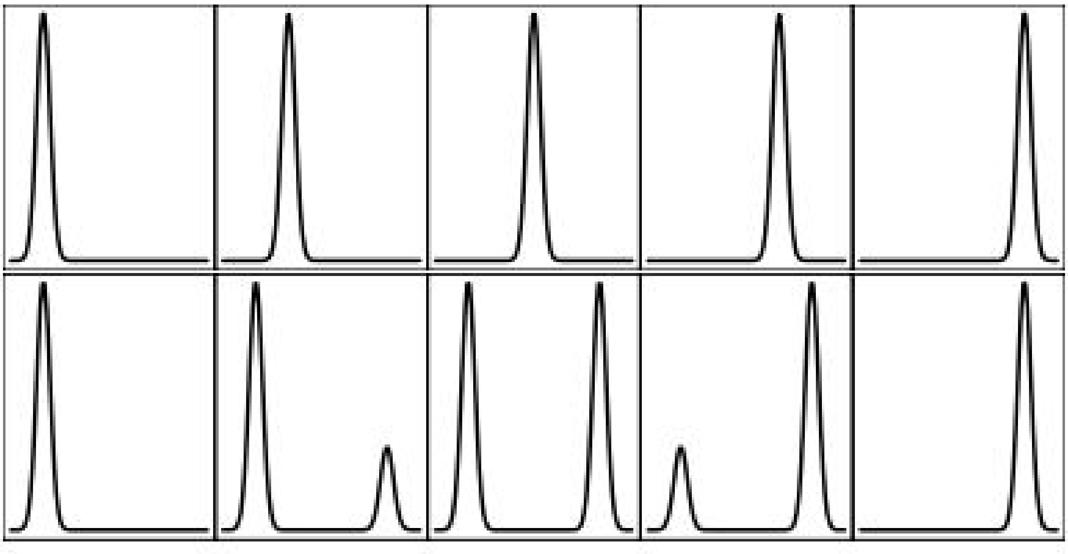


Figure 2.3: Comparison between two different paths of two distributions: the geodisc path in the upper row to a Euclidean path in the lower row.

This leads us to introduce the Wasserstein or KantorovichRubinstein metric. It is the distance used to calculate the distance between probability distributions in a certain metric space M. It can be expressed by Equation 2.9.

$$W_p(P,Q) = (\min_{J\epsilon J(P,Q)} \int \|x-y\|^p dJ(x,y))^{1/p} \tag{2.9}$$

where W$_p$ is the Wasserstein distance of order p, (x,y) ϵ R^d x R^d and J(P,Q) is the joint probability for (x,y) having P and Q as marginals.

Going back to Monge's problem, this metric calculates the minimum cost that is needed to move a certain mass of sand from an initial location (quarry) to the new location (construction cite). This cost is the product of the mass that we have and the distance to be traveled. The whole computational process respects the constraint of conservation of mass. The method is applied in many domains including probabilistic studies, statistics, shape analysis, computer science, machine learning [58, 59, 60]...

$$GH_{1,2} = L_2(LH_{1,2}) \tag{2.10}$$

where $GH_{1,2}$ is the global Hausdorff distance between the two shapes in images Im_1 and Im_2 and L_2 denotes the L_2 norm of $LH_{1,2}$ is the local Hausdorff distance.

Chapter 3

Methodology

The aim of this chapter is to illustrate the different steps of the proposed framework that will be able to detect and characterize the pollution sources in certain incidents. This framework will combine the different concepts discussed in Chapter 2.

3.1 Dispersion Model

3.1.1 General Description

Our purpose is to study the relation between a certain set of proposed parameters describing a source of air pollution on one hand, and its resulting concentration fields on the other hand. Based on that, we need a dispersion model that is able to predict how the pollutants from a certain source would move and accumulate in the surrounding environment and under the prevailing wind conditions.

This computational exercise is a major step in our main algorithm. For this purpose, we use a software named "GRAMM-GRAL" that stands for Graz Mesoscale Model- Graz Lagrangian Model [61, 62]. As the name implies, the software consists of two coupled models: GRAMM for the mesoscale wind field computations and GRAL for the microscale wind field along with the Lagrangian particle transport calculations. In fact, these two quantities are directly related where the transport of particles in the atmosphere will be governed by the prevailing wind field and will eventually lead to the creation a certain concentration field.

The software involves the creation of two nested domains that are defined as two uniform rectilinear grids: a coarse resolution domain for GRAMM and a finer resolution domain for GRAL. It also requires certain input data that properly describes the domain. These include:

- Domain topography.
- Shape files of buildings and their characteristics.
- Wind conditions: speed, direction, atmospheric stability class.
- Air pollution sources whether point, line or area sources, their locations, emission rates, stack properties...
- Surface properties including albedo, emissivity, heat conductivity...

After defining these inputs, a simulation could be done to obtain GRAM wind field depending on the topography and a GRAL flow field that defines the wind flow around the buildings at the urban scale. GRAL will also give us the resulting concentration field within its domain.

3.1.2 Setting Up the Domain

In our case, the studied domain is KAUST university in Thuwal,KSA which is located on the coast of the Red Sea at 80 kilometers north of Jeddah (Figure 3.1). Figure 3.3 represents the specific area of study where the prognostic GRAL model is implemented. The location background image is extracted from Google Earth and is georeferened using the software's Graphical User Interface (GUI) in a Universal Transverse Mercator (UTM) coordinate system of latitude and longitude. This coordinate system divides the Earth's sphere into 60 zones and projects them into a plane. This projection makes our domain in UTM zone 37N ranging between 501,200 and 523,700 towards the East and between 2,457,100 and 2,477,400 towards the North. Both domains constitute uniformly meshed Cartesian grids such that the outer GRAMM domain has a resolution of 100 meters and the inner GRAL domain has a fine resolution that can be as high as 2 meters.

After setting up the domains' borders and resolutions, it is necessary to define the features of our location. Based on that, topography and land use input files of our specific domain were extracted from global rasters. In case of topography, we extracted the elevation at each center of each grid cell of the GRAMM domain to create a topographic raster representation which is fed to the model. In addition, the CORINE (Coordination of Infomation on the Environment) values of land use were extracted at those same locations [63]. The corine value is a representative number that implies a certain combination of albedo, soil moisture, emissivity, surface roughness and conductivity. This combination of data is also fed to the model; by that, we now have the natural surface features of our domain. Since we are studying an urban domain, another major component is the man made structures primarily including KAUST buildings. The shape file showing the clear geometry of all 2,111 buildings is represented in Figure 3.2. This shape file is also input into the model to setup a complete domain that is ready to be studied. These buildings are georeferenced in the same coordinate UTM reference and imported over the base map as shown in black within GRAL domain in Figure 3.3.

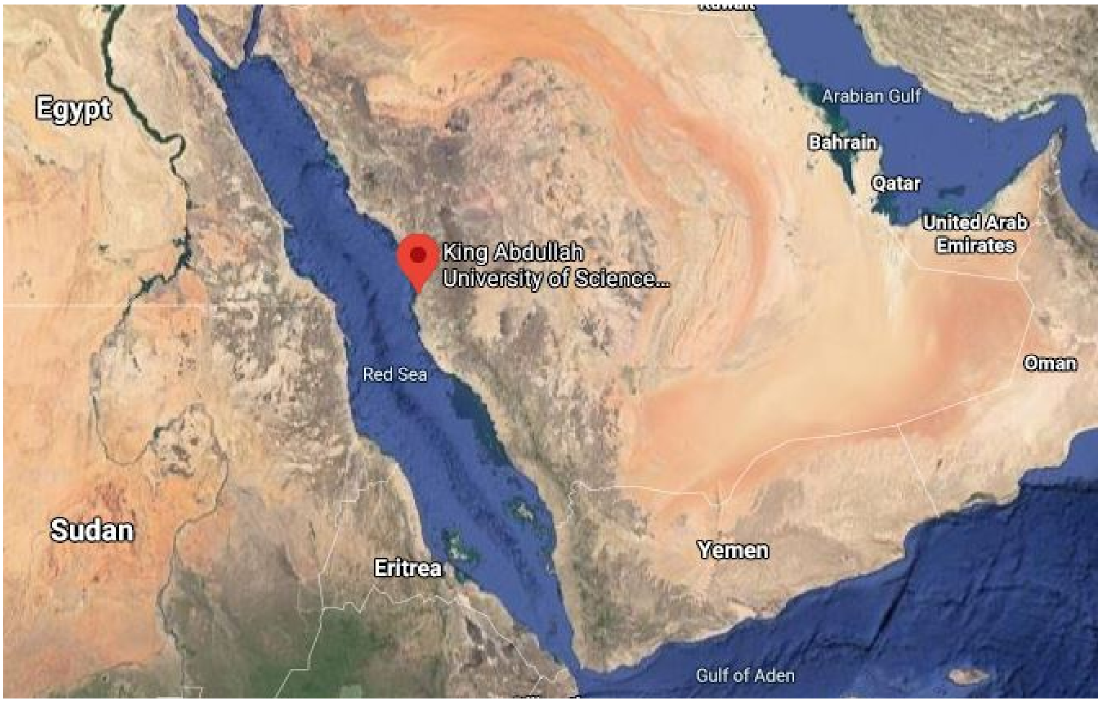


Figure 3.1: Geographical location of our domain.

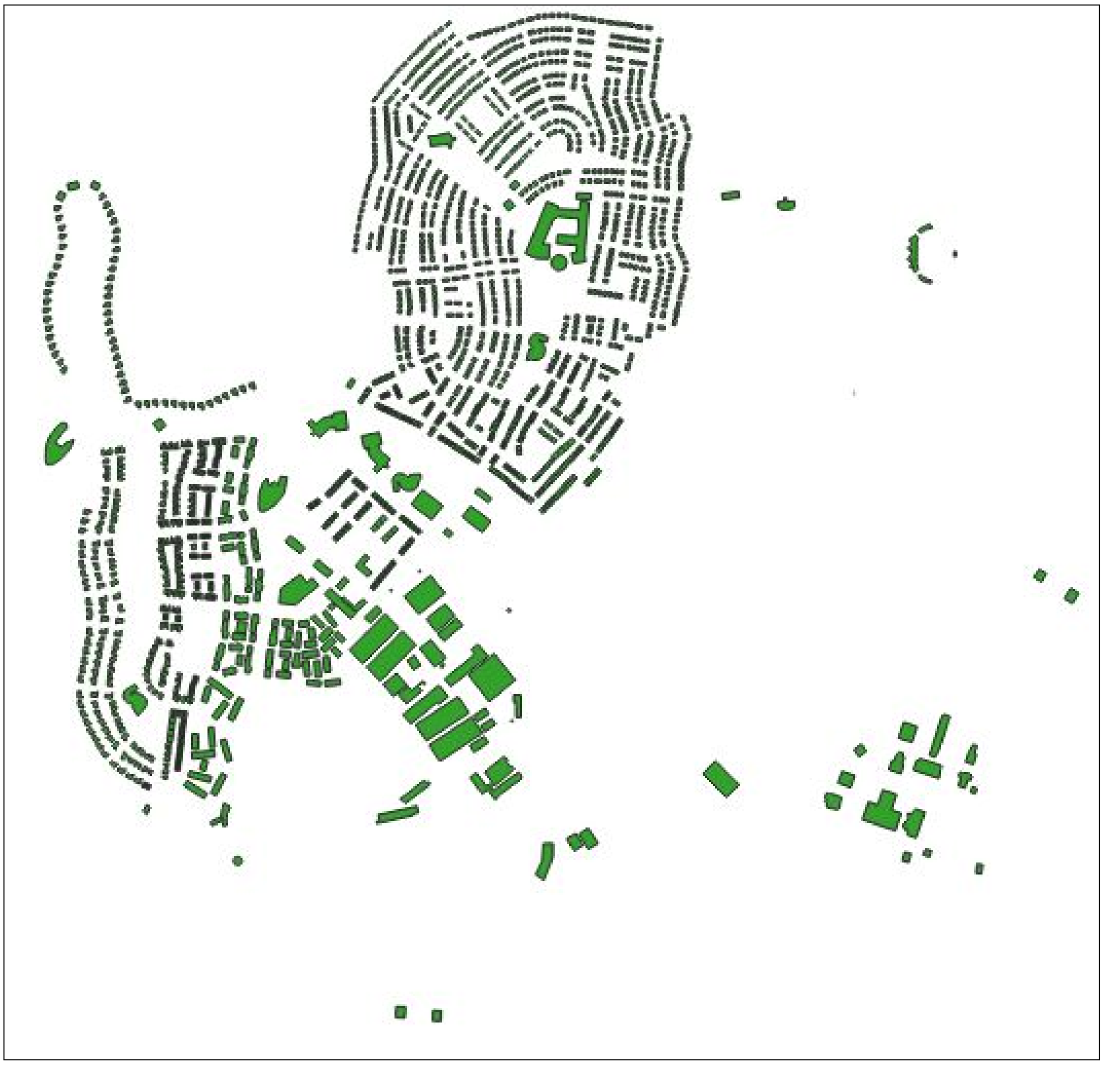


Figure 3.2: KAUST buildings shape file.

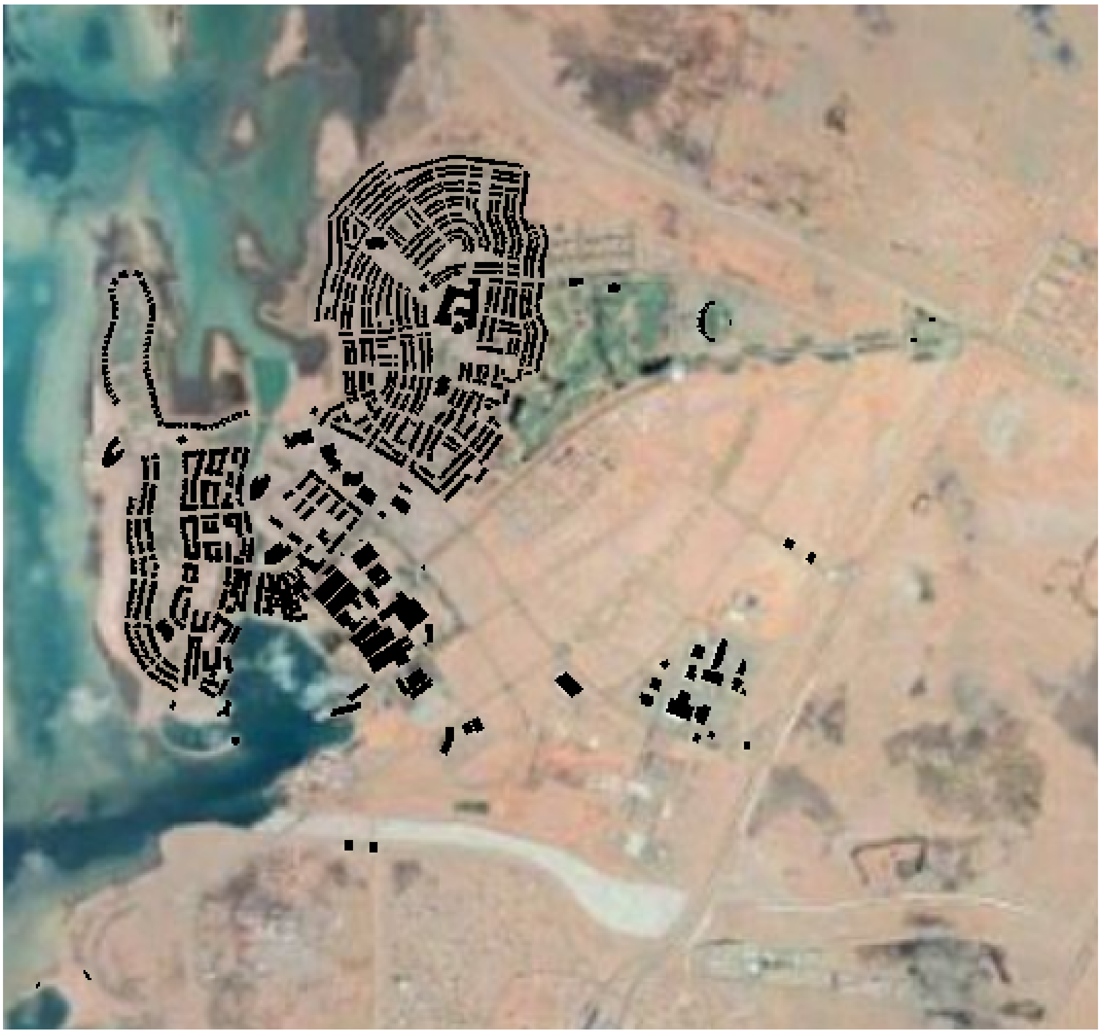


Figure 3.3: GRAL domain of study.

3.1.3 Running the Model

After setting up the domain, we are now ready to run the model. Usually GRAMM and GRAL models are run in series where we run GRAMM to get the prevailing mesoscale wind field and then we run GRAL to get the microscale wind field around the buildings and the resulting concentration fields in the presence of certain pollution sources. To do that, we must input the meteorological data at the boundary of our GRAMM domain. This data is in the form of a metfile containing the day, time, wind speed, wind direction and atmospheric stability class. In other words, this file contains the necessary boundary conditions that are used to get the flow and turbulence fields around the buildings in our specific topographic setup based on Navier Stokes equation and the k-ϵ turbulence closure model [61, 62].

The source(s) is defined as either point, line or area source. Its location, stack height, type of released contaminants, emission rates and emission duration are also all well defined. The final results that we get are average wind speeds and concentration values at each grid cell. The software provides its computed results in readable files that can be post processed and represented using other softwares such as the wind field represented in Figure 3.4. Also, the GUI of the software contains a post processing visualization tool where we can see its outputs as the concentration field shown in Figure 3.5.

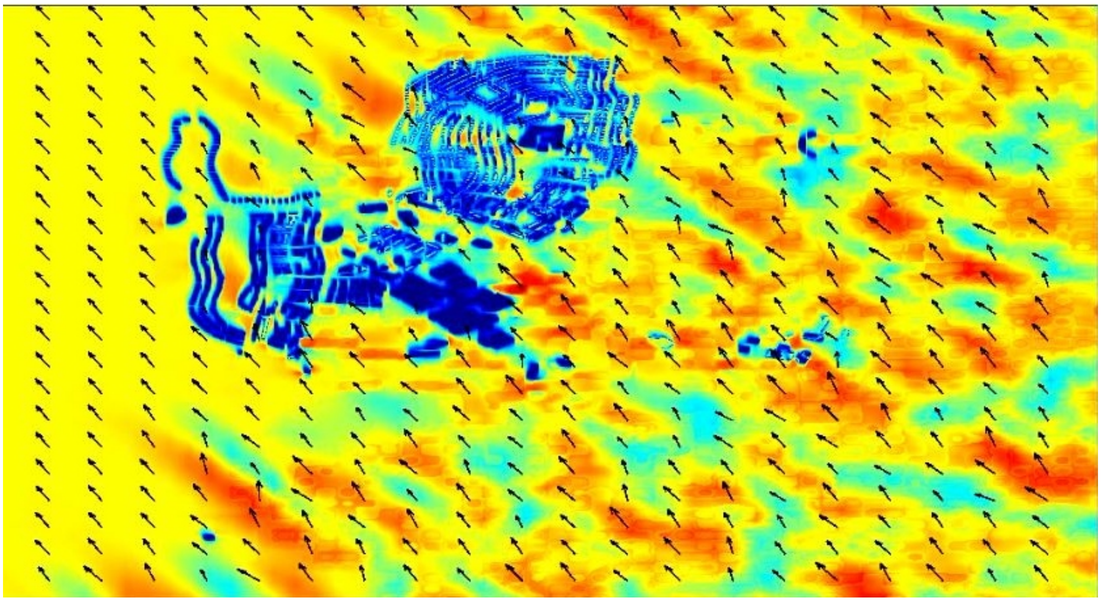


Figure 3.4: Instantaneous wind field computed by GRAMM-GRAL and visualized by MATLAB-R2019b.

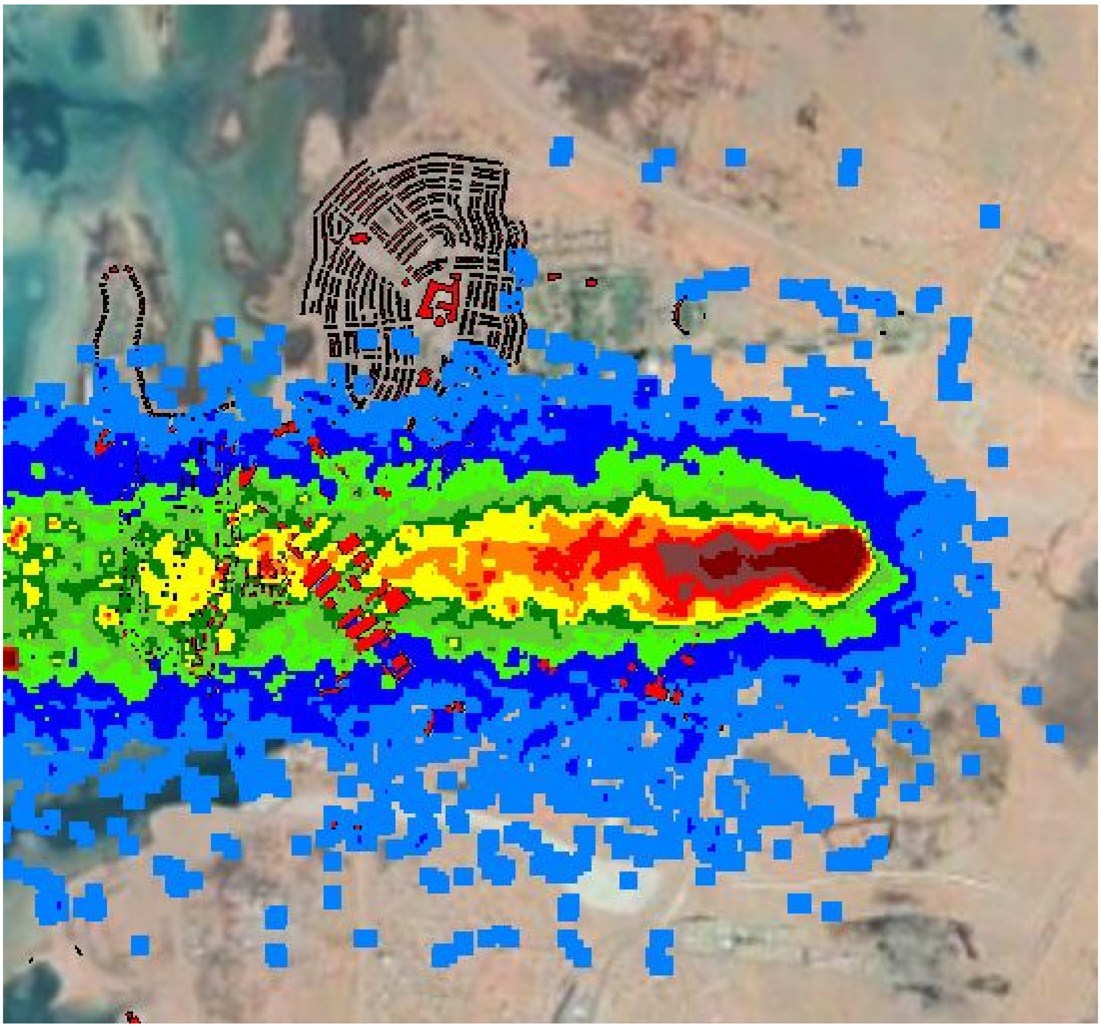


Figure 3.5: Instantaneous concentration field computed and visualized by GRAMM-GRAL.

3.2 Bayesian Inference of the Source Parameters

3.2.1 Background

Source detection is not an easy task as many processes are not directly reversible. In our algorithm, we intend to detect the source of air pollution and characterize it in a Bayesian reference. This framework will use Baye's rule to model the parameters of our source as random variables as discussed in Equation 2.5 subsection 2.3.2. These random variables will have P(M) as their prior probability distribution. In addition, the likelihood probability $P(D|M)$ will be used to have an idea about the discrepancies between the modeled and observed data. The product of these two probability distributions represent the posterior probability distribution $P(M|D)$ that will imply how probable is for a certain configuration of source parameters to be the true reference one.

3.2.2 The Prior Probability Distribution

The prior probability is expressed as the term P(M) in Equation 2.7. The distribution of this probability represents our prior knowledge about the event M. In our specific problem, the event M represents having a certain set of parameters describing the emission process in our model. These parameters include source location (latitude and longitude or x and y coordinates in UTM system), stack height, emission rate and emission duration.

Since, the event M is mainly composed of having five different parameters; we can say that M is the intersection of five different events (Equation 3.1).

$$M = M_1 \cap M_2 \cap M_3 \cap M_4 \cap M_5 \tag{3.1}$$

where M_i represents the event of having the parameter i in our emission process.

In addition, we assume that the events M_i are independent which allows us to express the prior probability as a product of the probabilities of having these events as shown in Equation 3.2.

$$P(M) = P(M_1 \cap M_2 \cap M_3 \cap M_4 \cap M_5) = \prod_{i=1}^{5} P(M_i) \tag{3.2}$$

In order to preserve the generality of the problem, it is better to use uninformative prior distributions. This reflects our limited knowledge about the proposals conveyed in event M. Based on that, we will use a uniform prior distribution over pre-defined boundaries as shown in Equation 3.3.

Parameter	Minimum Bound	Maximum Bound
X UTM Coordinate (in meters)	504600	521760
Y UTM Coordinate (in meters)	2459000	2475320
Stack Height (in meters)	2	80
Emission Rate (in kg/hr)	10	500
Emission Duration (in minutes)	0	360

Table 3.1: Bounds of the uniform prior probability of the five studied parameters.

$$P(M_i) = \begin{cases} \frac{1}{x_{M_i} - y_{M_i}} & \text{if } M_i \in [x_{M_i}; y_{M_i}] \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

The bounds of the prior distribution of each parameter are represented in Table 3.1. Since the value of $P(M_i)$ is always a constant, the prior probability will also be a constant and Equation 2.7 reduces to:

$$P(M|D) \propto P(D|M) \tag{3.4}$$

This implies that, in our case, the posterior probability and the likelihood probability have directly proportional distributions.

3.2.3 Observations

In order to infer the different parameters responsible for a certain release, we first need to obtain observations resulting from that release. The observations can be obtained in different forms by using different techniques. These first form of observation is a concentration field that gives an average concentration value at the center of each grid cell of our uniform rectilinear domain. Such fields can be obtained either from well established and rich sensing networks covering wide regions or from images and remote sensing techniques. Another form of observations is concentration contours that delineate the polluted regions without any indication of the concentration level at the indicated position. In addition, in locations where complete sensing networks are hard to install or operate, we may have a finite number of installed sensors. Thus, we can have an observation represented as concentration values at different scattered positions inside the domain. In our case, we will use the first form of observations resulting from synthetic data coming from GRAL model as described below:

Concentration Fields (denoted as F): This is a Cartesian grid having the same size and resolution of the GRAL domain grid. It contains the value of concentration of a certain pollutant at the center of each grid cell. The GRAL model allows us to extract 2D concentration grids from the original 3D Cartesian grid of the whole domain. The 2D extracted grids are obtained at selected elevations above the ground. These grids are written in the form of ASCII files.

3.2.4 Quantifying the Discrepancies

The inference process is an iterative process that predicts a new concentration distribution at each iteration. It is now necessary to quantify the discrepancies between every newly obtained model output and our reference observation. The choice of a suitable metric to measure these discrepancies depends on the nature of observations at hand. In our case, the below metric is a global metrics used in case of concentration fields.

Wasserstein Distance: As defined in section 2.4.2, it is the distance used to calculate the distance between two different distributions. It is obtained by minimizing the path required to translate one distribution to the other. We will be using this metric in order to get a measure of the discrepancies between two fields, each represented in a concentration grid as discussed in section 3.2.3. The dissimilarity is globally measured by using field displacement where each concentration grid is considered to be a density that can be displaced. This metric can overcome the double penalty effect of a local discrepancy estimator and was efficiently used to compare fields of accidentally released radioactive elements in the Fukushima-Daiichi accident [64].

This global metric is expressed by Equation 3.5

$$\delta_i = \delta(\lambda, \lambda_i) \tag{3.5}$$

where λ is the reference observation and λ_i the i^{th} output of the model. When an observation λ is a concentration field, δ represents the Wasserstein distance.

3.2.5 The Likelihood Probability

As mentioned in section 2.3.2, the likelihood probability $P(D|M)$ is the probability of having a certain set of data D given a certain combination of model parameters as defined by M. In other words, it is expresses how likely does the model predict the observation at hand given a certain set M of parameters. So, an indication about this probability needs a tool for the quantification of discrepancies between the model output and the original reference observation.

In this scope, for an observation, we will use the global metric discussed in section 3.2.4 in correspondence with the type of observation at hand and as expressed in Equation 3.5. The metric δ_i will decrease as the discrepancies between the two fields λ and λ_i decrease; in other words, as these two fields become similar. In terms of likelihood probability, less discrepancies mean higher probability for the field λ_i at hand to similar to λ. Hence, the likelihood probability distribution should increase as the value of the metric decreases and vice versa. Based on that, we will set our likelihood distribution to an exponential distribution as defined by Equation 3.6:

$$P(D|M)_{\delta_i} = \frac{1}{\beta} exp^{\frac{-\delta_i}{\beta}} \tag{3.6}$$

where β is the scale parameter that we assume to be constant and known. The value of β was selected to maximize the likelihood between the reference observation and the model output obtained by having its input parameters very close to the true ones. This distribution is defined for positive values of δ_i, which is always true for our metrics. In addition, its value approaches 1 as the value of δ_i goes to 0 and it approaches 0 as δ_i gets greater.

The likelihood probability represented in Equation 3.6 is used when we have only one reference observation. However, when we have n_C reference observations, we will assume that these are independent from one another and their likelihood probability will be expressed in Equation 3.7:

$$P(D|M)_{\delta_i} = \prod_{i=1}^{n_C} \frac{1}{\beta} exp^{\frac{-\delta_i}{\beta}} \tag{3.7}$$

3.2.6 Sampling the Posterior

To sample the posterior, we will use the Metropolis Hastings algorithm which was described in section 2.3.2. The steps used in our case can be described as follows:

1. Initialize the starting point s^0 by randomly sampling a value for each inferred parameter while respecting the pre-defined boundaries of our parameters' space. In our case, s^0 is a multi-dimensional vector having as components the sampled values of the inferred parameters (Equation 3.8).

$$s^n = s_i^n; i = 1, 2, ..., m \tag{3.8}$$

where m is the number of inferred parameters.

2. Start the iterative process.

a) Suppose that the current sample is s^n, sample a new candidate point s^{n+1} from a normal distribution centered at s^n.

$$s^{n+1} \sim N(s^n, \sigma) \tag{3.9}$$

b) Run GRAL model for the sampled set of source parameters s^{n+1} to obtain its resulting concentration pattern.

c) Calculate the likelihood probability $P(D|M)$ by measuring the distance between reference observation and the concentration pattern obtained from the sample of source parameters, s^{n+1}. This will give us an indication about the posterior probability $P(M|D)$.

d) Calculate the acceptance probability given by:

$$a = min(1, \frac{P(M_n|D)}{P(M_{n+1}|D)}) \tag{3.10}$$

e) Draw a random number α from the uniform distribution over the interval [0,1].

$$\alpha \sim U(0, 1) \tag{3.11}$$

f) Compare a and α to update the Markov chain accordingly. In the below, we denote by c^{n+1} the new element of the Markov chain.

$$c^{n+1} = \begin{cases} s^{n+1} & \text{if } \alpha < \text{a} \\ s^n & \text{otherwise} \end{cases} \tag{3.12}$$

Chapter 4

Numerical Experiments

In this section, we will define the implementation of a smart algorithm conveying the previous methodology. We will then represent the different numerical experiments conducted using this smart algorithm in a variety of experimental settings.

4.1 Smart Algorithm

The prior bounds represented in Table 3.1 state that the area of the domain containing the investigated probable location of the emission source is 17160m along the x direction and 16320m along the y direction. This means that we should run GRAL over a domain having an identical area as the one stated here. However, the challenge rising at this level is the high computational cost represented by the relatively long computational time required by GRAL to perform computations over such a domain. This time is estimated to be in the order of 60 seconds. In this scope, running GRAL model over a domain with a smaller area would definitely reduce the required computational time; hence, making our algorithm more feasible.

Moreover, based on some prior knowledge, our emission source lies in a smaller domain D1 comprising a subset of the original prior bounds denoted as D2. These two domains are represented in Figure 4.1.The properties of these two domain are represented in Table 4.1.

Having that said, we propose a smart algorithm that will utilize this prior knowledge in a computationally efficient manner. The smart algorithm works as follows:

1) Localize the proposed sample: Check whether the proposed sample at an iteration i lies inside D1 or outside D1.

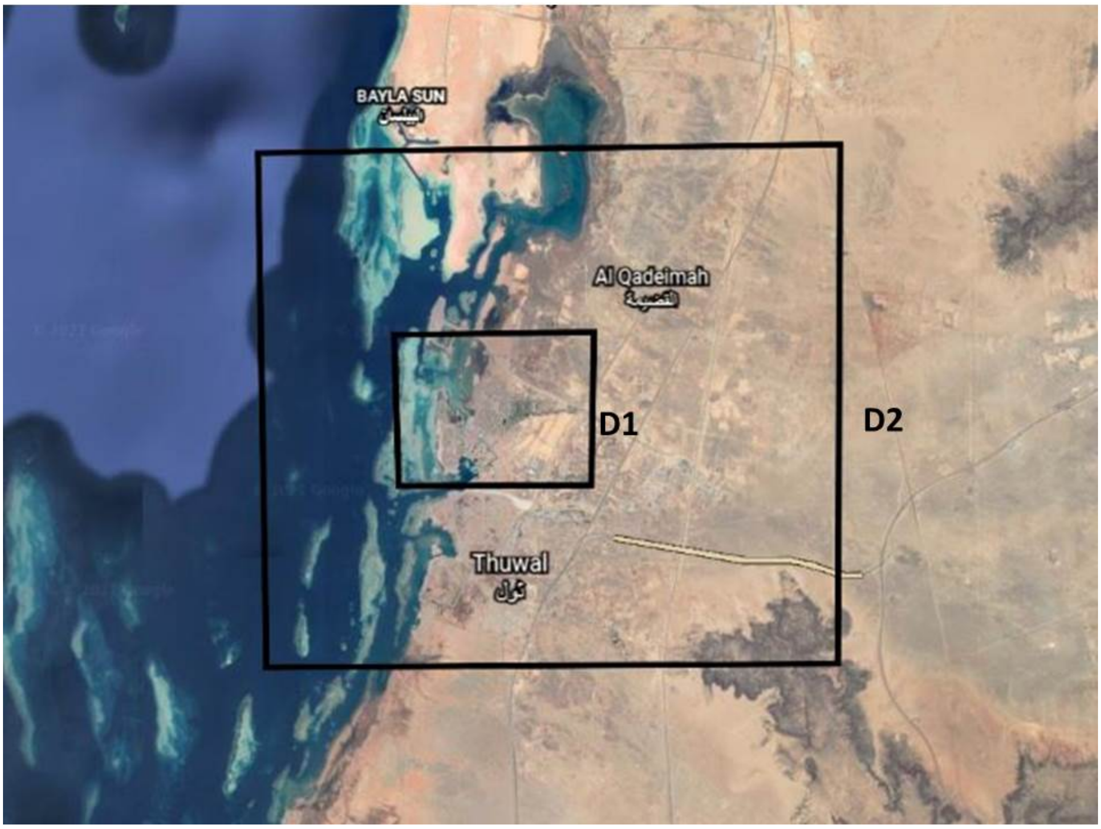


Figure 4.1: The geographic locations of the two domains D1 and D2 used by the smart algorithm.

Parameter	Domain D1	Domain D2
Minimum X UTM Coordinate (in meters)	508920	504600
Maximum X UTM Coordinate (in meters)	514680	521760
Minimum Y UTM Coordinate (in meters)	2464840	2459000
Maximum Y UTM Coordinate (in meters)	2470180	2475320
Area (in km^2)	31	280
GRAL run time (in seconds)	20	60

Table 4.1: Properties of the domains used in the smart algorithm.

2) Accordingly, decide where to run GRAL model:
- If the sample lies inside D1, run GRAL model over D1.
- If the sample lies outside D1, run GRAL model over D2.

After discovering the properties of the two domains D1 and D2, the smart algorithm is expected to make the framework more feasible, especially as our iterations come closer to a steady state. In such case, the bulk of the proposed samples will lie inside D1; hence, only 20 seconds will be required to run the GRAL domain.

4.2 Experimental Settings

In order to infer for the emission parameters, several numerical experiments will be conducted based on the concepts presented in Sections 3.1 and 3.2 and by using the smart algorithm proposed in Section 4.1. The experimental scenario assumes that a point source is located at x=513295m and y=2467205m in the UTM coordinate system. This source has a stack height of z=10 meters above the sea level and emits for d=240 minutes at an emission rate of q=100 kg/hr. To mimic real observations, all our synthetic observations are perturbed by an observational error of the form of an unbiased normal distribution having a standard deviation 500 $\mu.g/m^3$ which is around the mean concentration of the reference observation.

Our numerical experiments are designed in order to utilize one or more forms of observations that are explained in section 3.2.3. In each experiment and as described in section 3.2.4, a corresponding metric that suits the nature of the observation at hand is used to quantify the discrepancies between this reference observation and the predicted concentration pattern at each iteration. The same selectivity is applied at the level of the used likelihood expression out of those listed in section 3.2.5. Moreover, each experiment will be oriented towards inference of a certain set of source parameters. The numerical experiments carried out are listed below in Table 4.2.

Experiments 1,2 and 3 use a concentration field as its reference observation. This synthetic field is obtained above the urban environment represented in D1 (Figure 4.1) at an elevation of 10 meters above the ground and at a time instant T=360 minutes, that is two hours after the emission stops. This observation is represented below in Figure 4.2. The inference towards a different set of parameters is done in each experiment. At this level, we introduce an additional parameter m representing the mass of the release in kg. In fact, this parameter is the product of two of our main five parameters, the emission rate q (in kg/hr)

	Experiment	Type of	Used Metric	Likelihood	Parameters
	Observation		Expression	Inverting for	
Set I	1	F	Wasserstein distance	Equation 3.6	x,y,h
	2	F	Wasserstein distance	Equation 3.6	x,y,h,m
	3	F	Wasserstein distance	Equation 3.6	x,y,h,q,d
Set II	4	F	Wasserstein distance	Equation 3.7	x,y,h
	5	F	Wasserstein distance	Equation 3.7	x,y,h,m
	6	F	Wasserstein distance	Equation 3.7	x,y,h,q,d

Table 4.2: Settings of the numerical experiments conducted using the smart algorithm. The information listed in the second and fourth column are in accordance with sections 3.2.3 and 3.2.5 respectively.

and the emission duration d (in hr).

Experiments 4, 5 and 6 use observations having the same nature, a concentration field. More specifically, these experiments use two synthetic observations obtained at two different elevations of 10 meters and 25 meters above the ground at a time instant T=360 minutes. These two observations are represented below in Figure 4.3. The main difference between these experiments is the set of source parameters that we are inverting for. Experiment 4 inverts for the location of the point source expressed as x and y coordinates and the stack height h. Moreover, Experiment 5 inverts for the location of the point source as well as its strength represented by the mass of the release m (in kg). Then, Experiment 6 inverts for the location of the point source in addition to the emission rate, q and the emission duration, d.

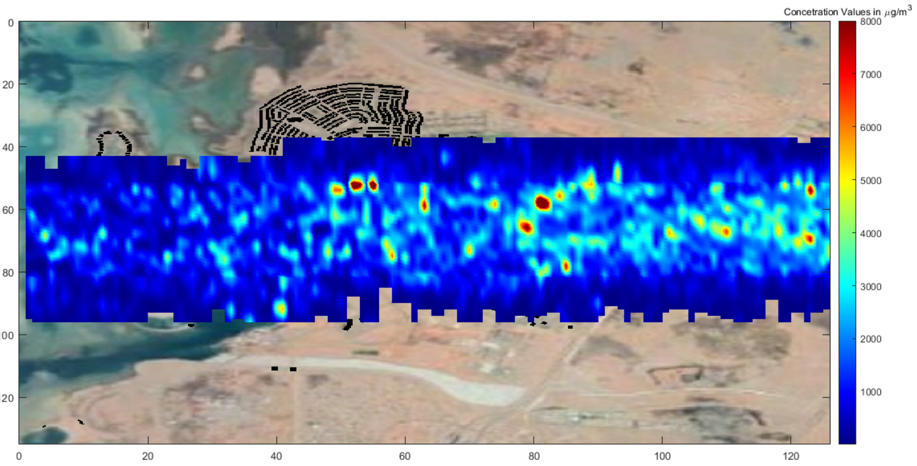


Figure 4.2: Concentration field used as reference observations for Set I collected at an elevation of 10 meters above the ground.

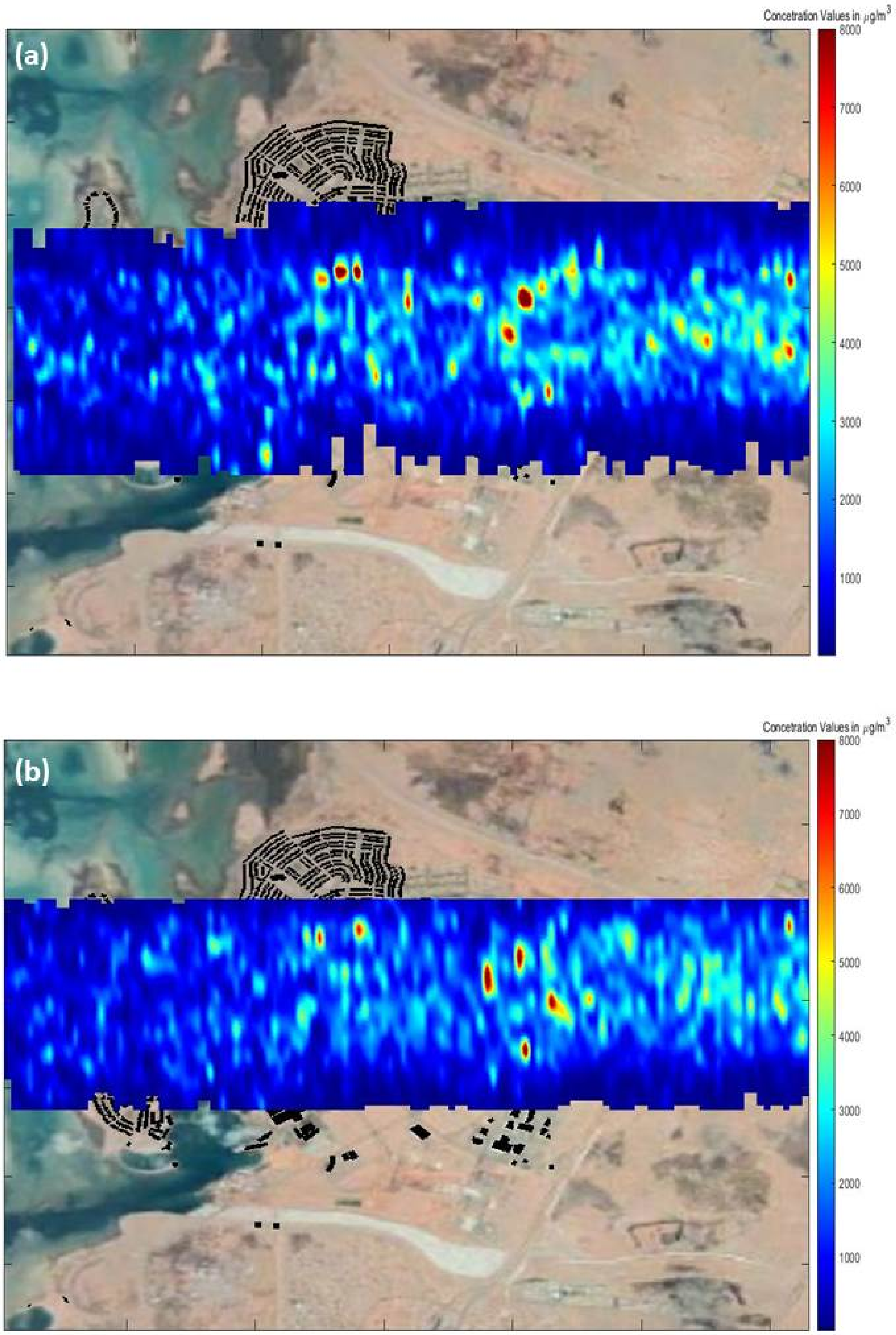


Figure 4.3: Concentration fields used as reference observations for Set II collected at an elevation of a) 10 meters above the ground b) 25 meters above the ground.

Chapter 5

Results and Discussion

This section is dedicated to show the results of the different experiments listed in section 4.2 and to analyze the performance of our proposed smart algorithm. Each subsection reveals the obtained Markov chain in each of the experiments. Moreover, the posterior probability distributions were extracted from these by using a kernel density estimator and their results are shown in a corner plot. In this corner plot, the off-diagonal subplots represent the joint posterior probability of each two corresponding parameters. In addition, the diagonal subplots show the posterior probability distribution of each parameter. The performance of the smart algorithm is analyzed by referring to the percentage of samples that lie inside D1 every 1,000 iterations.

5.1 Set I

5.1.1 Experiment 1

The smart algorithm was run over 20,000 iterations based on the experimental settings of experiment 1 in Table 4.2. The scale parameter of the likelihood expression, β, is set to 50. The trace plot of the Markov chain obtained from each of the three inferred parameters is shown in Figure 5.1 with an acceptance rate of 32%. The kernel density estimator resulted in a 3 dimensional matrix that was processed by 2D and 1D slicing to obtain the corresponding posterior probability distributions shown in the corner plot of Figure 5.2. Moreover, the step plot of the percentage of samples lying inside D1 is represented in Figure 5.3.

We notice that the our algorithm successfully inverted for the location of the emission source represented by its x and y coordinates. That is by which we obtained a narrow posterior distribution with its maximum being at the reference values of the two parameters x and y. At the level of h, we obtained a flat distribution over the range between 0 and 40 meters with several similar maximal values at heights that are away from our reference value. Regarding the smart algorithm, a gradual increase in the percentage of samples inside D1 occurs until reaching a plateau after 12,000 iteration. This plateau has a maximum of 69% indicating that the majority of our samples lies in D1, around the reference location of the emission source. The reduction of time achieved by the efficiency smart algorithm is shown in Table 5.1. It is worth noting that in addition to the time required by GRAL computations, an extra 20 seconds are needed at each iteration in order to compute the Wasserstein distance as discussed in section 3.2.4.

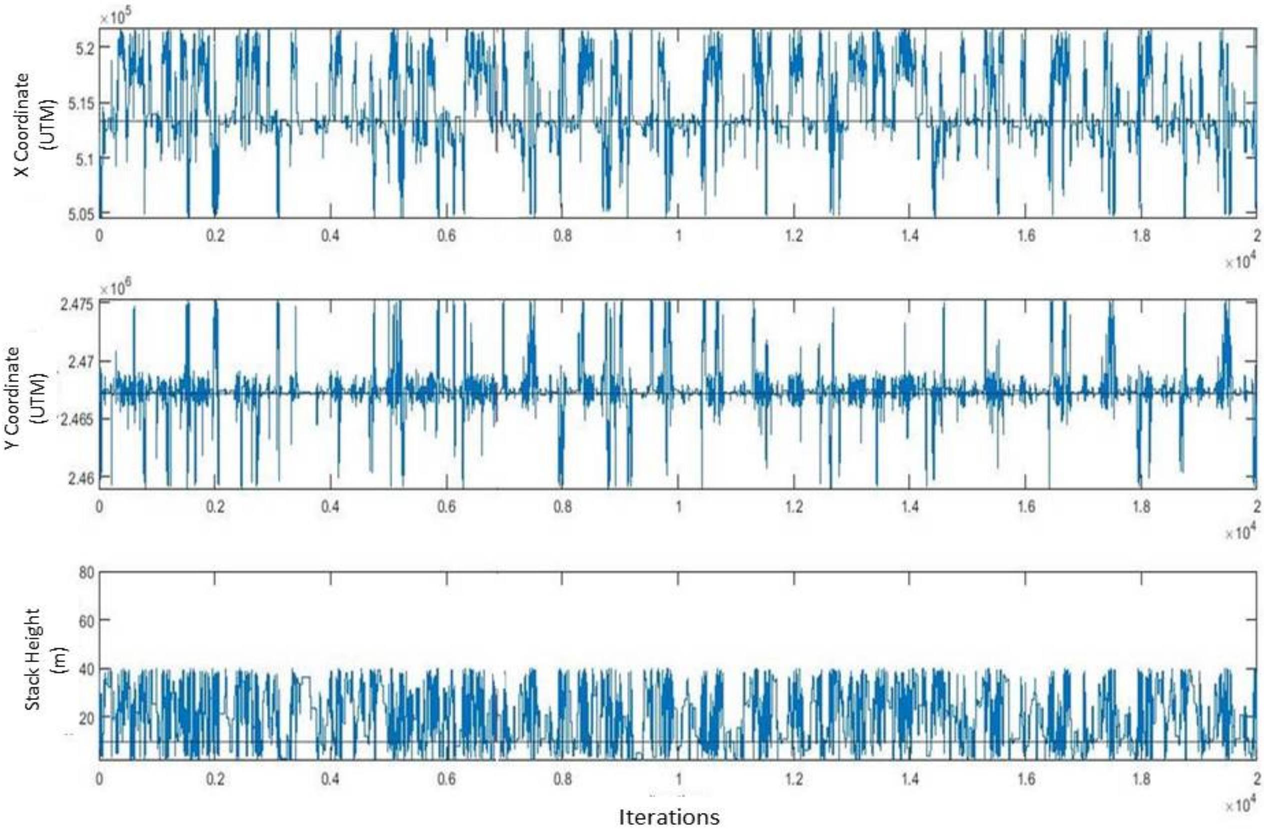


Figure 5.1: Markov chains obtained in Experiment 1. The horizontal black lines refer to the reference value of each parameter.

5.1.2 Experiment 2

The smart algorithm was run over 20,000 iterations based on the experimental settings of experiment 2 in Table 4.2. Similarly, the scale parameter of the like-

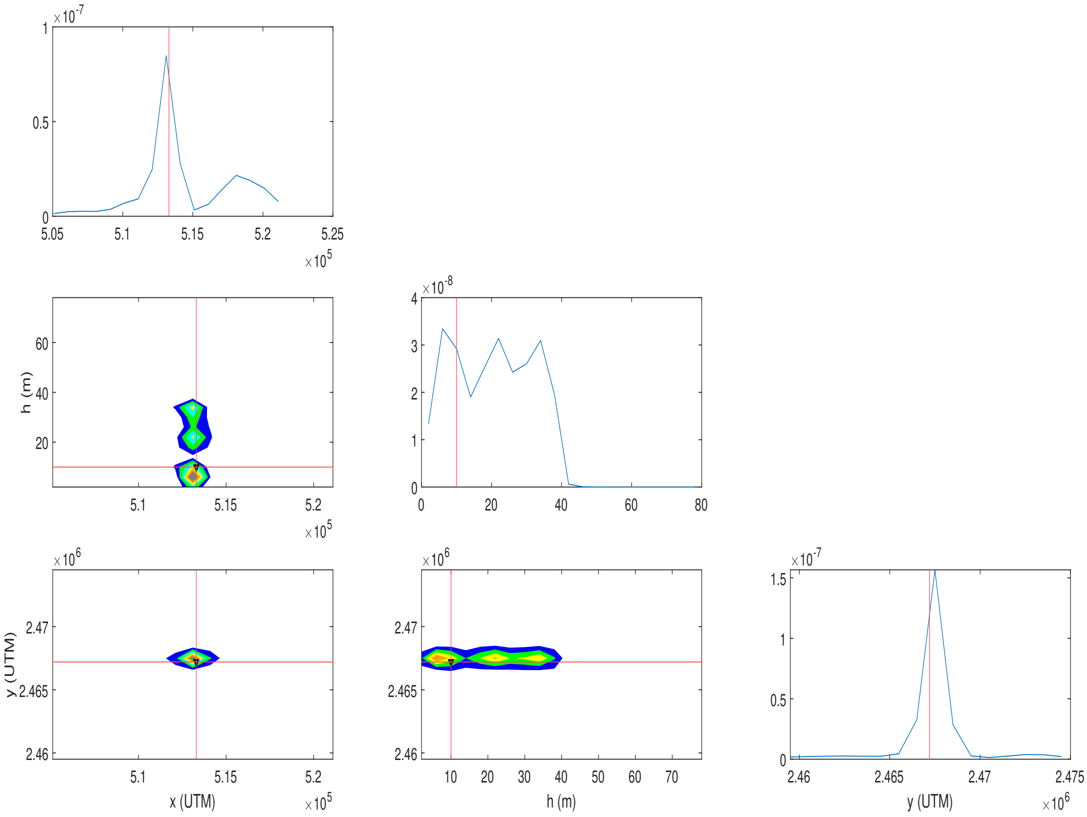


Figure 5.2: Corner plot of the obtained posterior probability distributions in experiment 1.The red lines represent the reference value of each of the inferred parameters.

	Inside D1	Outside D1
Time required per iteration (in seconds)	40	80
Average Percentage of samples in each domain	51.2	48.8
Original Time required (in hours)	444.4	
Time required by smart algorithm (in hours)	330.68	
Percentage reduction in time	25.6	

Table 5.1: Analysis of the computational time of the smart algorithm in Experiment 1.

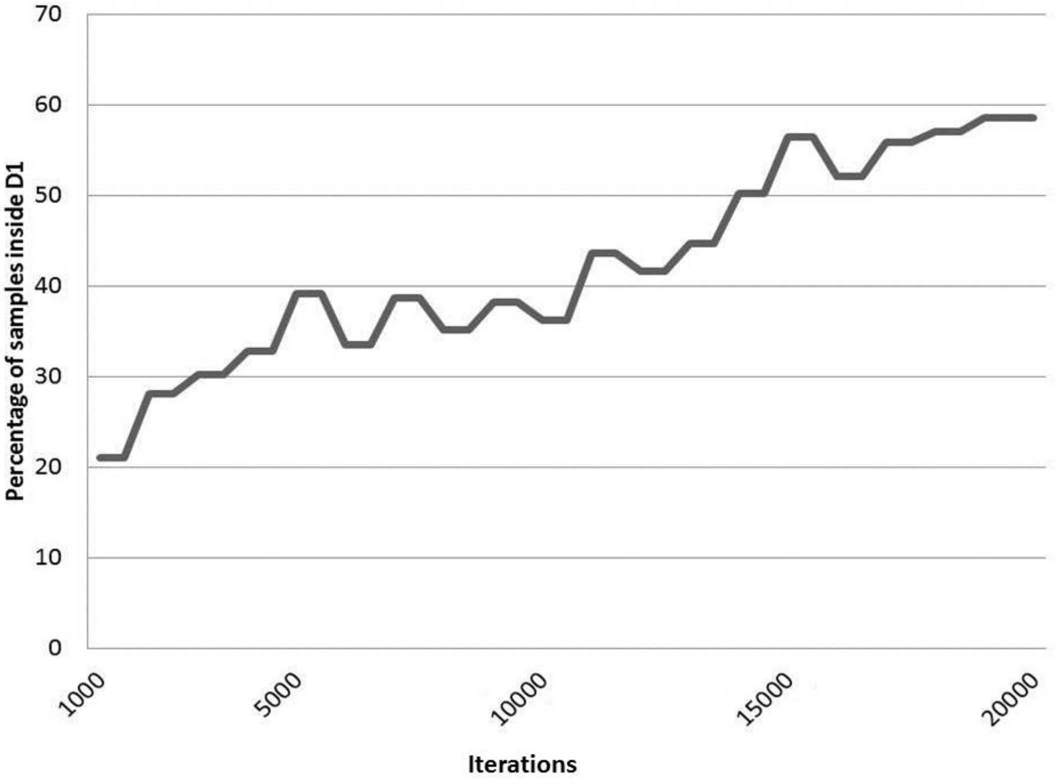


Figure 5.3: Step plot of the percentage of samples lying inside D1 in experiment 1.

	Inside D1	Outside D1
Time required per iteration (in seconds)	40	80
Average Percentage of samples in each domain	42.2	57.8
Original Time required (in hours)	444.4	
Time required by smart algorithm (in hours)	350.67	
Percentage reduction in time	21.1	

Table 5.2: Analysis of the computational time of the smart algorithm in Experiment 2.

lihood expression, β, is set to 50. In this experiment, we are inverting of the location of the source as well as the emission strength. The location is defined by the three parameters inverted for in Experiment 1 and the strength is defined by the mass of the release, m. The trace plot of Markov chain obtained from each of the four parameters is shown in Figure 5.4 with an acceptance rate of 36.58%. The kernel density estimator resulted in a 4 dimensional matrix that was processed by 2D and 1D slicing to obtain the corresponding posterior probability distributions shown in the corner plot of Figure 5.5. At the level of the smart algorithm, the step plot of the percentage of samples lying inside D1 is represented in Figure 5.6.

Our algorithm successfully inverted for the location of the emission source represented by its x and y coordinates, as well as for the released mass m. That by which we obtained a posterior distribution with its maximum being at the reference values of the three parameters x, y and m. At the level of h, we obtained a flat distribution over the prior bounds between 0 and 80 meters, but with its maximum being deviated from 10 meters, our reference value. Regarding the smart algorithm, a gradual increase in the percentage of samples inside D1 occurs until reaching a steady state after 15,000 iteration with a maximum of 59%. The improvement in computational time accomplished by the smart algorithm is shown in Table 5.2.

5.1.3 Experiment 3

In this experiment, we inverted for each of the five parameters under inspection by running the smart algorithm over 20,000 iterations. The scale parameter of the likelihood expression, β, is set to 50. The source location is similarly defined by x, y and h, whereas, the emission strength is now defined by the emission rate q and the emission duration d. Figure 5.7 represents the trace plot of the Markov chain obtained with an acceptance rate of 34%. Similarly, the kernel density estimator gave us a 5 dimensional matrix to which we applied 2D and 1D slicing. The resulting corner plot of the posterior probability distributions is represented in Figure 5.8. A step plot of the percentage of samples inside D1 is shown in

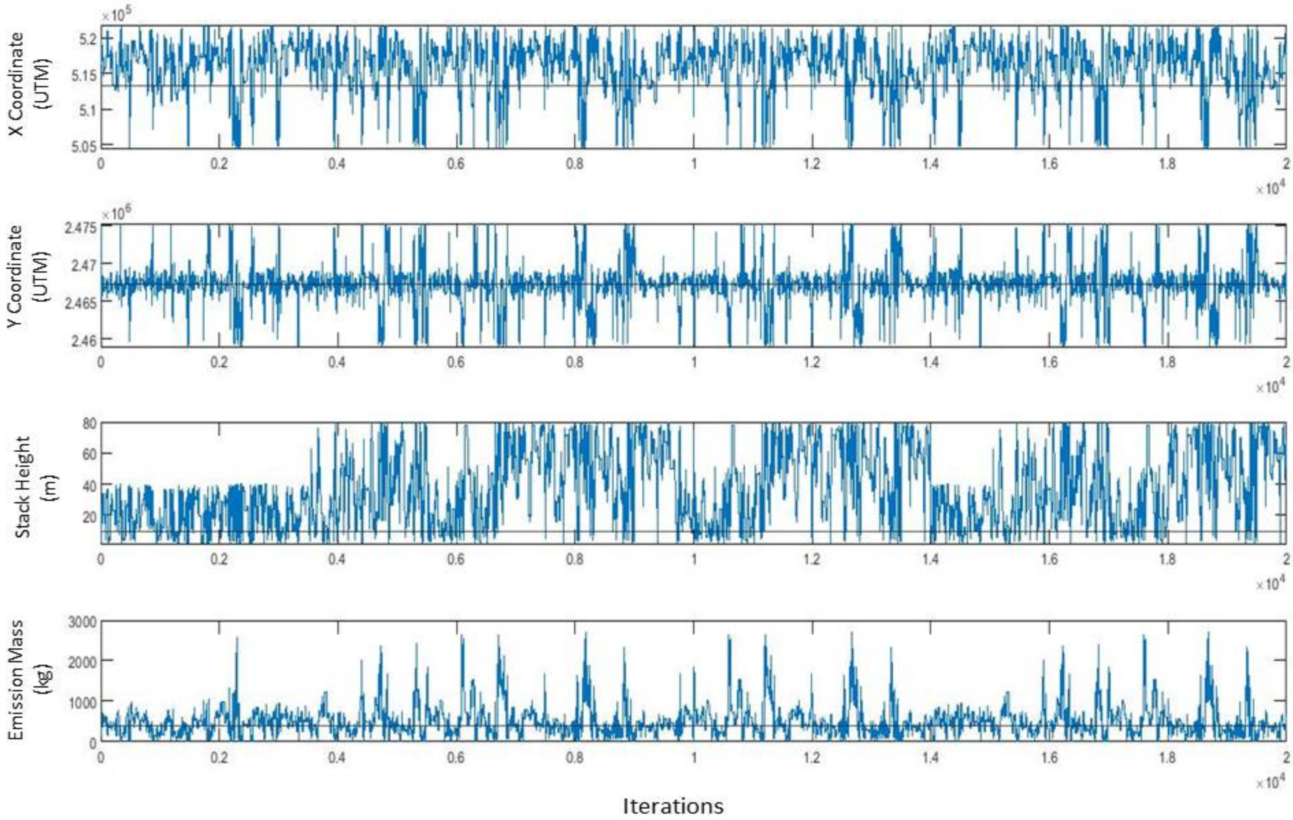


Figure 5.4: Markov chains obtained in Experiment 2. The horizontal black lines refer to the reference value of each parameter.

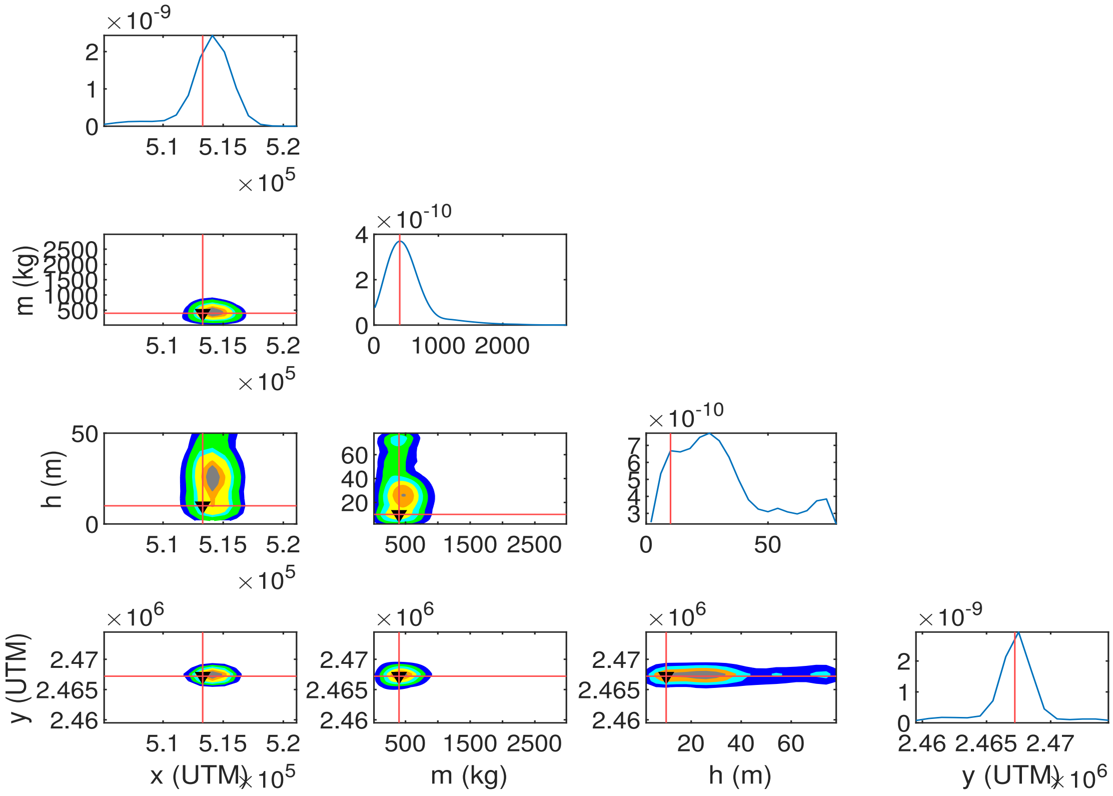


Figure 5.5: Corner plot of the obtained posterior probability distributions in experiment 2.The red lines represent the reference value of each of the inferred parameters.

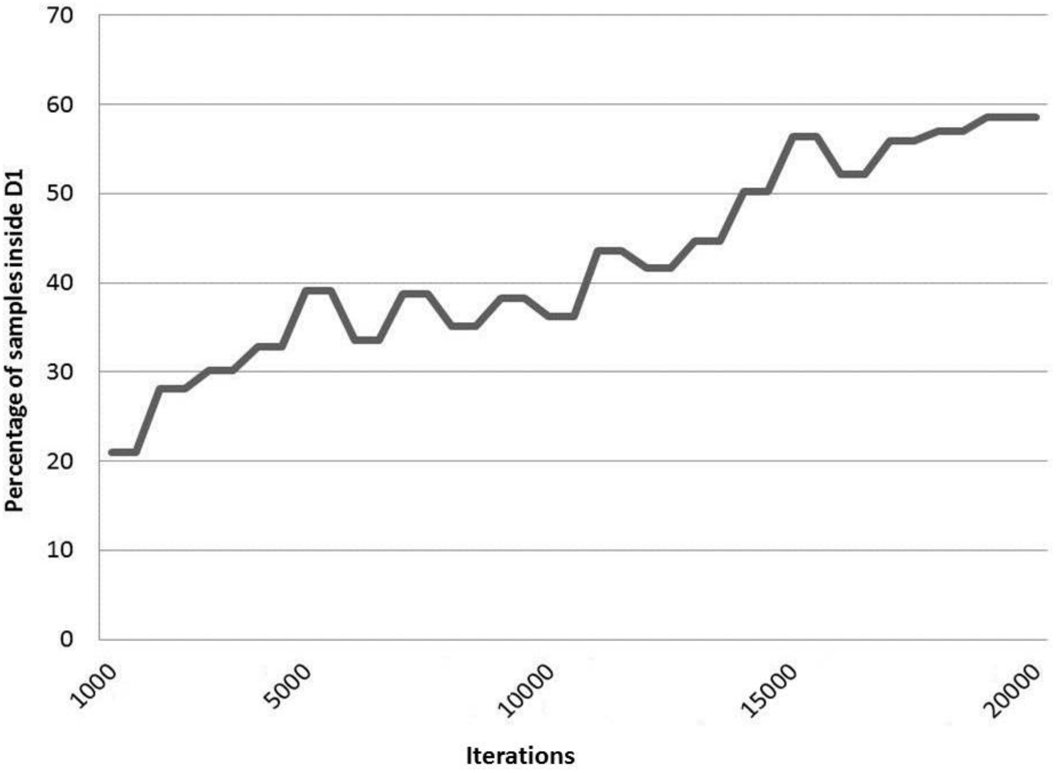


Figure 5.6: Step plot of the percentage of samples lying inside D1 in experiment 2.

	Inside D1	Outside D1
Time required per iteration (in seconds)	40	80
Average Percentage of samples in each domain	35.5	64.5
Original Time required (in hours)	444.4	
Time required by smart algorithm (in hours)	365.55	
Percentage reduction in time	18	

Table 5.3: Analysis of the computational time of the smart algorithm in Experiment 3.

Figure 5.9. In addition, the effect of the smart algorithm on computational time is represented in Table 5.3.

Just like the previous experiments, we got a good prediction of the source location in the 2D plane at the level of the two parameters x and y. The posterior probability of the stack height h remains to be a flat distribution with its maximum being over the range between 0 and 40 meters. At the level of the emission strength, presented by q and d, we notice that both posterior probabilities of these parameters have flat distributions over their bounds. Moreover, at the level of q, this distribution has its maximum value around the reference value unlike q whose posterior probability has its maximum deviated from the reference value. Moving to analyze the smart algorithm performance, we notice that the percentage of samples inside D1 increases gradually until reaching a steady state after around 13,000 iterations. This steady state is in the range between 40% and 50% with its maximum being 49%. This resulted in 18% reduction of the required computational time compared to the original required time.

5.1.4 Discussion of Results in Set I

After presenting the different results obtained in each of the first three experiments, we can say that our algorithm is accurately predicting the location of the emission source in the 2D plane; that is a good prediction of x and y. The third parameter defining the source location, h, is always having a flat posterior probability distribution. So, we have a weak indication about the reference value of the stack height. Coming to the emission strength studied in experiments 2 and 3, apparently the emission mass m was predicted in a better manner than its elementary components q and d. That is the bulk of accepted samples in the corresponding experiments may have their values of q and d away from the reference ones. However, their resulting product lies around the reference value of m.

On the other hand, the smart algorithm made our framework more computationally feasible. The efficiency of this algorithm increases as the number of

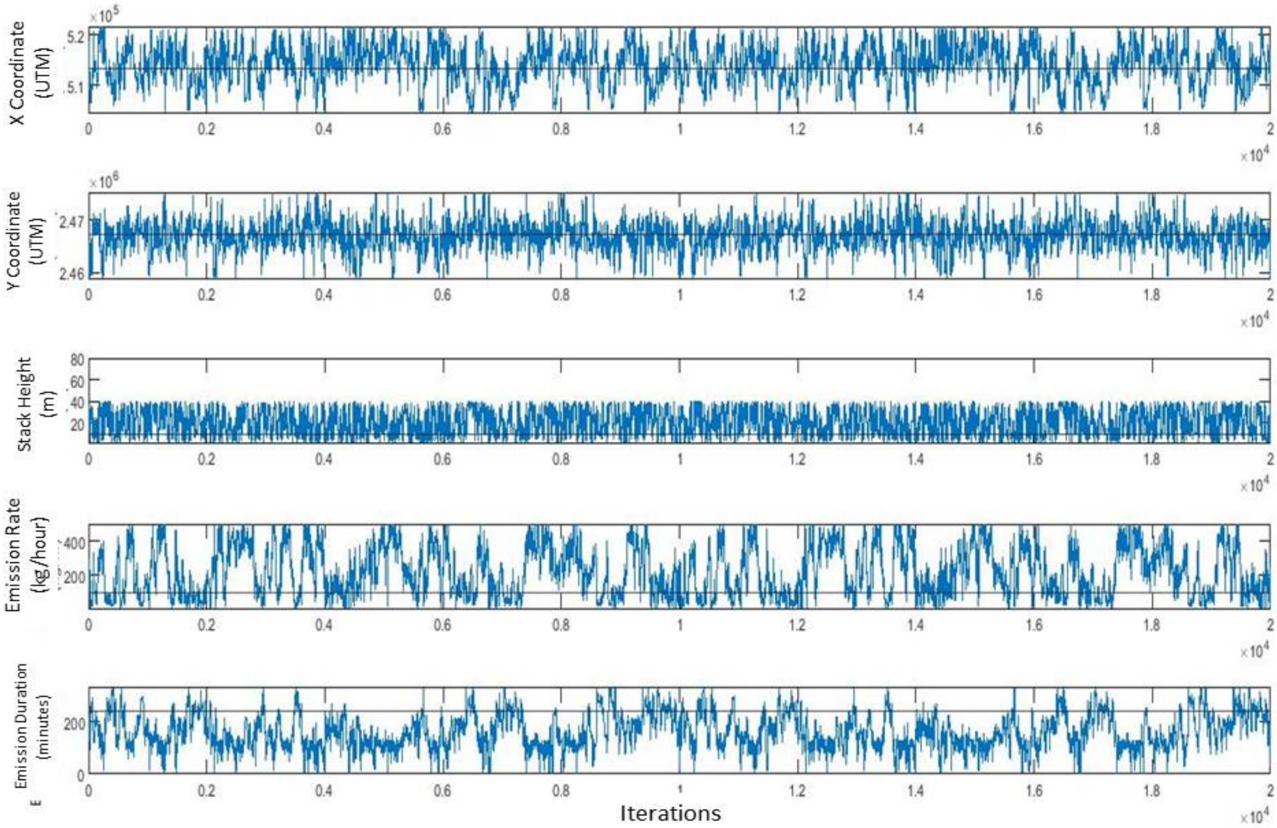


Figure 5.7: Markov chains obtained in Experiment 3. The horizontal black lines refer to the reference value of each parameter.

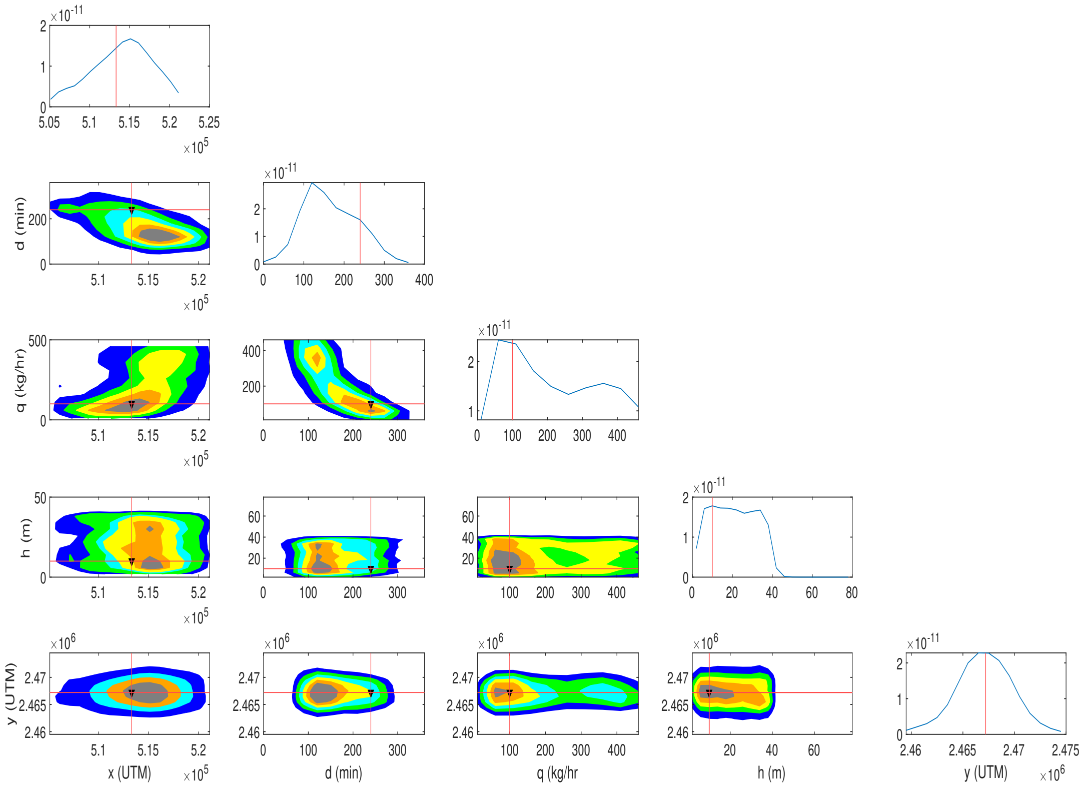


Figure 5.8: Corner plot of the obtained posterior probability distributions in experiment 3.The red lines represent the reference value of each of the inferred parameters.

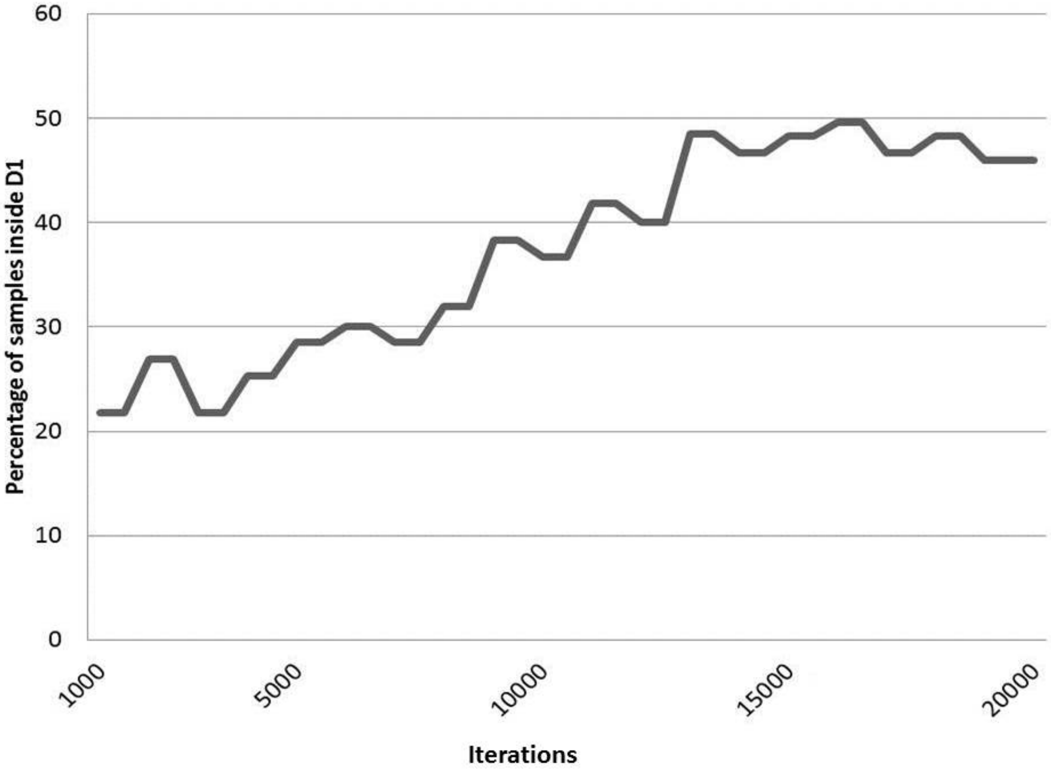


Figure 5.9: Step plot of the percentage of samples lying inside D1 in experiment 3.

iterations increases reaching a steady state which gives us an indication about a similar quasi-steady state of convergence of the corresponding Markov chains.

In order to improve the performance of our algorithm, we will try to obtain better predictions of all studied parameters and particularly of the emission stack height h by using observations F at different elevations.

5.2 Set II

5.2.1 Experiment 4

The smart algorithm was run over 20,000 iterations based on the experimental settings of experiment 4 in Table 4.2. The scale parameter of the likelihood expression, β, is set to 50. The trace plot of the Markov chain obtained from each of the three inferred parameters is shown in Figure 5.10 with an acceptance rate of 33.72%. The kernel density estimator was similarly used to post process the results and obtain the corresponding posterior probability distributions shown in the corner plot of Figure 5.11. In addition, a step plot of the percentage of samples lying inside D1 is represented in Figure 5.12.

The inference towards the source location in the 2D plane, represented by the x and y coordinates, resulted in an accurate prediction of the reference values. A narrow posterior distribution with its maximum being at the reference values of these two parameters is obtained. At the level of h, we obtained a flat distribution over the prior bounds between 0 and 50 meters with its maximum being at 14 meters, slightly away from our reference value. The smart algorithm resulted in a gradual increase in the percentage of samples inside D1 occurs until reaching a plateau after 15,000 iterations with its highest values in the range between 70% and 80%. Hence, the vast majority of our samples lies in D1, around the reference location of the emission source. The reduction of time achieved by the efficiency smart algorithm is shown in Table 5.4. It is worth noting that in addition to the time required by GRAL computations, an extra 40 seconds are needed at each iteration in order to compute the Wasserstein distance as discussed in section 3.2.4 at the level of each of the observations.

5.2.2 Experiment 5

Based on the experimental settings of experiment 5 in Table 4.2, the smart algorithm was run over 30,000 iterations. The scale parameter of the likelihood expression, β, is set to 50. The trace plot of the Markov chain obtained from each of the three inferred parameters is shown in Figure 5.13 with an acceptance rate of 30%. The kernel density estimator resulted in the posterior probability

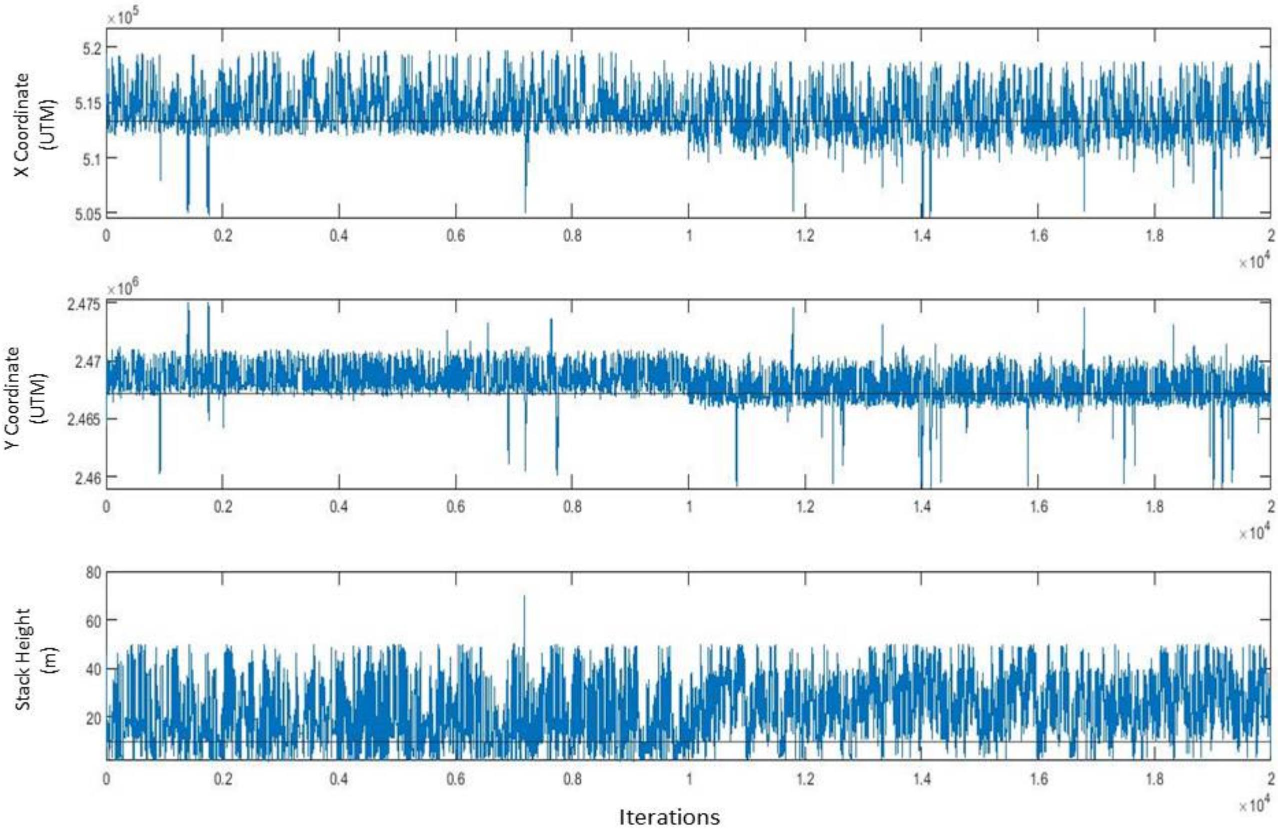


Figure 5.10: Markov chains obtained in Experiment 4. The horizontal black lines refer to the reference value of each parameter.

	Inside D1	Outside D1
Time required per iteration (in seconds)	60	100
Average Percentage of samples in each domain	63	37
Original Time required (in hours)	555.5	
Time required by smart algorithm (in hours)	415.55	
Percentage reduction in time	25.2	

Table 5.4: Analysis of the computational time of the smart algorithm in Experiment 4.

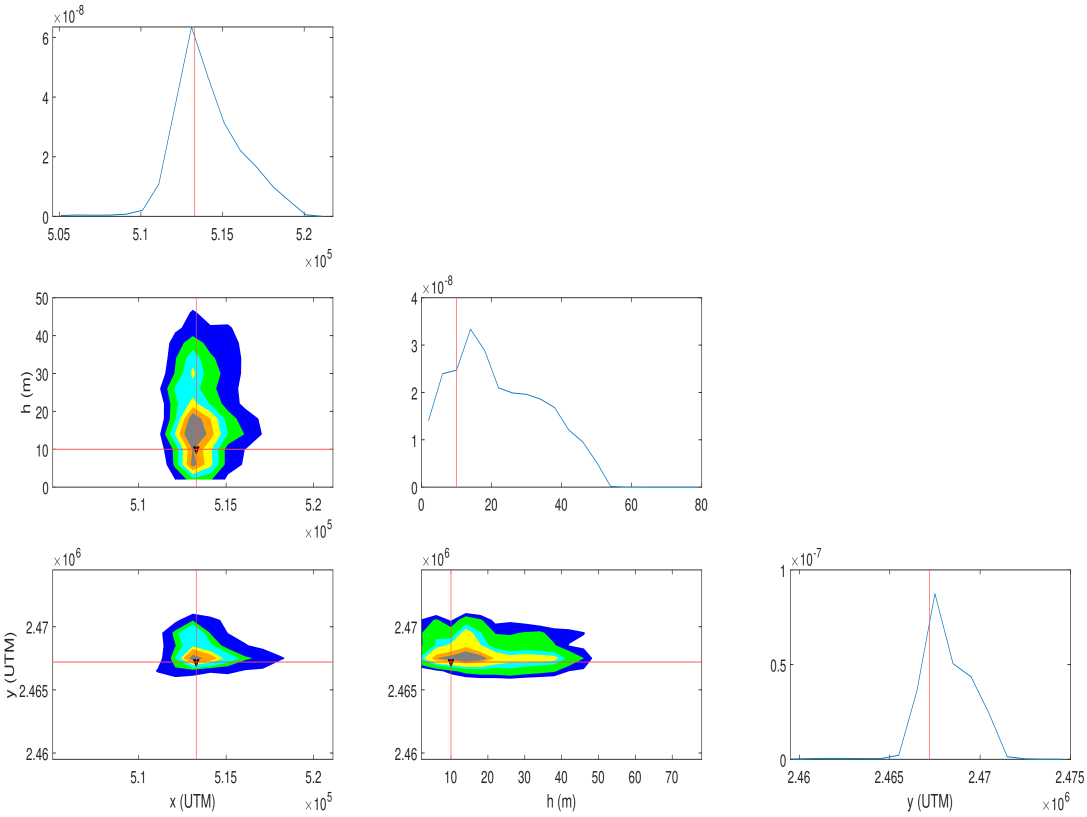


Figure 5.11: Corner plot of the obtained posterior probability distributions in experiment 4.The red lines represent the reference value of each of the inferred parameters.

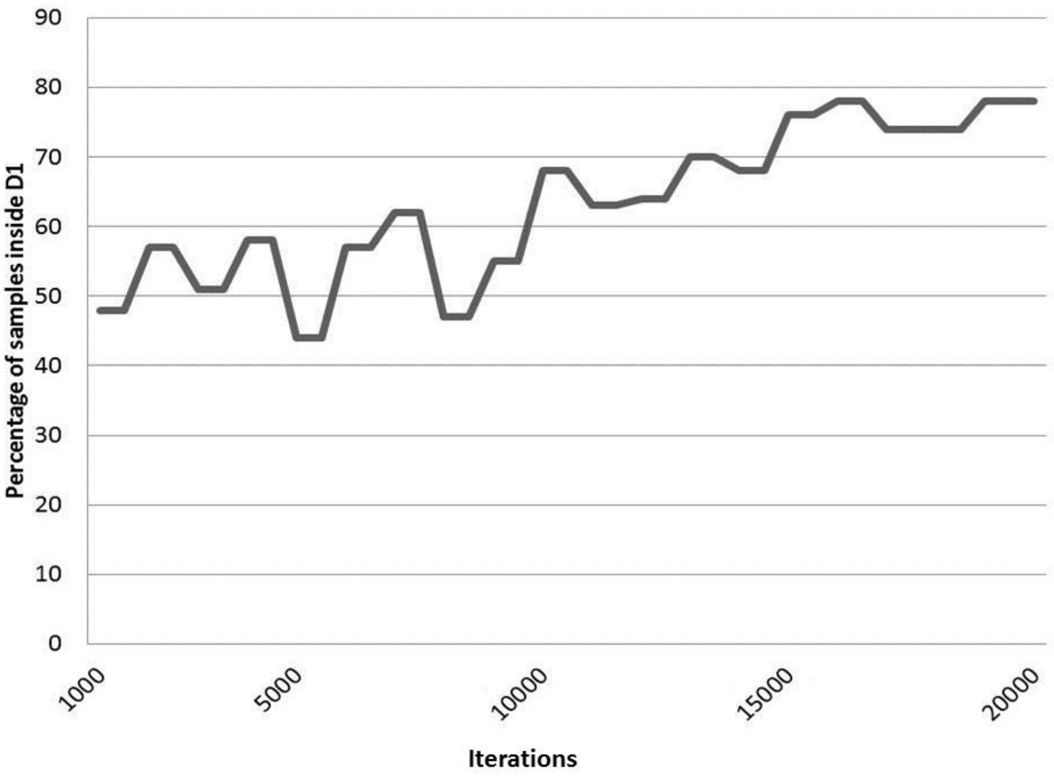


Figure 5.12: Step plot of the percentage of samples lying inside D1 in experiment 4.

distributions shown in the corner plot of Figure 5.14. Moreover, a step plot of the percentage of samples lying inside D1 is represented in Figure 5.15.

The inference towards the four studied parameters, represented by the x, y, h and m, resulted in an accurate prediction of our reference values. All the posterior distributions have their maxima at the reference values of these four parameters. Furthermore, the smart algorithm resulted in a gradual increase in the percentage of samples inside D1 occurs until having a steady state after 12,000 iterations. This steady state lies in the range between 60% and 70% giving an indication that our algorithm has its vast majority of our samples inside D1, around the reference location of the emission source. The reduction of time achieved by the efficiency smart algorithm is shown in Table 5.5.

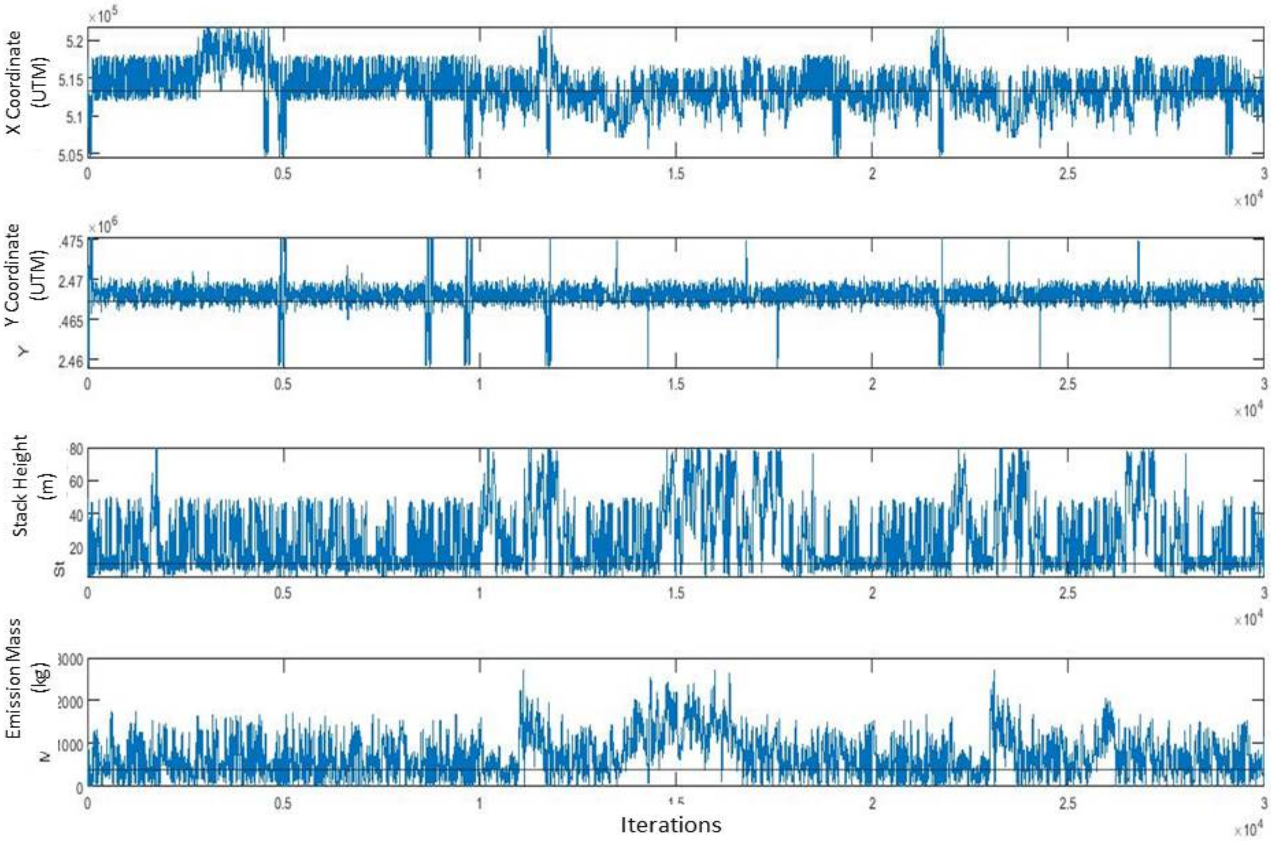


Figure 5.13: Markov chains obtained in Experiment 5. The horizontal black lines refer to the reference value of each parameter.

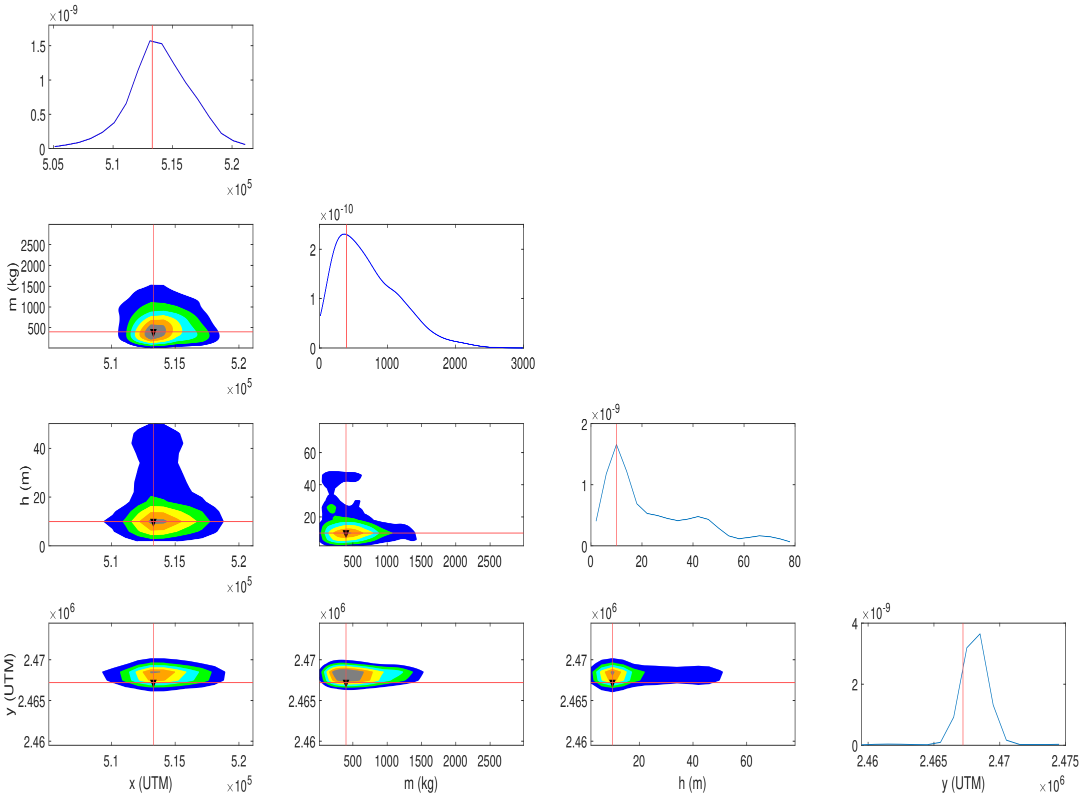


Figure 5.14: Corner plot of the obtained posterior probability distributions in experiment 5.The red lines represent the reference value of each of the inferred parameters.

	Inside D1	Outside D1
Time required per iteration (in seconds)	60	100
Average Percentage of samples in each domain	54.5	45.5
Original Time required (in hours)	833.34	
Time required by smart algorithm (in hours)	651.67	
Percentage reduction in time	21.8	

Table 5.5: Analysis of the computational time of the smart algorithm in Experiment 5.

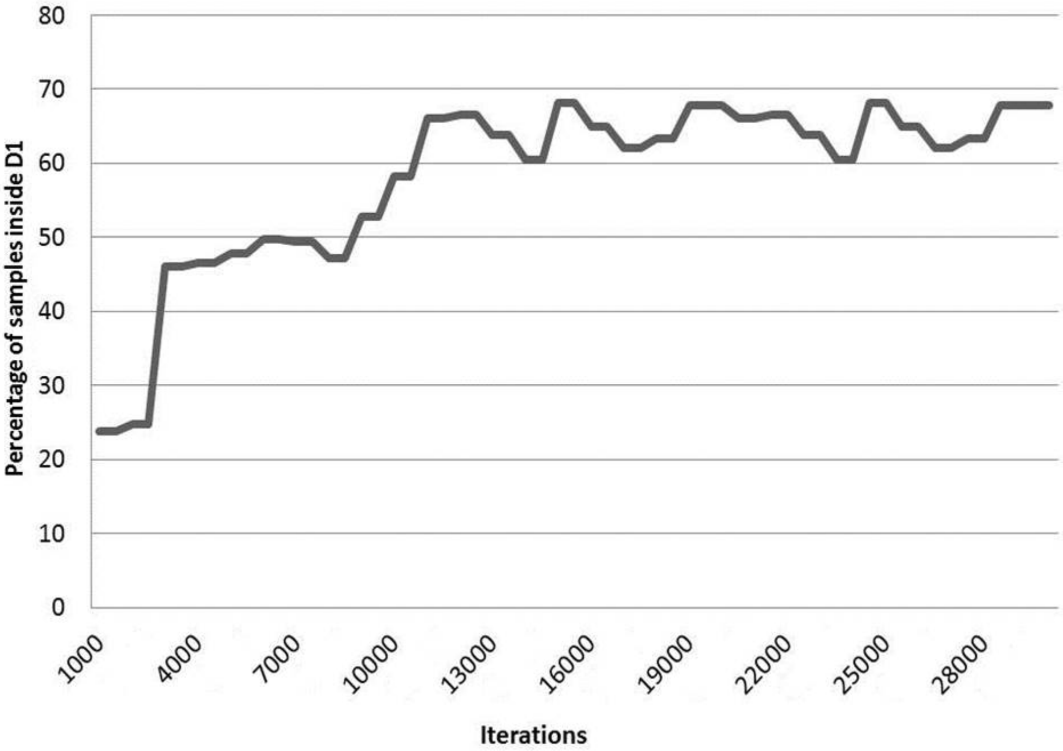


Figure 5.15: Step plot of the percentage of samples lying inside D1 in experiment 5.

	Inside D1	Outside D1
Time required per iteration (in seconds)	60	100
Average Percentage of samples in each domain	55	45
Original Time required (in hours)	833.34	
Time required by smart algorithm (in hours)	650	
Percentage reduction in time	22	

Table 5.6: Analysis of the computational time of the smart algorithm in Experiment 6.

5.2.3 Experiment 6

Referring to the experimental settings of experiment 6 in Table 4.2, the smart algorithm was run over 30,000 iterations. The scale parameter of the likelihood expression, β, is set to 50. The trace plot of the Markov chain obtained from each of the three inferred parameters is shown in Figure 5.16 with an acceptance rate of 31%. The kernel density estimator resulted in the posterior probability distributions shown in the corner plot of Figure 5.17. Moreover, a step plot of the percentage of samples lying inside D1 is represented in Figure 5.18.

Our algorithm accurately inverted for the source locations represented by x, y and h as well as the emission duration d. All the posterior distributions have their maxima at the reference values of these four parameters. However, we obtained a flat distribution of the posterior probability of q with several maxima. Furthermore, the smart algorithm resulted in a gradual increase in the percentage of samples inside D1 occurs until having a steady state after 13,000 iterations within the range between 60% and 70%. The reduction of time achieved by the efficiency smart algorithm is shown in Table 5.6.

5.2.4 Discussion of Results of Set II

Based on the obtained results in this set of experiments, the produced results give us good indications about the different parameters under inspection. The location of the emission source in the 2D plane, referred to as x and y, is being properly identified in all three experiments. In addition, the definition of the location in the 3D plane, characterized by h, is also being well predicted. At this level, a remarkable improvement is noticed in comparison to the results in Set I, i.e. after using two observations F at two different elevations instead of using a single observation F. Regarding the emission strength, the released mass m and emission duration d are also being properly identified unlike the emission rate q. Hence, we can say that although the accepted samples of q may be away from the reference value, the product of the proposed q and d at each iteration

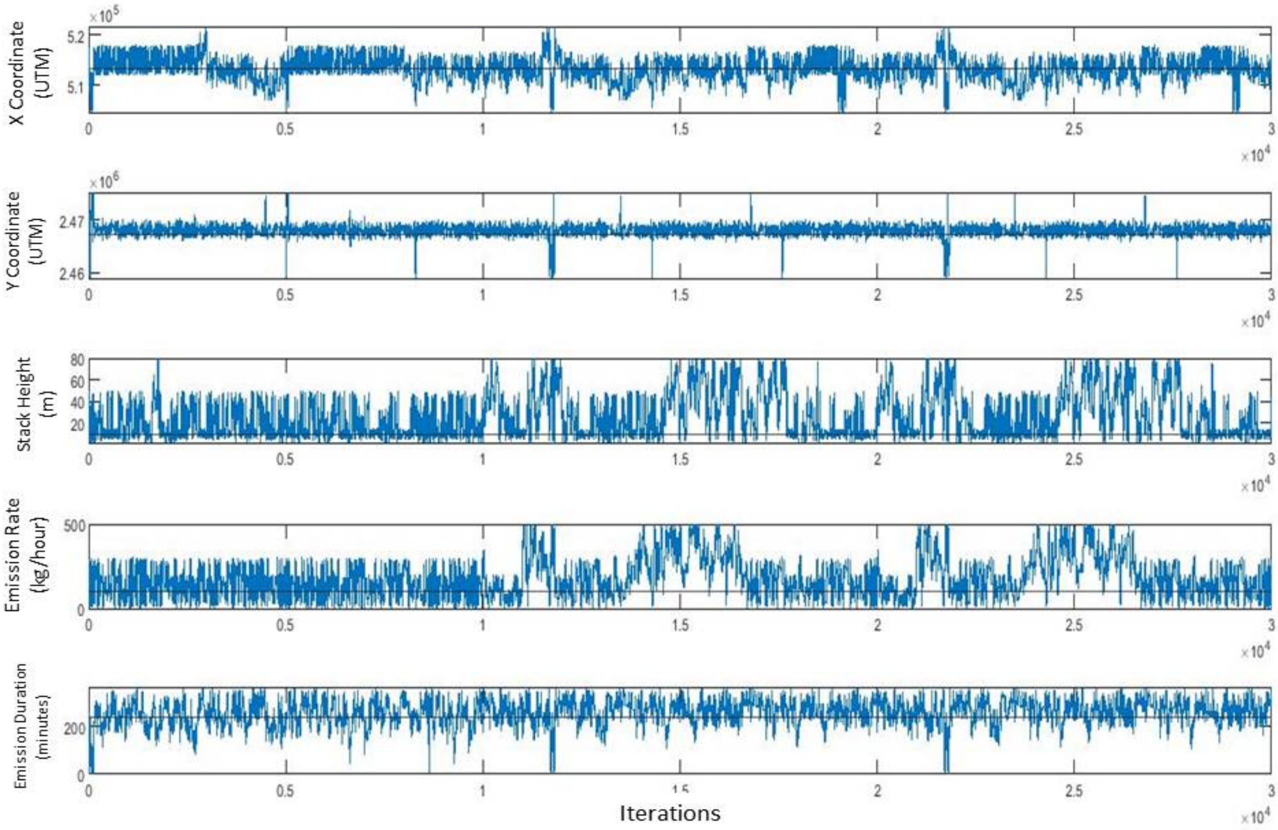


Figure 5.16: Markov chains obtained in Experiment 6. The horizontal black lines refer to the reference value of each parameter.

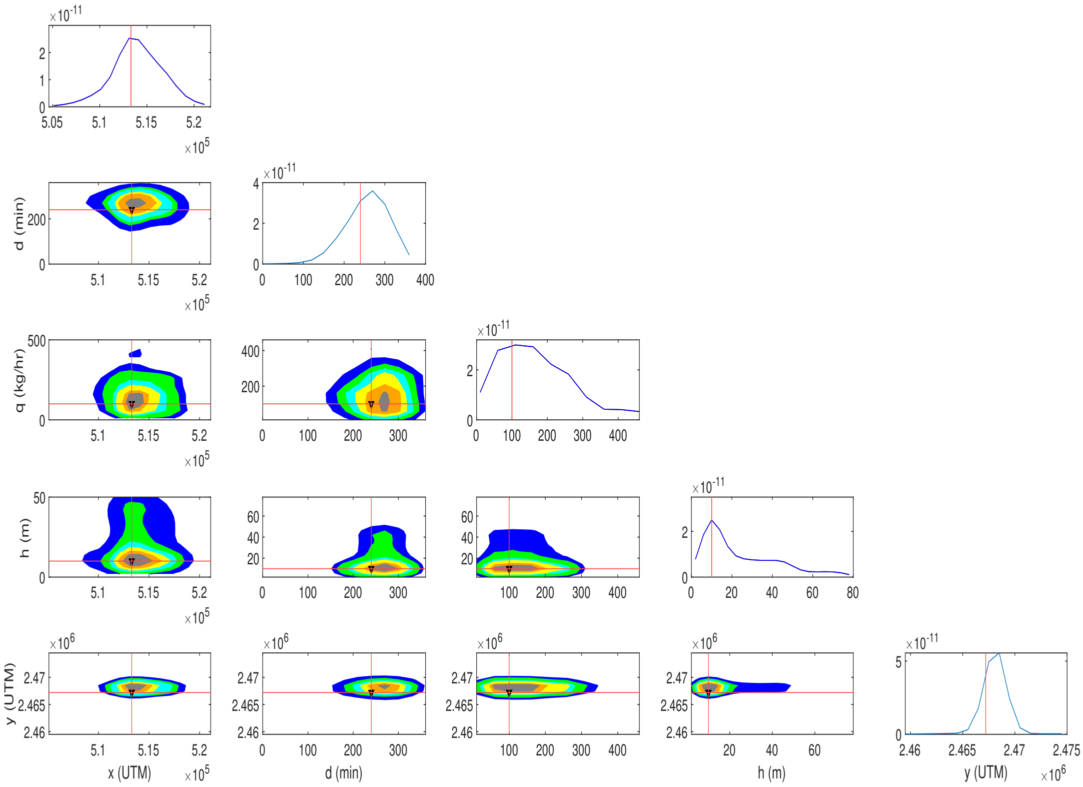


Figure 5.17: Corner plot of the obtained posterior probability distributions in experiment 6.The red lines represent the reference value of each of the inferred parameters.

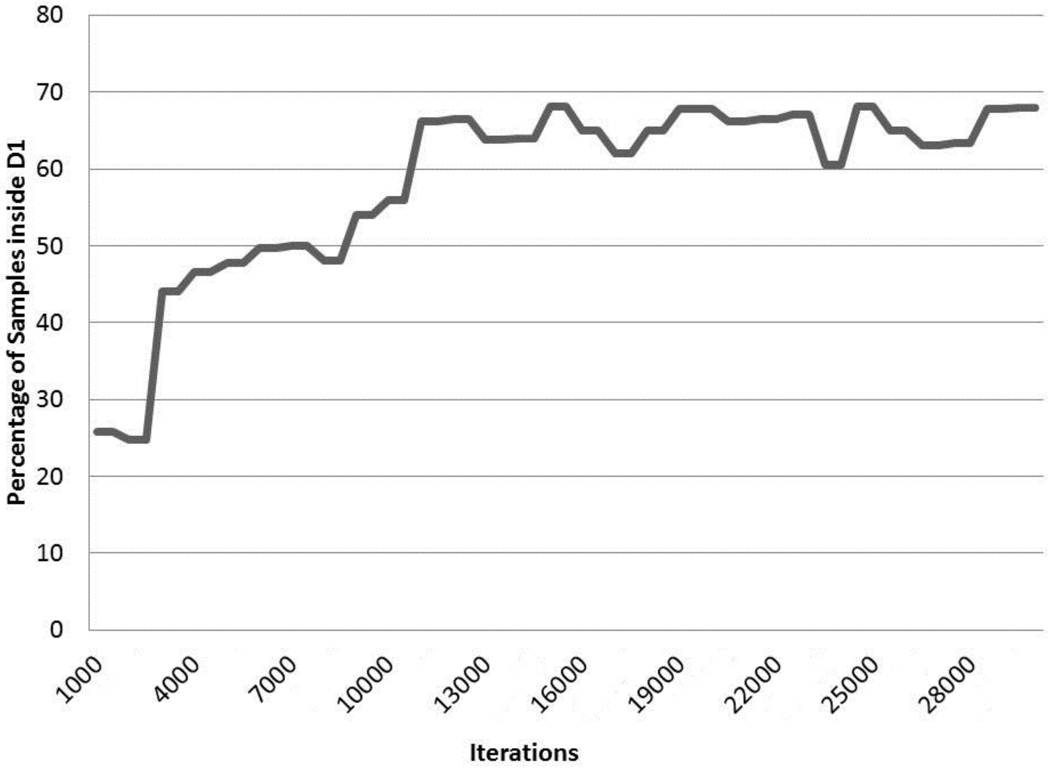


Figure 5.18: Step plot of the percentage of samples lying inside D1 in experiment 6.

is around the reference value of m. That is why the parameter m is being well predicted. Another aspect that is noticed in the Markov chains is simultaneous fluctuations in those of h and m in Experiments 5 and those of h and q in Experiment 6. This indicates the dependence between these two parameters where a larger stack height and a higher emission rate, eventually a higher emission mass, than our reference values will produce similar concentration fields as our reference observations.

At the level of the smart algorithm, it is worth noting that we achieved higher percentage of samples inside D1 in Set II than that in Set I. This reveals the relation between using two observations F and enhancing the efficiency of our smart algorithm and hence the feasibility of the framework.

After finishing these two sets of experiments, we can now proceed to test the reliability of our smart algorithm when using other forms of observations.

Chapter 6

Conclusion and Future Work

Air pollution is a common aspect witnessed in the different countries around the world. It harmfully affects all the living creatures in rural and urban environments as well as the natural habitats of the different animals. In this scope, it is very crucial to find the source or sources responsible for an observation indicating the presence of air pollution in a certain region. Characterizing the emission sources will allow agencies and governments to deal with the resulting pollution and mitigate its harmful effects. Also, it can serve as a tool to build a proper emergency plan in case of accidents and terrorist incidents.

In this research work, we developed a smart algorithm that is able to identify and characterize an emission source releasing air contaminants into the atmosphere. This framework uses Lagrangian atmospheric dispersion modeling along with Bayesian inference and stochastic Monte Carlo sampling. The iterative process will result in a Markov chain from which can extract the posterior probability distributions at the level of each inferred parameter. We implemented this smart algorithm over then domain of the urban environment of KAUST, KSA while inferring for five different parameters that reveal the emission source location and strength. The parameters include the x and y coordinates in the UTM coordinate system, the stack height, the emission rate and the emission duration. The observation used is in the form of a concentration field obtained at different elevations above the ground. The location in the 2D plane was properly identified in all experiments. However, the height prediction was properly done when using observations at two different elevations instead of one, that is in Set II instead of Set I. Good indications of the mass were also obtained in both Experiments 2 and 5, unlike its elementary components the emission rate and emission duration.

The framework is then efficient at predicting the different emission parameters. The smart algorithm is also efficient in terms of reduction computational time required by the basic framework where it reduced around 20% of the original required time making our algorithm more feasible. The increasing percentage of samples inside D1 was an additional factor indicating the quasi steady state of chain convergence.

Based on these results, many additional experiments can be done. These may consider different types of observations including concentration contours and concentration point-wise values. These may be obtained either from sensors or remote sensing equipment. Additionally, observations made be obtained at different times to check the effectiveness of our framework for such observations. Additionally, at the level of computational times, we may use a moving domain D1 instead of fixed one as we did in our work. The new position of D1 will depend mainly on the location of the sample itself. This algorithm may also be expanded to check uncertainty of the model meteorological boundary conditions responsible for driving the pollutants in the urban area at hand.

Bibliography

[1] L. Robertson, J. Langner, Source function estimate by means of variational data assimilation applied to the etex-i tracer experiment, Atmospheric Environment 32 (24) (1998) 4219–4225.

[2] J. A. Pudykiewicz, Application of adjoint tracer transport equations for evaluating source parameters, Atmospheric environment 32 (17) (1998) 3039–3050.

[3] C. T. Allen, G. S. Young, S. E. Haupt, Improving pollutant source characterization by better estimating wind direction with a genetic algorithm, Atmospheric Environment 41 (11) (2007) 2283–2289.

[4] C. T. Allen, S. E. Haupt, G. S. Young, Source characterization with a genetic algorithm–coupled dispersion–backward model incorporating scipuff, Journal of applied meteorology and climatology 46 (3) (2007) 273–287.

[5] G. Cervone, P. Franzese, Machine learning for the source detection of atmospheric emissions, Tech. rep., American Meteorological Society (2010).

[6] G. Cervone, P. Franzese, Non-darwinian evolution for the source detection of atmospheric releases, Atmospheric Environment 45 (26) (2011) 4497–4506.

[7] J.-P. Issartel, Emergence of a tracer source from air concentration measurements, a new strategy for linear assimilation, Atmospheric Chemistry and Physics 5 (1) (2005) 249–273.

[8] V. Winiarek, J. Vira, M. Bocquet, M. Sofiev, O. Saunier, Towards the operational estimation of a radiological plume using data assimilation after a radiological accidental atmospheric release, Atmospheric environment 45 (17) (2011) 2944–2955.

[9] A. Keats, E. Yee, F.-S. Lien, Bayesian inference for source determination with applications to a complex urban environment, Atmospheric environment 41 (3) (2007) 465–479.

[10] G. Johannesson, B. Hanley, J. Nitao, Dynamic bayesian models via monte carlo-an introduction with examples, Tech. rep., Lawrence Livermore National Lab., Livermore, CA (US) (2004).

[11] I. Senocak, N. W. Hengartner, M. B. Short, W. B. Daniel, Stochastic event reconstruction of atmospheric contaminant dispersion using bayesian inference, Atmospheric Environment 42 (33) (2008) 7718–7727.

[12] E. Yee, F.-S. Lien, A. Keats, R. DAmours, Bayesian inversion of concentration data: Source reconstruction in the adjoint representation of atmospheric diffusion, Journal of Wind Engineering and Industrial Aerodynamics 96 (10-11) (2008) 1805–1816.

[13] H. Rajaona, F. Septier, P. Armand, Y. Delignon, C. Olry, A. Albergel, J. Moussafir, An adaptive bayesian inference algorithm to estimate the parameters of a hazardous atmospheric release, Atmospheric Environment 122 (2015) 748–762.

[14] J. Lundquist, B. Kosovic, R. Belles, Synthetic event reconstruction experiments for defining sensor network characteristics, Tech. rep., Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States) (2005).

[15] L. Delle Monache, J. K. Lundquist, B. Kosović, G. Johannesson, K. M. Dyer, R. D. Aines, F. K. Chow, R. D. Belles, W. G. Hanley, S. C. Larsen, et al., Bayesian inference and markov chain monte carlo sampling to reconstruct a contaminant source on a continental scale, Journal of Applied Meteorology and Climatology 47 (10) (2008) 2600–2613.

[16] T. Ziehn, A. Nickless, P. Rayner, R. Law, G. Roff, P. Fraser, Greenhouse gas network design using backward lagrangian particle dispersion modelling-part 1: Methodology and australian test case (2014).

[17] P. Seibert, A. Frank, Source-receptor matrix calculation with a lagrangian particle dispersion model in backward mode, Atmospheric Chemistry and Physics 4 (1) (2004) 51–63.

[18] F. K. Chow, B. Kosović, S. Chan, Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations, Journal of applied meteorology and climatology 47 (6) (2008) 1553–1572.

[19] I. A. Kane, M. A. Clare, Dispersion, accumulation, and the ultimate fate of microplastics in deep-marine environments: A review and future directions, Frontiers in Earth Science 7 (2019) 80.

[20] Y.-J. Han, T. M. Holsen, P. K. Hopke, S.-M. Yi, Comparison between back-trajectory based modeling and lagrangian backward dispersion modeling for

locating sources of reactive gaseous mercury, Environmental science & technology 39 (6) (2005) 1715–1723.

[21] D. Fontaine, M. Hall, et al., Dispersion modeling of gas releases on offshore platforms, in: SPE Health, Safety and Environment in Oil and Gas Exploration and Production Conference, Society of Petroleum Engineers, 1991.

[22] D. Kahforoshan, E. Fatehifar, A. Babalou, A. Ebrahimin, A. Elkamel, J. Soltanmohammadzadeh, Modeling and evaluation of air pollution from a gaseous flare in an oil and gas processing area, in: WSEAS Conferences, Santander, 2008, pp. 180–186.

[23] M.-S. Park, S.-H. Park, J.-H. Chae, M.-H. Choi, Y. Song, M. Kang, J.-W. Roh, High-resolution urban observation network for user-specific meteorological information service in the seoul metropolitan area, south korea., Atmospheric Measurement Techniques 10 (4) (2017).

[24] A. A. Shusterman, V. E. Teige, A. J. Turner, C. Newman, J. Kim, R. C. Cohen, The berkeley atmospheric co2 observation network: Initial evaluation, Atmos. Chem. Phys 16 (21) (2016) 13449–13463.

[25] O. A. Popoola, D. Carruthers, C. Lad, V. B. Bright, M. I. Mead, M. E. Stettler, J. R. Saffell, R. L. Jones, Use of networks of low cost air quality sensors to quantify air quality in urban settings, Atmospheric environment 194 (2018) 58–70.

[26] C.-M. Gan, Y. Wu, B. Madhavan, B. Gross, F. Moshary, Application of active optical sensors to probe the vertical structure of the urban boundary layer and assess anomalies in air quality model pm2. 5 forecasts, Atmospheric environment 45 (37) (2011) 6613–6621.

[27] S. Srivastava, I. N. Sinha, Classification of air pollution dispersion models: a critical review, in: Proceedings of National Seminar on Environmental Engineering with special emphasis on Mining Environment, 2004.

[28] M. R. Beychok, Fundamentals of stack gas dispersion, MR Beychok, 2005.

[29] D. B. Turner, Workbook of atmospheric dispersion estimates: an introduction to dispersion modeling, CRC press, 1994.

[30] N. Atkins, Air pollution dispersion: Ventilation factor, Lyndon State College (2008).

[31] O. Sutton, The problem of diffusion in the lower atmosphere, Quarterly Journal of the Royal Meteorological Society 73 (317-318) (1947) 257–281.

[32] X.-X. Li, C.-H. Liu, D. Y. Leung, K. M. Lam, Recent progress in cfd modelling of wind field and pollutant transport in street canyons, Atmospheric Environment 40 (29) (2006) 5640–5658.

[33] C. J. Walcek, 8.7 lagrangian vs. eulerian dispersion modeling: Effects of wind shear on pollution dispersion.

[34] T. Wolf, L. H. Pettersson, I. Esau, A very high-resolution assessment and modelling of urban air quality.

[35] S. Di Sabatino, R. Buccolieri, B. Pulvirenti, R. Britter, Flow and pollutant dispersion in street canyons using fluent and adms-urban, Environmental Modeling & Assessment 13 (3) (2008) 369–381.

[36] J. Pullen, J. P. Boris, T. Young, G. Patnaik, J. Iselin, A comparison of contaminant plume statistics from a gaussian puff and urban cfd model for two large cities, Atmospheric Environment 39 (6) (2005) 1049–1068.

[37] J. Fensterstock, J. Kurtzweg, G. Ozolins, Reduction of air pollution potential through environmental planning, Journal of the air pollution control association 21 (7) (1971) 395–399.

[38] R. A. FINEBERG, Joint pipeline office (jpo) executive council (2007).

[39] J. Hadamard, Sur les problèmes aux dérivées partielles et leur signification physique, Princeton university bulletin (1902) 49–52.

[40] S. P. Parker, McGraw-Hill dictionary of scientific and technical terms, McGraw-Hill Book Co., 1989.

[41] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society: Series B (Methodological) 58 (1) (1996) 267–288.

[42] P. Bühlmann, S. Van De Geer, Statistics for high-dimensional data: methods, theory and applications, Springer Science & Business Media, 2011.

[43] T. Park, G. Casella, The bayesian lasso, Journal of the American Statistical Association 103 (482) (2008) 681–686.

[44] P. A. Vikhar, Evolutionary algorithms: A critical review and its future prospects, in: 2016 International conference on global trends in signal processing, information computing and communication (ICGTSPICC), IEEE, 2016, pp. 261–265.

[45] D. E. Goldberg, Genetic algorithms, Pearson Education India, 2006.

[46] L. Zhu, G. Gigerenzer, Children can solve bayesian problems: The role of representation in mental computation, Cognition 98 (3) (2006) 287–308.

[47] L. A. Foreman, A. F. Smith, I. W. Evett, Bayesian analysis of dna profiling data in forensic identification applications, Journal of the Royal Statistical Society: Series A (Statistics in Society) 160 (3) (1997) 429–459.

[48] M. Hutter, On universal prediction and bayesian confirmation, Theoretical Computer Science 384 (1) (2007) 33–48.

[49] G. Monge, Mémoire sur la théorie des déblais et des remblais, Histoire de l'Académie Royale des Sciences de Paris (1781).

[50] Q. Merigot, B. Thibert, Optimal transport: discretization and algorithms, arXiv preprint arXiv:2003.00855 (2020).

[51] B. Lévy, E. L. Schwindt, Notions of optimal transport theory and how to implement them on a computer, Computers & Graphics 72 (2018) 135–148.

[52] C. Villani, Topics in optimal transportation, no. 58, American Mathematical Soc., 2003.

[53] R. J. McCann, et al., Existence and uniqueness of monotone measure-preserving maps, Duke Mathematical Journal 80 (2) (1995) 309–324.

[54] G. Peyré, M. Cuturi, et al., Computational optimal transport. foundations and trends® in machine learning. 2019; 11 (5-6): 355–607.

[55] Q. Mérigot, J.-M. Mirebeau, Minimal geodesics along volume-preserving maps, through semidiscrete optimal transport, SIAM Journal on Numerical Analysis 54 (6) (2016) 3465–3492.

[56] L. Caffarelli, V. Oliker, Weak solutions of one inverse problem in geometric optics, Journal of Mathematical Sciences 154 (1) (2008) 39–49.

[57] S. Kolouri, S. R. Park, M. Thorpe, D. Slepcev, G. K. Rohde, Optimal mass transport: Signal processing and machine-learning applications, IEEE signal processing magazine 34 (4) (2017) 43–59.

[58] M. Petitjean, Chiral mixtures, Journal of Mathematical Physics 43 (8) (2002) 4147–4157.

[59] M. Petitjean, From shape similarity to shape complementarity: toward a docking theory, Journal of mathematical chemistry 35 (3) (2004) 147–158.

[60] S. Martin Arjovsky, L. Bottou, Wasserstein generative adversarial networks, in: Proceedings of the 34 th International Conference on Machine Learning, Sydney, Australia, 2017.

[61] A. D. S. LANDESREGIERUNG, Documentation of the lagrangian particle model gral (graz lagrangian model) vs. 19.1 (2018).

[62] A. D. S. LANDESREGIERUNG, Documentation of the prognostic mesoscale model gramm (graz mesoscale model) version 19.1 (2019).

[63] Corine land cover (2018).
URL `https://land.copernicus.eu/`

[64] A. Farchi, M. Bocquet, Y. Roustan, A. Mathieu, A. Quérel, Using the wasserstein distance to compare fields of pollutants: application to the radionuclide atmospheric dispersion of the fukushima-daiichi accident, Tellus B: Chemical and Physical Meteorology 68 (1) (2016) 31682.