



**EPFL**

---

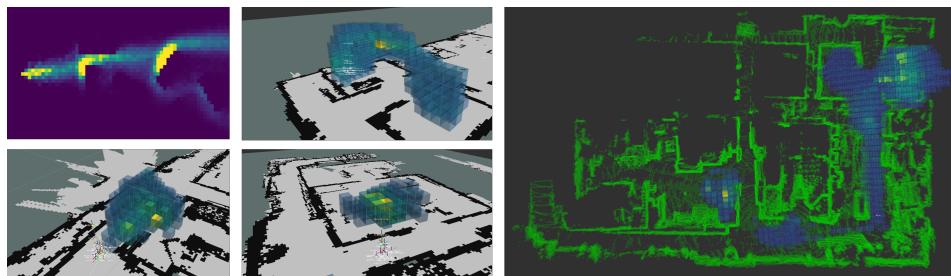
WS 2024-2025, IC-XXX, DISAL-SP111

Start: 09.09.2024  
Finish: 10.01.2025

---

## Gas Source Localization with Gaussian Markov Random Field

Marie Ethvignot



---

Professor: Alcherio Martinoli  
Assistant: Wanting Jin

---

This page intentionally left blank.

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Context . . . . .	4
1.2	Objectives . . . . .	4
<b>2</b>	<b>Methods</b>	<b>5</b>
2.1	Literature Review . . . . .	5
2.1.1	Context and Previous Approaches in Gas Distribution Mapping and Source Localization . . . . .	5
2.1.2	Structurally Aware Gas Distribution Mapping with Gaussian Belief Propagation . . . . .	6
2.2	Implementation and Evaluation of Gaussian Belief Propagation for Gas Distribution Mapping . . . . .	10
2.2.1	Implementation by the authors . . . . .	10
2.2.2	Adapting the Method for Evaluation. . . . .	11
<b>3</b>	<b>Results</b>	<b>16</b>
3.1	Influence of the sampling frequency . . . . .	16
3.2	Influence of the spatial frequency . . . . .	19
3.3	Influence of the obstacles . . . . .	21
3.4	Comparison to the Kernel DMV Method . . . . .	26
<b>4</b>	<b>Conclusion</b>	<b>31</b>
	<b>Bibliography</b>	<b>32</b>

# Chapter 1

## Introduction

### 1.1 Context

The use of robots in Gas Source Localization (GSL) tasks within hazardous environments offers a significant advantage by minimizing the risk to humans. These robots, equipped with advanced sensing capabilities for detecting gases and monitoring wind patterns, are required to autonomously navigate and localize gas sources in complex and often dangerous scenarios. GSL is particularly challenging in real-world settings due to the unpredictable nature of gas dispersion, which is governed by a variety of factors, like the airflow (advection) that directs the movement of gas molecules. The structure of the environment, along with obstacles, further complicates this process by shaping how air and gas flow through the space. Understanding and estimating the overall wind flow in the environment provides crucial insights into the gas distribution, which can greatly improve the efficiency and accuracy of GSL operations.

### 1.2 Objectives

The goal of this project is to gain a thorough understanding of Gaussian Markov Random Fields (GMRFs) and Gaussian Belief Propagation (GBP) algorithms, and to explore their potential applications in gas source localization and gas distribution mapping. To achieve this, the project is structured around two main objectives.

The first objective is to review the literature on the topic but more specifically to review the work presented in [1], which details a state-of-the-art Gaussian Belief Propagation algorithm designed for gas distribution mapping. This includes studying the theoretical foundations and previous methods to understand how the algorithm operates and its advancements over earlier approaches.

The second objective is to develop and adapt the proposed algorithm to test it on pre-recorded gas concentration measurements collected from specific environments. This step aims to evaluate the performance of the algorithm under controlled conditions and assess its strengths and limitations and its utility for gas distribution mapping and source localization tasks. Finally, the last step is to compare the algorithm's performance on this data with the Kernel DMV method's performance.

# Chapter 2

## Methods

In this chapter, we present the two main approaches that guided the execution of this project. First, we conduct a comprehensive review of the relevant literature, with a particular focus on the key paper [1] that forms the foundation of our work. This includes an in-depth exploration of the theoretical concepts of the study, such as Gaussian Markov Random Fields and Gaussian Belief Propagation, and their application to the problem of Gas Distribution Mapping and Gas Source Localization. Second, we detail the implementation and adaptation of this method developed to evaluate its performance in different scenarios and eventually compare it to a Kernel-based method.

### 2.1 Literature Review

#### 2.1.1 Context and Previous Approaches in Gas Distribution Mapping and Source Localization

The challenge of gas distribution mapping (GDM) and gas source localization has been a focus of extensive research in the field of mobile robotics. Various methods have been developed to tackle this problem, each offering advantages and limitations. In this section, we will review the state-of-the-art approaches in the literature for Gas Distribution Mapping.

**Kernel-Based Methods** One of the earliest and most widely used approaches for GDM is based on kernel density estimation. In kernel methods, first introduced by Lilienthal et al. [2], the environment is discretized into a grid, where each cell represents a predicted mean gas concentration. Measurements influence surrounding cells using a Gaussian-shaped kernel, with the magnitude of the contribution decreasing with distance. This method has been extended such as to the methods Kernel DM+V [3] and Kernel DM+V/W [4], which introduced variance estimation and incorporated wind direction measurements, respectively. Other extensions are time-decay versions like TD Kernel DM+V/W [5], which address the challenge of handling outdated measurements by reducing their influence over time.

Despite these advancements, kernel-based methods have fundamental limitations. Indeed, they lack the ability to model dependencies between grid cells, which prevents them from accounting for obstacles or environmental dynamics, such as wind patterns, in a probabilistic manner. Indeed, the method does not directly include the effects that obstacles have on the gas plume and cannot make predictions outside of the cutoff radius of the kernel.

**Gaussian Processes** Gaussian Processes (GPs) have been employed as an alternative to kernel-based methods due to their ability to predict continuous distributions over a domain. In the work of Stachniss et al. [6], GPs were used to model both background gas concentrations and localized plumes through a mixture of Gaussian processes. GPs avoid discretization errors

and provide smooth predictions, making them an attractive choice for GDM. However, they are computationally expensive, with computational complexity increasing significantly as the number of measurements grows. Furthermore, GPs are not well-suited for integrating obstacle information into the predicted map, limiting their utility in cluttered environments or large-scale scenarios.

**Gaussian Markov Random Fields-based methods** Gaussian Markov Random Fields (GMRFs) address several limitations of kernel methods and GPs by modeling spatial dependencies between grid cells. Indeed, in the GMRFs methods, the environment is represented as a graph where nodes correspond to grid cells, and edges are the dependencies between neighboring cells. Monroy et al. [7] proposed a time-variant GMRF approach that integrates obstacle information into the probabilistic framework and allows for the gas distribution map to be dynamically updated as new measurements are acquired. This method significantly improves the accuracy of GDM in complex environments by capturing the effects of obstacles and environmental changes. However, its reliance on global solvers makes it computationally demanding, with a complexity of  $O(n^{1.5})$ . This limits its scalability to large maps or real-time applications.

**Gaussian Belief Propagation method** Gaussian Belief Propagation (GBP) was introduced as a computationally efficient alternative for inference in GMRF-based models. Unlike traditional global solvers, GBP employs message-passing algorithms that perform local inference by propagating messages between neighboring nodes on the graph. This allows GBP to achieve faster convergence and scalability, making it suitable for real-time applications.

Rhodes et al. [8] demonstrated the use of GBP for gas distribution mapping, introducing a wildfire message scheduling algorithm to optimize the computational process. GBP allows the distribution map to be updated continuously as new data is received, rather than waiting for all data to be processed at once, which greatly reduces computational load. However, initial work on GBP was limited to 2D mapping scenarios and required a globally defined map for inference.

The method of interest in this project [1] builds upon the advancements of GBP to overcome the limitations of their previous work [8]. It extends GBP to 3D mapping, improves computational efficiency, and integrates real-world applicability on a dynamic robotic platform. Unlike Monroy et al.'s GMRF approach [7], which requires resolving the entire map per iteration, the GBP-based method adopted here uses localized message passing to achieve scalability and robustness, even in complex environments. The theory behind this method will be discussed in the following subsection.

### 2.1.2 Structurally Aware Gas Distribution Mapping with Gaussian Belief Propagation

We will now dive into the theory of the method of our interest [1] to understand how they model and solve the Gas Distribution Mapping problem using GMRFs and GBP.

**Gaussian Markov Random Fields** A Gaussian Markov Random Field (GMRF) is a probabilistic model that represents spatially correlated Gaussian-distributed variables using a graph structure. The fundamental principle of a GMRF is the Markov property, which ensures that each variable only directly depends on its neighbors. This property leads to a sparse precision matrix, which is the inverse of the covariance matrix. The sparsity of the precision matrix reflects local dependencies between variables and allows for computationally efficient inference, even in large-scale systems.

In the context of gas distribution mapping (GDM), a GMRF is well-suited to model the gas concentration distribution. Indeed, as gas dispersion in an environment is a spatially correlated

process, the gas concentration at one point depends on its surroundings. So, by associating each spatial location (grid cell) with a random variable and defining dependencies between neighboring cells through graph edges, a GMRF effectively represents the spatial correlations in the gas field. The joint probability distribution of the gas concentrations is modeled as a multivariate Gaussian distribution, which allows for the use of efficient inference techniques such as Gaussian Belief Propagation.

**Factor Graph Representation.** In the method proposed by Rhodes et al., the gas distribution mapping (GDM) problem is formulated as an inference task over a factor graph, where the goal is to estimate the most likely gas concentration values at discrete spatial locations (cells) in a 2D or 3D space. This is done by calculating the maximum a posteriori (MAP) estimate of the variables (gas concentrations) across the map.

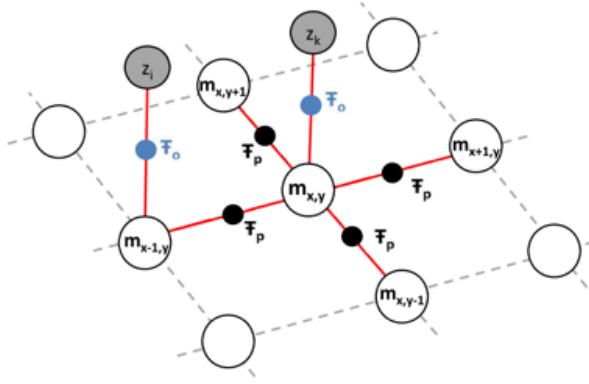


Figure 2.1: [7] Representation of the factor graph for gas distribution mapping. Factors ( $\mathcal{F}$ ) connect robot poses ( $m_{x,y}$ ) to observations ( $z_k$ ) and neighboring poses ( $m_{x+1,y}$ ,  $m_{x,y+1}$ ), illustrating the relationships enforced by observation ( $f_o$ ) and propagation ( $f_p$ ) constraints.

A factor graph is a bipartite graph where variables are connected to functions (factors) that define their relationships or constraints. In this representation, variable nodes correspond to gas concentrations at specific spatial locations (grid cells), and factor nodes represent relationships or constraints that define how these concentrations interact with each other or behave.

The factor graph of this method uses three types of factors to model the problem:

- **Observation Factors ( $f_o$ ):** These factors link the gas concentration sensor measurements to the estimated gas concentration at the corresponding location in the model. They ensure that the model incorporates the information provided by real-world measurements while accounting for uncertainties such as sensor noise and time-dependent inaccuracies.
- **Regularization Factors ( $f_r$ ):** These factors define the relationships between neighboring nodes in the graph by modeling the physical spatial correlations of gas dispersion. Gas concentrations at nearby locations are often similar, and regularization factors enforce this smoothness. Moreover, these factors account for obstacles in the environment, such as walls, which block the direct correlation between nodes on opposite sides.
- **Default Factors ( $f_d$ ):** These factors provide a base for gas concentrations in areas where no sensor measurements are available. They lightly pull the estimated concentrations toward a default value, often zero, to make sure that the model remains stable in regions with no or almost no data.

Together, these factors ensure that the factor graph accounts for sensor data, spatial relationships, and default assumptions.

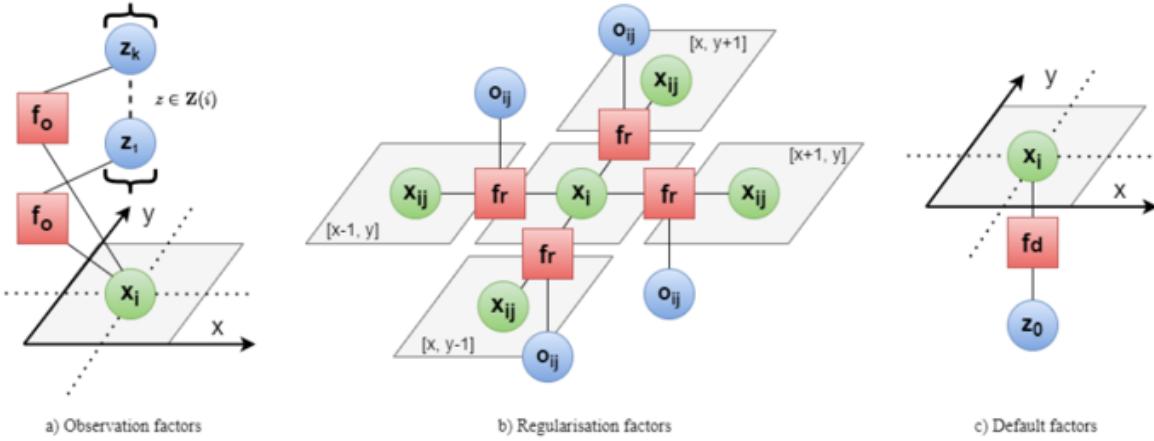


Figure 2.2: [8] Overview of factor types in a gas distribution mapping factor graph. (a) Observation factors  $f_o$  connect the robot pose  $x_i$  to sensor observations  $z_k$ . (b) Regularization factors  $f_r$  enforce spatial consistency among neighboring robot poses  $x_{ij}$ . (c) Default factors  $f_d$  provide prior information on the robot pose  $x_i$ .

**Gaussian Belief Propagation** To perform inference on the factor graph, the method uses Gaussian Belief Propagation (GBP), a message-passing algorithm, instead of direct solving [7]. GBP is effective for GMRFs because of the Gaussian properties of the variables. The algorithm starts with an initialization phase, where each variable node sends prior messages to its neighboring factor nodes. During the message-passing phase, nodes iteratively exchange messages that encode information about their beliefs about gas concentrations. Each message is represented as a Gaussian distribution with a mean and a variance.

The belief at each node is updated iteratively with the incoming messages from its neighbors, and the process continues until convergence. The convergence is achieved when the changes in messages between iterations fall below a predefined threshold (which is a hyperparameter of the algorithm), i.e. when the incoming information is not important enough. The final beliefs represent the marginal distributions of the gas concentrations at each location.

The authors compared and evaluated different message-passing algorithm in their previous work [8] which led them to develop in this method a hybrid message-passing algorithm to efficiently solve the factor graph while maintaining accuracy. It combines two techniques: a wildfire scheduling strategy, which focuses computation on active regions of the graph where changes are significant, and a residual-based approach, which prioritizes updates based on the error or inconsistency at each node.

This message-passing approach makes the method usable for large maps with high computational efficiency. The localized nature of the computations avoids the need to solve large linear systems directly (which [7] does), which would be computationally difficult in real-time applications. Additionally, the method dynamically grows the factor graph as new measurements are collected, making it suitable for real-time applications.

**Hyperparameters of the Method.** The performance of the Gaussian Belief Propagation (GBP) method is influenced by several hyperparameters that control its behavior and results. One key hyperparameter is the threshold for message updates, which determines when message-

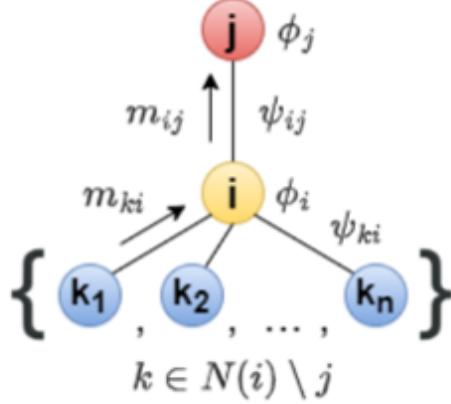


Figure 2.3: [1]Message-passing process in the factor graph. Node  $i$  receives messages ( $m_{ki}$ ) from neighboring nodes  $k_1, k_2, \dots, k_n$  (excluding  $j$ ) and sends a message ( $m_{ij}$ ) to node  $j$ . Factors  $\psi_{ij}$  and  $\phi_i$  represent the potential functions that define the interaction between connected nodes.

passing updates are considered significant enough to propagate further. A smaller threshold can lead to higher accuracy by ensuring more detailed convergence but increases computational time. Another important hyperparameter is the variance of the regularization factors, which controls the strength of the spatial correlations between neighboring nodes. Adjusting this variance affects the smoothness of the estimated gas distribution, with lower values enforcing stronger correlations and higher values allowing for more localized variations. The sensor noise variance and time-dependent uncertainty variance in the observation factors also play critical roles, as they determine how much weight is given to sensor data versus prior assumptions. Finally, the residual prioritization factor in the hybrid message-passing algorithm influences the order of updates, allowing the method to focus on nodes with larger errors for faster convergence. Tuning these hyperparameters is essential for balancing accuracy, computational efficiency, and robustness in different environments.

**Contributions of the method** This method makes several advancements over previous work. First, it extends gas distribution mapping (GDM) from 2D to 3D, enabling inference in more complex environments. Another improvement from their previous work is the hybrid message-passing algorithm that combines wildfire and residual scheduling to improve convergence speed and efficiency when handling multiple point sampling sensors. The approach also dynamically builds the factor graph based on information content, allowing it to adapt to the environment in real time. Finally, all of these innovations are integrated into a robotic framework capable of performing 3D gas mapping, simultaneous localization and mapping (SLAM), and obstacle-aware inference onboard the robot in an unknown environment.

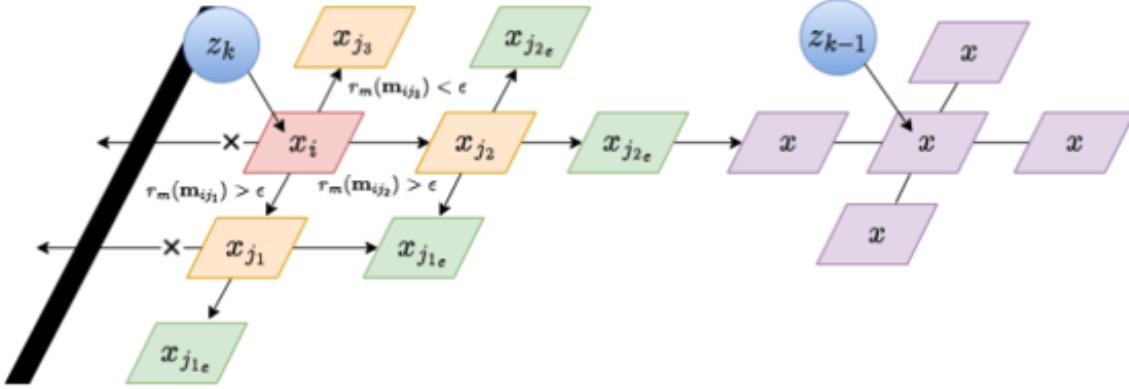


Figure 2.4: [1] Information theoretic local factor graph growth around sequential measurements. The graph illustrates how gas concentration estimates  $x_i$  (variables of the factor graph) are updated when new measurements ( $z_k$ ) are introduced. It shows the growth of the factor graph, where connections between variables are established based on the comparison between the threshold  $\epsilon$  and the residuals  $r_n(m_{ij})$  and spatial constraints (obstacle represented as a black line).

## 2.2 Implementation and Evaluation of Gaussian Belief Propagation for Gas Distribution Mapping

### 2.2.1 Implementation by the authors

To begin, we thoroughly reviewed and analyzed the code provided by the authors of the Gaussian Belief Propagation (GBP) method. This involved running their implementation to verify that it performed as described in their paper, particularly its ability to efficiently map gas distributions in complex 3D environments in real-time and onboard a mobile robot. This step ensured that we understood the structure and functionality of their approach before actually testing and evaluating it.

To gain a deeper understanding of the method, we first reviewed and tested the authors' implementation from their publicly available GitHub repository [9]. The core of the method is written in Python and relies on the Gaussian Belief Propagation (GBP) algorithm presented in their paper [1] for solving the factor graph representation of the gas distribution mapping (GDM) problem. The implementation uses the Robot Operating System (ROS), a widely used framework in robotics, to handle data flow, integration with robotic platforms, and real-time operations.

In their framework, ROS is used for handling sensor data acquisition, robot localization, and communication between different components of the system. The factor graph, which represents the gas concentration field and environmental constraints, is dynamically constructed and updated based on incoming data, such as gas sensor readings and robot positions. Obstacles in the environment are incorporated into the graph using an occupancy map and are detected by two LIDAR sensors onboard the robot that provide point-clouds data in real-time. To ensure having an accurate 3D gas concentration estimation, three gas concentration sensors are placed at different heights on an antenna on the robot. The method processes measurements dynamically as they arrive, but due to practical factors like sensor latency or communication delays, measurements often arrive in small batches. Unfortunately, neither the paper nor the

code specifies how the robot decides which measurements to take, the path-planning strategy, or the sampling frequency, focusing instead on processing the measurements and solving the factor graph once data is provided. So do not know for example if the robot goes towards the direction where the maximum concentration is measured or takes more measurements when it arrives in a high concentration area.

We ran their implementation on the example datasets provided in the repository. These datasets are ROS bags that played the data from a pre-recorded real-time 3D scenario where the robot explores an unknown complex environment with obstacles with different gas source locations. By doing so, we confirmed that their method successfully constructs 2D and 3D gas distribution maps in real time, efficiently solving the inference problem using their hybrid Gaussian Belief Propagation algorithm. The output consists of a visual 3D map of the environment containing the estimated gas concentrations updated in real-time. This initial exploration was crucial to understand how their approach works but we were unable to evaluate and to compare it to the kernel-based method ourselves. This is why we needed to adapt their method and algorithm to be able to test it on our data, evaluate it and compare it to the kernel-based method.

### 2.2.2 Adapting the Method for Evaluation.

To evaluate the method, we created our own implementation, where we incorporated the key elements of their Gaussian Belief Propagation algorithm and adapted it for testing on pre-recorded gas concentration measurement data instead of on real-time data. Moreover, it should be noted that in our implementation, only the wildfire message-passing algorithm is used since, in the original work, the residual belief propagation message-passing algorithm is used to complement it when the wildfire loop is done and no new measurement needs to be processed. Our goal was to test the algorithm in specific scenarios, evaluate its performance (in Gas Source Localization and Gas Distribution Mapping), and ultimately compare it to a kernel-based method applied to the same data. This implementation allowed for flexibility in testing various conditions and assessing the robustness of the algorithm.

**Data Used for Evaluation.** The evaluation was performed using gas concentration measurements collected from different environments, each with distinct gas source locations and spatial configurations. However for all of these environment, the size of the environment was the same, a space of 20x4 meters. For each dataset, the following details were included:

- **Gas concentration readings ( $C$ ):** Measured gas levels at specific points in the environment.
- **Robot positions ( $x, y, z$ ):** The spatial coordinates of the robot at the time of each measurement.
- **Obstacles location:** A .txt file containing the number of obstacles present in the environment and the coordinates of the diagonal corners of each of these obstacles.

In order to test the algorithm on different scenarios, we not only used different maps with different gas source locations and different obstacles positions but we also tested, for each map, different sampling frequency (using only 1 out of 5 measurements for example) and different spatial frequency (using spaced measurements in the x direction). Using these different scenarios helped us simulate the measurements taken by a real robot and evaluate the performance of the algorithm in difficult settings, to see where it fails.

**Structure of Our Implementation.** The implementation was structured in four modules, as follows, to adapt the original algorithm for systematic evaluation:

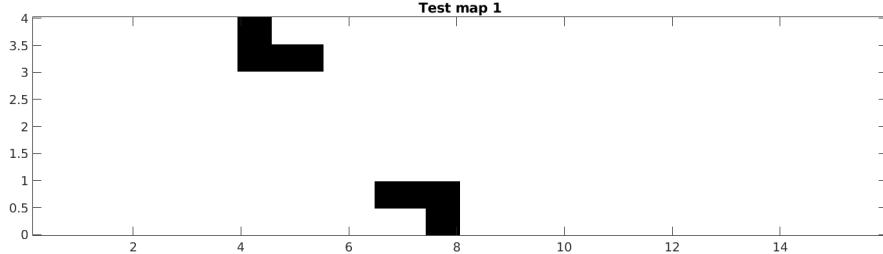


Figure 2.5: Example of map that we use for testing (obstacles in black).

1. **Input Handling:** The code reads pre-recorded measurement data, including gas concentrations and robot positions. A grid is built to match the one used for the recording of measurements: each measurement is the center of a grid cell. The environment where measurements have been taken is divided into a discretized space with 64 cells in the x direction and 64 in the y direction. The same grid is used for all scenarios. This module also creates an occupancy grid for the same grid to encode the cells occupied by an obstacle.
2. **Gaussian Belief Propagation:** This module contains the core algorithm for dynamic factor graph construction and performs belief propagation on batches of measurements. In our implementation, only the wildfire scheduling message-passing algorithm is used since, in the original method, the residual belief propagation one is only used when there is no new measurements when the wildfire loop is done.
3. **Scenario Testing:** This module allows for testing various scenarios by adjusting one of the two following parameters (see Figure 2.6 for a visualization of the measurements when we change these parameters):
  - **Sampling frequency:** How often gas concentration measurements are taken over time (in number of grid cells)
  - **Spatial resolution:** How spaced in the x-direction the measurements are (in number of grid cells).

These scenarios helped identify the conditions under which the algorithm performs well and where it fails.

**Metrics for Evaluation.** To evaluate the performance of the Gaussian Belief Propagation algorithm under different parameters, settings, and measurement conditions, the following four metrics were used.

- **Root Mean Square Error (RMSE):** This metric measures the difference between the estimated gas concentrations ( $\hat{C}_i$ ) and the ground truth values ( $C_i$ ), providing an overall measure of accuracy. It is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (C_i - \hat{C}_i)^2},$$

where  $N$  is the total number of measurement points.

- **Shape Coverage Metric (SCM):** This metric, introduced in the paper [10], evaluates how well the algorithm captures the overall shape of the gas distribution. It focuses on

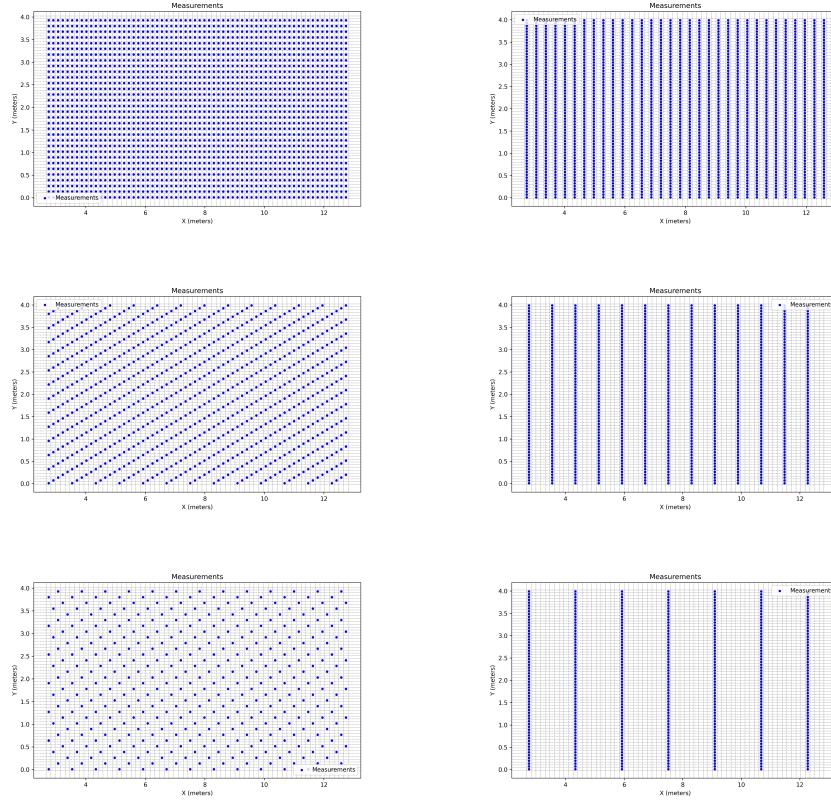


Figure 2.6: Measurements with a sampling frequency of 2, 5 and 10 (left column) and measurements with a spatial frequency of 2, 5 and 10 (right column)

comparing the spatial extent and boundaries of the estimated plume with the ground truth. The SCM is computed as:

$$\text{SCM} = \frac{|E \cap G|}{|G|},$$

where  $E$  is the set of grid cells in the estimated gas distribution above a concentration threshold,  $G$  is the corresponding set of grid cells in the ground truth distribution, and  $|E \cap G|$  represents the intersection of these two sets. The SCM provides a value between 0 and 1, where higher values indicate better coverage of the plume shape.

- **Gas Source Localization Error:** To evaluate the accuracy of the algorithm in identifying the gas source location, we used a gas source localization error metric. This metric measures the Euclidean distance between the estimated source location  $(\hat{x}_s, \hat{y}_s)$  and the true source location  $(x_s, y_s)$ , and is defined as:

$$\text{Source Error} = \sqrt{(\hat{x}_s - x_s)^2 + (\hat{y}_s - y_s)^2}.$$

The metric provides a direct assessment of how well the algorithm can localize the gas source based on the estimated gas distribution map.

- **Computational Time:** This metric assesses the algorithm's efficiency by measuring the time required to generate the distribution map. It provides an indication of the method's usability for real-time applications.

**Hyperparameter Tuning.** Before running tests and evaluating the performance of the algorithm, we performed hyperparameter tuning to identify the optimal parameters for our specific case. This involved testing different values for key parameters, including the update threshold ( $\epsilon$ ), observation weight ( $\lambda_o$ ), and regularization weight ( $\lambda_p$ ). These parameters control different aspects of the Gaussian Belief Propagation algorithm:

- $\lambda_p$  (regularization weight): controls the strength of the spatial correlations between neighboring nodes in the factor graph. A smaller  $\lambda_p$  induces stronger correlations, resulting in smoother gas distribution estimates, while a larger  $\lambda_p$  allows for more localized variations in the map.
- $\lambda_o$  (observation weight): controls how much influence the sensor measurements have on the estimated gas concentrations. A higher  $\lambda_o$  increases the weight given to the observed data, making the algorithm rely more on the measurements, while a lower  $\lambda_o$  reduces their impact, allowing prior information and spatial correlations to dominate.
- $\epsilon$  (convergence threshold): This parameter sets the threshold for message updates during belief propagation. A smaller  $\epsilon$  results in more precise convergence at the cost of increased computational time, while a larger  $\epsilon$  speeds up the process but may lead to less accurate results.

We explored different combinations of these parameters by running the Gaussian Belief Propagation (GBP) algorithm on the same scenario, evaluating its performance using metrics such as the root mean square error (RMSE) of gas concentrations and the source localization error. The scenario used for this operation was one without obstacles but with a sampling frequency of 10, to have sparse data like in a real-life scenario. After processing each set of parameters, we found that the optimal hyperparameters in our case were:  $\lambda_p = 0.5$ ,  $\lambda_o = 15.0$ , and  $\epsilon = 1e-3$  (for the original method they were set to  $\lambda_p = 0.5$ ,  $\lambda_o = 10.0$ , and  $\epsilon = 1e-4$ ). Table 2.1 sums up the results of our hyperparameter tuning.

Epsilon	Lambda_o	Lambda_p	RMSE	Source Error
0.0001	5.0	1.0	1039.97	0.1586
0.0001	5.0	5.0	1328.61	0.1586
0.0001	10.0	1.0	978.68	0.1586
0.0001	10.0	5.0	1181.59	0.1586
0.001	5.0	1.0	1039.96	0.1586
0.001	5.0	5.0	1328.14	0.1586
0.001	10.0	1.0	978.65	0.1586
0.001	10.0	5.0	1181.43	0.1586
0.01	5.0	1.0	1039.94	0.1586
0.01	5.0	5.0	1328.61	0.1586
0.01	10.0	1.0	978.65	0.1586
0.01	10.0	5.0	1181.32	0.1586
0.0001	10.0	0.5	943.60	0.1586
0.0001	15.0	0.5	931.52	0.1586
0.0001	15.0	1.0	956.03	0.1586
0.001	10.0	0.5	943.58	0.1586
0.001	15.0	0.5	931.50	0.1586
0.001	15.0	1.0	955.99	0.1586
0.01	10.0	0.5	943.58	0.1586
0.01	15.0	0.5	931.50	0.1586

0.01	15.0	1.0	955.99	0.1586
0.0001	5.0	0.5	978.09	0.1586
0.0001	15.0	5.0	1112.47	0.1586
0.001	5.0	0.5	978.08	0.1586
0.001	15.0	5.0	1112.43	0.1586
0.01	5.0	0.5	978.08	0.1586
0.01	15.0	5.0	1112.42	0.1586

Table 2.1: Hyperparameter tuning results

**Summary of Contributions.** Through this implementation, we were able to evaluate the performance of the Gaussian Belief Propagation algorithm under realistic and controlled conditions. By comparing it to the kernel-based method on the same datasets, we gained a better understanding of its advantages, limitations, and areas for improvement. This project not only highlights the potential of this algorithm for real-time gas distribution mapping and gas source localization but also provides a framework for future testing and adaptation in complex environments.

# Chapter 3

## Results

In this chapter we present an extensive evaluation of the Gaussian Belief Propagation method. As explained in the Methods chapter, our analysis is structured in four parts, each focusing on a specific aspect of the method's behavior and performance for gas distribution mapping and gas source localization.

In the first section, the effect of the sampling frequency (as defined in the Methods section and not as its literal meaning) on the gas concentration maps and source estimation is studied to see how it affects the gas distribution mapping and gas source localization results. The second section explores the impact of varying spatial frequency (in the x direction only), providing insights into the trade-offs between computational efficiency and map accuracy. In the third section we investigate the influence of obstacles on the algorithm's performance. To this end, the method was tested on four different maps: three with obstacles of varying size and location and one without obstacles. Each map was paired with four different gas source locations, allowing for a detailed analysis of how obstacles affect the estimated gas concentration maps and the accuracy of source localization. Finally, the fourth section compares the proposed method with the Kernel DMV method.

To evaluate the performance of the method in all cases and to compare it to the Kernel Method, we used four metrics, as explained in the Methods chapter: the root-mean-square error (RMSE) of the estimated gas concentration, which measures the accuracy of the generated maps; the source coverage metric (SCM), which evaluates the method's ability to capture the spatial extent of the gas plume; the source localization error (SLE), which quantifies the error between the estimated source location and the true source location; and the computational time.

### 3.1 Influence of the sampling frequency

As a reminder, in this analysis, "sampling frequency" refers to the frequency of the extracted measurements from the file containing all measurements (a frequency of five means we took one out of five measurements from the measurements file for the test). Because of how the measurements were recorded, a larger "sampling frequency" results in measurements being more spaced along the y direction. So an increase in the sampling frequency results in a decrease in the number of measurements. The results, summarized in Figure 3.2 (plots of the gas concentration maps) and Figure 3.1 (plots of the evaluation metrics), clearly demonstrate the influence of varying sampling frequency on gas distribution mapping and gas source localization.

At low sampling frequencies, the algorithm can rely on more measurements, resulting in better gas source localization and gas distribution mapping. Indeed, the SCM metric is higher in this

case, as the algorithm preserves the general shape of the gas plume. Indeed, we observe on the plots that the gas plume is narrower and better defined for low sampling frequencies, aligning with the lower RMSE values. However, computational time is maximal, as the algorithm needs to process more measurements.

At higher sampling frequencies (sparser measurements), the estimated gas plume appears wider and less defined. This occurs because the algorithm has fewer measurements to work with, requiring more extrapolation and smoothing to estimate the gas distribution over unmeasured regions. The lack of detailed measurements leads to an overgeneralized gas plume shape, which spreads wider to account for potential uncertainties in the unmeasured areas. Consequently, the shape coverage metric (SCM) decreases and the RMSE of the gas concentration increases. The source localization error increases too, except for a single decrease with a sample frequency of 30. This outlier can be explained by the fact that sometimes the algorithm is "lucky" when one of the few measurements is one that is close to the gas source. This is why we should focus on global trends. As the number of measurements decreases, the computational time decreases as well, as expected.

These results highlight the importance of balancing sampling frequency to achieve the desired trade-off between computational efficiency and accuracy. The choice of sampling frequency should depend on the specific goals of the gas distribution mapping task, such as whether the priority is computational efficiency or precise source localization or gas plume mapping.

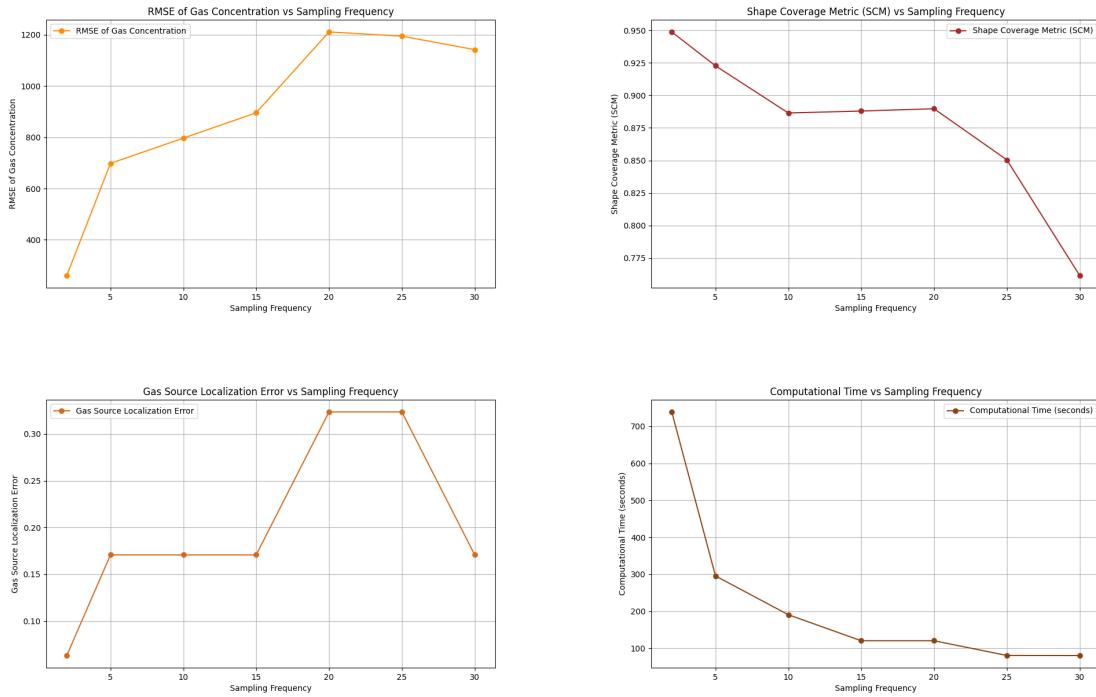


Figure 3.1: Plots of the RMSE, SCM, Source location error and computational time against sampling frequency

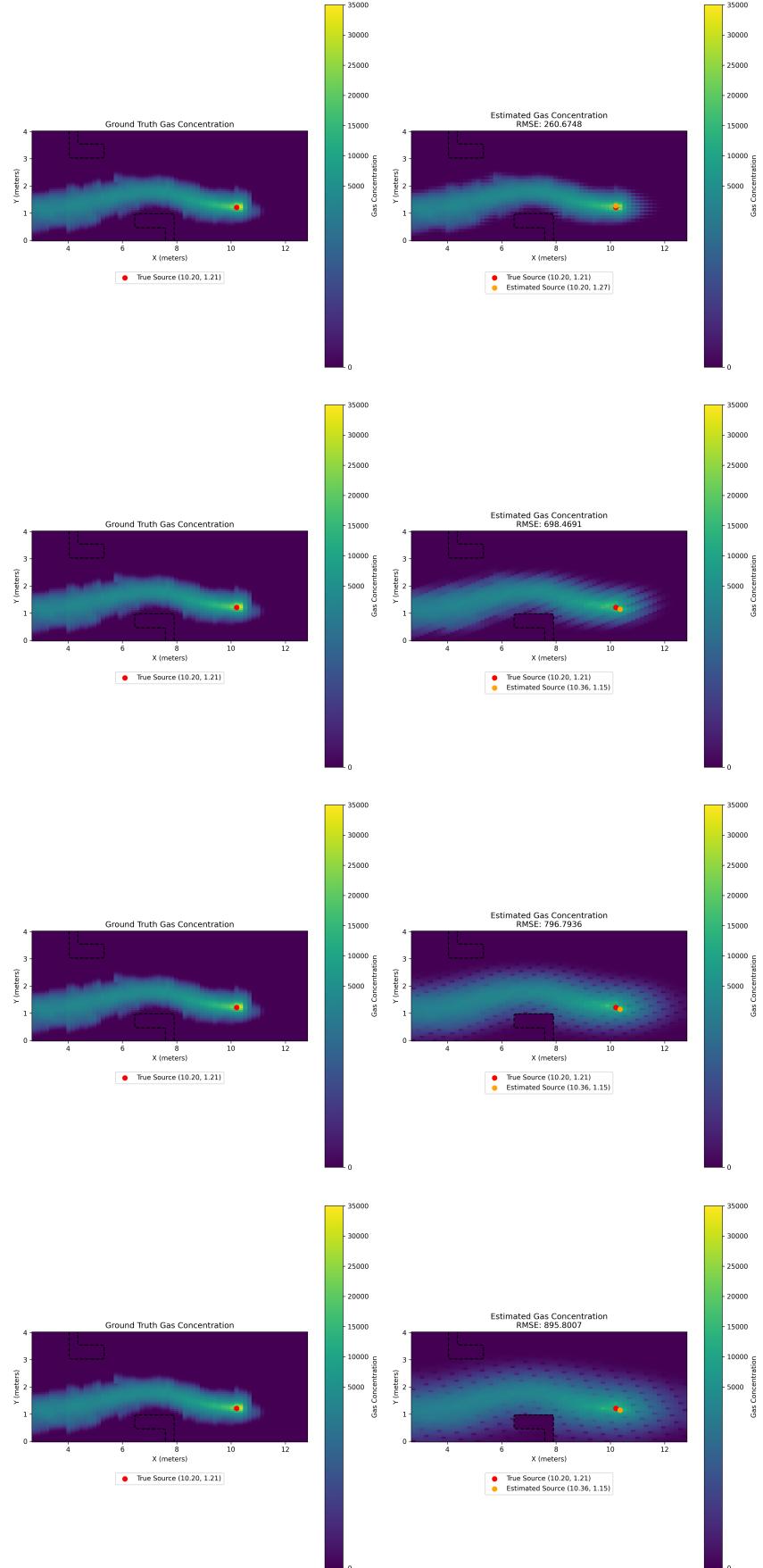


Figure 3.2: Plots of the gas concentration estimation over the map for the sampling frequencies of 2, 5, 10, 15 (from top to bottom on the right column). For each plot, the ground truth is plotted on the left

## 3.2 Influence of the spatial frequency

As a reminder, the "spatial frequency" in this analysis is defined as the sparsity of measurement lines along the x-direction while maintaining maximal measurements along the y-direction. As can be visualized on the gas concentration estimation maps with varying spatial frequencies on Figure 3.3 and the metrics plots on Figure 3.4, the spatial frequency of the measurements significantly impacts the estimated gas plume and source localization.

At higher spatial frequencies (e.g., 10 or 15), where the measurements are more widely spaced along the x-axis, the reconstructed gas plume becomes less detailed, appearing patchy and less accurate in the estimated gas concentration maps (except for the x for which there are measurements, clearly visible on the plots). This is due to the reduced amount of data available for interpolation, leading to higher RMSE values, as shown in the metrics plot, and a wider plume.

In contrast, lower spatial frequencies (e.g., 2 or 5), where measurement lines are more densely distributed along the x-axis, result in more accurate and smoother gas concentration maps. The increased density of measurements provides more data points, enabling better interpolation and representation of the gas plume. Consequently, lower spatial frequency is associated with smaller RMSE values and a more accurate representation of the true gas plume.

The gas source localization error also shows a strong dependence on spatial frequency. Globally, at higher spatial frequencies (widely spaced measurements), the source localization error increases significantly (except for some outliers due to the potential "luck" of having, in the few measurements available, some close to the source, as discussed in the previous section). This is likely due to the lack of sufficient measurements in the x-direction to triangulate the source accurately. At lower spatial frequencies, the localization error decreases, showing better source identification due to the richer data.

Similarly, the shape coverage metric (SCM) improves with decreasing spatial frequency, indicating that lower spatial frequencies capture the shape fo the gas plume more effectively. We also observe that the computational time decreases as spatial frequency increases, as fewer measurements reduce the computational load. While this may be a practical advantage, the trade-off is a significant reduction in mapping and localization accuracy.

These trends are clearly visible in the maps of the reconstructed gas plumes (Figures X-Y) and the metrics plots (Figure Z) and are very similar to the ones presented in the previous analysis. As expected, the analysis for both type of "frequencies" shows us the trade-off between the accuracy of the gas source localization and gas plume mapping and the computational efficiency, depending on the application's requirements.

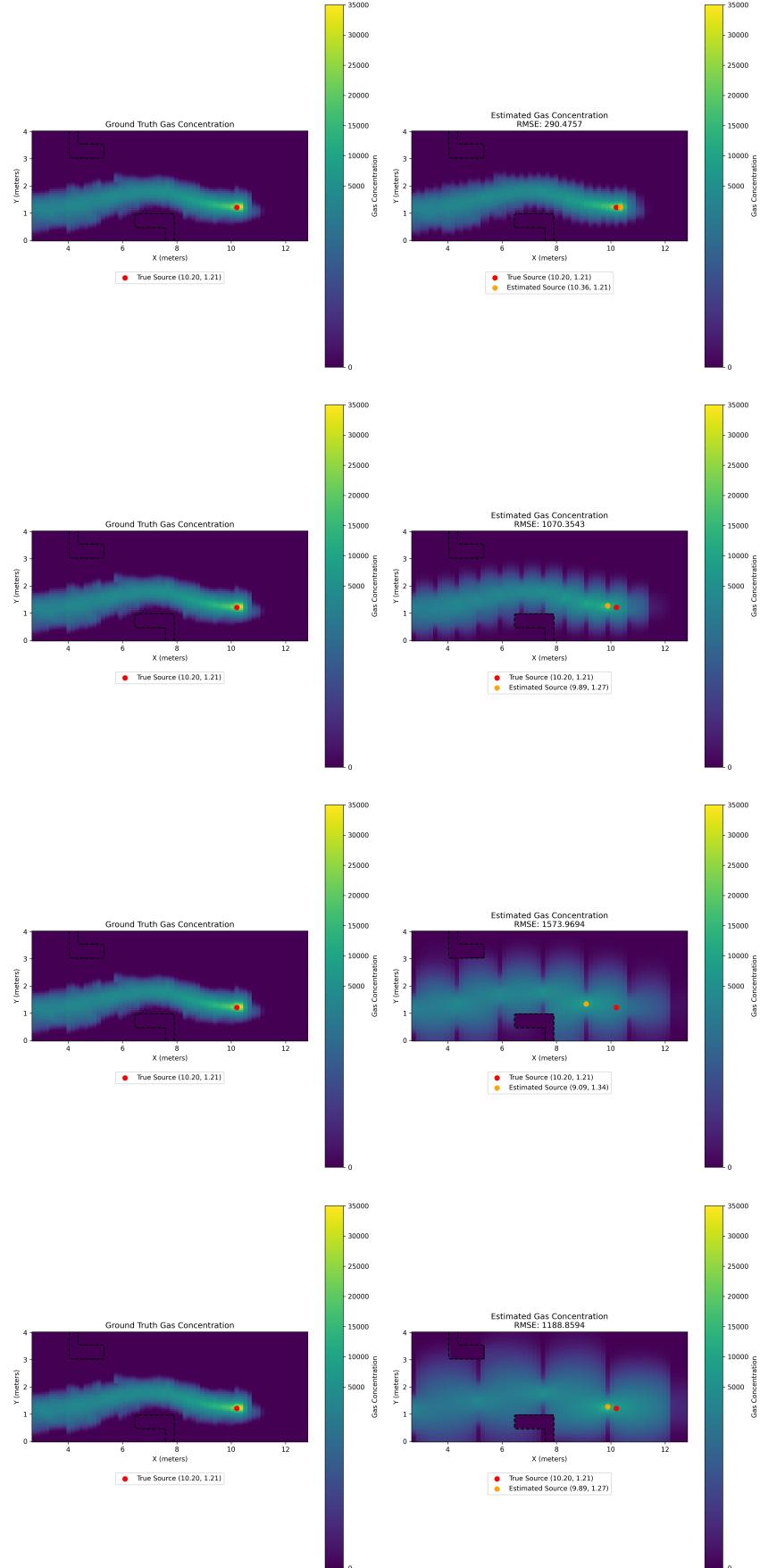


Figure 3.3: Plots of the gas concentration estimation over the map for the spatial frequencies of 2, 5, 10, 15 (from top to bottom on the right column). For each plot, the ground truth is plotted on the left

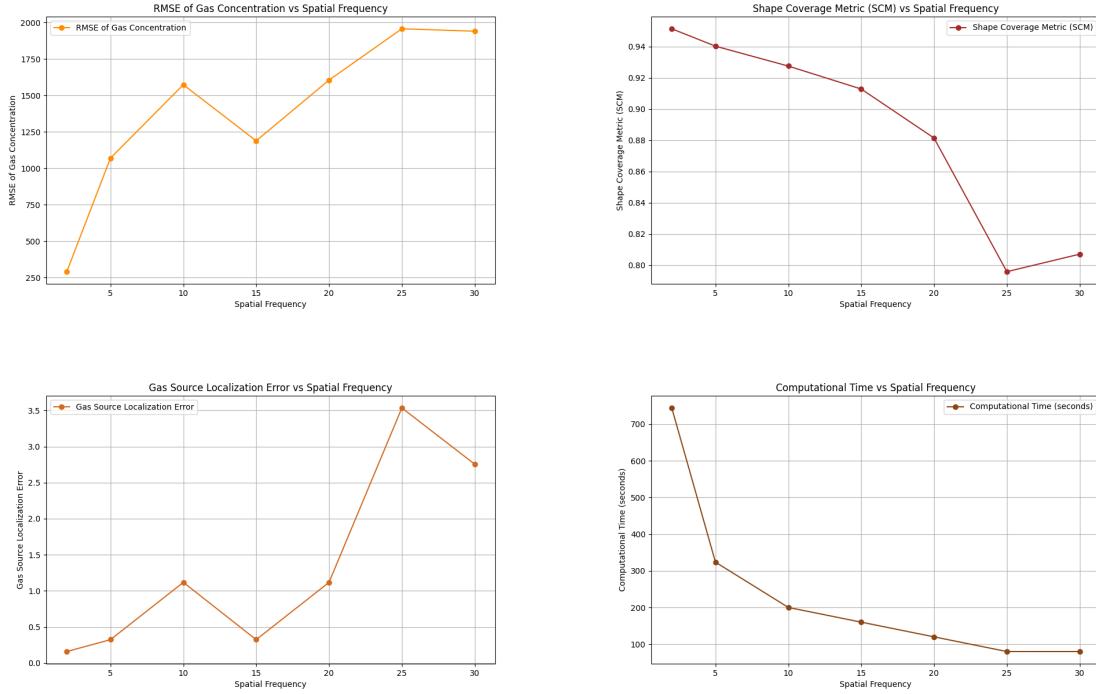


Figure 3.4: Plots of the RMSE, SCM, Source location error and computation time against the spatial frequency

### 3.3 Influence of the obstacles

This section explores the impact of obstacles in the environment on the gas distribution mapping results and source localization. Four maps were tested: Map A, which has no obstacles, and Maps B, C, and D, which include varying obstacle configurations and different gas source locations (see Figure 3.5). The analysis of the gas concentration plots across these maps shows that the presence of obstacles does not significantly change the general shape of the gas plume. Instead, the trends in plume spread and source localization appear to be driven primarily by the sampling and spatial frequencies of the measurements. As noted in the previous sections, at lower sampling frequencies (more measurements), the gas plume is narrower and more precise across all maps, while at higher sampling frequencies (fewer measurements), the plume is wider and less defined, regardless of the presence or configuration of obstacles. The example of the estimated gas concentrations for Map D for different sampling frequencies and different spatial frequencies are shown on Figure 3.6 and Figure 3.7 respectively.

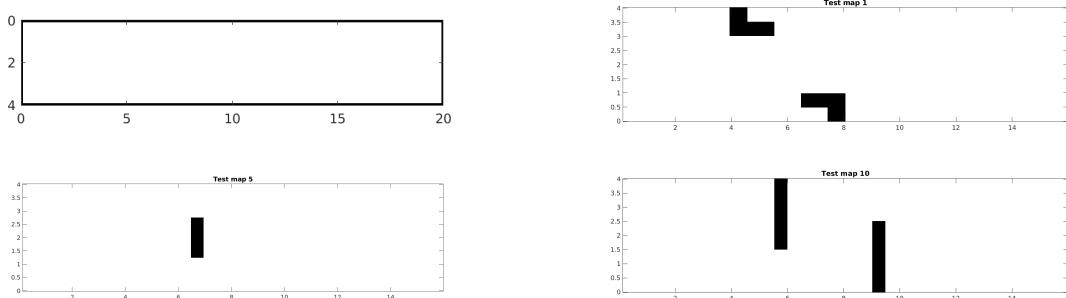


Figure 3.5: Images of the four maps tested (A, B, C, D from left to right and top to bottom)

The metrics (Figure 3.8 and Figure 3.9) provide quantitative insights into this observation. Indeed, the RMSE and source localization error globally follow the same trends across all four maps, showing an increase with lower sampling and spatial frequencies. Surprisingly, Map A (no obstacles) shows a globally higher RMSE over sampling frequencies than other maps (besides one outlier). However, another difference is particularly noticeable for the Shape Coverage Metric (SCM) (over sampling and spatial frequencies), where Map A outperforms the other maps (besides an outlier), indicating a slightly better representation of the true plume shape in the obstacle-free environment for most sampling and spatial frequencies.

The computational time shows minimal variation across the four maps, reinforcing that obstacles have negligible influence on the algorithm’s efficiency. Overall, the performances vary across the maps and across the metrics and we cannot establish a direct influence of the obstacles on the results. The dominant factors affecting the results are the sampling and spatial frequencies, which drive the resolution and accuracy of the estimated gas plume and source localization. These findings highlight the robustness of the method to environmental configurations. However, it should be noted that the comparison was done across maps with different obstacles and gas source locations. Future work may include a comparison between maps with the same gas source location with and without obstacles.

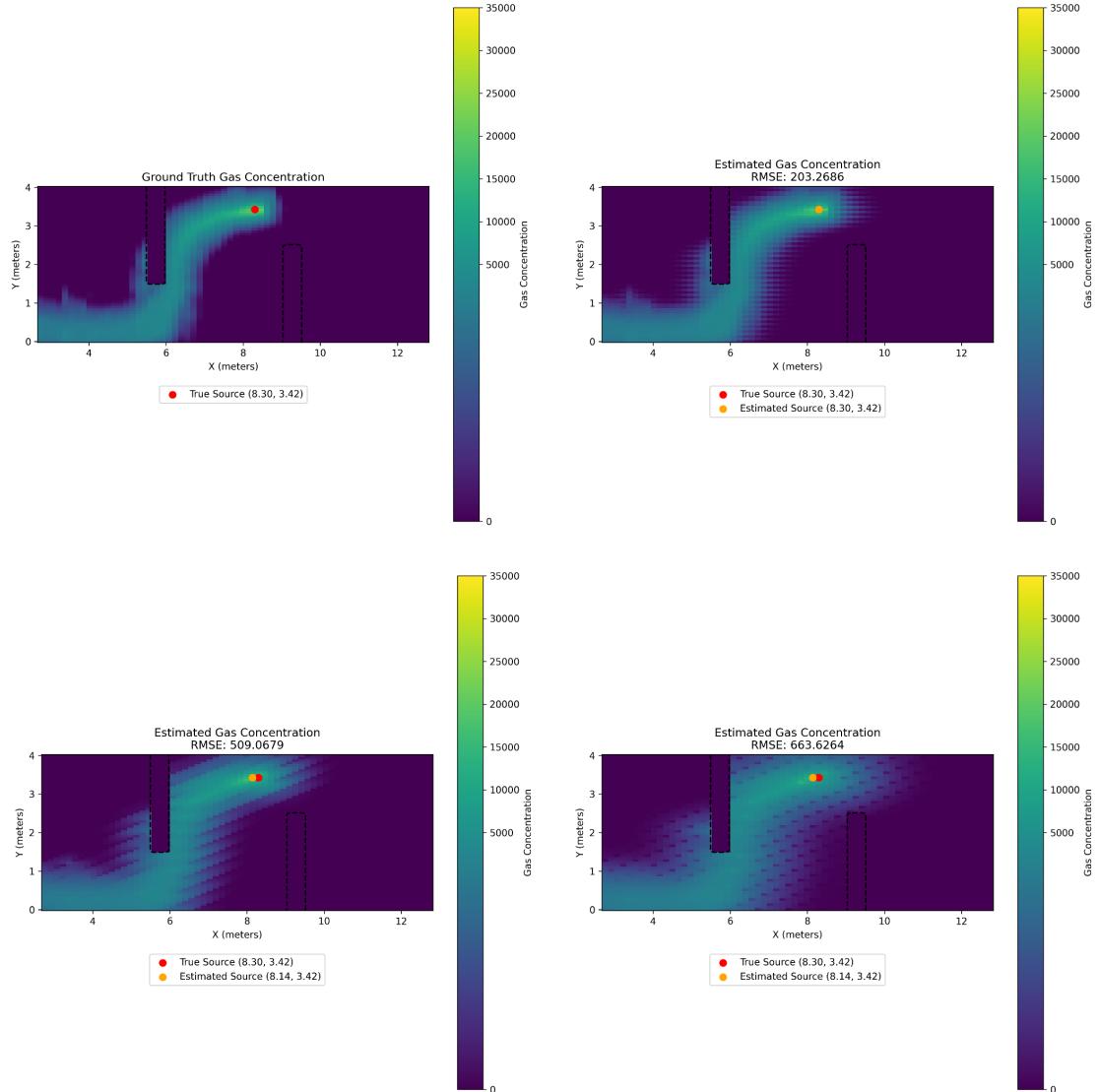


Figure 3.6: Ground truth (top left) and estimated gas concentration of the map D for the sampling frequencies 2 (top right), 5 (bottom left) and 10 (bottom right)

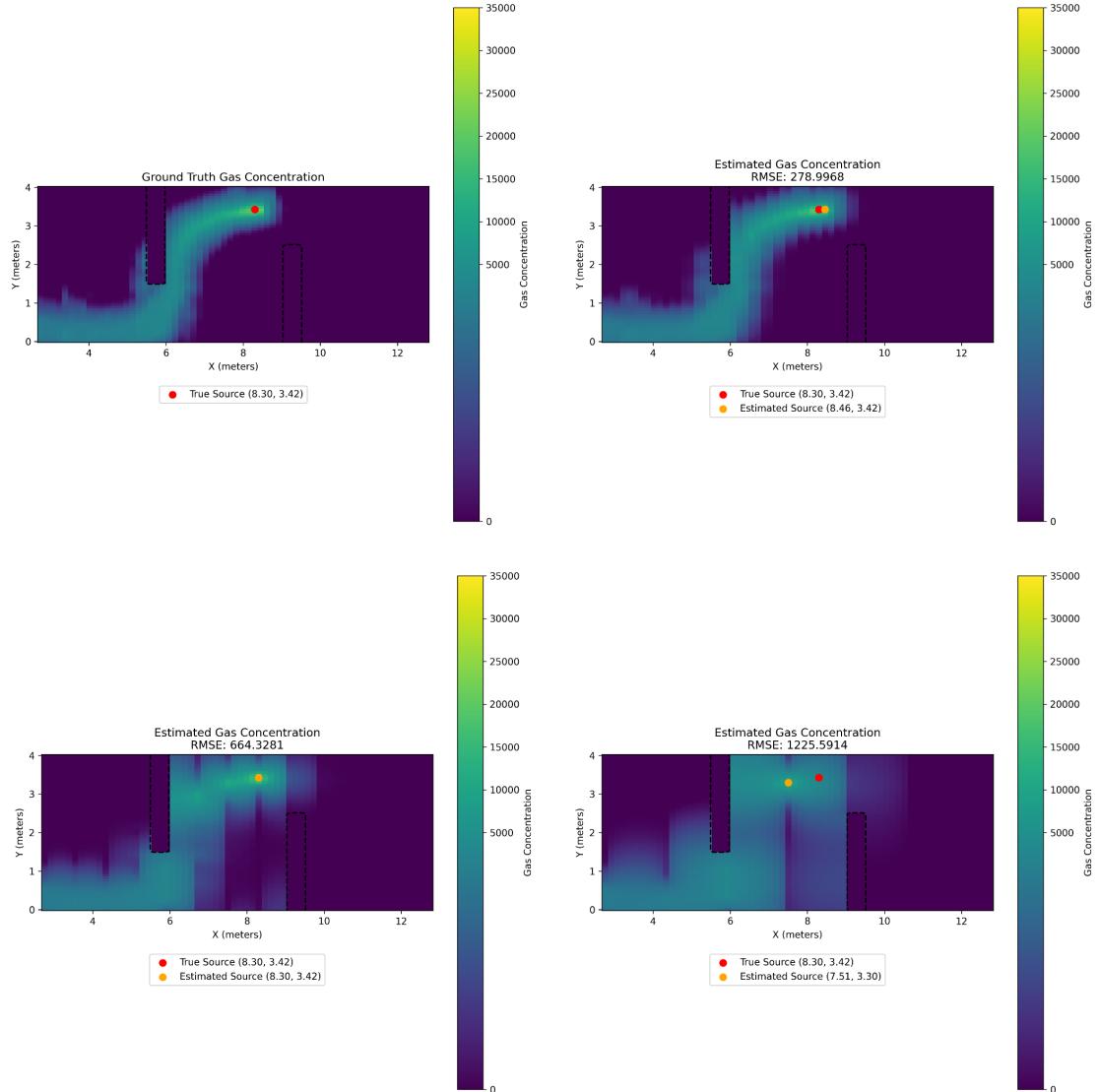


Figure 3.7: Ground truth (top left) and estimated gas concentration of the map D for the spatial frequencies 2 (top right), 5 (bottom left) and 10 (bottom right)

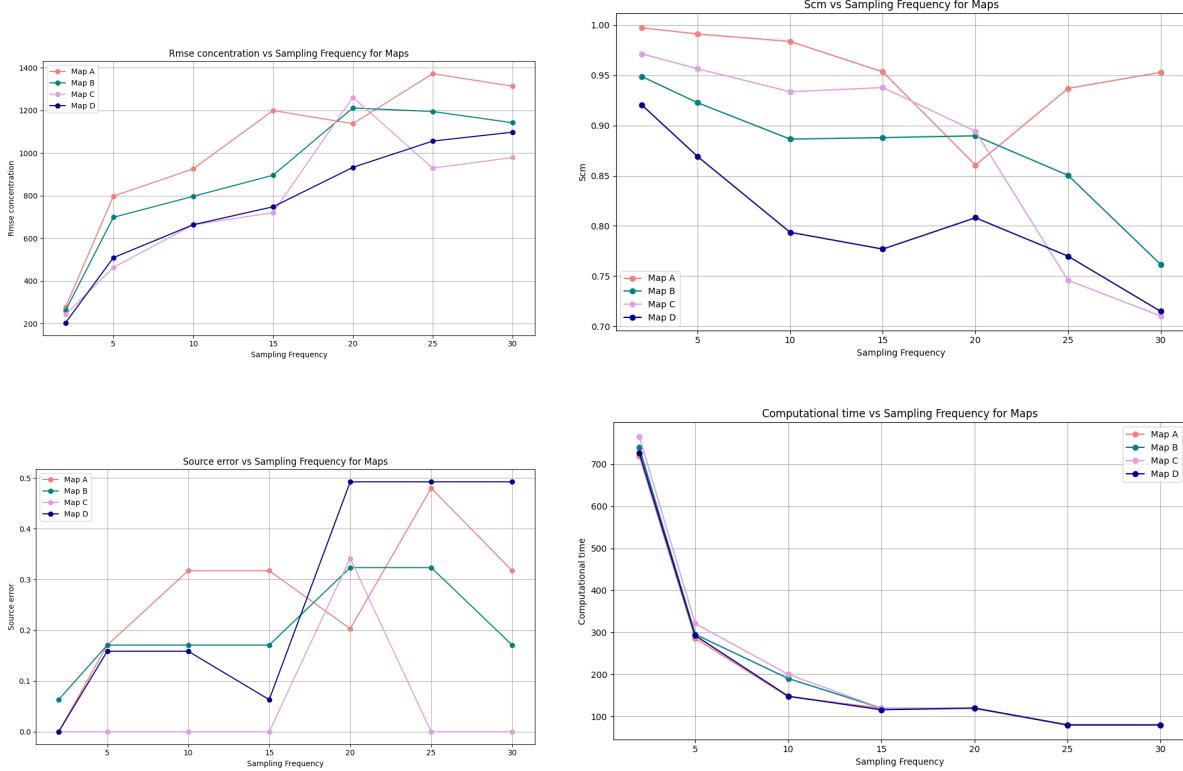


Figure 3.8: Comparison of RMSE, SCM, Source localization error and the computational time for different sampling frequencies for each map

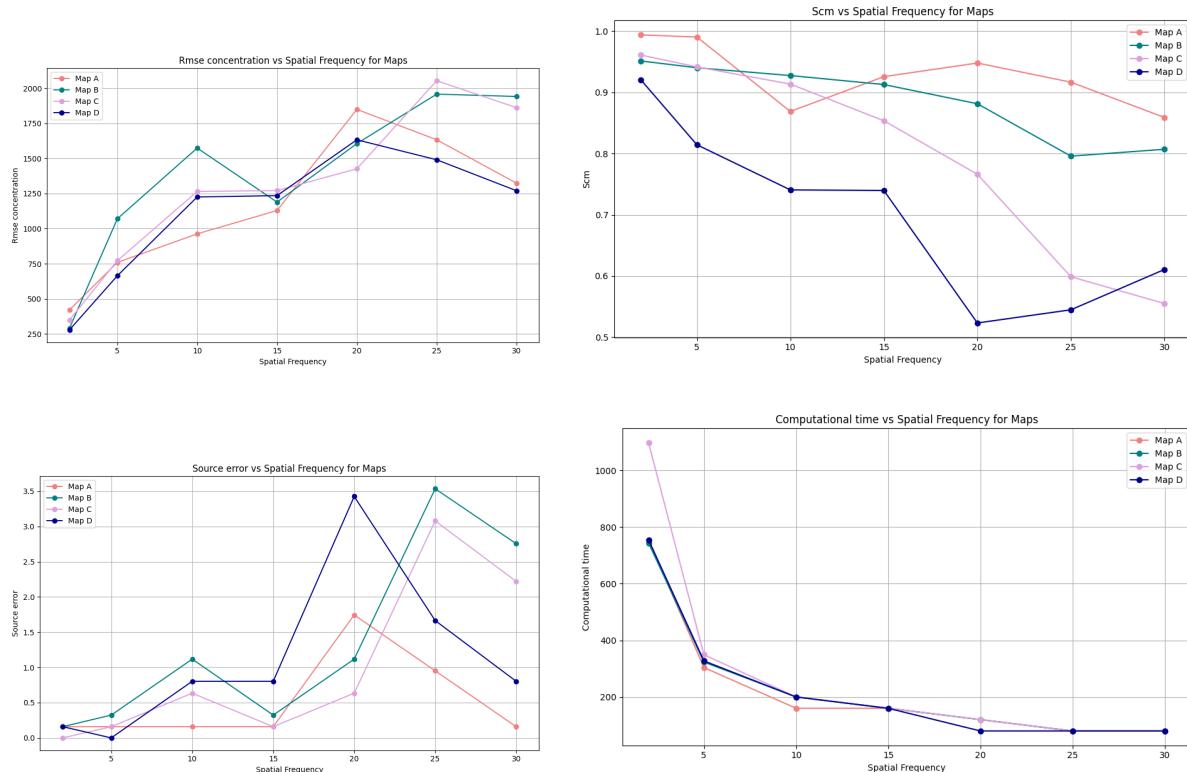


Figure 3.9: Comparison of RMSE, SCM, Source localization error and the computational time for different spatial frequencies for each map

### 3.4 Comparison to the Kernel DMV Method

Finally, to compare the performance of the Gaussian Belief Propagation (GBP) method and the Kernel DMV method, we analyzed their behavior across varying sampling and spatial frequencies on the obstacle-free map (map A), using the metrics of RMSE, SCM, source localization error, and computational time. The results, illustrated in 3.12 to 3.14, provide a detailed comparative analysis. The ground truth gas concentration is represented on Figure 3.10. We observe

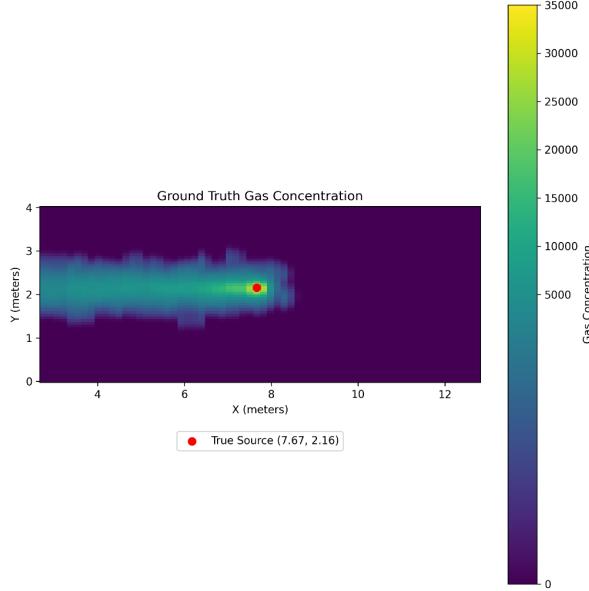


Figure 3.10: Ground truth concentration of the map used for comparing the Kernel DMV method and GBP (with a log scale between 0 and 5000)

that the gas plumes generated by the Kernel DMV and GaBP methods show distinct patterns as frequencies increase (fewer measurements). The Kernel DMV method produces fragmented and discontinuous plumes at higher frequencies due to its reliance on dense measurements for smooth interpolation. In contrast, the GaBP method maintains smoother and more continuous plumes even with sparse measurements, thanks to its probabilistic framework and ability to incorporate spatial priors. However, at very high frequencies, the GBP method exhibits broader, less detailed plumes, while the Kernel method exhibits tighter plumes.

The GBP method demonstrated a slightly superior gas distribution mapping performance, particularly at higher frequencies (both spatial and sampling), as evidenced by lower RMSE values in the corresponding plots for most frequencies between 10 and 30. This highlights GBP's robustness in accurately reconstructing the gas plume with sparser measurements. This out-performance can be seen on the Shape Coverage metric plots. Indeed, as could be expected when looking at the resulting gas plumes, the shape coverage metric of the Kernel method is significantly lower than the one of the GBP method (even though the GBP method shows surprising trends).

When examining the source localization error, the GBP method generally achieved lower errors compared to the Kernel DMV method for the varying spatial frequencies but not particularly for the different sampling frequencies.

From a computational efficiency perspective, the GBP method required significantly more computational resources, as reflected in the computational time plots. This trade-off is critical when selecting a method for real-time applications, where computational efficiency might be prioritized over accuracy.

In summary, the GBP method slightly outperformed the Kernel DMV method in terms of gas plume reconstruction (especially at higher frequencies) and not particularly for the source localization error (only for different spatial frequencies). However, this comes at a cost of increased computational time. But it's important to keep in mind that we might not have found the optimal parameters for each hyperparameter, as we only tested three values for each and selected the combination with the smallest RMSE. Future work could focus on further tuning and testing additional values, though this was not feasible in our case due to the computational cost of such evaluations. Other metrics could also be used for further evaluation and comparison of the two methods.

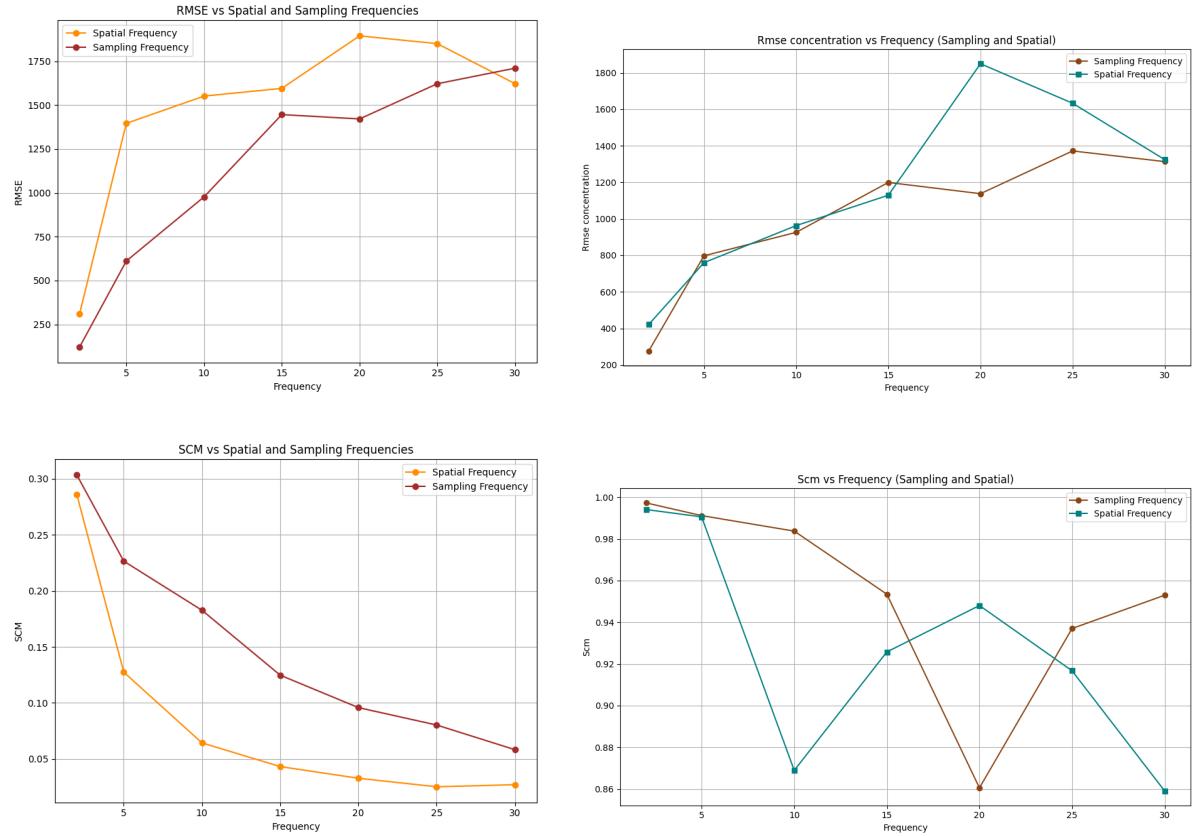


Figure 3.11: Comparison of the RMSE and SCM for spatial and sampling frequencies between the Kernel DMV Method (left column) and the GBP Method (right column)

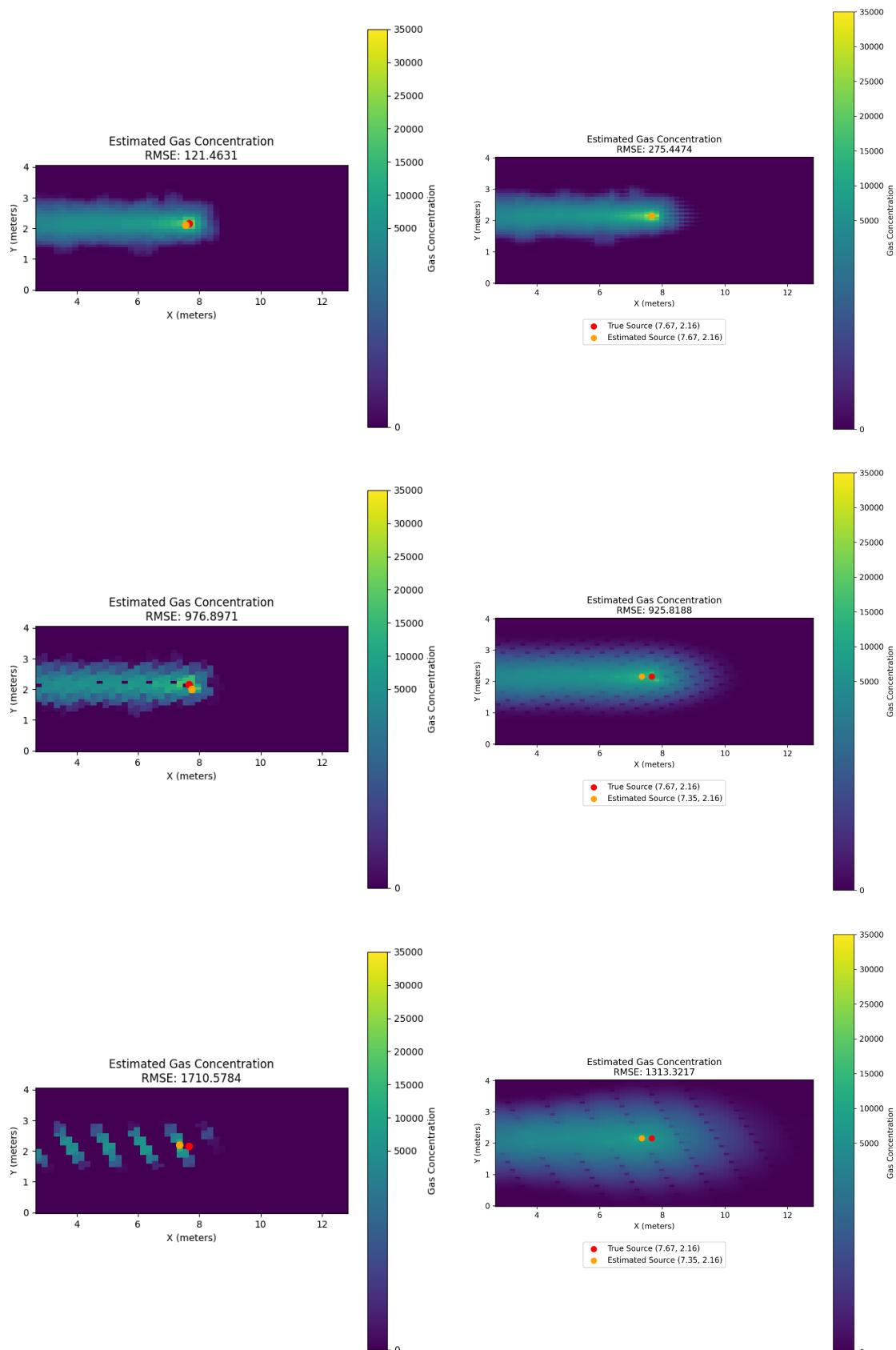


Figure 3.12: Comparison of the gas concentration estimation for different sampling frequencies (2, 10, and 30 from top to bottom) between the Kernel DMV Method (left column) and the GBP Method (right column)

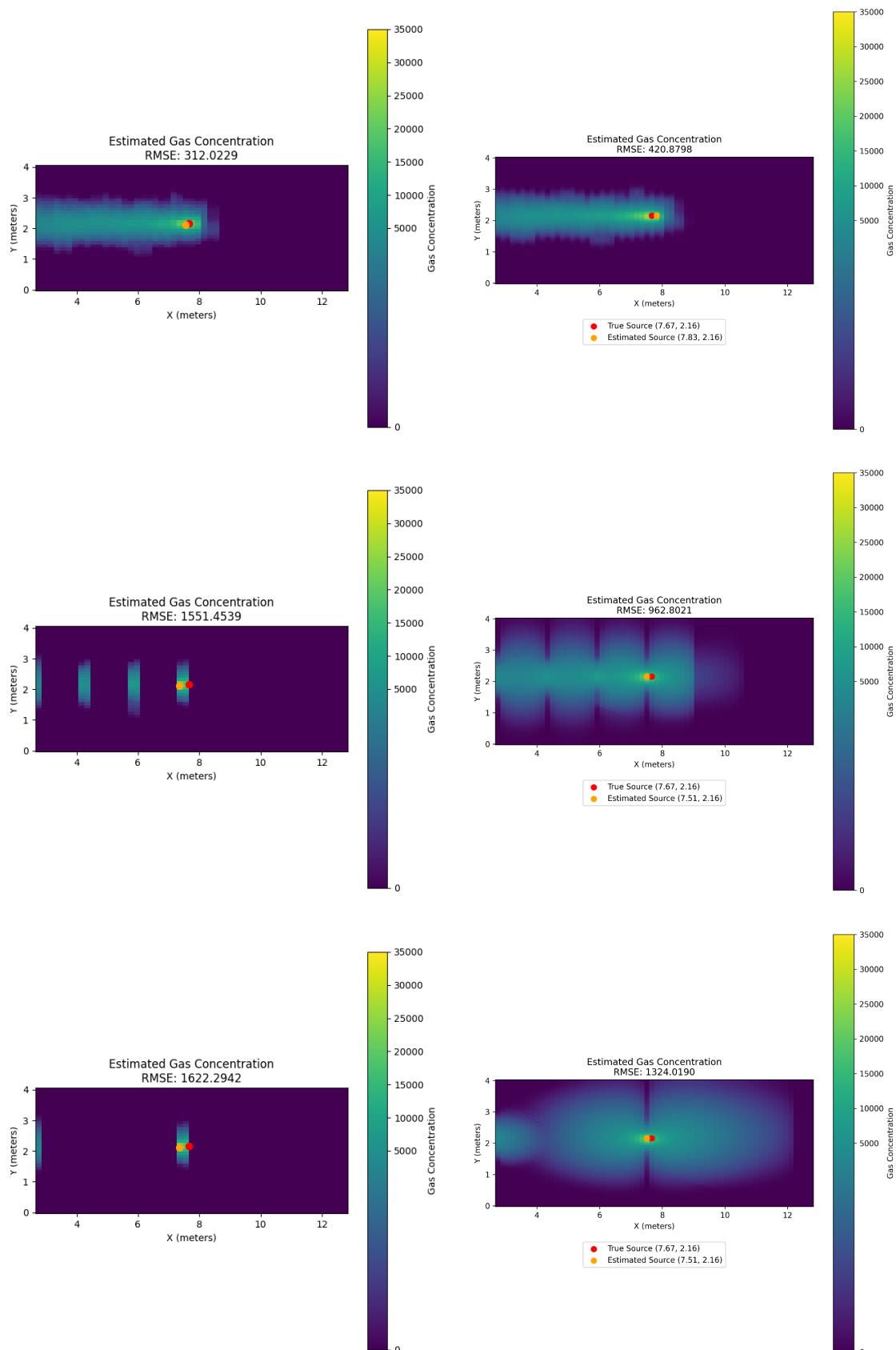


Figure 3.13: Comparison of the gas concentration estimation for different spatial frequencies (2, 10, and 30 from top to bottom) between the Kernel DMV Method (left column) and the GBP Method (right column)

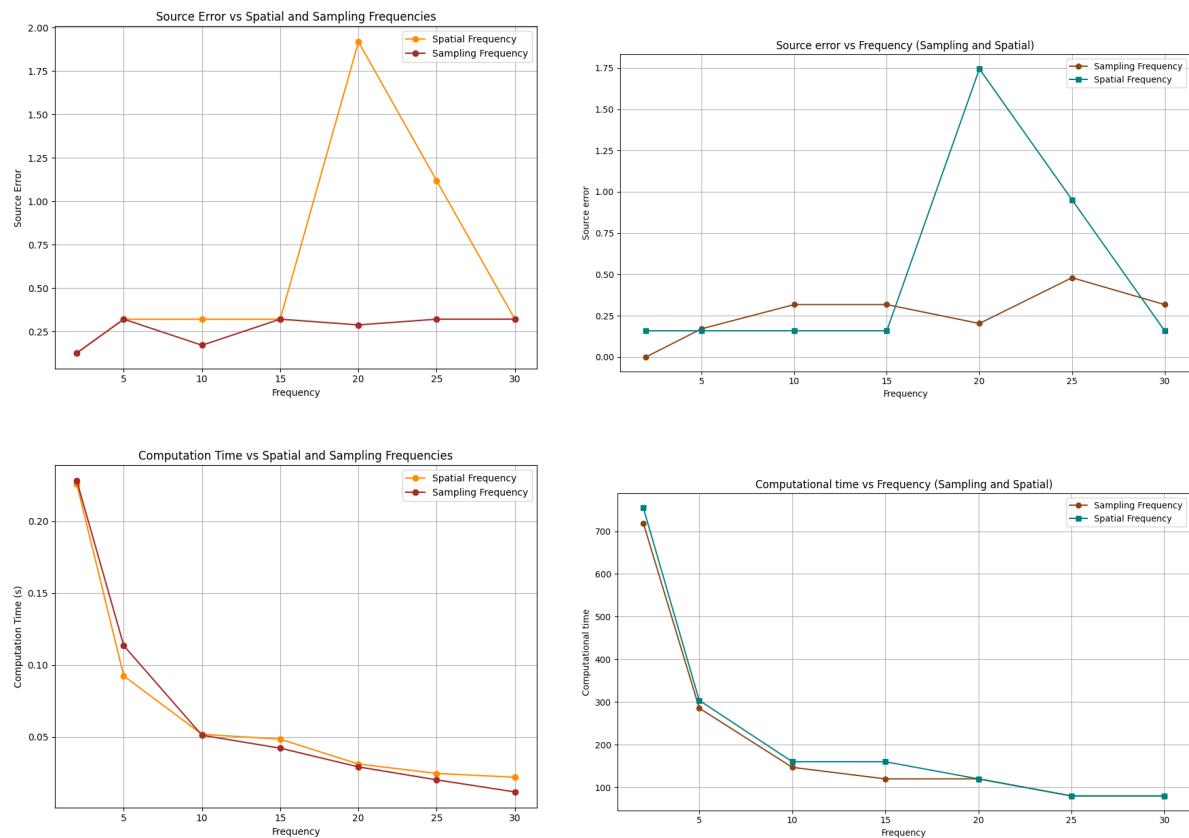


Figure 3.14: Comparison of the Source localization error and the computational time for spatial and sampling frequencies between the Kernel DMV Method (left column) and the GBP Method (right column)

# Chapter 4

## Conclusion

The first objective of this project was to investigate and evaluate the use of Gaussian Markov Random Fields (GMRFs) and Gaussian Belief Propagation (GBP) for gas distribution mapping and gas source localization. The work began with a comprehensive understanding of GMRFs and GBP through an extensive literature review and hands-on testing of the existing implementation. This groundwork allowed us to explore the strengths and limitations of the GBP method proposed by Rhodes et al. and provided insight into its computational framework and real-time applicability.

Building on this understanding, we adapted the GBP method to work with pre-recorded gas concentration data in specific environments. This required significant modifications to the original codebase to accommodate different data structures and scenarios and adapt it from a real-time ROS implementation. By incorporating flexibility into the implementation, we tested the method under varying sampling and spatial frequencies, as well as in environments with and without obstacles, allowing for a robust assessment of its capabilities.

The results of our evaluation demonstrated the algorithm's ability to estimate gas concentration maps and localize gas sources rather accurately, particularly at lower sampling and spatial frequencies. However, these benefits came at the cost of increased computational time. The influence of obstacles was found to be minimal, with performance primarily dictated by the density of measurements. In comparison to the Kernel DMV method, GBP showed slightly superior performance in terms of gas plume reconstruction when fewer measurements are available, though it required greater computational resources.

While the findings of this project highlight the potential of GBP for gas distribution mapping and gas source localization, they also point to areas for future exploration. Hyperparameter tuning was limited by computational constraints, leaving room for further optimization to enhance both accuracy and efficiency, by testing different parameters. Moreover, for our implementation, only the wildfire message-passing algorithm was used (since the residual belief propagation in the original work is called when there is no new measurement), future work should evaluate the performances of the algorithm when using this second message-passing algorithm as well. This would be particularly important for an evaluation with real-time measurement processing on a real-time application, which would be a relevant way to evaluate and compare the GBP method to the Kernel-based method (combined with a path-planning algorithm for example, to "choose" where to take measurements). Additionally, the comparison of maps with and without obstacles could benefit from experiments that isolate the effects of obstacles more precisely (using the same gas source location for all maps for example). Finally, further enhancements to this method could also include other measurements such as wind measurements.

# Bibliography

- [1] L. C. Rhodes, C. and W.-H. Chen, “Structurally aware 3d gas distribution mapping using belief propagation: A real-time algorithm for robotic deployment,” in *2022 IEEE/RS International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022. The work extends Gaussian Belief Propagation (GBP) to 3D gas distribution mapping with real-time applicability.
- [2] A. J. Lilenthal and T. Duckett, “A method for gas distribution mapping using a mobile robot,” *IEEE Sensors Journal*, 2003.
- [3] J. Smith and A. Jones, “Kernel dm+v: A framework for gas distribution modeling and prediction,” 2010.
- [4] Author(s), “Kernel dm+v/w: Incorporating wind information and variance estimation into gas distribution mapping,” *Journal/Conference Name*, Year. Citation details need to be completed.
- [5] S. Brown and D. Lee, “Td kernel dm+v/w for time-decay gas distribution mapping with wind integration,” 2015.
- [6] C. Stachniss *et al.*, “Gaussian processes for mobile robot gas distribution mapping,” *Robotics Journal*, 2008.
- [7] J. Monroy, J. Blanco, and J. Gonzalez-Jimenez, “Time-variant gas distribution mapping with obstacle information,” *Autonomous Robots*, vol. 40, no. 1, pp. 1–16, 2016.
- [8] C. Rhodes, C. Liu, and W.-h. Chen, “Scalable probabilistic gas distribution mapping using gaussian belief propagation,” in *IROS*, 2022.
- [9] C. Rhodes, “Gaussian belief propagation for 2d and 3d gas distribution mapping.” [https://github.com/callum-rhodes/gabp<sub>m</sub>apping](https://github.com/callum-rhodes/gabp_mapping), 2024. Accessed : 2025 – 01 – 08.
- [10] C. Ercolani, L. Tang, A. A. Humne, and A. Martinoli, “Clustering and informative path planning for 3d gas distribution mapping: Algorithms and performance evaluation,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8960–8966, IEEE, 2021.