GaSLAM: An Algorithm for Simultaneous Gas Source Localization and Gas Distribution Mapping in 3D

Chiara Ercolani[1], Lixuan Tang[1] and Alcherio Martinoli[1]

Abstract— **Chemical gas dispersion poses considerable threat to humans, animals and the environment. The research areas of gas source localization and gas distribution mapping aim to localize the source of gas leaks and map the gas plume respectively, in order to help the coordination of swift rescue missions. Although very similar, these two areas are often treated separately in literature. In some cases, inferences on the gas distribution are made a posteriori from the source location, or vice-versa. In this paper, we introduce GaSLAM, a methodology that couples the estimation of the gas map and the source location using two state of the art algorithms with a novel navigation strategy based on informative quantities. The synergistic approach allows our algorithm to achieve a good estimation of both objectives and push the navigation strategies towards informative areas of the experimental volume. We validate the algorithm in simulation and with physical experiments in varying environmental conditions. We show that the algorithm improves on the source location estimate compared to a similar approach found in literature, and is able to deliver good quality maps of the gas distribution.**

I. INTRODUCTION

Chemical leaks can cause uncontrolled dispersion of gas in the air, endangering animals, humans and the environment. Two research areas are concerned with studying the gas dispersion phenomenon: Gas Distribution Mapping (GDM) aims to create a map of the gas dispersion in a given environment, while Gas Source Localization (GSL) aims to identify the source of the dispersion. The recent advances in the fields of robotics, chemical sensing and embedded systems allow both GDM and GSL to be performed by robotic platforms. Ground robots have been primarily employed for GDM and GSL [1], [2]. However, flying vehicles are increasing in popularity because of their versatility and volume coverage, even though they are often constrained to a 2D movement [3]–[5]. Recent research in these fields aims to allow for exploration of a tridimensional volume, in order to provide more information about the gas dispersion.

Rotary-winged Unmanned Aerial Vehicles (UAVs), endowed with sensing capabilities, are often used for GDM and GSL [4]–[7]. While these vehicles have great maneuverability properties, the turbulence of their propellers, called "wake", interacts with the gas dispersion and can hinder gas detection [8], [9]. An interesting alternative is provided by Nano Aerial Vehicles (NAVs, up to 15cm [10]). Their smaller propellers cause less disturbance to the gas plume [3], [11]–[13]. The drawback of NAVs is their limited flight time capability.

A variety of approaches have been proposed to achieve GDM [14]. The Kernel DM+V method learns a 2D statistical model of the gas distribution, providing the mean and the variance of the map [15]. This model was extended to 3D in [16] and tested with a ground robot following a pre-planned trajectory. In our previous work [12], we presented an adaptive path planning strategy that, coupled with the map estimation provided by the 3D Kernel DM+V/W algorithm, delivered good quality maps of the environment. Since this algorithm is less computationally expensive than other alternatives and we successfully used it in 3D in previous work, we leverage it in this paper as well.

GSL algorithms often take inspiration from strategies observed in nature [2], focusing in particular on the reliance on olfaction for foraging purposes observed in living species. Bio-inspired algorithms require low computational and memory resources, and are very reactive. Examples of bio-inspired algorithms deployed on UAVs and NAVs are [11], [17], [18]. Gradient-based approaches also require low computational resources [19]. In this class of algorithms, the plume is assumed to have a smooth concentration gradient that can be followed easily by a robot.

A class of GSL algorithms frequently used in recent years is the probabilistic one, where a probability distribution models the belief about the source location. The belief is updated after obtaining each new observation using Bayesian estimation, until the probability distribution can be described by a Dirac distribution. The most notable algorithms belonging to this category are Infotaxis [20] and Source-Term Estimation (STE) [21]. Probabilistic algorithms can estimate several environmental parameters beyond source location, such as the release rate of the gas source [5]. Moreover, the estimation has a measure of uncertainty that can be used to quantify its quality. Finally, different types and configurations of sensing nodes can be used to gather the necessary data, involving single or multiple nodes endowed with different degrees of mobility. The biggest drawback of these methods is their high computational cost.

In the scope of this work, an STE algorithm based on [22] is used for the estimation of the source location. The algorithm collects gas samples of the distribution and fits them to a plume model, with the objective of estimating the parameters of the model. STE algorithms have mostly been tested in simulation and in 2D [5], but have also been tested in 3D simulation [23].

In literature, GDM and GSL approaches are often tackled

*This work was funded by the Swiss National Science Foundation under grant 200020_175809.

[1]The authors are with the Distributed Intelligent Systems and Algorithms Laboratory, School of Architecture, Civil and Environmental Engineering, École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland

separately. In some instances, conclusions on the source location are drawn by the gas concentration map, where higher concentration indicates source presence [3], or a gas map is created a posteriori, knowing the source location [5].

The estimation part of GDM and GSL is often coupled with non-adaptive path planning approaches, where the robot follows a predefined route. However, path planning strategies where the robot moves to zones that are richer in information are more efficient and scale better to larger environments [12]. An example of an adaptive path planning approach based on informative quantities for STE is presented in [22]. In this work, Partially Observable Markov Decision Processes are used to select the goal position that would provide the highest amount of information among four candidate positions in 2D. In [24], the Kullback-Leibler divergence (KLD) [25] is used to predict the most informative goal position. Adaptive path planning strategies for GDM are found in [26], where several strategies based on Informative Path Planning (IPP) are tested in simulation, and in [12], where approaches that couple IPP with a clustering approach are used to achieve GDM in 3D.

In this paper, we introduce GaSLAM, Simultaneous Gas Localization and Mapping, a methodology that explores the synergies created by the simultaneous application of state of the art GDM and GSL methods. The benefits of our approach are two-fold: the underlying plume model used by the GSL method serves as an indicator of the gaseous zones throughout the whole map, while the GDM method provides an averaged gas estimation of the explored areas, which confirms the presence of gas. Moreover, the averaged gas data from GDM can be used for the GSL estimation, instead of adopting a navigation strategy that stops to gather several samples at each goal position.

Concretely, the contributions of the paper are:

- A methodology to combine state of the art GSL and GDM techniques to provide a good estimate of the gas map and of the source location within it in 3D.
- A navigation strategy that relies on the synergies between the two algorithms and on informative quantities.
- A novel source declaration algorithm that fits our methodology.

We tested our method in a high-fidelity simulator and with real experiments in varying environmental conditions in a three dimensional volume.

II. APPROACH AND ALGORITHMS

In this section, we provide an overview of GaSLAM. Furthermore, we give details about the GDM and GSL techniques employed in this paper, the strategies put in place to combine them effectively, and the navigation approach used throughout the work. Finally, the source declaration method used in this work will be presented.

A. Overview

The objective of GaSLAM is to provide a satisfactory gas map of a 3D volume and localize the origin of the gas dispersion by exploring synergies between a GDM and a GSL algorithm. Concretely, we combine knowledge acquired by both algorithms and exploit their respective estimation processes for an effective navigation strategy. The gas concentration map provided by the plume model used by the STE will be referred to as *STE map*, while the gas concentration map provided by the 3D Kernel DM+V/W will be referred to as *Kernel map*. Both maps are divided into N cells and have the same size and resolution.

In our algorithm, the Kernel map is continuously updated with new readings throughout the whole flight time budget B_t. When a goal position is reached, the estimation of the parameters of the STE is updated, and the STE map is created using the plume model and the current estimate of the parameters. The data used by the STE parameter estimation is the Kernel map, which provides an averaged estimation of the gas distribution. The STE map is then used, together with the Kernel map, to identify the next goal position through the proposed navigation algorithm. Once the confidence on the quality of the current gas distribution and source location estimations is high, the algorithm starts logging the source location estimations. At the end of the flight, a source position is declared using the estimations recorded, and a final gas map is produced by combining the information contained in the Kernel map and in the STE map.

An overview of GaSLAM is provided in Algorithm 1 and detailed in the following sections.

Algorithm 1 GaSLAM: Simultaneous Gas Source Localization and Distribution Mapping Algorithm

```
while t < B_t do
    Update Kernel map
    if Goal reached then
        Update STE
        Generate STE map
        Navigation (Algorithm 2)
    end if
    if High confidence in estimation then
        Start Source Recording (S_r = True)
    end if
end while
Declare Source and Create Final Map
```

B. 3D Kernel DM+V/W Mapping

The 3D Kernel DM+V/W algorithm uses a multivariate Gaussian weighting function to create a gas map of the environment [16]. A Gaussian kernel is used to compute the weights at the distance between where the measurement μ_i is taken and the center $x^{(k)}$ of cell k, with $k \in N$:

$$w_i^{(k)} = \mathcal{N}(|x^{(k)} - \mu_i|, \sigma_0) = \frac{1}{(2\pi)^{\frac{3}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x^{(k)}-\mu_i)^T \Sigma^{-1}(x^{(k)}-\mu_i)} \quad (1)$$

where Σ is the covariance matrix, which dictates the shape of the kernel. Σ depends on the stretching coefficient γ, which

determines the uncertainty of the wind estimate, and on the kernel width σ_0, which affects the amount of extrapolation on individual readings. Although the 3D Kernel DM+V/W algorithm also takes into account wind information, in the scope of this work we use wind direction and intensity information known a priori in our setup.

C. Source Term Estimation

STE is a probabilistic framework that aims to estimate a set of parameters of the gas distribution. To do so, it relies on a plume concentration model. We use the Pseudo-Gaussian concentration model employed in [27]:

$$C(m) = \frac{Q}{2\pi\bar{u}\sigma_y(x-x_s)\sigma_z(x-x_s)} e^{-\frac{(y-y_s)^2}{2\sigma_y^2(x-x_s)} - \frac{(z-z_s)^2}{2\sigma_z^2(x-x_s)}} \quad \forall x \geq x_s \tag{2}$$

where $m = (Q, x_s, y_s, z_s, \sigma_y, \sigma_z)$ and corresponds to the set of parameters that the model is trying to estimate, namely the release rate Q, the source position (x_s,y_s,z_s), and the standard deviation of the gas dispersion on Y and Z (σ_y and σ_z, respectively). $\bar{u}$ indicates the average wind speed, which is known a priori in our setup.

The parameters of the model are estimated using the probabilistic methodology described in [22]. A Bayesian formulation is used to estimate the posterior distribution of the parameters m, $P(m|D)$, where D is the sampled gas data. A Metropolis-Hasting method [28] is used as an approximation method to evaluate the posterior function through efficient sampling. When an estimation of the model parameters m is available, GaSLAM uses Equation (2) to infer the values of gas concentration throughout the whole map. The generated gas map corresponds to the STE map in this work.

In [22], the prior $P(m)$ of the Bayesian formulation is defined as a uniform distribution for all parameters, while the likelihood $P(D|m)$ is defined as:

$$P(D|m) \propto exp\left(-\frac{1}{2} \sum_{i=0}^{N} \frac{(D_i - C_i(m))^2}{\sigma_M^2 + \sigma_D^2} \right) \tag{3}$$

where σ_M and σ_D are the standard deviations of model and measurement error, which are assumed to follow a normal distribution with zero mean. The likelihood, which represents the probability that data D is obtained given a set m of parameters, is computed for each grid cell i. $C_i(m)$ is the gas concentration value of the cell i computed by Equation (2) for m parameters. In [22], D_i is the average of 50 concentration values gathered by the sensor in a given location, and corresponds to the actual measured gas value. However, the flight time constraints of the robotic platform employed in this work, coupled with the propeller disturbance, discourage a sampling strategy that has the robot collect samples for several seconds at a given location. We therefore collect samples continuously while in flight and feed them to the 3D Kernel DM+V/W, which creates an averaged gas map. When a goal position is reached, the STE algorithm estimates the model parameters using the averaged gas map provided by the 3D Kernel DM+V/W.

D. Navigation

The navigation strategy is inspired by the one proposed in [22] and chooses the next goal position by combining two vectors, one promoting exploration of the volume and one promoting exploitation of the information already gathered:

$$V_{goal} = (1-\alpha)V_{explore} + \alpha V_{exploit} \tag{4}$$

where α is a coefficient that changes dynamically and encodes the reliability of the information gathered.

The exploration vector $V_{explore}$ points from the current position to the cell $i_{explore}$, which has the highest KLD to confidence ratio:

$$i_{explore} = \underset{i \in N}{argmax} \frac{KLD_i}{c_i} \tag{5}$$

The KLD measures the difference between two probability distributions, P and Q [25], and is computed as:

$$D_{KL}(P||Q) = \sum_i P(i) log_2 \left(\frac{P(i)}{Q(i)} \right) \tag{6}$$

where P is the probability distribution of the gas samples in each grid cell and Q is the estimated next step probability distribution. Higher values of KLD indicate that the samples taken so far are not homogeneous in value, and more samples should be gathered in that area to reach a higher consensus on the gas value. Examples of usage of the KLD as a reward functions for IPP approaches are [12], [22], [24].

The confidence value c is computed for each cell in the scope of the 3D Kernel DM+V/W algorithm as:

$$c_i = 1 - exp\left(-\frac{\Omega_i}{\sigma_\Omega^2} \right) \; \forall i \in N \tag{7}$$

where Ω is the weight of the gas readings of the cell and σ_Ω is a scaling coefficient. The value of c is used in the original 3D Kernel DM +V/W algorithm to scale the concentration values of each cell. In this work, instead, we use it to direct exploration towards areas of lower confidence.

The exploitation vector is computed as follows:

$$V_{exploit} = \begin{cases} V_{source} & \text{if } E > Th_E \text{ or } S_r = False \\ V_{diff} & \text{if } E < Th_E \text{ and } S_r = True \end{cases}$$

where E is the entropy of the posterior probability of the STE, Th_E is a threshold picked with preliminary experiments, and S_r indicates whether the recording of source guesses has started. When E is above Th_E, the confidence on the source guess is low. Thus, the exploitative vector pushes to move towards the currently guessed source location to decrease the uncertainty around the guess. When good confidence is reached on the estimation, the exploitation vector is equal to V_{diff}, which points to the highest difference between the STE map and the Kernel map, d_i, computed as:

$$d_i = |k_i - s_i| \ \forall i \in N_g \tag{8}$$

where k_i is the gas value of the Kernel map and s_i is the gas value of the STE map. N_g is a subset of N and indicates cells that are identified as gaseous both in the Kernel map and in the STE map. To identify if a cell contains gas or not, we use a K-means segmentation algorithm to divide the Kernel map and the STE map into gas and no-gas zones.

The coefficient α, which guides the exploration-exploitation trade-off of Equation (4), is computed as:

$$\alpha = \sqrt{\frac{N_g}{N_{kg}}} \tag{9}$$

where N_{kg} is the number of cells identified as gaseous in the Kernel map by the K-means segmentation. This coefficient is higher if the overlap between the gaseous zone of the STE map and of the Kernel map is high. A high coefficient indicates that the parameters estimated by the STE are able to generate a map, based on the Pseudo-Gaussian plume model, that matches the Kernel map dispersion. The value of α is always contained between 0 and 1, and the root square is used to enhance smaller overlaps of gas zones in order to push the algorithm towards exploitation to confirm the estimation of the parameters.

Concretely, the navigation algorithm presents an initial explorative phase, followed by an exploitative phase where the robot navigates within the plume and its edges. The navigation algorithm is summarized in Algorithm 2.

Algorithm 2 Navigation Algorithm

```
K-means Segmentation
Compute V_explore
Compute α
if E < Th_E and S_r = True then
    Compute V_diff
else
    Compute V_source
end if
Update V_exploit
Update goal
```

E. Source Declaration

In GSL methods, the source declaration is usually the last step of the algorithm, when the confidence in the estimation is deemed acceptable. In our scenario, the drone continues to fly for a set amount of time and guesses on the source location continue to be made according to the new data. The source declaration algorithm proposed in this work saves all the guesses made on source location after an acceptable level of confidence in the estimation is reached and then uses the RANSAC algorithm [29] to produce the final source position estimation.

The estimation of the source parameters is considered good when the entropy of the posterior E falls below a threshold Th_E [22]. Moreover, values of α above 0 indicate some overlap between the STE map and the Kernel map. Therefore, when the conditions $E < Th_E$ and $\alpha > 0$ is satisfied, the recording of the source positions starts.

When the time budget B_t elapses, the RANSAC algorithm produces the final source position estimate by dividing the recorded source guesses into outliers and inliers. It achieves this by computing how many recorded positions fall within a set distance S_{th} from each recorded position. The algorithm then finds which recorded position has the highest number of neighbors, labels them as inliers, and computes the final declared source as the mean of the inlier positions weighted by the posterior entropy value associated with them. The threshold S_{th} is set to:

$$S_{th} = 2\sqrt{(x_r^2 + y_r^2 + z_r^2)} \tag{10}$$

where x_r, y_r and z_r are the resolutions of the source guesses on the three axis respectively.

F. Final Map Creation

The final gas map is created at the end of the algorithm by integrating the STE map with the Kernel map. Namely, all cells that are considered gaseous by the K-means segmentation of the Kernel map will be integrated in the STE map, even if they are not considered to be as gaseous by the STE map. Moreover, all the cells that are considered non-gaseous will be attributed a value that corresponds to the average Kernel map reading of non-gaseous zones. We chose this integration method because it entrusts the Kernel map with having more accurate gas predictions, but still refers to the STE map for the general shape of the plume.

III. EXPERIMENTAL EVALUATION

This section introduces the platform employed for this work, presents an overview of the algorithmic parameters chosen for this work, describes the environmental conditions in which the algorithm was tested, and outlines the evaluation metrics. Finally, it presents the simulation and experimental setups.

A. NAV Platform

The robotic platform employed for this work is a Crazyflie V2.1 (CF2, Bitcraze AB, Malmö, Sweden). The NAV is enhanced with a custom Printed Circuit Board (PCB) hosting a MICS-5521 CO/VOC sensor from SGX Sensortech Technologies with a sampling frequency of 10 Hz. The sensor is placed at the bottom of the drone, since better sensing performance was achieved previously with this configuration [11]. A Motion Capture System (MCS) from Motion Analysis is used for localization. The total payload of the NAV amounts to 6.5g, which includes five MCS markers and the custom PCB. The velocity of the NAV is kept constant at 0.5 m/s for a total flight time B_t, 4 minutes and 30 seconds in our case. The velocity was selected to allow the NAV to gather a satisfactory number of samples within a grid cell, while still keeping a fast enough speed for exploration. The CF2 communicated with a PC serving as ground station over

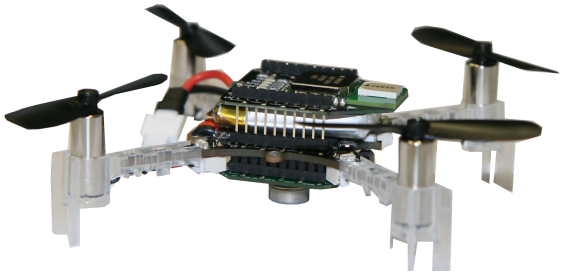

Fig. 1: Crazyflie V2.1 with a custom PCB for gas sensing at the bottom.

the 2.4 GHz ISM radio band. The position data, acquired by the MCS, is sent to the CF2 through the radio link. All algorithmic computations are carried out off-board.

B. Algorithmic Parameters

The parameters of the 3D Kernel DM+V/W algorithm were chosen in accordance to previous work [12]. The model and measurement errors of the likelihood σ_M and σ_D are merged into one single parameter σ_E which takes into account the error that we expect to observe because of the effect of the wake of the propellers on the gas dispersion and the subsequent divergence of the gas plume from the model. The value was determined with preliminary experiments.

Although the STE algorithm provides an estimation of several parameters of the gas distribution, in this work we chose to focus on analyzing the estimation of the source location parameters. The estimation of the other parameters will be used for future work, where it will be coupled with the gas map estimation of the 3D Kernel DM+V/W to provide gas values closer to the ground truth.

C. Environmental Parameters

In order to assess the performance and robustness of the algorithm in different environmental conditions, two parameters were taken into account: wind speed and source release rate.

The wind speed impacts the width of the plume. Stronger winds generate a narrower plume, which is harder to detect, especially in the exploration phase. However, guesses on the source location should be more precise after gas particles are detected. Lower wind speeds instead generate a wider plume, which might be detected faster, but source guesses might be further from the true source position. In this paper, we test our algorithm in wind speeds equal to 0.2 m/s and 0.7 m/s, consistent with real world scenarios.

The source release rate parameter impacts the density of the gas plume. A higher source release rate should aid the gas detection, especially when using a vehicle with propellers. During the experiments, a stationary electric pumping device is used to release a mix of ethanol and air in the experimental volume. The source release rate values chosen for the evaluation are 0.5 L/min (high) and 0.25 L/min (low). The air to ethanol ratio of the mix is unknown.

The environmental configurations are shown in Table I.

Label	Wind Speed	Release Rate
A	0.2 m/s	low
B	0.2 m/s	high
C	0.7 m/s	low
D	0.7 m/s	high

TABLE I: Environmental configurations

D. Evaluation Metrics

The outcome of the source localization is evaluated using the mean of the error between the true source position and the declared source position in the three coordinates, computed for each experimental setting as:

$$PE = \sqrt{((x - x_s)^2 + (y - y_s)^2 + (z - z_s)^2)} \tag{11}$$

The final map is evaluated using two metrics: the RMSE and the shape coverage. The RMSE is used to quantify the difference between the final gas map and the ground truth map of the gas. It is computed as:

$$RMSE = \sqrt{\frac{\Sigma_{i=1}^{N}(d_i - g_i)^2}{N}} \tag{12}$$

where d_i is the final gas value of each grid cell and g_i is the corresponding ground truth value. We compare the RMSE of the final map with the one of the STE map, $RMSE_{STE}$, to show how our method improves map estimation. Note that we do not compare to the RMSE of the Kernel map, since this provides an estimate of the explored areas only.

Although the RMSE gives a good sense of the quality of the map, the fluctuations of the readings can sometimes cause high discrepancy with the ground truth values, especially in areas of high gas. For this reason, in our previous work [12], we introduced the shape coverage metric, which aims to describe if the shape of the plume of the final map overlaps with the ground truth. It is computed as the probability of a cell to be identified correctly as containing gas or not according to the ground truth:

$$SC = p(d_i \geq Th_d | g_i \geq Th_g || d_i < Th_d | g_i < Th_g) \ \forall i \in N \tag{13}$$

where Th_d and Th_g are the thresholds used to determine whether or not a cell contains gas. Please refer to [12] for a discussion on these thresholds.

To ease the comparison between the different methods, the metrics are combined in one overall metric:

$$M = (SC * |1 - RMSE|)/PE \tag{14}$$

Higher values of this metric indicate better performance. We consider runs where the source position in X is further than 5 meters from the true position as failed and we do not take them into account in the computation of the metrics. The X axis is the one parallel to the wind in our setup.

E. Simulation Setup

We used Webots [30], a high-fidelity open-source simulator, for the simulation experiments. The wind and gas fields were simulated using a gas dispersion plugin [31], which

provides reasonably realistic environmental conditions based on the filament-based plume dispersion model proposed in [32]. Ten simulations were run for each environmental condition using a simulated CF2 equipped with a gas sensor. In order to challenge the algorithm, the source position is randomized in Y and Z, the crosswind plane.

The simulator does not model the effect that the wake of the propellers has on the gas dispersion. Moreover, the gas dispersion is model-based and might not correspond exactly to a real plume. In [12], we highlighted some of the limitations of simulation with respect to real experiments in the context of gas mapping. In order to provide a comprehensive evaluation of our algorithms, we validated our simulation results with physical experiments.

F. Wind Tunnel Setup

Physical experiments were conducted in a wind tunnel of volume 18x4x2 m^3. A honeycomb filter is used to laminarize the adjustable wind speed inside the tunnel. The limitations in terms of flight time of the NAV employed for this work prompted us to constrain the experimental volume to 7x2x0.5 m^3, which is comparable to other approaches found in literature given the size and capabilities of the flying platform. The experiments are carried out in a quasi-laminar regime, higher turbulence in the wind or different environmental conditions might cause different results. The initial position of the robot corresponds to the right corner of the volume opposite to the source. Ten experiments were carried out for each environmental condition.

We acquired a ground truth map of the gas distribution using an array of static sensors mounted on a traversing system (a 3-axis robotic manipulator present in many wind tunnels). The same type of gas sensor as the one mounted on the NAV was used. The sensors sample at 10 Hz for 2 seconds at the center of each grid cell. The data is then averaged after eliminating outliers. The data coming from the static sensors is calibrated against the NAV sensor in order to compensate for small manufacturing differences.

IV. RESULTS

This section presents the results of GaSLAM in simulation and with physical experiments. The source localization results are qualitatively compared to the ones obtained in [22] with the same setup and STE algorithm, but using a ground robot and a different navigation strategy. An indication of the improvement on gas map quality with our algorithm is given by the comparison of the RMSE to the $RMSE_{STE}$.

A. Simulation Results

The results of the source position estimation obtained in simulation are presented in Figure 2. It can be seen that the error is generally low for all axes and it is similar across different environmental conditions, consistently with the findings in [22].

An overview of the evaluation metrics in simulation is reported in Table II. The RMSE metric shows that the estimated gas dispersion differs the most from the ground

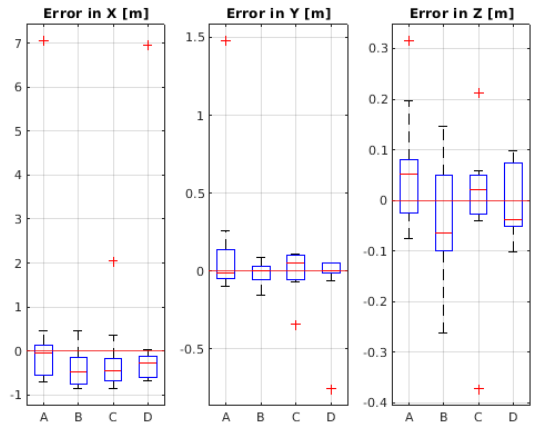

Fig. 2: Error with respect to the true source position in X,Y,Z for different environmental conditions in simulation.

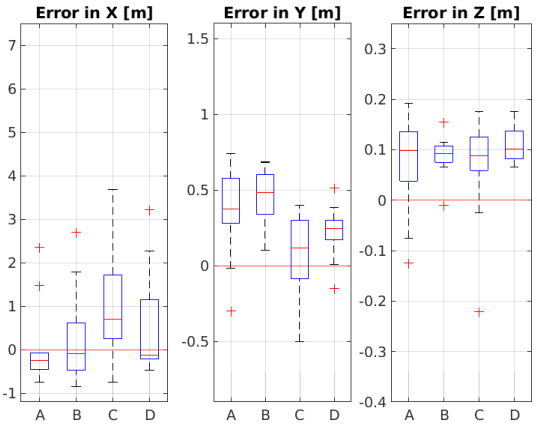

Fig. 3: Error with respect to the true source position in X,Y,Z for different environmental conditions in physical experiments

truth in scenario B, with low wind and high source release rate. This is probably due to the fact that the whole shape of the plume is not captured well, which is also confirmed by the corresponding shape coverage metric. The $RMSE_{STE}$ shows that integrating the Kernel map and the STE map only yields slight improvements in simulated scenarios with high winds. The shape coverage highlights that the shape of the plume is better captured in higher wind speed conditions. The PE metric highlights that scenarios A and D are the ones where the algorithm performs best. Two runs out of forty were considered as failed in simulation, yielding a success rate of 95%.

Setup	RMSE	$RMSE_{STE}$	SC	PE	M
A	0.0486	0.0384	0.7321	0.3732	1.8663
B	0.0883	0.0541	0.6856	0.5359	1.1665
C	0.0294	0.0296	0.8219	0.6789	1.1750
D	0.0477	0.0638	0.8214	0.3700	2.1141

TABLE II: Evaluation metrics - average of 10 runs in simulation

B. Real Experimental Results

The results obtained with real experiments regarding the location of the gas source are presented in Figure 3. The

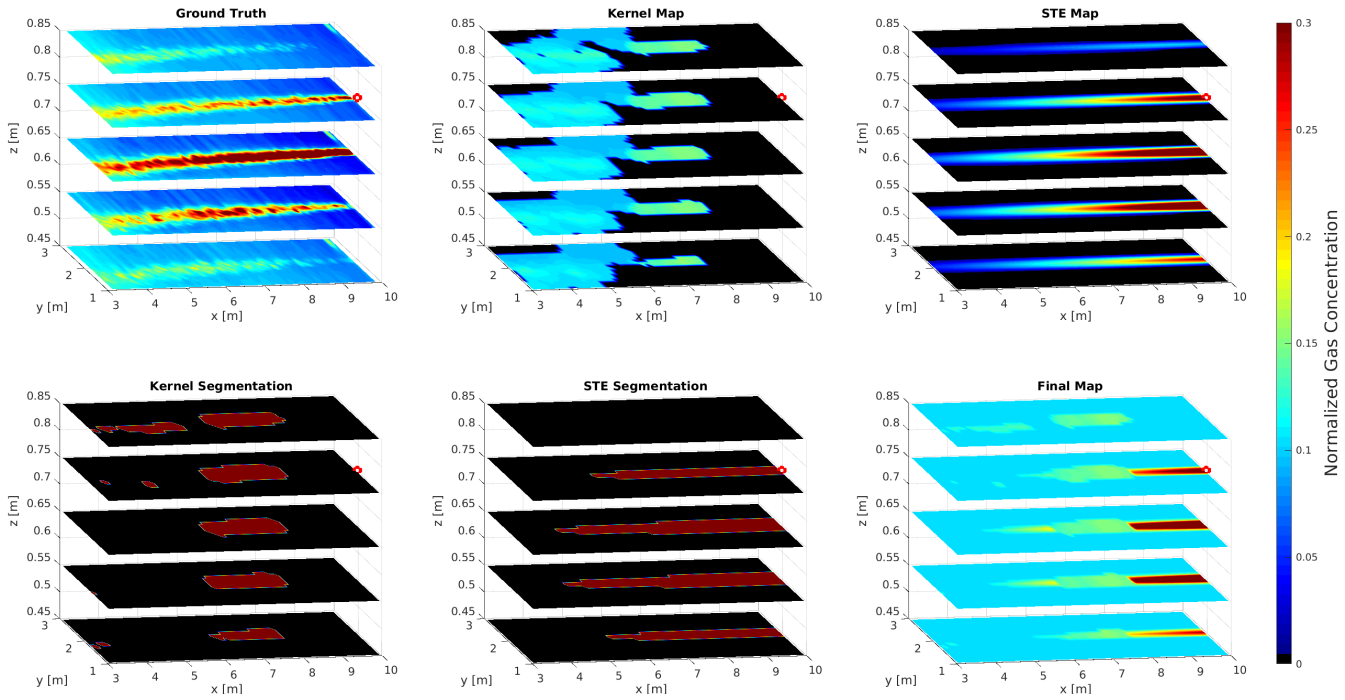

Fig. 4: Examples of different maps used in the algorithm for an experimental run with high wind and high release rate during physical experiments. The true source position is indicated by a red cyrcle.

variance of the results is bigger compared to simulation, which is expected. The error on the X axis is, on average, worse for the experiments carried out in high wind and low source release rate, probably because the drone had a hard time acquiring a lot of meaningful gas data in this configuration. The error in the Y axis is lower for both configurations with higher wind. This is due to the fact that the plume is narrower with higher wind. The error in Z is similar across all approaches. In our previous work [12], we showed that the Kernel maps can suffer from a slight error in Z due to the Kernel interpolation. It would be interesting to investigate further if there is a correlation between this and the error in Z for the source position. Our algorithm obtained better estimation of the source position compared to the results obtained with ground robots in [22] for low wind scenarios, and comparable results for high wind scenarios. This can be explained by the addition of a more comprehensive source declaration process, and by the more dynamic navigation method that constantly re-evaluates the exploration-exploitation trade-off.

An example of maps generated by GaSLAM can be found in Figure 4. The ground truth map, the Kernel map and the STE map are shown next to each other. The black parts of the Kernel map indicate cells for which there were no readings and therefore were not explored. From the Kernel Map, it can be seen that more exploration of non gaseous zones happens further from the source, while the drone converges to gaseous zones as it moves closer to the source position. The segmented maps obtained with the K-means algorithm are also shown, highlighting the zones that are considered as gaseous by the algorithm. The final map highlights that the inclusion of the average value of non-gaseous cells of the Kernel map corresponds more closely to the equivalent no-gas cells in the ground truth. The shape of the plume is also fairly well highlighted. With this method we obtain a final map of the whole experimental volume, in contrast with GDM only approaches, where only a map of the volume covered is obtained.

The metrics are reported in Table III. Scenarios with high winds have higher shape coverage and lower RMSE, indicating that the overall quality of the map is higher. This confirms the trend identified in simulation. In contrast to simulation, a reduction in the RMSE can be observed in the final map produced with GaSLAM compared to the final STE map across all scenarios. Moreover, the shape coverage is higher than the best performing scenarios of our GDM-only approach in [12], tested in a scenario identical to D. These findings confirm that the integration of Kernel map and STE map is useful to improve the quality of the map. The localization error instead is the highest for lower wind and lower source release rate. According to the overall metric, the algorithm performs best with high wind and high source release rate, which was to be expected given the fact that a denser and narrower plume can be sensed by the drone and described more easily by the STE model. No runs were considered to have failed in physical experiments. In general, the algorithm seems to be fairly robust to different environmental conditions, with the exception of low wind and low release rate. More experiments in harsher conditions are needed to reinforce this claim.

Setup	RMSE	$RMSE_{STE}$	SC	PE	M
A	0.1026	0.1103	0.7948	0.8988	0.7936
B	0.1145	0.1275	0.8093	0.9939	0.7211
C	0.0929	0.1166	0.8711	1.2852	0.6148
D	0.0754	0.1007	0.8523	0.9298	0.8475

TABLE III: Evaluation metrics - average of 10 runs in physical experiments

V. CONCLUSIONS AND OUTLOOK

In this paper, we present GaSLAM, an algorithm that organically combines state of the art methods in GDM and GSL with the objective of obtaining a map of the gas distribution and localizing the source of the gas dispersion within it. This method leverages the information coming from both algorithms to maximize the knowledge of the environment and uses it for an adaptive navigation strategy based on informative quantities. The method was tested in different environmental conditions and presented a fair level of robustness to most of them. We show that GaSLAM can produce good source localization and mapping estimations, improving on comparable GSL-only and GDM-only approaches carried out in the same setup.

We believe that the general framework of this method, consisting of the synergies between STE and 3D Kernel DM+V/W algorithms and the adaptive navigation method based on a continuous trade-off between exploration and exploitation, could be the foundation of other methods aiming for simultaneous gas distribution mapping and source localization. Future work will focus on the improvement of this method within the same framework by analyzing, for example, the employment of different informative quantities in the navigation method, and by integrating the source release rate estimate of STE into the algorithm.

REFERENCES

[1] L. Marques, U. Nunes, and A. T. de Almeida, "Olfaction-based mobile robot navigation," *Thin solid films*, vol. 418, no. 1, pp. 51–58, 2002.

[2] X. Chen and J. Huang, "Odor source localization algorithms on mobile robots: a review and future outlook," *Robotics and Autonomous Systems*, vol. 112, pp. 123–136, 2019.

[3] J. Burgués, V. Hernández, A. J. Lilienthal, and S. Marco, "Smelling nano aerial vehicle for gas source localization and mapping," *Sensors*, vol. 19, no. 3, p. 478, 2019.

[4] P. P. Neumann, M. S. I. V. H. Bennetts, and I. M. Bartholmai, "Adaptive gas source localization strategies and gas distribution mapping using a gas-sensitive micro-drone," *Technology (BMWi)*, vol. 4, no. 5, p. 6, 2012.

[5] M. Hutchinson, C. Liu, and W.-H. Chen, "Source term estimation of a hazardous airborne release using an unmanned aerial vehicle," *Journal of Field Robotics*, vol. 36, no. 4, pp. 797–817, 2019.

[6] T. Kersnovski, F. Gonzalez, and K. Morton, "A UAV system for autonomous target detection and gas sensing," *IEEE Aerospace Conference*, (12 pages), 2017, DOI 10.1109/AERO.2017.7943675.

[7] L. Bing, M. Qing-Hao, W. Jia-Ying, S. Biao, and W. Ying, "Three-dimensional gas distribution mapping with a micro-drone," in *34th Chinese Control Conference*, 2015, pp. 6011–6015.

[8] M. Rossi and D. Brunelli, "Gas sensing on unmanned vehicles: Challenges and opportunities," in *IEEE New Generation of Circuits and Systems Conference*, 2017, pp. 117–120.

[9] T. Villa, F. Salimi, K. Morton, L. Morawska, and F. Gonzalez, "Development and validation of a UAV based system for air pollution measurements," *Sensors*, vol. 16, no. 12, p. 2202, 2016.

[10] M. Hassanalian and A. Abdelkefi, "Classifications, applications, and design challenges of drones: A review," *Progress in Aerospace Sciences*, vol. 91, pp. 99–131, 2017.

[11] C. Ercolani and A. Martinoli, "3D odor source localization using a micro aerial vehicle: System design and performance evaluation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 6194–6200.

[12] C. Ercolani, L. Tang, A. A. Humne, and A. Martinoli, "Clustering and informative path planning for 3D gas distribution mapping: Algorithms and performance evaluation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5310–5317, 2022.

[13] S. Shigaki, M. Fikri, and D. Kurabayashi, "Design and experimental evaluation of an odor sensing method for a pocket-sized quadcopter," *Sensors*, vol. 18, no. 11, p. 3720, 2018.

[14] J. Burgués and S. Marco, "Environmental chemical sensing using small drones: A review," *Science of the Total Environment*, vol. 748, p. 141172, 2020.

[15] A. J. Lilienthal, M. Reggente, M. Trincavelli, J. L. Blanco, and J. Gonzalez, "A statistical approach to gas distribution modelling with mobile robots-the kernel DM+V algorithm," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 570–576.

[16] M. Reggente and A. J. Lilienthal, "The 3D-kernel DM+V/W algorithm: Using wind information in three dimensional gas distribution modelling with a mobile robot," in *IEEE International Conference on Sensors*, 2010, pp. 999–1004.

[17] P. P. Neumann, V. Hernandez Bennetts, A. J. Lilienthal, M. Bartholmai, and J. H. Schiller, "Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms," *Advanced Robotics*, vol. 27, no. 9, pp. 725–738, 2013.

[18] S. Shigaki, T. Sakurai, N. Ando, D. Kurabayashi, and R. Kanzaki, "Time-varying moth-inspired algorithm for chemical plume tracing in turbulent environment," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 76–83, 2017.

[19] V. Genovese, P. Dario, R. Magni, and L. Odetti, "Self organizing behavior and swarm intelligence in a pack of mobile miniature robots in search of pollutants," in *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems*, vol. 3. IEEE, 1992, pp. 1575–1582.

[20] M. Vergassola, E. Villermaux, and B. I. Shraiman, "'infotaxis' as a strategy for searching without gradients," *Nature*, vol. 445, no. 7126, pp. 406–409, 2007.

[21] M. Hutchinson, H. Oh, and W.-H. Chen, "A review of source term estimation methods for atmospheric dispersion events using static or mobile sensors," *Information Fusion*, vol. 36, pp. 130–148, 2017.

[22] F. Rahbar, A. Marjovi, and A. Martinoli, "An algorithm for odor source localization based on source term estimation," in *IEEE International Conference on Robotics and Automation*, 2019, pp. 973–979.

[23] ——, "Design and performance evaluation of an algorithm based on source term estimation for odor source localization," *Sensors*, vol. 19, no. 3, p. 656, 2019.

[24] M. Hutchinson, C. Liu, and W.-H. Chen, "Information-based search for an atmospheric release using a mobile robot: Algorithm and experiments," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 6, pp. 2388–2402, 2018.

[25] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[26] C. Rhodes, C. Liu, and W.-H. Chen, "Informative path planning for gas distribution mapping in cluttered environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 6726–6732.

[27] S. P. Arya *et al.*, *Air pollution meteorology and dispersion*. Oxford University Press New York, 1999, vol. 310.

[28] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The journal of chemical physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[29] K. G. Derpanis, "Overview of the ransac algorithm," *Image Rochester NY*, vol. 4, no. 1, pp. 2–3, 2010.

[30] O. Michel, "Cyberbotics Ltd. Webots™: professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, 2004.

[31] Wikibooks, "Webots odor simulation — wikibooks, the free textbook project," https://en.wikibooks.org/w/index.php?title=Webots OdorSimulation&oldid=1966420, 2010, accessed: 2021-09-01.

[32] J. A. Farrell, S. Pang, and W. Li, "Plume mapping via hidden Markov methods," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 33, no. 6, pp. 850–863, 2003.