

Particle Source Localization with a Low-Cost Robotic Sensor System: Algorithmic Design and Performance Evaluation

Xinxing Chen, Ali Marjovi, Jian Huang, Alcherio Martinoli

Abstract—In this paper we tackle the problem of finding the source of particulate matter with a mobile robot equipped with a low-cost multi-channel optical particle counting sensor. The proposed method is based on the Infotaxis odor source localization algorithm and makes multiple modifications to adapt it to particle plumes. In particular, we propose three simple but efficient ways to fuse multiple probability maps associated with various particle sizes and use the resulted integrated map to guide the robot’s movements. A refined measurement data collection is conducted in a wind tunnel to fit the particle plume model. The method with three proposed integration strategies is evaluated in simulation and in the wind tunnel emulating realistic environmental conditions in a repeatable fashion. In particular, we have investigated the impact of two environmental parameters - the wind speed and source release rate on the algorithm performance. The proposed algorithm with the weighted multi-modality map integration strategy outperforms the original Infotaxis and the other two variants. In high wind speed, the proposed algorithm is able on average to estimate the source location with less than 1 meter error in the 80 m^2 wind tunnel arena.

Index Terms—particle source localization, low-cost optical particle counter, mobile robotics, Bayesian algorithm, wind tunnel experiments

I. INTRODUCTION

OR odor source localization is the problem of finding the source of a chemical plume. For most of the living organisms in the world, locating an odor source is an inborn ability. It helps animals locate their food or prey and avoid predators. For example, dogs have a much more sensitive nose than human beings, which makes them way superior to humans when searching for poisonous chemicals and explosives. Therefore, dogs and other animals have been deployed to find chemical sources, but they require a substantial amount of training, could get hurt in dangerous scenarios, and get tired after a few hours of search operation.

This work was supported by the China Scholarship Council (201706160029), in part by the Swiss National Science Foundation under grant 200020 175809/1, and in part by the Fundamental Research Funds for the Central Universities (HUST: 2019kfyXKJC019).

Xinxing Chen and Jian Huang are with the Key Laboratory of Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: cxx@hust.edu.cn; huang_jan@mail.hust.edu.cn).

Ali Marjovi and Alcherio Martinoli are with the Distributed Intelligent Systems and Algorithms Laboratory (DISAL), School of Architecture, Civil, and Environmental Engineering, École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland (e-mail: ali.marjovi@epfl.ch; alcherio.martinoli@epfl.ch).

This work has been entirely carried out at DISAL, EPFL.

Manuscript received May 12, 2020; accepted June 3, 2020.

Therefore, since the early 1990s, many researchers have dedicated attention to the design of a robotic system capable of finding a source of chemical release [1]. Localizing the source of an airborne particle plume presents similarities with the widely-discussed chemical odor source localization problem but the different nature of the particle plume suggests a significant adaptation of the methods used in the chemical odor source localization problem. Moreover, dogs cannot locate particle sources because most of the particle plumes are odorless. Finally, the problem of particle source localization is still in its early stage, with almost no related works.

Because of the limited literature on particle source localization, we will review in the following some of the most relevant methods used in odor source localization. The whole process of robotic odor source localization can be divided into three sub-tasks:

- Plume finding, which means searching the environment in order to sense the presence of an odor;
- Plume tracking, which means following the plume to find the odor source;
- Source declaration, which means estimating the actual location of the source.

Source declaration is either carried out leveraging other sensing modalities such as vision, or established according to certain rules (such as measuring a concentration threshold) set by a human supervisor. Most odor source localization algorithms tackle only the first two phases.

Different from some other source tracking problems (e.g., radiation source and sound source localization), in which the intensity of the signal is highly related to the distance from the sensor to the source [2], [3], odor/particle source localization presents a higher level of complexity. The challenges come from the variety of environmental parameters related to the dispersion of plumes (e.g., wind field, source release rate, gravity settling effect). The resulting odor/particle concentration field can hardly be processed analytically. Although researchers are actively overcoming the difficulties, robotic odor, and even more particle, source localization solutions are still far from industrial applications.

We can divide the existing odor source localization algorithms for single robots into three categories [4], [5]: gradient-based algorithms, bio-inspired algorithms and probabilistic algorithms.

Starting from the early 1990s, contributions on gradient-based algorithms [1], [6]–[9] were preliminary attempts in the research area. Researchers designed algorithms to guide

robots to the odor source exploiting the concentration gradient and wind direction. In these early works, robotic odor source localization did not perform as well as its biological counterparts. One important factor is that an odor plume is propagated through wind in a typically turbulent regime. This makes the odor concentration distribution far from showing a smooth gradient along the wind direction; it is indeed intermittent, and characterized by great concentration fluctuations. Thus, it is hard to collect accurate odor gradients in general and it is even more difficult to do so in short time windows.

Bio-inspired algorithms are motivated by the amazing ability of living creatures (such as moth [10]–[13], lobster [14]–[16], and dung beetle [17]) to find odor sources. These algorithms are relatively computationally cheap. On the other hand, Bennetts et al. [18] argued that biomimetic approaches to locate odor sources are of limited use, state-of-the-art sensing and locomotion abilities of robots have large differences in comparison to those of animals.

Probabilistic algorithms including Bayesian inference [19]–[26], hidden Markov [27] and kernel methods [28], [29] model the odor source location as a probability distribution. Through continuous observations, the probability distribution will converge towards a Dirac function at the source. Probabilistic and map-based algorithms do not rely heavily on the concentration gradient, and they are flexible in their forms. Moreover, as hardware technologies have radically increased their performances in the last decades, their computational cost is no longer a major burden.

Most of the previous works focus on chemical odor source localization and use Volatile Organic Compound (VOC) sensors to measure the concentration of these substances (e.g., ethanol or acetone). However, in many situations, such as burning some materials, or even more severe - in fire disasters, the source does not release a chemical substance that can be detected by VOC sensors, but rather airborne particles. These particles are often odorless and common chemical sensors are not able to sense them. Therefore, solving the particle source localization problem requires substantially changes in the sensing technology and in turn adapting the related localization algorithms.

The most common techniques for monitoring airborne particle use static and fairly expensive sensor nodes [30]–[32], which are reliable, but also characterized by limited portability, a feature that together with the cost limit their deployment in large quantities. Recently, some low-cost small-sized Particle Matter (PM) sensors have become commercially available. Because of their cost and dimensions, these sensors can be deployed in large quantities and on small-scale mobile robotic assets. Although there are a few papers testing the reliability of these low-cost PM sensors on monitoring air pollution [33], [34], there are not reported applications in the field of robotic particle source localization. Therefore, to the best of our knowledge, our effort can be considered among the frontier attempts in addressing the challenge of finding and tracking a PM source using a mobile robot. Some previous works on robotic fire detection [35], [36] used carbon dioxide sensors, but they may not work well if the combustion is not complete. PM source localization can be an alternative solution for this

case.

While the odor source localization is an actively investigated topic in the robotics and partially in the sensor communities (though still a niche one), to the best of our knowledge, there was no previous work on robotic particle source localization at the time of our submission. Therefore, our contribution can be considered a proof-of-concept study and a solution to the above issues, at the system level, supported by repeatable quantitative results gathered by leveraging recently available low-cost particle sensing technology. Within this context, we have decided to create a baseline paper by adapting the existing Infotaxis algorithm [23] to airborne particle source localization. In addition to a different nature of the source, the adaptation intended to leverage also the sensing richness characterizing particle sensors able to produce quantitative data for different particle sizes. Therefore, the overall adaptation involves a different plume modeling parametrization, map integration strategies, and a thorough experimental evaluation in simulation and reality under different environmental conditions. To this purpose, we studied the sensing characteristics of a typical low-cost multi-modality particle sensor, fitted a particle plume model and accordingly proposed some variants of the Infotaxis algorithm, with different probability map integration strategies for airborne particle source localization. An information-greedy robot navigation approach is adopted in this paper instead of sophisticated exploration strategies such as stochastic optimal control with path integrals [37] and Partially Observable Markov Decision Process [38] balancing exploration and exploitation, because the aim of the proposed algorithm was to estimate the location of the particle source with a minimum number of iterations. Further study on the balance between spatial exploration and exploitation strategies will be conducted in future work.

The main contributions of this paper are threefold. A first and significant contribution of this paper is an experimental investigation of particle plume modeling which reveals differences between chemical plume and particle plume tracking. A second contribution is concerned with the adaptation of the Infotaxis algorithm to three different probability map integration strategies, exploiting all the modalities produced by a multi-size particle sensor. A third contribution is to evaluate the proposed algorithm under different environmental conditions and compare its performance with those of the original Infotaxis algorithm directly applied to particle source localization. Since the wind plays a significant role in the dispersion of particles, in this paper we do not consider windless conditions.

In Section II, we show how particle plumes differ from chemical plumes and introduce the low-cost particle sensor we used in this work. In Section III, we present the original Infotaxis algorithm and the adaptations we made to accommodate multiple sensing modalities. In Section IV, the plume models of different sizes of airborne particles are fitted into the pseudo-Gaussian model and the evaluation results of the proposed algorithm with the fitted plume model in simulation and physical reality are presented.

II. PARTICLE PLUME DISPERSION AND SENSING

One of the main differences between particulate and chemical plumes comes from their correspondingly different physical properties of the source, the involved physical transportation of the airborne components, and the corresponding available sensing technology. Usually odor fields reported in the robotic literature are generated by a single source emitting a single chemical compound. Therefore, the concentration measurement involves a single sensing modality. On the other hand, particle sources tend to emit a cocktail of particles of different sizes and therefore the available sensing technology, including low-cost sensors, provides natively a number of related metrics (e.g. number, mass or surface area) related to these different sizes. Therefore the problem of particle source localization becomes a multi-modal search, while odor source localization is typically a single modality search. Additionally, the dispersion of the particles in the air is also different since it is dependent on the size and the mass of the particles. Large particles fall down faster so they travel more locally. Small particles are more likely to travel further, and we can detect them far from the particle source [39], [40]. This is a very important feature that has guided us in the design of our algorithm for particle source localization.

A. The low-cost particle sensor

One of the key elements in any search scenario is to choose a proper sensor. After studying multiple low-cost sensors available on the market, we chose the Plantower PMS 7003 [41]. The Plantower sensor is cheap, small in size and is supposed to measure particles with diameter from 300 nm to 10 μm . It uses laser scattering to count particles, that is, casting the laser on the suspended particles in the air, receiving and measuring the scattered light at a certain angle to compute the size of particles and the number of particles in a certain volume of air according to Mie scattering theory [42]. It has shown reliable enough results in real world tests [33], [34], and was reported to be suitable for monitoring short-lived airborne particle concentrations especially when coupled with wind data.

The Plantower sensor measures 12 different modalities at the same time, as summarized in Table I. Among the 12 sensing channels, the ones labeled with *CF1* (Calibration Factor 1) were not relevant to our environment since normalized for a specific industrial environment. The concentration of particles (channel 4 to 6) were also reflected by the number of particles (channel 9, 10 and 12). Therefore, we only chose the last six channels (in bold) as ‘‘observations’’ in the rest of the paper.

B. Particle plume model

There are various plume models, such as Gaussian Processes [43], the filament-based atmospheric dispersion model (also known as the Advection-Diffusion model) [44] and the pseudo-Gaussian model [45], [46]. We have chosen the pseudo-Gaussian model to model the smoke particle plume because it is widely used and relatively easy to be fitted. For a continuous point source in steady air flow, the following pseudo-Gaussian

TABLE I
MEASUREMENT MODALITIES OF PLANTOWER PMS 7003

No.	Measurement value	Abbreviation
1	PM 1 concentration ($CF = 1$)	c_{1_CF1}
2	PM 2.5 concentration ($CF = 1$)	$c_{2.5_CF1}$
3	PM 10 concentration ($CF = 1$)	c_{10_CF1}
4	PM 1 concentration (in atmospheric environment)	c_{1_A}
5	PM 2.5 concentration (in atmospheric environment)	$c_{2.5_A}$
6	PM 10 concentration (in atmospheric environment)	c_{10_A}
7	The number of particles ($d > 0.3\mu\text{m}$) in 0.1 liter of air	$n_{0.3}$
8	The number of particles ($d > 0.5\mu\text{m}$) in 0.1 liter of air	$n_{0.5}$
9	The number of particles ($d > 1\mu\text{m}$) in 0.1 liter of air	n_1
10	The number of particles ($d > 2.5\mu\text{m}$) in 0.1 liter of air	$n_{2.5}$
11	The number of particles ($d > 5\mu\text{m}$) in 0.1 liter of air	n_5
12	The number of particles ($d > 10\mu\text{m}$) in 0.1 liter of air	n_{10}

model can estimate the concentration reading at position (x, y, z) [47]:

$$C_{exp}(x, y, z) = \frac{Q}{2\pi\tilde{u}\sigma_y\sigma_z} e^{-\frac{y^2}{2\sigma_y^2} - \frac{z^2}{2\sigma_z^2}} \quad (1)$$

where x , y and z represent the relative position along the wind direction (X-axis), crosswind horizontal direction (Y-axis) and crosswind vertical direction (Z-axis), respectively. Q is the plume source release rate, \tilde{u} is the wind speed, σ_y and σ_z are the standard deviations along crosswind horizontal direction and crosswind vertical direction respectively, which can be fitted into functions of x . This model applies to both chemical and particulate plumes, but with different parameter values. Detailed plume model fitting and quantitative analysis are presented in Section IV-B.

III. ADAPTING THE INFOTAXIS ALGORITHM TO PARTICLES SOURCE LOCALIZATION

Our source localization solution is based on the Infotaxis algorithm, which is a well-established probabilistic algorithm for odor source localization proposed by Vergassola et al. [23]. This algorithm can be very efficient and flexible since it can be implemented without a very accurate plume model, and it holds a belief of the source location, which can help locate the source before the robot reaches its proximity. In this section, we will explain how the original Infotaxis algorithm works, and also how we have adapted it to the multi-modal sensing technology leveraged in this paper for particle plumes.

Infotaxis is based on a typical Bayesian inference method. The robot keeps a probability distribution S for the location of the source. In a 2D environment with one point source, the probability of the position (x, y) to be the source can be expressed as $P(x, y)$. In this environment, S is a map of $P(x, y)$. The entropy of the probability distribution map is:

$$Entropy(S) = - \sum P(x, y) \log(P(x, y)) \quad (2)$$

The higher the entropy is, the more uncertain the source location is.

The robot, while moving in the environment, senses a series of concentration $\{C_0, C_1, \dots\}$. Using a plume model

(discussed in Section IV-B), the probability of getting a certain concentration level C_k at the position (x, y) is expressed as:

$$P(C_k) = \sum P(C_k | source = (x_s, y_s)) P(x_s, y_s) \quad (3)$$

At the beginning of the search, the probability of every point to be the odor source is the same, so the probability map is uniformly distributed and thus the entropy is at its maximum.

The Infotaxis algorithm can be divided into four processes that keep iterating: observation, updating the probability map, evaluating future potential observation points (targets), and robot movement. Below, we explain the algorithm in detail.

A. Observation and probability map updating

At every step i , the robot will take a concentration measurement (observation) $O_i = (x, y, C_k)$.

Assuming $P(C_k | source = (x_s, y_s))$ satisfies Poisson distribution (the same as [25], [48]), it can be calculated by

$$P(C_k | source = (x_s, y_s)) = \frac{1}{C_k!} \times C_{exp}^{C_k} \times e^{-C_{exp}} \quad (4)$$

where C_k and C_{exp} represent the measured and the expected concentration level respectively. C_{exp} can be calculated by Eq. (1).

Let $P_i(x_s, y_s)$ represent the posterior probability at step i that the source is located at (x_s, y_s) . According to Bayesian inference,

$$P_i(x_s, y_s) = P(C_k | source = (x_s, y_s)) \cdot P_{i-1}(x_s, y_s) \quad (5)$$

Combining Eq. (4) and Eq. (5), the probability map can be updated by:

$$P_i = P_{i-1} \times C_{exp}^{C_k} \times e^{-C_{exp}} \quad (6)$$

With the particle sensor discussed in Section IV-A, we are able to sense the number of particles with different sizes. By considering the number of particles as equivalent to concentration, we fitted particle plumes into the pseudo-Gaussian concentration model. Furthermore, since more than one measurement value per location can be read through the sensor, multiple probability maps can be generated.

B. Target evaluation and robot movement

There are several potential target positions for the robot's next step. In this paper, we set 10 targets (5 in the front and 5 in the back) for the robot's movement (shown in Fig. 1). The robot can reach T5 and T6 through a straight line, T3, T4, T7, T8 along a curve with curvature 0.05, and T1, T2, T9, T10 along a curve with curvature 0.1. The distance to every target is the same and set to 0.3 m in this paper.

If the robot goes to the n th target located at (x_n, y_n) , the probability to get a concentration C_k can be expressed using Eq. (3). If $O_{i+1} = (x_n, y_n, C_k)$, the estimated S_{i+1} becomes $S_{i+1} = f(S_i, (x_n, y_n, C_k))$. The entropy of the estimated probability map S_{i+1} can be calculated with Eq. (2). The expected entropy yielded by this movement is:

$$Entropy_{exp} = \sum_k Entropy(S_{i+1}) P(C_k) \quad (7)$$

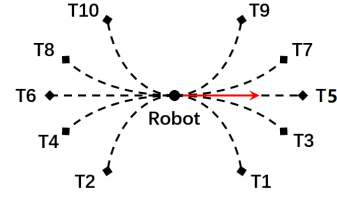


Fig. 1. Targets for robot's movement. The red arrow is the robot's heading direction. T1 ~ T10 are targets.

The robot will assess all the potential targets and all the potential observations; move towards the target yielding the smallest entropy, which means it will gain most information about the source location and therefore decrease the related uncertainty.

With repeated observations and movements, the entropy will keep decreasing until the robot finds a position with a high probability to be the source.

C. Adaptation to particle plumes

As mentioned before, for particle plumes we have multiple (in our case six) measurement channels and so we can generate multiple probability maps, one per channel. However, the Infotaxis algorithm uses only one single probability map. The immediate question is therefore how to adapt the Infotaxis algorithm to a multi-modal sensing landscape. Here below we illustrate four possible strategies to address this question.

1) *Single modality map*: A naïve method is to use only the most representative sensing modality in the algorithm and ignore the other ones. In this case, we use the smallest particle size measurements (in our case $n_{0.3}$) since they are the most dispersed ones allowing the robot to construct the most extended map, including regions far from the source (see details in Section IV-A).

2) *Unweighted multi-modality map (Integration Strategy I)*: A smarter but reasonably simple way to fuse all the measurements is to sum with equal weight all the values related to different particle sizes into a single map.

This form aggregates all the individual maps but does not bias the result of the fusion towards any one of them.

3) *Weighted multi-modality map (Integration Strategy II)*: Another equally interesting aggregation form is the weighted sum of all the six original maps, with the reciprocal of the entropy of each map representing the weight of that map in the integrated one. The probability distribution of the new map is therefore expressed as:

$$P(x, y) = \sum_{i=1}^6 \frac{P_i(x, y)}{Entropy(S_i)} \quad (8)$$

where $i = 1, 2, \dots, 6$ represents the index of the six probability maps we generated from the observation series directly. S_i represents the i th probability map.

Using the inverse entropy as the coefficient of the weighted sum assigns more importance to the probability maps with lower entropy but does not entirely ignore probability maps with high uncertainty, which prevents the probability distribution to

converge too quickly. The integrated probability map is used as the probability map in the target evaluation stage.

4) Multi-modality voting map (Integration Strategy III):

Multi-modality voting also takes all six sensing modalities into account. In this case, six single probability maps are updated and recorded simultaneously at every time step. For each of them, the target evaluation is conducted separately and the best movement target for each map will get a vote. The robot will eventually move towards the target with most votes.

More sophisticated methods to fuse the considered six sensing channels would be possible, for instance by developing dedicated measurement models for each modality. However, this would require the availability of additional ground truth measurement equipment and a thorough calibration campaign that is out of scope of this paper.

D. Plume finding stage and particle concentration levels

To map the particle number values measured by the sensor to concentration levels (C_k in Eq. (3)) used in the algorithm, we first measure the baseline particle concentration in the clean air and then define multiple levels of concentrations for higher values. Similar to [25], [48], we set the robot to do a crosswind scan as its plume finding stage and record concentration values at the beginning of the experiments. The robot will come to the highest concentration point during this crosswind scan and start the algorithm. The concentration baseline is set to a value slightly larger than the lowest measured value in the crosswind row. We set the same fixed baseline B for all six modalities based on the measurement we get from the dominant particle size (which is the smallest size particle). Based on B , we define 10 concentration levels similar to the previous works [25], [48].

E. End of the algorithm

Our goal is to locate the position of the source by calculating the probability distribution, so it is not necessary to let the robot reach the source. We define the following conditions for ending the algorithm:

- the entropy of the integrated probability map becomes lower than a threshold;
- the highest probability point in the map does not move for a number of consecutive measurements;
- the robot arrives at the source;
- the robot goes to the boundaries of the arena.

Only one of these conditions needs to be met to terminate the algorithm. The ending strategy is similar to our previous work [25], but we adjusted the related threshold according to the peculiarities of particle plumes.

F. Computational complexity

Assuming the searching area is modeled as a grid, we denote the total number of cells with A . Assume also that N is the number of potential targets, K the number of concentration levels and I is the number of sensing modalities. The computational complexity of the algorithm entails two components: the computational cost of updating the probability

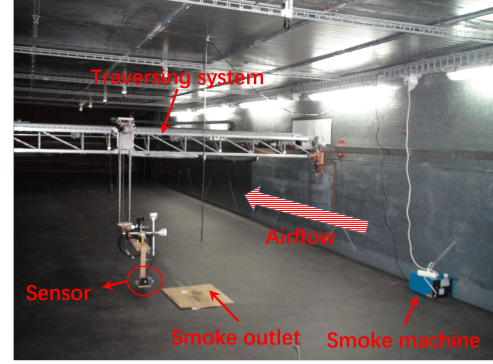


Fig. 2. Wind tunnel environment

map which is a function of the number of cells, i.e. $O(A)$, and that of evaluating the best target and move to it, which is a function of the number of potential targets, the number of concentrations levels, the number of sensing modalities, and the number of cells involved in the probability map estimation at the next step. Therefore, with a single modality the overall computational cost is $O(A + A \cdot N \cdot K)$. The computational complexity of the algorithm adapted with Integration Strategies I and II is $O(I \cdot A + A \cdot N \cdot K)$. The computational complexity of the algorithm adapted with Integration Strategy III is $O(I \cdot A + I \cdot A \cdot N \cdot K)$. The difference between Integration Strategies I, II and III comes from their different sequential order of map integration and target evaluation.

We believe it is worth to maintain 6 maps because it does not increase the computational complexity greatly, especially for Integration Strategies I and II. We set $N = 10$, $K = 10$, $I = 6$ in this paper. The above equations of computation complexity are $O(101 \cdot A)$, $O(106 \cdot A)$ and $O(606 \cdot A)$, respectively. By maintaining 6 maps, the simplicity of computation is sacrificed a bit in order to greatly promote the source localization accuracy.

IV. PLUME MODEL FITTING AND ALGORITHM EVALUATION

We conducted a refined data collection in a wind tunnel and fitted the particle plume model into the pseudo Gaussian model. We have evaluated the algorithm in simulation and also in the wind tunnel. In simulation, we have compared the performance of the proposed algorithm based on the three multi-modal map integration approaches mentioned above with the original Infotaxis algorithm using a single modality map.

A. Experimental setup

Our controlled environment is represented by a multi-channel wind tunnel. The channel we selected for this work has a dimension of $20\text{ m} \times 4\text{ m} \times 2\text{ m}$ and can generate air flow in this particular channel with any arbitrary speed between 0.1 m/s and 5 m/s . Thus, we are able to conduct extensive repeated experiments with different environmental settings. The wind channel is equipped with a traversing system moving in three dimensions (shown in Fig. 2). A smoke machine is placed inside the wind channel as an airborne particle source, burning a special oil to generate particles. The tip of the smoke machine is placed 6 cm above the floor.

TABLE II
FITTING PARAMETERS

	$\frac{Q}{2\pi u}$	$\sigma_y = ax^b$		$\sigma_z = ax + \frac{b}{x} + c$		
		a	b	a	b	c
$n_{0.3}$	2597.8	0.0909	0.3073	0.9127	6.4255	-3.9602
$n_{0.5}$	898.2	0.0928	0.2966	1.0014	6.9873	-4.3382
n_1	3837.6	0.0925	0.3123	7.8199	47.7052	-32.7684
$n_{2.5}$	6403.1	0.0912	0.3379	29.216	166.7206	-121.8657
n_5	5754.2	0.09	0.3379	56.247	263.1706	-217.9539
n_{10}	1462.4	0.0881	0.3513	40.1203	150.197	-145.8391

TABLE III
RMSE OF σ_y AND σ_z

RMSE	σ_y		σ_z	
	Power function	Linear function	Nike function	Linear function
$n_{0.3}$	0.0196	0.0210	0.4852	0.9548
$n_{0.5}$	0.0205	0.0216	0.4871	0.8858
n_1	0.0217	0.0226	2.9222	6.6466
$n_{2.5}$	0.0237	0.0241	9.6936	21.3806
n_5	0.0252	0.0252	20.7342	44.0463
n_{10}	0.0259	0.0266	18.2744	30.8333

B. Plume model fitting

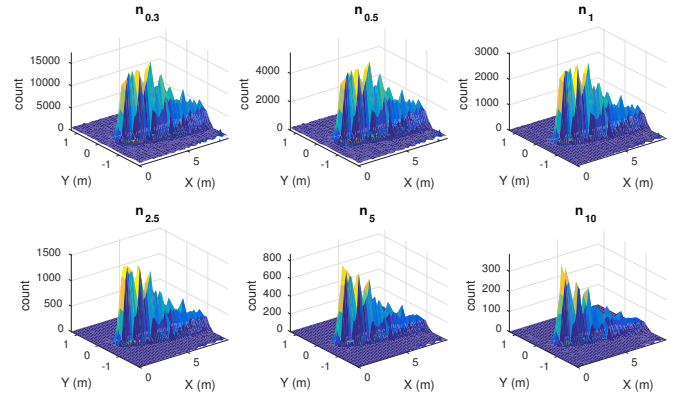
To fit the particle plume into the pseudo Gaussian model, measurements of a real particle distribution is needed. To this purpose a refined scan in 3D space was carried out in the wind tunnel. The Plantower sensor was mounted on the tip of the traversing system. The wind speed was set to 1.4 m/s. The scan covered a volume of $9.20m \times 3.25m \times 0.2m$ and measurements were taken at $50 \times 20 \times 5$ location points uniformly distributed within this volume. The volume that was in close proximity of the smoke machine, our particle plume source, was not scanned for sensor safety reasons.

Multiple slices of the obtained particle map perpendicular to the X-axis were used to calculate σ_y and σ_z . In a previous work fitting the chemical plume [25], both σ_y and σ_z were fitted into linear functions of x . In this paper, σ_y was fitted into the empirical power law function [49] of x and σ_z is fitted into the Nike function of x for decreasing error.

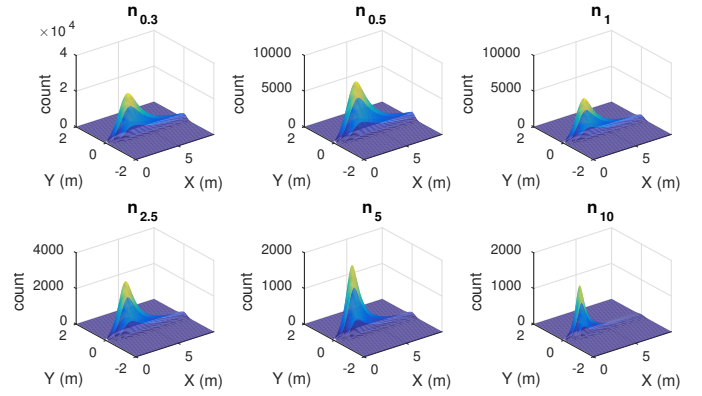
The fitting result on the $z = 0$ plane is shown in Fig. 3. The fitting parameters are shown in Table II. The RMSE of σ_y and σ_z are presented in Table III and compared with that of linear fitting. It can be seen that when σ_y and σ_z are fitted into the power law function and the Nike function respectively, the RMSE is significantly lower.

We see that the pseudo-Gaussian model that is typically used for odor plume dispersion modeling can also be adapted to the particle plumes, with different forms of σ_y and σ_z . However, the major difference is that for particle plumes, since the sensing is multi-modal, we have multiple pseudo-Gaussian models describing various particle sizes.

Our own measurement campaign confirms the finding reported in the literature mentioned above [39], [40]: *the smaller are the particles, the larger is their transportation range*.



(a) Scanned plume



(b) Fitted plume

Fig. 3. Figure (a) is the measured plume distribution in the wind tunnel. The figures show the floor plane. Each measurement value is the mean of 12 samples. Figure (b) shows the fitted pseudo-Gaussian plume. $n_{0.3} \sim n_{10}$ correspond to the last six channels reported in Table I. The particle plume source is placed at (0,0).

C. Simulation

Before evaluating the proposed method in the physical reality, we have developed and tested the algorithm in simulation using Webots [50], a high-fidelity, open-source robotic simulator.

1) *Setup*: The setup (shown in Fig. 4) is entirely similar to the wind tunnel mentioned in Section IV-A. In the coordinate system of Fig. 4, the particle source is located at (10.6, 0.2) (in meters, the same below). A simulated Khepera IV robot equipped with a simulated particle sensor was used in the simulations. We use the mean μ and the standard deviation σ of the concentration value in the wind tunnel scan to generate random values with Gaussian probability distribution $N(\mu, \sigma^2)$ to simulate the particle sensor's output. In our paper we assume that the robot knows its pose all the time: in reality, we achieved that by tracking the robot with an external camera system and forwarding the absolute positioning information to the robot through a narrowband communication channel; in simulation such configuration is emulated through the use of a supervisory functionality available in the simulator. However, this is not at all representing a limitation of the algorithm: such localization functionality could be easily replaced by a SLAM method for instance by constructing a graph-structured

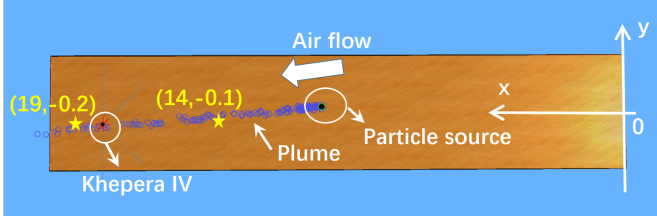


Fig. 4. Webots setup. The blue oxygens represent particle plumes.

map along with an exploration process [51] or by leveraging vision-based navigation using signed distance fields [52]. As mentioned in Section I, the proposed algorithm focuses on the process of plume tracking while the plume finding strategy is implemented in a simple way: the robot will perform a crosswind scan, identify the highest concentration point, and start the particle source localization algorithm from inside the plume. Although the plume finding strategy might appear simple, some systematic tests in simulation characterized by a random start of the robot outside the plume have demonstrated both the robustness of our plume finding strategy as well the consistency in the performance of the overall particle source localization. To study the algorithm performance when the robot starts from a different location (close to or far from the source), we compared two cases in which the robot's initial position was $(19, -0.2)$ and $(14, -0.1)$, respectively (marked by stars in Fig. 4). The robot's velocity is set to 0.34 m/s in the movement stage. For each set of experiments, 50 runs were carried out.

2) *Metrics*: The performance of the algorithm has been evaluated using two metrics: *distance error* (i.e. the distance between the maximum probability position in the map and the actual position of the source when the algorithm ends) and the *number of iterations*, where each iteration is defined as the whole process of the robot sampling in a position, updating the probability map, taking decision to go to a target point, and moving to the next position. The former of these two metrics can show the algorithm's accuracy while the latter can illustrate its efficiency over time.

3) *Evaluation of the proposed method*: One of the modifications we did to the Infotaxis algorithm was to integrate the probability maps. In the simulation stage, we studied the difference among the four strategies explained in Section III-C. The results are shown in Fig. 5.

When the robot starts from $(19, -0.2)$, in terms of distance error, the algorithm based on a single modality map performs reasonably well for the four cases of $n_{0.3}$ to $n_{2.5}$. This is because small particles travel further and the robot can sense high concentrations even at the beginning of the runs, when the robot is far away from the source. But for single-modality strategies based on n_5 and n_{10} maps, since the large particles do not travel very far from the source, the robot could sense only low values far from the source and this was reflected in the results. On the other hand, for both Integration Strategy I and II, the distance error was less than 1 m in most of the runs, which represents a much superior performance in comparison to strategies based on any single modality. Integration Strategy

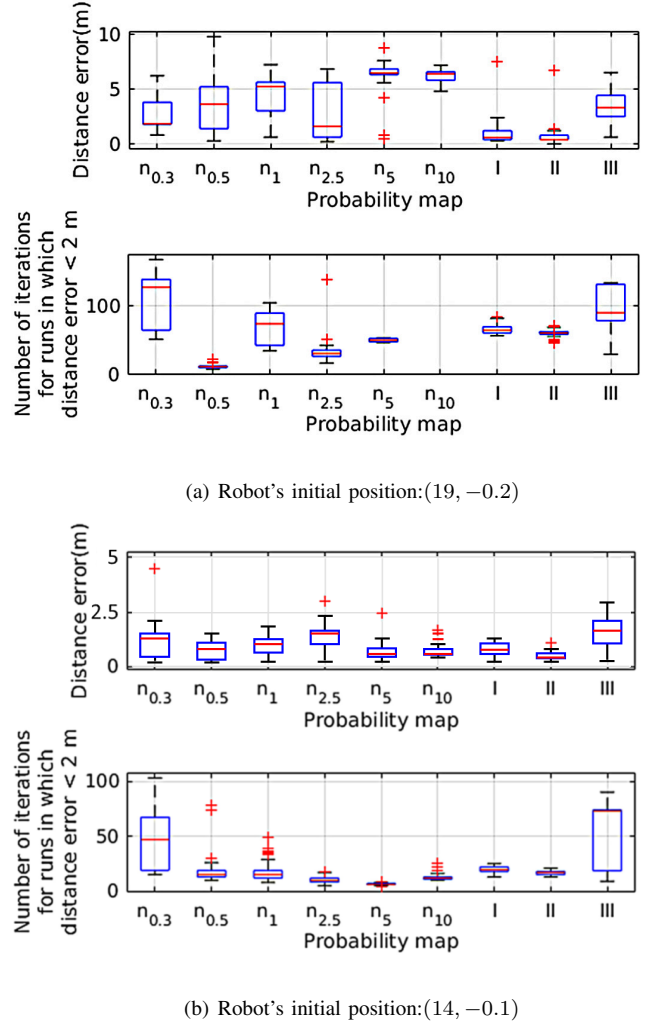


Fig. 5. Performance evaluation in Webots for single probability maps, and the three different integration strategies reported in Section III-C. $n_{0.3} \sim n_{10}$ are six single probability maps respectively. I: unweighted multi-modal map. II: weighted multi-modal map. III: multi-modal voting map. (In the boxplot, the middle bar indicates the median.)

II shows slightly higher performance than Integration Strategy I. Integration Strategy III does not yield as good results as I and II. The reason why Integration Strategy II shows better results is that it takes into account the uncertainty of the individual probability maps - more weight is assigned to maps with less uncertainty, thus the robot has higher chance to make the right choice for its next movement. Integration Strategy I assigns equal weights to all maps, in which case probability maps with small uncertainty cannot give the robot more guidance than those with large uncertainty. For Integration Strategy III, since some maps may be not as reliable as others when their entropies are high, voting results can even contradict the action yielding minimum expected entropy. This explains why the performance of Integration Strategy III is worse than the other two. In terms of number of iterations, we only consider runs in which the distance error is smaller than 2 m (which can be considered as an acceptable result). The methods with Integration Strategy I and II show an average value comparable to the runs based on single-modality maps. In a few cases of single maps, the

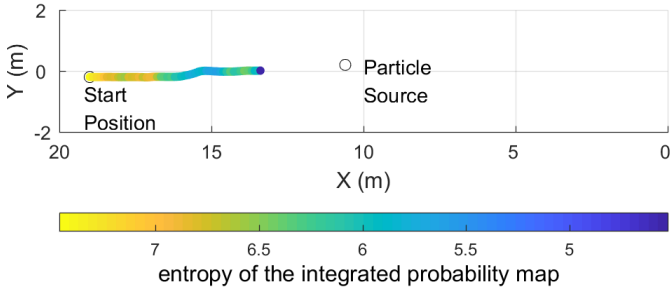


Fig. 6. Trajectory of the robot running the source localization algorithm in a simulation run. X and Y are coordinates in the simulated wind tunnel. The color code on the trajectory is related to the entropy of the integrated probability map.

number of iterations was lower than others because the robot got limited information and ran into walls, thus the algorithm was terminated sooner, but the estimated source location was close to the actual one by coincidence. Integration Strategy II also shows slightly better results than I and III on this metric.

When the robot starts from $(14, -0.1)$, a location relatively close to the airborne particle source, the source localization is much more accurate. The distance errors in all settings except for Integration Strategy III are similar - less than $1.5 m$. Among them, Integration Strategy II still yields the best result. In terms of number of iterations, Integration Strategy II also outperforms the other integration strategies and some of those leveraging single-modality maps.

Fig. 6 shows the robot's trajectory when working with Integration Strategy II in a simulation run. We can see that the robot almost goes along a straight line and the algorithm stops before the robots reach the particle source when the robot has already acquired sufficient information to calculate the particle source location. The trajectory demonstrates the efficiency of the proposed algorithm.

From Fig. 5, it can be seen that the performance of the algorithm with the three integrated strategies is less dependent on the initial position, while the performance of the single-map algorithm can be affected a lot. In fact, because the dispersion of larger particles is limited (discussed in Section IV. B), when the robot starts from $(19, -0.2)$, i.e. relatively far from the particle source, it can only observe a very low concentration level of them at the beginning. When guided by a single modality map, the robot fails to estimate the source location properly, loses its direction, and might even collide with environmental boundaries. This leads to high distance error for single modality maps as shown Fig. 5(a), especially for large particles such as n_5 and n_{10} . When the robot starts from $(14, -0.1)$, i.e. closer to the particle source, all particles sizes can be observed from the beginning. Thus, the localization accuracy is enhanced, even for single maps. This difference is especially striking for n_5 and n_{10} maps.

4) *Probability map evolution*: Here we look into how the probability maps change at every iteration when running the Infotaxis algorithm with Integration Strategy II. In Fig. 7, the observation and the entropy of the probability distribution are shown for a simulation run. In Fig. 8, we represent the

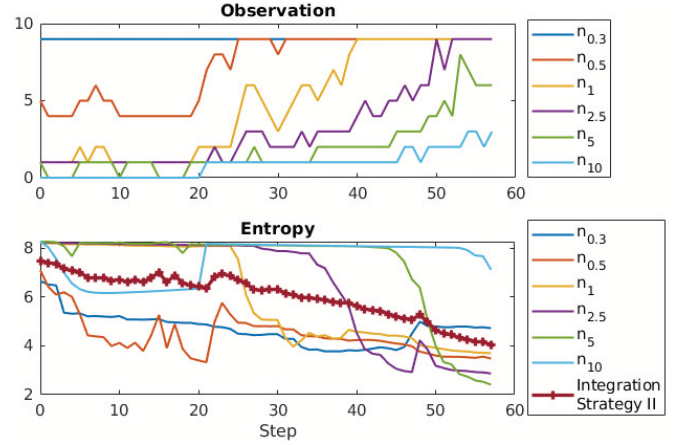


Fig. 7. Observation and entropy at every iteration

probability maps evolution process in this run. At the beginning, only $n_{0.3}$ and $n_{0.5}$ could be continuously detected. With the observation input, the probability maps of $n_{0.3}$ and $n_{0.5}$ kept being updated and their associated entropy decreased quickly. However, low entropy does not result necessarily in an accurate source localization. From Fig. 8(b) and (c), it can be seen that even if $n_{0.3}$ and $n_{0.5}$ maps have a low entropy, the grid point with highest probability of source location is still far away from the source. If the algorithm only takes $n_{0.3}$ or $n_{0.5}$ maps into consideration, it will terminate more quickly (based on the termination conditions in Section III. E) and estimate the source location with a larger error. For n_1 and $n_{2.5}$, the robot can sense some patchy and weak concentration values. For n_5 and n_{10} , the concentration was always below the baseline at the early stage. These six probability maps are of great diversity. Using the weighted aggregation of probability maps, the algorithm never ends before reaching the region occupied also by large particles because the entropy of the integrated map will not decrease to the preset threshold until all the single maps present a low entropy. To decrease the entropy of all maps (especially n_5 and n_{10} , whose entropy is usually large at the beginning because the robot cannot observe large particles far from the source), the robot has to move closer to the source and observe some high concentration of large particles. This prevents the algorithm from terminating too quickly when the robot is still far from the source, thus reducing the source localization error. When the robot is able to detect all six different particle sizes, the entropy of every probability map decreases quickly, and the robot can locate the source position accurately. In a few cases, due to the patchiness of the plume, the robot detects concentrations which vary a lot from its expectations calculated by the plume model, which leads to a sharp increase in the entropy.

D. Wind tunnel experiments

To evaluate the method in the physical world under various environmental conditions, we have conducted extensive experiments in our wind tunnel.

1) *Robotic platform*: A Khepera IV robot (see Fig. 9) was used in the wind tunnel experiments. The Plantower PMS 7003

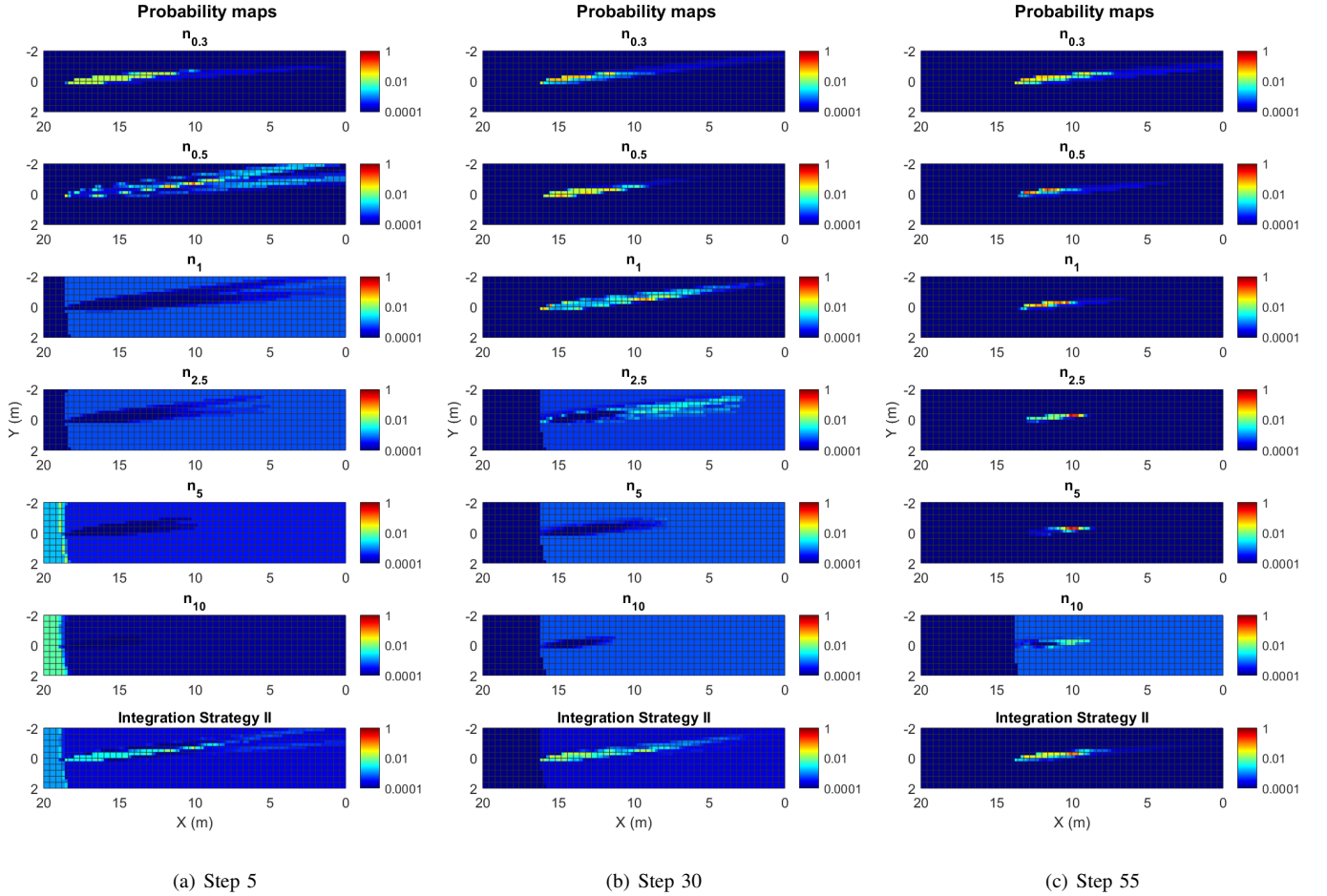


Fig. 8. (a)-(c) is the distribution of the six single probability maps, $n_{0.3} \sim n_{10}$ and the weighted multi-modality map (from top to bottom). The redder one is, the higher probability this point is the plume source. X and Y are coordinates in the simulated wind tunnel.

sensor was mechanically connected to the robot. It was driven by an Arduino Uno board, which communicated with the robot through UART. The measurement data was sent to the robot once per second. The wind direction sensor board was the same as in [48]. When running the proposed algorithm, the robot measures the wind direction and the particle concentrations, estimates the source location, evaluates its movements, chooses the best one, and actuates the wheels at every step. The red and green LED markers on the top of the robot were captured by cameras on the top of the wind tunnel to track the robot's trajectory.

We conducted 10 runs for every environmental condition.

2) *Performance evaluation*: From the simulations, we concluded that among the single-modality map variants, the measurements related to the smallest particles ($n_{0.3}$) led to the best results. Moreover, we could also conclude that among the multi-modality integration strategies the best one was that based on a weighted aggregation (Integration Strategy II). Therefore, in the wind tunnel experiments we only considered the probability map of $n_{0.3}$ for the single-modality strategy and the weighted multi-modal integration strategy.

The results are shown in Fig. 10. The number of iterations for the variant leveraging an integrated map is much higher, because when working with the single-modality strategy, the

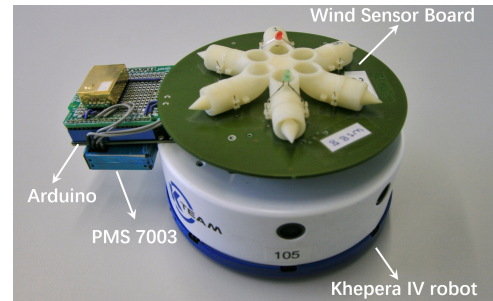


Fig. 9. Khepera IV robot with the Plantower sensor mechanically anchored to it.

robot always measures high concentration of $n_{0.3}$ and does not get enough guidance from its observations, thus runs to the boundaries quickly and ends the algorithm. The distance error shows that the Integration Strategy II delivers much more accurate source localization than that achieved through single-modality map. This result confirms those obtained in simulation reported in Section IV-C.

We present the robot's trajectory in an experiment run in Fig. 11. We can see that the robot's trajectory is similar to the one in simulation, although the algorithm ends when the robot gets closer to the smoke machine.

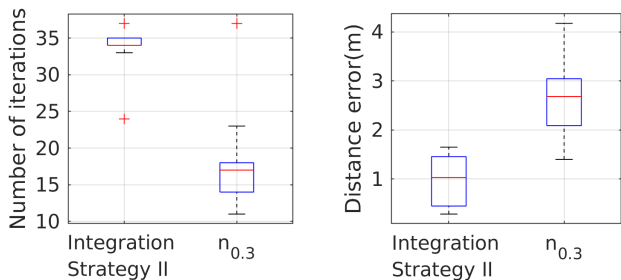


Fig. 10. Performance evaluation in the wind tunnel for single modality and the weighted multi-modal strategy (Integration Strategy II).

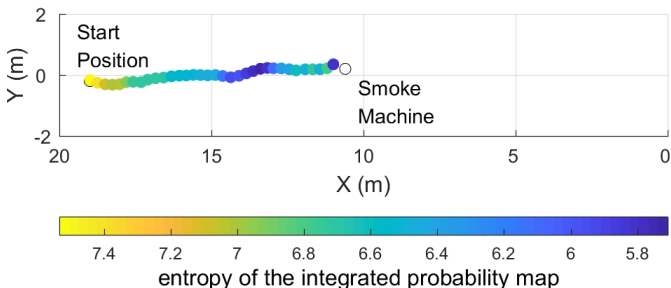


Fig. 11. Trajectory of the robot running the proposed algorithm in an experiment run. X and Y are the coordinates in the wind tunnel. The color code on the trajectory is related to the entropy of the integrated probability map.

TABLE IV
ENVIRONMENTAL CONDITIONS FOR THE RUNS IN THE WIND TUNNEL

Setting	Source release rate	Wind speed
A	low (nominal value)	fast (1 m/s)
B	high (150% of nominal value)	fast (1 m/s)
C	low (nominal value)	slow (0.2 m/s)
D	high (150% of nominal value)	slow (0.2 m/s)

3) *Influence of environmental conditions:* We conducted experiments in the wind tunnel to test the robustness of the algorithm under different environmental parameter settings shown in Table IV. When decreasing the release rate or increasing the wind speed, the plume would become narrower and straighter downstream. Otherwise, the plume was wider. According to Beaufort scale [53], the wind speed under 0.5 m/s can be considered "calm", thus Settings C and D are used to evaluate the performance of the algorithm in less windy conditions. For Settings A - D, we used exclusively Integration Strategy II to guide the robot.

The results are shown in Fig. 12. The number of iterations shows that when both the wind speed and the particle source release rate are high (Setting B) the algorithm is much faster than in the other conditions. The other settings have more or less the same number of iterations. In most of runs, the distance error for Settings A and B is below 1 m . A sporadically higher localization error than in simulation might be due to a noisier wind sensor in reality. For setting C and D, the average (median) distance error (around 0.7 m and 2.5 m respectively)

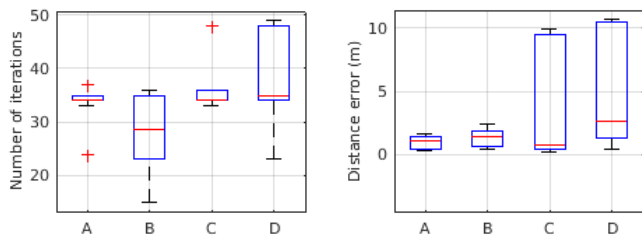


Fig. 12. Performance evaluation in the wind tunnel in different environmental settings.

is acceptable compared to the dimension of the experimental area, but the performance variation across runs is too large, which implies that there are several cases in which the robot misestimated the source location. This is because under weak wind, the robot was not able to gather enough observations when being far away from the source.

It should be mentioned that, for all experiments, the same plume distribution model described in Section IV-B was used (i.e. high release rate for the source and fast wind speed), which does not fit exactly the dispersion of the particles in all the environmental conditions mentioned in Table IV. Moreover, it is worth noticing that in our wind tunnel, due to a mild manufacturing asymmetry, the wind has a noticeable angle towards one side of the channel. The compensation of the biased wind direction when fitting the plume model might be imperfect. On the other hand, the obtained results indirectly point out that there is no need to have perfect plume models and the proposed algorithm is robust to such modeling inaccuracies.

The drawbacks of the proposed multi-modality Infotaxis algorithm are similar to those of the original Infotaxis algorithm. The algorithm depends on the accurate global localization of the robot. The performance of the algorithm gracefully degrades with an increased robot localization error and inaccuracies in the plume model, subject to existing localization techniques. When the localization of the robot or the plume model is not accurate, the source location probability maps will be updated through the plume model with higher uncertainty at each step. The reliability of the maps decreases, so the accuracy of the estimated source location will be affected. The performance of the algorithm in dynamic/ lower speed (closer to diffusion) wind field still needs to be evaluated. Compared to the original Infotaxis algorithm, the computation load is heavier when updating probability maps for all six modalities at each iteration, but our algorithm can still run fully on-board on a small-scale, resource-constrained robot such as the Khepera IV.

V. CONCLUSION AND OUTLOOK

We successfully adapted the Infotaxis algorithm to search for particle sources with a low-cost optical sensor. The algorithm was evaluated in simulation and in a wind tunnel in different environmental settings (in terms of source release rate and wind speed).

Overall, the results obtained show that the proposed algorithm is efficient, can achieve good source localization accuracy and is robust to different environmental settings. In particular, at high wind speed, the algorithm is able on average to locate

the source with less than 1 meter error in an 80 m^2 arena. With the wind speed decreasing, the average (median) error distance would increase to 2.5 m, but it is still acceptable in a large area. The variant based on the Integration Strategy II is clearly superior to a naïve use of the Infotaxis algorithm leveraging a map based on one of the six available sensing modalities.

In the future, we are planning to study how the proposed algorithm will behave in a multiple source scenario. The impact of the plume dispersion process on the efficiency of the algorithm will be evaluated in future work. Further study on the robustness of the proposed algorithm in dynamic/lower speed (diffusion regime) wind field will be conducted. We will also evaluate the algorithm in scenarios with obstacles. A quantitative analysis on the impact of robot localization on the performance of the algorithm will be also presented in future work. And we are going to study a self-adaptive way to set the concentration baseline and adjust the parameters of the plume model. Keeping balance between greedily exploit information and further exploring the environment will also be a promising research direction.

REFERENCES

- [1] R. Rozas, J. Morales, and D. Vega, "Artificial smell detection for robotic navigation," in *IEEE Int. Conf. on Advanced Robotics*, 1991, pp. 1730–1733.
- [2] T. Basten, H. de Bree, and S. Sadasivan, "Acoustic eyes, a novel sound source localization and monitoring technique with 3d sound probes," in *presentation at ISMA 2008 International Conference on Noise and Vibration Engineering*, 2008.
- [3] Z. Liu and S. Abbaszadeh, "Double q-learning for radiation source detection," *Sensors*, vol. 19, no. 4, p. 960, 2019.
- [4] H. Ishida, Y. Wada, and H. Matsukura, "Chemical sensing in robotic applications: A review," *IEEE Sensors Journal*, vol. 12, no. 11, pp. 3163–3173, 2012.
- [5] 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.
- [6] R. A. Russell, D. Thiel, R. Deveza, and A. Mackay-Sim, "A robotic system to locate hazardous chemical leaks," in *IEEE Int. Conf. on Robotics and Automation*, vol. 1, 1995, pp. 556–561.
- [7] H. Ishida, K.-i. Suetsugu, T. Nakamoto, and T. Morizumi, "Study of autonomous mobile sensing system for localization of odor source using gas sensors and anemometric sensors," *Sensors and Actuators A: Physical*, vol. 45, no. 2, pp. 153–157, 1994.
- [8] 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.
- [9] L. Marques and A. T. De Almeida, "Electronic nose-based odour source localization," in *IEEE Int. Workshop on Advanced Motion Control*, 2000, pp. 36–40.
- [10] J. M. B. Calvo, S. B. i Badia, H. T. Simó, and P. F. Verschure, "The real-world localization and classification of multiple odours using a biologically based neurorobotics approach," in *IEEE Int. Joint Conf. on Neural Networks*, 2010, pp. 1–7.
- [11] Y. Kuwana, S. Nagasawa, I. Shimoyama, and R. Kanzaki, "Synthesis of the pheromone-oriented behaviour of silkworm moths by a mobile robot with moth antennae as pheromone sensors," *Biosensors and Bioelectronics*, vol. 14, no. 2, pp. 195–202, 1999.
- [12] A. Lilienthal, D. Reimann, and A. Zell, "Gas source tracing with a mobile robot using an adapted moth strategy," in *Autonome Mobile Systeme*. Springer, 2003, pp. 150–160.
- [13] L. L. López, V. Vouloutsis, A. E. Chimeno, E. Marcos, S. B. i Badia, Z. Mathews, P. F. Verschure, A. Ziyatdinov, and A. P. i Lluna, "Moth-like chemo-source localization and classification on an indoor autonomous robot," in *On Biomimetics*, A. L. D. Pramatarova, Ed. Rijeka: IntechOpen, 2011, ch. 21.
- [14] T. Consi, J. Atema, C. Goudey, J. Cho, and C. Chryssostomidis, "AUV guidance with chemical signals," in *IEEE Int. Symp. on Autonomous Underwater Vehicle Technology*, 1994, pp. 450–455.
- [15] F. W. Grasso, J. A. Basil, and J. Atema, "Toward the convergence: robot and lobster perspectives of tracking odors to their source in the turbulent marine environment," in *IEEE Int. on Intelligent Control*, 1998, pp. 259–264.
- [16] F. W. Grasso and J. Atema, "Integration of flow and chemical sensing for guidance of autonomous marine robots in turbulent flows," *Environmental Fluid Mechanics*, vol. 2, no. 1-2, pp. 95–114, 2002.
- [17] R. A. Russell, A. Bab-Hadiashar, R. L. Shepherd, and G. G. Wallace, "A comparison of reactive robot chemotaxis algorithms," *Robotics and Autonomous Systems*, vol. 45, no. 2, pp. 83–97, 2003.
- [18] V. Hernandez Bennetts, A. J. Lilienthal, P. Neumann, and M. Trincavelli, "Mobile robots for localizing gas emission sources on landfill sites: is bio-inspiration the way to go?" *Frontiers in Neuroengineering*, vol. 4, p. 20, 2012.
- [19] J. L. Blanco, J. G. Monroy, A. Lilienthal, and J. Gonzalez-Jimenez, "A kalman filter based approach to probabilistic gas distribution mapping," in *Annual ACM Symp. on Applied Computing*. ACM, 2013, pp. 217–222.
- [20] D. Grünbaum and M. A. Willis, "Spatial memory-based behaviors for locating sources of odor plumes," *Movement ecology*, vol. 3, no. 1, p. 11, 2015.
- [21] H. Hajieghrary, M. A. Hsieh, and I. B. Schwartz, "Multi-agent search for source localization in a turbulent medium," *Physics Letters A*, vol. 380, no. 20, pp. 1698–1705, 2016.
- [22] J.-G. Li, Q.-H. Meng, Y. Wang, and M. Zeng, "Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm," *Autonomous Robots*, vol. 30, no. 3, pp. 281–292, 2011.
- [23] M. Vergassola, E. Villermaux, and B. I. Shraiman, "'infotaxis' as a strategy for searching without gradients," *Nature*, vol. 445, no. 7126, p. 406, 2007.
- [24] N. Voges, A. Chaffiol, P. Lucas, and D. Martinez, "Reactive searching and infotaxis in odor source localization," *PLoS computational biology*, vol. 10, no. 10, p. e1003861, 2014.
- [25] J. Ruddick, A. Marjovi, F. Rahbar, and A. Martinoli, "Design and performance evaluation of an infotaxis-based three-dimensional algorithm for odor source localization," in *IEEE/RJS International Conference on Intelligent Robots and Systems*, 2018, pp. 1413–1420.
- [26] X. Chen and J. Huang, "Combining particle filter algorithm with bio-inspired anemotaxis behavior: a smoke plume tracking method and its robotic experiment validation," *Measurement*, vol. 154, pp. 1–11, 2020.
- [27] J. A. Farrell, S. Pang, and W. Li, "Plume mapping via hidden markov methods," *IEEE Trans. on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 33, no. 6, pp. 850–863, 2003.
- [28] 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/RJS Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 570–576.
- [29] 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 Sensors*, 2010, pp. 999–1004.
- [30] M. Maksimović, "Evaluating the optimal sensor placement for smoke detection," *Yugoslav Journal of Operations Research*, vol. 26, no. 1, 2016.
- [31] Q. Wu, J. Cao, C. Zhou, J. Huang, Z. Li, S.-M. Cheng, J. Cheng, and G. Pan, "Intelligent smoke alarm system with wireless sensor network using zigbee," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.
- [32] Y. Chu, V. Kodur, and D. Liang, "A probabilistic inferential algorithm to determine fire source location based on inversion of multidimensional fire parameters," *Fire technology*, vol. 53, no. 3, pp. 1077–1100, 2017.
- [33] M. Badura, P. Batog, A. Drzeniecka-Osiadacz, and P. Modzel, "Evaluation of low-cost sensors for ambient pm2.5 monitoring," *Journal of Sensors*, vol. 2018, Article ID 5096540, 16 pages, 2018.
- [34] F. M. Bulot, S. J. Johnston, P. J. Basford, N. H. Easton, M. Apetroaie-Cristea, G. L. Foster, A. K. Morris, S. J. Cox, and M. Loxham, "Long-term field comparison of multiple low-cost particulate matter sensors in an outdoor urban environment," *Scientific Reports*, vol. 9, no. 1, p. 7497, 2019.
- [35] Y. Xing, T. A. Vincent, H. Fan, E. Schaffernicht, V. H. Bennetts, A. J. Lilienthal, M. Cole, and J. W. Gardner, "Firenose on mobile robot in harsh environments," *IEEE Sensors Journal*, 2019.
- [36] A. Gaur, A. Singh, A. Kumar, K. S. Kulkarni, S. Lala, K. Kapoor, V. Srivastava, A. Kumar, and S. C. Mukhopadhyay, "Fire sensing technologies: A review," *IEEE Sensors Journal*, vol. 19, no. 9, pp. 3191–3202, 2019.
- [37] D. A. Duecker, A. R. Geist, E. Kreuzer, and E. Solowjow, "Learning environmental field exploration with computationally constrained un-

derwater robots: Gaussian processes meet stochastic optimal control,” *Sensors*, vol. 19, no. 9, p. 2094, 2019.

- [38] P. Morere, R. Marchant, and F. Ramos, “Sequential bayesian optimization as a pomdp for environment monitoring with uavs,” in *2017 IEEE International Conference on Robotics and Automation*. IEEE, 2017, pp. 6381–6388.
- [39] N. Gao and J. Niu, “Modeling particle dispersion and deposition in indoor environments,” *Atmospheric environment*, vol. 41, no. 18, pp. 3862–3876, 2007.
- [40] B. Zhao, C. Yang, X. Yang, and S. Liu, “Particle dispersion and deposition in ventilated rooms: testing and evaluation of different eulerian and lagrangian models,” *Building and Environment*, vol. 43, no. 4, pp. 388–397, 2008.
- [41] “PlanTower PMS 7003, www.plantower.com,” 2018.
- [42] G. Mie, “Beiträge zur optik trüber medien, speziell kolloidaler metallösungen,” *Annalen der physik*, vol. 330, no. 3, pp. 377–445, 1908.
- [43] C. Stachniss, C. Plagemann, and A. J. Lilienthal, “Learning gas distribution models using sparse gaussian process mixtures,” *Autonomous Robots*, vol. 26, no. 2-3, pp. 187–202, 2009.
- [44] J. A. Farrell, J. Murlis, X. Long, W. Li, and R. T. Cardé, “Filament-based atmospheric dispersion model to achieve short time-scale structure of odor plumes,” *Environmental fluid mechanics*, vol. 2, no. 1-2, pp. 143–169, 2002.
- [45] A. Marjovi and L. Marques, “Optimal swarm formation for odor plume finding,” *IEEE transactions on cybernetics*, vol. 44, no. 12, pp. 2302–2315, 2014.
- [46] F. Rahbar, A. Marjovi, and A. Martinoli, “An algorithm for odor source localization based on source term estimation,” in *International Conference on Robotics and Automation*. IEEE, 2019, pp. 973–979.
- [47] S. P. Arya *et al.*, *Air pollution meteorology and dispersion*. Oxford University Press New York, 1999, vol. 310.
- [48] T. Lochmatter, “Bio-inspired and probabilistic algorithms for distributed odor source localization using mobile robots,” Ph.D. dissertation, No. 4628, School of Architecture, Civil and Environmental Engineering, EPFL, 2010.
- [49] S. R. Hanna, “A simple method of calculating dispersion from urban area sources,” *Journal of the Air Pollution Control Association*, vol. 21, no. 12, pp. 774–777, 1971.
- [50] O. Michel, “Cyberbotics ltd. webots™: professional mobile robot simulation,” *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, p. 5, 2004.
- [51] C. Wang, W. Chi, Y. Sun, and M. Q.-H. Meng, “Autonomous robotic exploration by incremental road map construction,” *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 4, pp. 1720–1731, 2019.
- [52] H. Huang, Y. Sun, H. Ye, and M. Liu, “Metric monocular localization using signed distance fields,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2019, pp. 1195–1201.
- [53] Wikipedia contributors, “Beaufort scale — Wikipedia, the free encyclopedia,” 2020, [Online; accessed 1-April-2020]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Beaufort_scale&oldid=948400305



Xinxing Chen graduated from Huazhong University of Science and Technology (HUST), China and received the Bachelor of Engineering degree from HUST in 2015. She is currently a Ph.D. student at School of Artificial Intelligence and Automation, HUST. Her main research interests include mobile robots and wireless sensor networks.



Ali Marjovi Ali Marjovi received his PhD degree from Institute of Systems and Robotics, University of Coimbra in 2013. He has been a postdoctoral researcher at the Distributed Intelligent Systems and Algorithms Laboratory at the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. His research interests include mobile robotics, machine learning, embedded systems, IoT and data science.



Jian Huang graduated from Huazhong University of Science and Technology (HUST), Wuhan, China, in 1997 and received the Master of Engineering degree and the Ph.D. degree from HUST in 2000 and 2005, respectively. From 2006 to 2008, he was a Postdoctoral Researcher at the Department of Micro-Nano System Engineering and the Department of Mechano-Informatics and Systems, Nagoya University, Japan. In 2015, he was a research fellow in Nagoya University supported by JSPS invitation fellowship. He is currently a full professor with the School of Artificial Intelligence and Automation, HUST. He is also a guest professor in Nagoya University of Japan and University Paris-Est Créteil (UPEC) of France. His main research interests include rehabilitation robotics, robotic assembly, networked control systems and bioinformatics.



Alcherio Martinoli has a M.Sc. in Electrical Engineering from the Swiss Federal Institute of Technology in Zurich (ETHZ), and a Ph.D. in Computer Science from the Swiss Federal Institute of Technology in Lausanne (EPFL). He is currently an Associate Professor at the School of Architecture, Civil, and Environmental Engineering and the head of the Distributed Intelligent Systems and Algorithms Laboratory. Before joining EPFL he carried out research activities at the Institute of Biomedical Engineering of the ETHZ, at the Institute of Industrial Automation of the Spanish Research Council in Madrid, Spain, and at the California Institute of Technology, Pasadena, U.S.A. His research interests focus on methods to design, control, model, and optimize distributed cyber-physical systems, including multi-robot systems, sensor and actuator networks, and intelligent vehicles.