Author's accepted manuscript: International Conference on Applications of Intelligent Systems (APPIS), Las Palmas de Gran Canaria, Spain (2020).

https://doi.org/10.1145/3378184.3378220

An Evaluation of Gas Source Localization Algorithms for Mobile Robots

Pepe Ojeda, Javier Monroy, Javier Gonzalez-Jimenez
Machine Perception and Intelligent Robotics group (MAPIR). Dept. of System Engineering and Automation. Biomedical Research Institute of Málaga (IBIMA). University of Málaga. Spain
{ojedamorala,jgmonroy,javiergonzalez}@uma.es

ABSTRACT

This paper presents a comparative evaluation of different approaches to the problem of gas source localization (GSL) with a mobile robot. Concretely, four state-of-the-art algorithms are implemented and evaluated: Surge-Cast, Spiral, Surge-Spiral and a Probabilistic (Particle Filter-based) method. The experiments have been carried out with the gas dispersion simulator GADEN and the robotic tools offered by ROS (Robotic Operating System) under diverse, realistic environments that feature obstacles and turbulent airflows. Our study reveals, among other results, that Surge-Spiral out-performs Surge-Cast under turbulent wind conditions, and the particle filter approach becomes advantageous only when the assumption of a homogeneous wind holds. We believe that these findings can help the research community to decide on the most appropriate GSL method for a given application. Besides, the code that implements these algorithms is made publicly available.

CCS CONCEPTS

• **Computer systems organization** → Robotics.

KEYWORDS

Intelligent Robotics, Machine Olfaction, Gas Detection, Gas Source Localization

ACM Reference Format:

Pepe Ojeda, Javier Monroy, Javier Gonzalez-Jimenez. 2020. An Evaluation of Gas Source Localization Algorithms for Mobile Robots. In *Proceedings of the 3rd International Conference on Applications of Intelligent Systems (APPIS 2020), January 7–9, 2020, Las Palmas de Gran Canaria, Spain.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3378184.3378220

1 INTRODUCTION

Robotic olfaction (RO) refers to mobile robots with the capability to perceive gases in the environment. RO is a research field with many important technological and scientific challenges yet to be solved [21]. Recent advances in the design and manufacturing of portable gas sensing devices, usually referred to as electronic noses (*e-noses*) [1, 7], as well as in applying and adapting signal processing techniques to this field [3, 6] have already positioned RO as a promising solution for many practical applications.

In general, the problems addressed by RO can be sorted into three main topics: identification or classification of a measured volatile [25], estimation of the gas dispersal through the creation of distribution maps [5, 27] and the localization of the gas emission source (from now on, Gas Source Localization or GSL) [10, 15, 24].

In this work we focus on GSL, a problem with many practical applications, including the localization of gas pipe leaks in industrial facilities [19], the identification and location of illegal substances for contraband interception [3, 33], or the pinpoint of pollution-related emissions that may affect high density population areas [28] among others.

Traditionally, this problem has been tackled with the help of animals, deploying networks of fixed chemical sensors (e.g. pollution monitoring stations [31]), or by means of portable handheld gas sensors carried by expert operators [34]. Even though these solutions might be acceptable for some scenarios, often the scale of the problem (i.e. large search areas where a fixed network of sensors is impractical) or the dangerous environmental conditions the operators or animal assistants would be exposed to (i.e. toxic gases, high temperatures) make them unfeasible. For these reasons, it is interesting to have mobile robots equipped with chemical sensors to take the role of the operator/animal in the search process [9, 33].

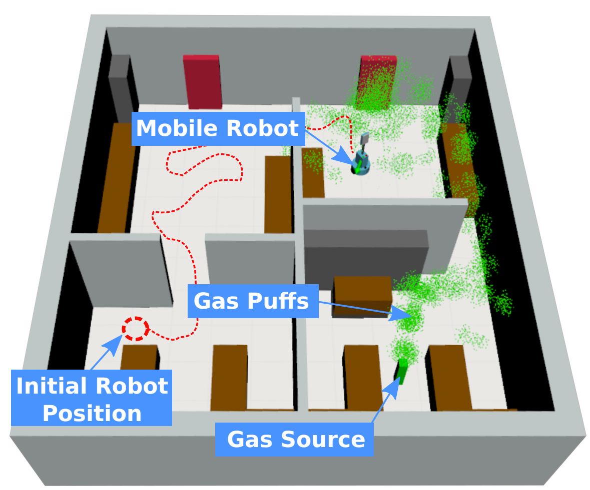


Figure 1: Illustration of a GSL task. A mobile robot, able to sample the gas concentration and the wind vector, explores the environment looking for the location of the gas source in a realistic office-like environment.

APPIS 2020, January 7–9, 2020, Las Palmas de Gran Canaria, Spain

ACM ISBN 978-1-4503-7630-3/20/01…$15.00
https://doi.org/10.1145/3378184.3378220

Author's accepted manuscript: International Conference on Applications of Intelligent Systems (APPIS), Las Palmas de Gran Canaria, Spain (2020).

https://doi.org/10.1145/3378184.3378220

Fig. 1 illustrates this task featuring a mobile robot that continuously samples the environment trying to locate the gas emission point. As can be seen, the presence of obstacles in the environment generates turbulence that breaks down the gas dispersion into multiple gas puffs.

Multiple approaches have been proposed along the last decades to tackle the problem of GSL with a mobile robot. Because of the complexity of gas dispersion phenomena, many of these proposals make assumptions about the nature of the source, the environmental conditions (characteristics of the airflow, presence of obstacles) or the sensors the robot is equipped with. A common drawback in these works is for the performed experimentation to be limited to simplified environmental conditions, as well as a lack of objective comparisons with other methods that try to present a solution to the same problem. This issue makes it difficult to evaluate the proposed methods, or to select the most convenient method for each specific application, as well as hindering the development of new alternative solutions.

This work contributes an experimental evaluation of the performance of some state-of-the-art GSL algorithms for mobile robotics under different environmental conditions. The goal is to provide an objective comparison that helps the scientific community to select the most appropriate GSL methods for specific applications. To perform this evaluation we rely on simulation tools, which will allow us to compare the performance of the different methods under identical conditions and to set up multiple configurations that would be unfeasible with real experiments.

All the algorithms described in this work have been implemented using tools from the Robotic Operating System (ROS) [26], which makes the implementations compatible with most common robotic hardware and simulation tools. Moreover, the code for the implementations has been made publicly available [1].

2 STATE OF THE ART

This section review some of the most popular approaches to gas source localization. According to whether the assumption of having a gas dispersion model is made or not, we classify them into to main categories: model-based approaches and reactive approaches. For a more detailed taxonomy, please see [13].

2.1 Model-based Methods

When the environment presents a laminar airflow, gas is distributed in the form of a plume [20], but turbulences break the plume into "patches" that move independently. This means gas does not necessarily disperse in straightforward patterns, so it is possible for a gas patch measured far from the source to have followed an unpredictable path, particularly in complex environments that feature obstacles [32, 35]. This makes it difficult to estimate where the gas measured at a given location might have been released from.

Due to these difficulties, algorithms based on statistical inference to estimate the location of the source usually rely on previous assumptions that simplify the environmental conditions (such as laminar/homogeneous/constant airflow) [10, 14]. By assuming these conditions are met, it becomes possible to develop analytical models for the dispersion of the gas, permitting probabilistic searches. However, even though these assumptions might perform well for certain narrow applications, they are not generally realistic.

In most cases, probabilistic methods use these analytical models to create a Probability Density Function (PDF) that gives the likeliness of a given point of the environment being the actual location of the source, and then rely on some statistical inference technique (Bayesian inference [30], a particle filter [14]) to modify this probability function after each measurement.

2.2 Reactive Methods

Because of the difficulty to analytically describe the dispersion of gases, many proposed algorithms do not try to estimate the location of the source, but rather to devise a movement strategy that allows the robot to reactively navigate towards it. The behavior of certain types of insects has been an important inspiration for these reactive strategies [4, 16, 18]. These bioinspired algorithms can be divided into chemotactic and anemotactic strategies.

Chemotactic strategies use the measured gas concentration to guide the movements of the robot. They are usually designed to be utilized in environments where there is no strong airflow and gas is mostly dispersed through diffusion, which creates a concentration gradient. Some of the most utilized chemotactic strategies include [18] the movement pattern of the **E.coli bacteria** [29], which is a biased random walk; **Spiral** [4], a strategy that uses a growing spiral movement to get closer to the source, and restarts the spiral when the gas measurements indicate it is closer to the source than it was before (see section 3.1); and **Gradient Climbing** [29], which has several variants, but revolves around the use of two different gas sensors to measure the gas concentration in several points at each time instant to determine the direction of the concentration gradient, among others.

Anemotactic strategies use wind information to guide the movements of the robot, and therefore are appropriate for environments with a strong, measurable airflow, where advection is a main component of the gas dispersion.

The most notable subcategory of anemotactic algorithms is plume-tracking [12], which assumes the existence of a downwind gas plume and tries to find the source by moving through it. Some examples of plume-tracking algorithms include the **Silkworm Moth** algorithm [15] , which combines short straight movements when detecting odour with upwind zig-zagging and circular movements when not detecting it; the **Dung Beetle** algorithm [12], which similarly uses upwind zig-zag movements, but uses them to track the plume rather than to recover it; and surge-based algorithms like **Surge-Cast** [17] (see section 3.2) and **Surge-Spiral** [8] (see section 3.3).

Although many such bioinspired algorithms have been demonstrated to perform well in simplified test environments, it is currently unclear whether they are appropriate for the more complex environments required for real applications [2].

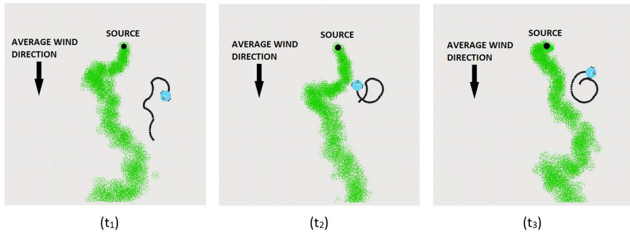
3 IMPLEMENTED ALGORITHMS

Among the multiple approaches dealing with the problem of GSL, in this work we have selected and implemented four of them which will be used for the comparative study (Figure 2). In this section we

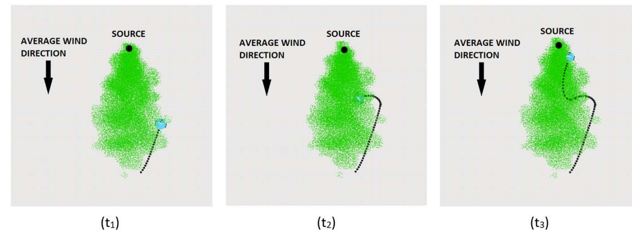
[1] https://github.com/MAPIRlab/Gas-Source-Localization

Author's accepted manuscript: International Conference on Applications of Intelligent Systems (APPIS), Las Palmas de Gran Canaria, Spain (2020).

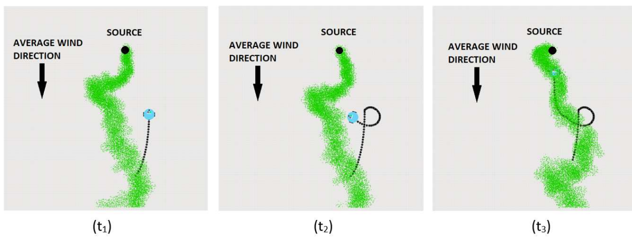
https://doi.org/10.1145/3378184.3378220



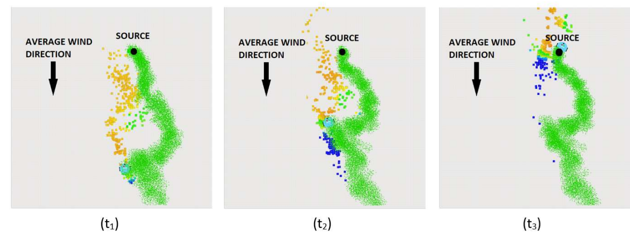
(a) Execution of the Spiral algorithm. The robot follows a spiralling pattern until it infers from the gas measurement that it is closer to the source than it was when it began moving, and then it restarts the spiral. Each of the subfigures shows a different time instant.



(b) Execution of the Surge-Cast algorithm. The robot moves directly upwind when it measures gas, and crosswind when it does not. Each of the subfigures shows a different time instant.



(c) Execution of the Surge Spiral algorithm. The robot moves upwind and maintains the same direction while it continues to measure gas. When it loses the gas, it performs a spiral to try and recover it. Each of the subfigures shows a different time instant.



(d) Example of the execution of the particle filter algorithm. The dots represent particles, and their colour represents their weight, red being the highest and blue the lowest. Each of the subfigures shows a different time instant.

Figure 2: Snapshots of the execution of the four algorithms considered in the comparative study.

provide an overview of each selected method and describe, when necessary, the implementation details.

3.1 Spiral

The Spiral method, presented by Ferri et al. [4] in 2009, is a chemotactic and reactive GSL algorithm. This means that only information from the gas sensors is exploited to resolve the robot movement towards the gas source. It defines a simple strategy by which the robot always follows a spiraling movement pattern, and the only decision to be made is whether to continue the current spiral or to start a new one. In order to decide the next step, a heuristic parameter called Proximity Index is calculated from the gas concentration measurements (see [4]).

The reasoning behind the algorithm is that, while inside of a gas plume, the restarting spirals create a general movement in the direction of increasing Proximity Index values, which moves the robot towards the source; and when the plume is lost or broken into patches, the spiraling movement covers the area around the robot allowing it to find gas again.

3.2 Surge-Cast Plume Tracking

Surge-Cast [17] is a plume tracking approach based on the idea that, if gas is being distributed in the form of a plume, once the plume has been found the robot has only to follow it in order to find the source.

This particular instance of plume tracking utilizes a state machine with two core states, defined as follows:

- Surge: Used while the robot is within the gas plume. While in this state, the robot moves in a straight line directly upwind.
- Cast: Used when the robot loses the plume. The robot performs a crosswind swipe to try and find the gas plume, stopping as soon as gas is found.

The implementation used for this study uses some auxiliary states that have been defined in order to make the search process more robust (see [22]). In this implementation, the surge movement stops after a given distance to resample the wind direction, or as soon as the plume is lost.

3.3 Surge-Spiral Plume Tracking

The Surge-Spiral method, presented by Hayes et al. [8] is another approach of a plume tracking strategy that combines both Surge-Cast and Spiral. In this particular case, the Spiral method is not used to move the robot towards the source, but merely to regain the plume once lost.

The presented implementation follows the specifications in [8]. Differing from the traditional implementation of Surge-Cast, the surge phase is not interrupted as soon as the robot stops measuring gas and, during a surge, any successive hits will reset the surge distance without re-sampling the wind direction. This makes for a less strict tracking of the plume, meaning the robot is allowed to move outside of the plume more easily than in the case of the Surge-Cast algorithm. Therefore, the spiraling movement is chosen to substitute the crosswind cast, since it is a more robust, albeit slower, method to regain the plume compared to casting [16].

Author's accepted manuscript: International Conference on Applications of Intelligent Systems (APPIS), Las Palmas de Gran Canaria, Spain (2020).

https://doi.org/10.1145/3378184.3378220

3.4 Particle Filter Based GSL

This algorithm, presented by Li et al. [14] in 2011, is an example of an statistical inference based method. It is based on the use of a particle filter to estimate the location of the gas source without the need to physically reach it, that is, the source can be declared as found from far away.

In order to sample and weigh the particles, a probability density function for the source location that depends on the gas and wind measurements is defined. As explained previously, defining such a function for a complex environment with obstacles and turbulent wind flows is still an open question, and for that reason it is necessary to consider some simplifications about the environmental conditions. Concretely, this approach assumes that the wind can change over time, but is homogeneous in all the environment. This assumption is admissible for outdoors environments without obstacles, where air currents are large and can encompass the entire work space of the search, but not in more complex environments, particularly indoors.Yet, we include this method in our comparative study to analyze to what extend the non-compliance of the environmental assumptions affect the result.

As in the original version of this method, the particle filter algorithm does not influence the movement of the robot. Instead, a reactive algorithm (Surge-Spiral, see section 3.3) is utilized to guide the movements while the particle filter is executed.

4 EXPERIMENTS

All the experiments included in this work have been carried out using GADEN [23] and Stage[2], since the use of simulation tools not only allows to easily set up experiments in diverse environments, but also facilitates reproducing the experiments by granting control over the many variables that influence gas dispersion. All the configuration files and generated environments are available as part of the GADEN project[3].

It should be noted that, for the purposes of these experiments, the problem of source declaration (discerning when the source has been found) has not been taken into consideration. Therefore, the search will be considered a success whenever the robot, or, for the case of the Particle Filter algorithm, the average of the estimations, gets within a set distance (in this case, 0.5 meters) of the source. The search will be considered a failure, on the other hand, if the source has not been found after a given time (set, for these experiments, to 600 seconds).

The problem of finding the plume in the first place has equally not been considered, since none of these methods define any particular exploration behaviors. Therefore, the robot will in all cases be initially positioned where it can measure gas.

Finally, regarding the perception system, the simulated robot is equipped with a gas sensor to measure the gas concentration and a 2D anemometer for sensing the wind speed and direction (both affected by Gaussian noise), and a laser scanner for self-navigation.

[2]http://wiki.ros.org/stage
[3]https://github.com/MAPIRlab/gaden

4.1 Exp. 1: Weak Airflow in an Open Environment

This experiment is meant to evaluate the performance of each of the methods in an environment where diffusion is a main component of the gas dispersion. In an environment without obstacles (10m x 10m), gas is released from a given point and allowed to distribute through the effect of a low speed, unstable airflow.

It is trivial that, in the absence of a measurable airflow, all methods other than Spiral will fail. Therefore, even though the wind speed is set to values low enough that sensor noise could pose a problem for anemotactic strategies, it has been kept high enough for them to function.

Results are shown in Table 1.

4.2 Exp. 2: Homogeneous Time-Dependent Airflow in Empty Environment

This experiment replicates the conditions described in [14] for the particle filter algorithm. The same empty 10m x 10m environment from the previous experiment is used, but with a strong airflow that is homogeneous in the entire workspace and changes over time. This is designed to be an admissible simplification of the way the wind affects the gas source localization process in a completely open outdoors environment.

As can be observed in Fig. 3, the characteristics of this airflow cause the gas to disperse in the form of a narrow, bent plume that gradually changes shape and position over time.

Results are shown in Table 2.

4.3 Exp. 3: Steady Airflow With a Central Obstacle

This experiment introduces the presence of an obstacle in the environment. As can be observed in Fig. 3, the disruption of the airflow caused by the obstacle makes for a more complex problem where the plume is broken and moves close to the walls.

Results are shown in Table 3.

4.4 Exp. 4: Steady Airflow in Maze-Like Environment

This experiment is set in a more complex environment that resembles a maze (Figure 3), with several interior walls forming a narrow zig-zagging path from the gas source to the initial location of the robot. The shape of the environments makes it so that the gas is not able to form a complex dispersion pattern, but also poses a difficulty for the robot's navigation, since none of the chosen algorithms use map information to plan the navigation goals.

Results are shown in Table 4.

5 CONCLUSIONS AND FUTURE WORK

This work has presented a comparative experimental study under simulation of some of the most popular Gas Source Localization methods. Using the gas dispersion simulator GADEN, and the navigation and sensing tools offered by ROS, four algorithms namely Surge Cast [11], Spiral [4], Surge Spiral [8] and the Particle Filter based GSL presented by Li et al. [14], have been implemented and then evaluated in different environmental conditions, ranging from

Author's accepted manuscript: International Conference on Applications of Intelligent Systems (APPIS), Las Palmas de Gran Canaria, Spain (2020).

https://doi.org/10.1145/3378184.3378220

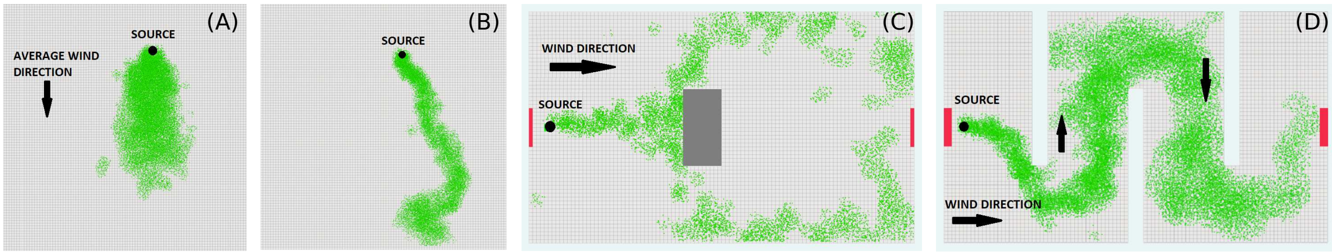


Figure 3: Illustration of the simulated environments considered along the multiple experiments: (A) weak airflow in an open environment, (B) homogeneous, time-varying airflow in an open environment, (C) steady airflow with a central obstacle, and (D) steady airflow in a maze-like environment.

Table 1: Results under a weak airflow.

Algorithm	Success rate	Average time ± stdev (s)	Average distance ± stdev (m)
Surge-Cast	5/5	82.60 ± 11.59	9.42 ± 2.60
Spiral	5/5	264.8 ± 151.6	20.28 ± 10.31
Surge-Spiral	5/5	128.52 ± 85.23	23.78 ± 13.76
Particle Filter	5/5	125.47 ± 44.81	23.09 ± 7.63

Table 2: Results under homogeneous time-dependent airflow.

Algorithm	Success rate	Average time ± stdev (s)	Average distance ± stdev (m)
Surge-Cast	5/5	114.40 ± 43.17	18.19 ± 7.75
Spiral	3/5	203.90 ± 34.65	14.91 ± 3.64
Surge-Spiral	5/5	50.50 ± 7.21	8.87 ± 1.33
Particle Filter	5/5	75.42 ± 49.95	13.75 ± 8.30

Table 3: Results in an environment that features a central obstacle.

Algorithm	Success rate	Average time ± stdev (s)	Average distance ± stdev (m)
Surge-Cast	5/5	77.76 ± 17.65	13.93 ± 3.30
Spiral	0/5	-	-
Surge-Spiral	5/5	83.88 ± 21.15	18.73 ± 4.74
Particle Filter	5/5	112.52 ± 5.23	17.51 ± 1.27

open spaces with homogeneous airflows to more complex environments featuring obstacles and turbulent wind. From the results of these simulated experiments, it can be concluded that:

- **Surge-Cast Plume Tracking** is a versatile algorithm that shows good results in all the experiments that have been carried out. This algorithm works best when a clear, stable plume exists, but has shown to be able to handle non ideal situations as well. Results show that in the absence of an ideal plume it is particularly relevant that the parameters of the algorithm (most notably, the duration of the measurement phase) are adjusted for the specific conditions of the environment.
- **Spiral** is the only one of the implemented algorithms that can be used in environments without a measurable airflow, since it does not require any wind information. However, it must be concluded from the results obtained that in case the environment does feature a measurable airflow, even if it is weak and relatively unstable, anemotactic strategies outperform Spiral. Also, due to its restrictive movement pattern, Spiral has shown to not be able to navigate complex environments.
- **Surge-Spiral Plume Tracking** is similar to Surge Cast, and has managed to reliably find the source in every environment. Because of its less strict policy with respect to staying inside of the gas plume, it is able to out-perform Surge Cast in environments where the plume is broken into patches or displaced by a strong wind. On the other hand, it is on average slower in environments with a stable plume because leaving the plume more easily forces it to spend more time trying to regain it.
- **Li *et al.*'s Particle Filter algorithm** shows good results when the specific conditions for which it was designed (that is, a near homogeneous wind) are met, but does not have a good performance in more complex environments, only being able to give a correct estimation of the source position after the robot physically reaches it through the Surge Spiral movement strategy. It must be concluded, then, that it is

Table 4: Results in a maze-like environment.

Algorithm	Success rate	Average time ± stdev (s)	Average distance ± stdev (m)
Surge-Cast	5/5	182.42 ± 55.64	21.45 ± 1.74
Spiral	0/5	-	-
Surge-Spiral	5/5	194.98 ± 23.52	26.59 ± 3.52
Particle Filter	5/5	229.30 ± 27.65	28.55 ± 2.39

only appropriate for use in outdoors environments where the assumption of a homogeneous wind is admissible.
It should be pointed out that the particle filter algorithm is the only of these four that offers a way to perform source declaration, and it remains to be tested whether that specific application might perform well in complex environments.

From the conclusions and the ideas discussed in this work, multiple proposals for future investigation can be drawn, including the corroboration of the results by performing real-world experiments; increasing the number and variety of GSL algorithms, creating an open-source repository that would allow for more extensive testing; or extending the scope of the experiments to include the plume finding and source declaration sub-problems.

REFERENCES

[1] J W Gardner Bartlett and P N. 2000. Electronic Noses. Principles and Applications. *Measurement Science and Technology* 11, 7 (jul 2000), 1087–1087. https://doi.org/10.1088/0957-0233/11/7/702
[2] Victor Hernandez Bennetts, Achim J. Lilienthal, Patrick P. Neumann, and Marco Trincavelli. 2012. Mobile robots for localizing gas emission sources on landfill sites: Is bio-inspiration the way to go? *Frontiers in Neuroengineering* 4, JANUARY (2012). https://doi.org/10.3389/fneng.2011.00020
[3] Javier Burgués and Santiago Marco Colás. 2019. *Signal Processing and Machine Learning for Gas Sensors: Gas Source Localization with a Nano-Drone.* Ph.D. Dissertation. Universitat de Barcelona.
[4] Gabriele Ferri, Emanuele Caselli, Virgilio Mattoli, Alessio Mondini, Barbara Mazzolai, and Paolo Dario. 2009. SPIRAL: A novel biologically-inspired algorithm for gas/odor source localization in an indoor environment with no strong airflow. *Robotics and Autonomous Systems* 57, 4 (apr 2009), 393–402. https://doi.org/10.1016/J.ROBOT.2008.07.004
[5] Javier G. Monroy, Jose Luis Blanco, and Javier Gonzalez-Jimenez. 2016. Time-variant gas distribution mapping with obstacle information. *Autonomous Robots* 40, 1 (jan 2016), 1–16. https://doi.org/10.1007/s10514-015-9437-0
[6] Andres Gongora, Javier Monroy, and Javier Gonzalez-Jimenez. 2018. An Electronic Architecture for Multipurpose Artificial Noses. *Journal of Sensors* 2018 (feb 2018), 1–9. https://doi.org/10.1155/2018/5427693
[7] Javier Gonzalez-Jimenez, Javier G. Monroy, and Jose Luis Blanco. 2011. The multi-chamber electronic nose-an improved olfaction sensor for mobile robotics. *Sensors* 11, 6 (jun 2011), 6145–6164. https://doi.org/10.3390/s110606145
[8] Adam T. Hayes, Alcherio Martinoli, and Rodney M. Goodman. 2002. Distributed odor source localization. *IEEE Sensors Journal* 2, 3 (2002), 260–271. https://doi.org/10.1109/JSEN.2002.800682
[9] Curtis Humphrey and JULIE A. ADAMS. 2009. Robotic tasks for CBRNE incident response. *Advanced Robotics* (2009).
[10] Michael Hutchinson, Cunjia Liu, and Wen Hua Chen. 2018. Information-Based Search for an Atmospheric Release Using a Mobile Robot: Algorithm and Experiments. *IEEE Transactions on Control Systems Technology* (2018), 1–15. https://doi.org/10.1109/TCST.2018.2860548
[11] Hiroshi Ishida, Takamichi Nakamoto, Toyosaka Moriizumi, Timo Kikas, and Jiri Janata. 2001. Plume-Tracking Robots: A New Application of Chemical Sensors. *The Biological Bulletin* 200, 2 (2001), 222–226. https://doi.org/10.2307/1543320
[12] H. Ishida, K. Suetsugu, T. Nakamoto, and T. Moriizumi. 1994. Study of autonomous mobile sensing system for localization of odor source using gas sensors and anemometric sensors. *Sensors and Actuators: A. Physical* 45, 2 (nov 1994), 153–157. https://doi.org/10.1016/0924-4247(94)00829-9
[13] Gideon Kowadlo and R. Andrew Russell. 2008. Robot odor localization: A taxonomy and survey. *International Journal of Robotics Research* 27, 8 (aug 2008), 869–894. https://doi.org/10.1177/0278364908095118
[14] Ji Gong Li, Qing Hao Meng, Yang Wang, and Ming Zeng. 2011. Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm. *Autonomous Robots* 30, 3 (2011), 281–292. https://doi.org/10.1007/s10514-011-9219-2
[15] Alexander Liberzon, Kyra Harrington, Nimrod Daniel, Roi Gurka, Ally Harari, and Gregory Zilman. 2018. Moth-inspired navigation algorithm in a turbulent odor plume from a pulsating source. *PLoS ONE* 13, 6 (jun 2018), e0198422. https://doi.org/10.1371/journal.pone.0198422
[16] Thomas Lochmatter and Alcherio Martinoli. 2009. Theoretical analysis of three bio-inspired plume tracking algorithms. *Proceedings - IEEE International Conference on Robotics and Automation* (2009), 2661–2668. https://doi.org/10.1109/ROBOT.2009.5152686
[17] Thomas Lochmatter and Alcherio Martinoli. 2009. Tracking Odor Plumes in a Laminar Wind Field with Bio-inspired Algorithms. In *Springer Tracts in Advanced Robotics*. Vol. 54. Springer, Berlin, Heidelberg, 473–482. https://doi.org/10.1007/978-3-642-00196-3_54
[18] João Macedo, Lino Marques, and Ernesto Costa. 2019. A Comparative Study of Bio-Inspired Odour Source Localisation Strategies from the State-Action Perspective. *Sensors* 19, 10 (2019), 2231. https://doi.org/10.3390/s19102231
[19] Dani Martínez, Tomàs Pallejà, Javier Moreno, Marcel Tresanchez, Mercè Teixidó, Davinia Font, Antonio Pardo, Santiago Marco, and Jordi Palacín. 2014. A mobile robot agent for gas leak source detection. In *Advances in Intelligent Systems and Computing*. Vol. 293. Springer, Cham, 19–25. https://doi.org/10.1007/978-3-319-07476-4_3
[20] C. W. Miller and L. M. Hively. 1987. A review of validation studies for the Gaussian plume atmospheric dispersion model. *Nuclear Safety* 28, 4 (1987), 522–531.
[21] Javier Monroy and Javier Gonzalez-Jimenez. 2018. Towards odor-sensitive mobile robots. In *Electronic Nose Technologies and Advances in Machine Olfaction.* 244–263. https://doi.org/10.4018/978-1-5225-3862-2.ch012
[22] Javier Monroy, Javier Gonzalez-Jimenez, Jose-Raul Ruiz-Sarmiento, and Javier Gonzalez-Jimenez. 2019. An Evaluation of Plume Tracking as a Strategy for Gas Source Localization in Turbulent Wind Flows. *2019 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN)* (may 2019), 9–11. https://doi.org/10.1109/ISOEN.2019.8823147
[23] Javier Monroy, Victor Hernandez-Bennetts, Han Fan, Achim Lilienthal, and Javier Gonzalez-Jimenez. 2017. GADEN: A 3D gas dispersion simulator for mobile robot olfaction in realistic environments. *Sensors (Switzerland)* 17, 7 (2017), 1–16. https://doi.org/10.3390/s17071479
[24] Javier Monroy, Jose Raul Ruiz-Sarmiento, Francisco Angel Moreno, Francisco Melendez-Fernandez, Cipriano Galindo, and Javier Gonzalez-Jimenez. 2018. A semantic-based gas source localization with a mobile robot combining vision and chemical sensing. *Sensors (Switzerland)* 18, 12 (2018), 1–20. https://doi.org/10.3390/s18124174
[25] Javier G. Monroy, Esteban J. Palomo, Ezequiel López-Rubio, and Javier Gonzalez-Jimenez. 2016. Continuous chemical classification in uncontrolled environments with sliding windows. *Chemometrics and Intelligent Laboratory Systems* 158, 158 (nov 2016), 117–129. https://doi.org/10.1016/j.chemolab.2016.08.011
[26] M. Quigley, Gerkey B., Conley K., Faust J., Foote T., Leibs J., Berger E., Wheeler R., and Andrew Ng. 2009. ROS: an open-source Robot Operating System. *ICRA workshop on open source software* 3.2 (2009), 5. https://doi.org/10.1109/IECON.2015.7392843 arXiv:1106.4561
[27] Matteo Reggente and Achim J. Lilienthal. 2009. Using local wind information for gas distribution mapping in outdoor environments with a mobile robot. In *Proceedings of IEEE Sensors.* IEEE, 1715–1720. https://doi.org/10.1109/ICSENS.2009.5398498
[28] Matteo Reggente, Alessio Mondini, Gabriele Ferri, Barbara Mazzolai, Alessandro Manzi, Matteo Gabelletti, Paolo Dario, and Achim Lilienthal. 2010. The DustBot System: Using Mobile Robots to Monitor Pollution in Pedestrian Area. *Chemical Engineering Transactions* 23 (2010), 273–278. https://doi.org/10.3303/CET1023046
[29] R. Andrew Russell, Alireza Bab-Hadiashar, Rod L. Shepherd, and Gordon G. Wallace. 2003. A comparison of reactive robot chemotaxis algorithms. *Robotics and Autonomous Systems* 45, 2 (nov 2003), 83–97. https://doi.org/10.1016/S0921-8890(03)00120-9
[30] C. Sanchez-Garrido, J. Monroy, and J. Gonzalez-Jimenez. 2018. Probabilistic estimation of the gas source location in indoor environments by combining gas and wind observations. *Frontiers in Artificial Intelligence and Applications* 310 (2018), 110–121. https://doi.org/10.3233/978-1-61499-929-4-110
[31] Philipp Schneider, Nuria Castell, Matthias Vogt, Franck R. Dauge, William A. Lahoz, and Alena Bartonova. 2017. Mapping urban air quality in near real-time using observations from low-cost sensors and model information. *Environment International* 106, May (2017), 234–247. https://doi.org/10.1016/j.envint.2017.05.005
[32] Spyros Sklavounos and Fotis Rigas. 2004. Validation of turbulence models in heavy gas dispersion over obstacles. *Journal of Hazardous Materials* 108, 1-2 (apr 2004), 9–20. https://doi.org/10.1016/j.jhazmat.2004.01.005
[33] James P. Trevelyan, Sung-Chul Kang, and William R. Hamel. 2008. Robotics in Hazardous Applications. In *Springer Handbook of Robotics.* Springer Berlin Heidelberg, Berlin, Heidelberg, 1101–1126. https://doi.org/10.1007/978-3-540-30301-5_49
[34] Denise Michèle Wilson, Scan Hoyt, Jiri Janata, Karl Booksh, and Louis Obando. 2001. Chemical sensors for portable, handheld field instruments. *IEEE Sensors Journal* 1, 4 (2001), 256–274. https://doi.org/10.1109/7361.983465
[35] Bin Zhao, Caiqing Yang, Xudong Yang, and Shuangke Liu. 2008. Particle dispersion and deposition in ventilated rooms: Testing and evaluation of different Eulerian and Lagrangian models. *Building and Environment* 43, 4 (apr 2008), 388–397. https://doi.org/10.1016/j.buildenv.2007.01.005