

A Comparison of Casting and Spiraling Algorithms for Odor Source Localization in Laminar Flow

Thomas Lochmatter, Xavier Raemy, Loïc Matthey, Saurabh Indra and Alcherio Martinoli

Abstract—We compare two well-known algorithms for locating odor sources in environments with a main wind flow. Their plume tracking performance is tested through systematic experiments with real robots in a wind tunnel under laminar flow condition. We present the system setup and show the wind and odor profiles. The results are then compared in terms of time and distance to reach the source, as well as speed in upwind direction. We conclude that the spiral-surge algorithm yields significantly better results than the casting algorithm, and discuss possible rationales behind this performance difference.

I. INTRODUCTION

The olfactory sense is a very important sense for many animals. Smell is used for food scavenging and inspection, mating, kin recognition, hunting and predation avoidance [1]. Humans also use olfaction in their daily life, although their nose is not particularly sensitive. In the animal kingdom, dogs and rats are known to have very sensitive noses. Because of this ability, these animals are used in search and rescue operations, on airports or country borders to search for drugs or explosives, and for humanitarian demining [2] [3] [4] [5]. In all these applications, animals are trying to locate the source of a chemical substance (odor) in the air.

With the advances in mobile robotics and electronic odor sensor technology [6] [7], such tasks could in the future be accomplished by sniffing robots [8] [9]. Replacing animals by mobile robots could reduce the cost significantly, as breeding and training animals is time- and resource-intensive. In addition, animals get tired and sometimes need guidance, which can be dangerous when searching for explosives.

A robotic system to track down odor sources requires the following components:

- ▷ A mobile robot able to move in the target area.
- ▷ An appropriate electrochemical sensory module, able to detect the target substance at its expected concentration.
- ▷ Optionally a sensor indicating the direction and speed of the wind.
- ▷ An odor source localization algorithm.

We have designed and developed a complete system with off-the-shelf components and focus our research on the algorithmic aspect of the problem.

All authors are with the Swarm-Intelligent Systems Group, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland. (thomas.lochmatter@epfl.ch, alcherio.martinoli@epfl.ch)

This work was supported by the National Competence Center in Research on Mobile Information and Communication Systems NCCR-MICS, a center supported by the Swiss NSF under grant number 5005-67322.

Alcherio Martinoli is currently sponsored by a Swiss NSF grant (contract Nr. PP002-68647).

Odor source localization algorithms are generally split into three parts [10]:

- 1) *Exploration*, to acquire a scent.
- 2) *Plume tracking* towards the odor source.
- 3) *Source declaration*.

In this paper, we compare two *plume tracking* algorithms: spiral-surge [10] [11] [12] [13] and casting [14] [15] [16] [17] [18] [19] [20] in a controlled laminar wind flow scenario. Both algorithms and many variants have been studied in simulation and with real robots, but to our knowledge, no fair comparison using the same real robotic platform has been made to date.

Comparing already existing performance results of different contributions is virtually impossible, as the experiments were carried out in different setups and environments, or with different plume simulators. The wind profile has a major impact on odor tracking, as the dispersion of molecules in the air highly depends on it. Thus, a direct comparison between results of different studies is hardly possible.

In addition, the spiral-surge algorithm has never been tested in a precisely characterized wind field (in [10], for instance, wind was generated with five off-the-shelf fans), and many studies of the casting algorithm have only considered simulations or mathematical models and did not use real robots and sensors.

In this study, we present comparable results from real-robot experiments for both the casting algorithm and the spiral-surge algorithm. We ran all experiments under the same environmental conditions in a wind tunnel, with the same laminar and (almost) homogeneous wind profile, and used the same robot, the same type of plume and the same sensors for both algorithms. The wind tunnel allows us to reproduce the same conditions at will, and to keep them stable during all experiments, which would not be possible when running experiments under uncontrolled indoor or outdoor fluid dynamic conditions.

The remainder of the paper is organized as follows: we first introduce the two algorithms and discuss some general properties in Section II. In Section III and Section IV, we present our robotic platform and experimental setup. The results are discussed in Section V. Finally, we conclude our paper in Section VI.

II. ALGORITHMS

Casting and spiraling are both bio-inspired techniques [8] [12] that have been ported to real robots, based on a combination of moving upwind in the plume and locally searching to reacquire the plume when it is lost.

Both algorithms use only binary odor information, that is, they either *perceive the odor* or *do not perceive any odor*, but ignore different concentrations levels. Commonly, the measured concentration is thresholded to obtain this binary value, but more elaborate processing could be used as well.

Finally, both algorithms need a wind sensor to measure the direction the wind is coming from. As molecules are mainly transported by advection, this piece of information is very valuable, and - especially for the casting algorithm - as important as the odor sensor. The wind speed is ignored.

Note that we are only interested in the plume tracking behavior. Hence, we assume that the robot has found the plume already when the experiments start, and declare failure if it gets too far away from the plume. This allows us to rule out any effect of the arena geometry, which could greatly influence the results if random search techniques were used.

Similarly, we omit source declaration and call experiments successful if the robot has come in physical vicinity of the source.

A. Casting (Zig-Zagging)

We have implemented the same casting algorithm as Li et al. presented in [14]. As shown in Figure 1, a robot in the plume moves upwind with an angle β until it is out of the plume for a certain distance (or time), denoted d_{lost} . Once the plume is lost, it turns and moves crosswind until it hits an odor packet, and then moves upwind with angle β again.

The wind direction is measured each time the robot switches to plume reacquisition, and when it encounters the plume again.

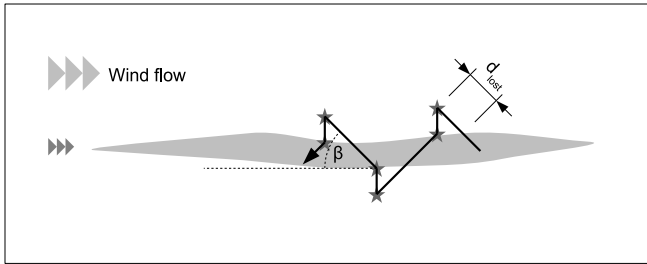


Fig. 1. Sketch of the casting algorithm. The stars indicate where the wind direction is measured.

B. Spiral-Surge

The spiral-surge algorithm is similar to Hayes' algorithm presented in [10], except that here it is only used for plume tracking, and not for plume search.

A robot in the plume moves straight upwind until it loses the plume for a distance d_{lost} . It then tries to reacquire the plume by moving along an Archimedean spiral with gap size d_{gap} until it reacquires the plume. Unlike in [10], we start our spiral in upwind direction, as drawn in Figure 2.

The wind direction is measured when the robot switches from spiraling to upwind surge.

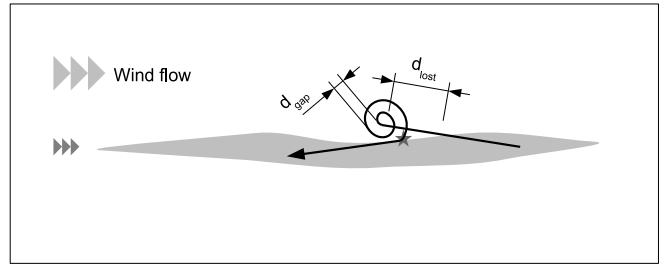


Fig. 2. Sketch of the spiral-surge algorithm. The star indicates where the wind direction is measured.

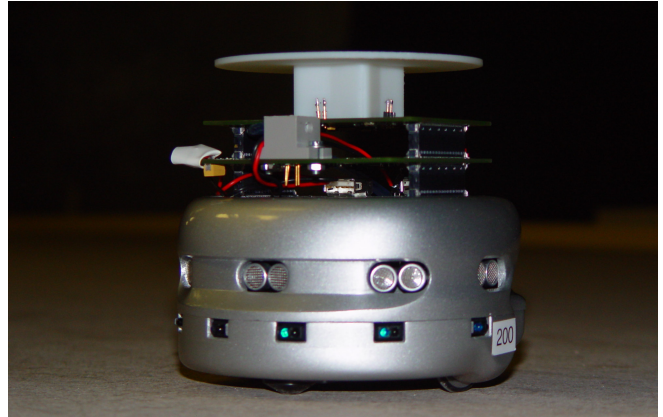


Fig. 3. Khepera III robot with an additional odor sensor, wind sensor, and voltage regulation board.

III. ROBOTIC PLATFORM

We are using a Khepera III robot, produced by K-Team SA, Switzerland, with development assistance from the Swarm-Intelligent Systems Group at EPFL. The robot was extended with an odor and a wind sensor board that we designed in-house.

The Khepera III robot (see Figure 3) is a differential-drive robot of 12 cm diameter and 7 cm height (without any additional modules). It hosts a KoreBot board running an embedded Linux and a WLAN 802.11 CompactFlash card. The robot is equipped with two DC motors with optical encoders enabling the use of odometry on a flat surface, 9 infrared sensors for obstacle avoidance, two infrared floor sensors, and 5 ultrasound sensors. Other boards can be stacked on top.

We developed two such sensor boards for detecting ethanol vapor and wind, plus an additional voltage regulation board. The odor sensor board is equipped with a MiCS-5521 volatile organic compound (VOC) sensor from MicroChemical Systems SA, Switzerland. The sensitive surface inside this sensor has a very fast response time (≈ 0.1 s) as compared to other VOC sensors on the market, which is a clear advantage for real-time robotic applications. The sensor is not very selective, though: it reacts not only to ethanol, but also to many other volatile organic compounds and hydrocarbons (at different sensitivities). The sensitivity to alcohol - although not provided in the technical datasheet - seems to be comparable to other state-of-the-art VOC sensors, and therefore

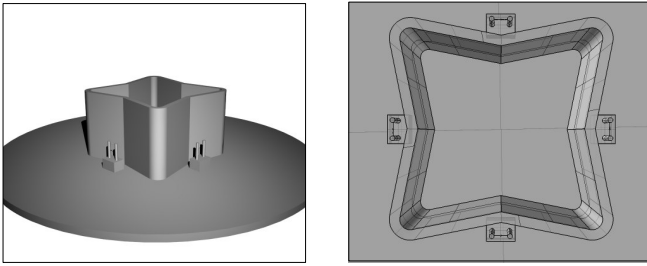


Fig. 4. Model of the wind sensor board (without top cover). Four tiny NTC thermistors are placed around a star-shape obstacle.

in the 1-10 ppm range.

To take advantage of the fast response time of the sensitive surface, air is continuously drawn into the sensor using a small gas pump (NMP 05 S, KNF Neuberger). Without the pump, it would take significantly longer for the plume to enter the sensor package and reach the surface. The setup is very similar to the olfaction system of humans or animals, except that air is not exhaled through the same pipes as it enters.

The wind sensor consists of 4 NTC thermistors (Honeywell 111-202CAK-H01) placed around a star-like shape, as shown in Figure 4, and connected to a microcontroller. The thermistors are heated to approximately 85 °C and cool down as a function of the wind flow, therefore changing their resistance. The obstacle in the center deflects the wind so that it is possible to infer the wind direction (in 2 dimensions) and speed from the measured resistance pattern. The wind direction accuracy is in the order of 5 to 10 degrees, except for a few critical angles where the error occasionally went up to 25 degrees. The design is heavily inspired by the wind sensor of Ishida et al. [23], but achieves a finer resolution.

Both boards are connected to the main processor with the I²C bus, and raw sampling rates as seen from Linux are in the order of 100 Hz. The overall perception-to-action loop of our program runs at roughly 5 Hz.

While the bare robot running Linux has an autonomy of a bit more than 2 h, the odor and wind modules reduce this autonomy to approximately 1 h 30 min of continuous use.

IV. EXPERIMENTAL SETUP

The experiments were carried out in a 16 m long and 4 m wide wind tunnel. The effective space (arena) used for the experiments was approximately 12 m by 3 m. The wind tunnel is equipped with a 3-axis traversing system, which allows to scan the whole tunnel with any sensor probe.

A. Wind

We generated laminar wind at roughly 1 m/s. The wind speed, shown in Figure 5, was almost homogeneous with differences of 0.2 m/s between the maximum and minimum speeds.

The wind speed was only measured in one direction. The odor profile, measured with the robot described in Section III and shown in Figure 6, reveals however that the wind was drifting slightly to the east of the wind tunnel, at a rate of

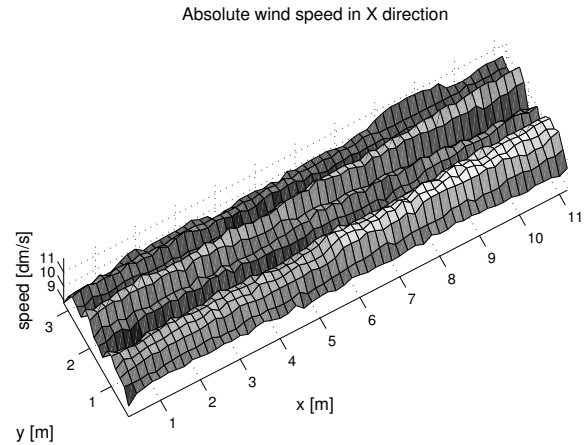


Fig. 5. Wind profile in x direction. Each point is an average of 25 measurements with an Extech 407119 hot-wire anemometer mounted on the traversing system of the wind tunnel. The grid has a resolution of 10 cm in both X and Y direction. The wind speed was measured at the height of the robot's wind sensor board.

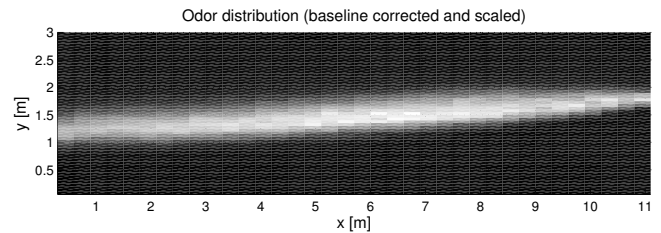


Fig. 6. Odor profile in the arena. Each measurement point is an average over about 20 seconds. The grid has a resolution of 30 cm in X direction, and 5 cm in y direction. The odor was measured at the height of the robot's odor sensor board using the traversing system of the wind tunnel.

about 5 cm/m. This is presumably due to the asymmetric outlet of the tunnel, but we believe that this drift did not significantly affect the results.

B. Odor

We use ethanol molecules (C₂H₅OH) as odor, for the following reasons:

- ▷ Molecules, in contrast to particles (smoke) or droplets (aerosols), perfectly follow the air flow, as their own mass, and therefore their own momentum, is negligible.
- ▷ Ethanol evaporates quickly (without heating) and does not leave any traces.
- ▷ Ethanol is cheap, readily available and harmless in the small quantities used here (especially in a continuous flow wind tunnel).
- ▷ Ethanol sensors are readily available as well.
- ▷ Ethanol is harmless for any electronic and mechanic equipment used in the wind tunnel.
- ▷ Ethanol can be pumped with standard gas pumps, such as those we are using for the odor source or on the robot.

The main disadvantage of using volatile compound molecules is their invisibility. Hence, it is not possible to get

an instantaneous picture of the plume, as would be possible with smoke, for example.

C. Odor Source

A functional drawing of our odor source is shown in Figure 7. A bottle containing about 0.1 l of industrial ethanol at 94% purity is connected with a tube to a pump (LIQUIPORT NF 100 KT.18 RC, KNF Neuberger). Note that even though this pump's primary intent is for liquids, it works fine with gases as well. In the first part of the tube, we added some wick, which is long enough to reach the bottom of the bottle. A second tube allows air to flow into the bottle.

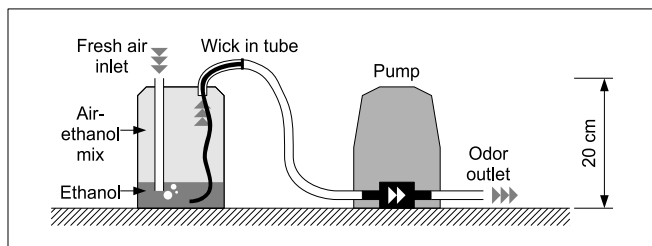


Fig. 7. Schematic drawing of the odor source. The arrows indicate the air flow direction.

As ethanol evaporates fairly quickly at room temperature, the upper part of the bottle will quickly fill up with an air-ethanol mix. This phenomenon is amplified by the wick, which always remains wet due to capillary forces. The pump then ejects this air-ethanol mix.

As long as the pump is running at a steady speed (1.2 l/min in our experiments), the ethanol concentration is constant, as the size of the air-ethanol interface and the temperature do not change significantly.

D. Setup

A schematic of our setup is shown in Figure 8. The distance between the starting zone and the target area was 11 m, and the odor source was placed about 20 cm behind the target area.

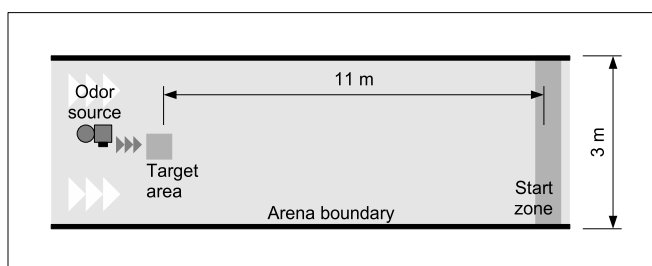


Fig. 8. Schematic drawing of our arena (not to scale).

The robot was launched in the starting zone, and randomly moved in this area until it smelled the ethanol. Once its sensors measured an odor concentration above a certain fixed threshold (chosen experimentally), the actual experiment started. All counters were set to zero and the robot tried to track the odor plume towards the source, with one of the algorithms presented in Section II.

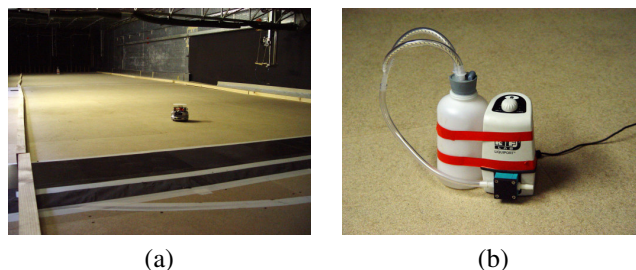


Fig. 9. (a) Picture of the arena. (b) Picture of the odor source.

If the robot completely lost the plume and hit the arena border, the run was abandoned and considered unsuccessful. If the robot reached the target area in front of the source (while smelling ethanol at the same time), the run was considered successful, and distance and time were logged.

The time was recorded with the internal clock of the robot's CPU, and the distance using the wheel encoder values. While the clock is certainly very precise, we estimate the error on the distance measurement to be in the order of 1%.

V. RESULTS AND DISCUSSION

We ran 10 experiments for each of the following configurations:

| | Algorithm | Parameter |
|---|--------------|-----------------------------|
| A | Casting | $\beta = 15$ [deg] |
| B | Casting | $\beta = 30$ [deg] |
| C | Casting | $\beta = 45$ [deg] |
| D | Spiral-surge | $d_{\text{gap}} = 0.21$ [m] |
| E | Spiral-surge | $d_{\text{gap}} = 0.36$ [m] |
| F | Spiral-surge | $d_{\text{gap}} = 0.57$ [m] |
| G | Spiral-surge | $d_{\text{gap}} = 0.85$ [m] |

The forward speed of the robot (on straight lines) was approximately 7.1 cm/s and the plume lost distance was set to $d_{\text{lost}} = 10$ cm for all experiments. The plume threshold could not be kept constant, as the sensor values were fluctuating slowly throughout the day. From time to time, therefore, it was reset to about 90% of the peak odor values measured in the starting zone. The threshold is thus independent of the configuration.

A. Performance Comparison and Observations

Table I lists the success ratio as well as other values of interest averaged over the successful runs. Note that the values in column A have to be taken with a grain of salt, as this is an average over two successful runs only. "Advancement" refers to the distance to the target, and "distance/advancement" therefore denotes how much distance a robot has to cover to come 1 m closer to the source.

Mean distances and times are furthermore plotted with their confidence intervals in Figure 10 and Figure 11.

The casting algorithm is quite successful for higher values of β , but fails completely for $\beta = 15$ deg. Bigger upwind angles, however, make the algorithm slower, as the plume has to be crossed many more times until the source is reached.

TABLE I
MEAN VALUES (EXCEPT FOR THE SUCCESS RATIO) OF ALL CONFIGURATIONS.

| Configuration | A | B | C | D | E | F | G |
|----------------------------|-----------|-----------|----------|-----------|-----------|-----------|-----------|
| Success ratio | 0.2 | 0.8 | 0.7 | 0.9 | 0.9 | 0.8 | 1 |
| Distance to target area | 13.7647 | 19.5307 | 23.6424 | 16.0107 | 14.5767 | 15.16 | 16.7082 |
| Time to target area | 218.0595 | 314.7298 | 378.9157 | 239.6247 | 220.6653 | 236.2997 | 272.8486 |
| Distance in plume [m] | 11.4554 | 13.91 | 16.6027 | 12.2803 | 11.5093 | 11.5567 | 11.3893 |
| Ratio in plume | 0.83451 % | 0.71377 % | 0.7032 % | 0.78716 % | 0.79962 % | 0.79024 % | 0.71386 % |
| Distance not in plume [m] | 2.3093 | 5.6207 | 7.0397 | 3.7304 | 3.0674 | 3.6033 | 5.3188 |
| Ratio not in plume | 0.16549 % | 0.28623 % | 0.2968 % | 0.21284 % | 0.20038 % | 0.20976 % | 0.28614 % |
| Distance/advancement [m/m] | 1.2513 | 1.7755 | 2.1493 | 1.4555 | 1.3252 | 1.3782 | 1.5189 |
| Advancement speed [m/s] | 0.050953 | 0.035269 | 0.029206 | 0.048145 | 0.050991 | 0.04974 | 0.043252 |
| Mean robot speed [m/s] | 0.063139 | 0.062065 | 0.062442 | 0.067057 | 0.06622 | 0.064928 | 0.06193 |

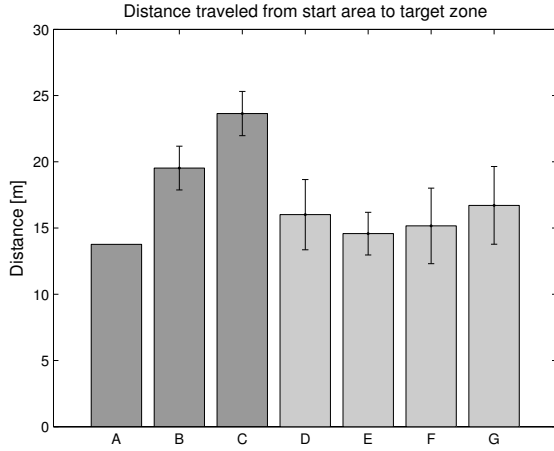


Fig. 10. Distance (mean with 95 % confidence interval for normal data) that the robot traveled from the start of the experiment until it reached the target area. Only successful runs were included in the analysis. The confidence interval for configuration A is not shown because only two runs were successful.

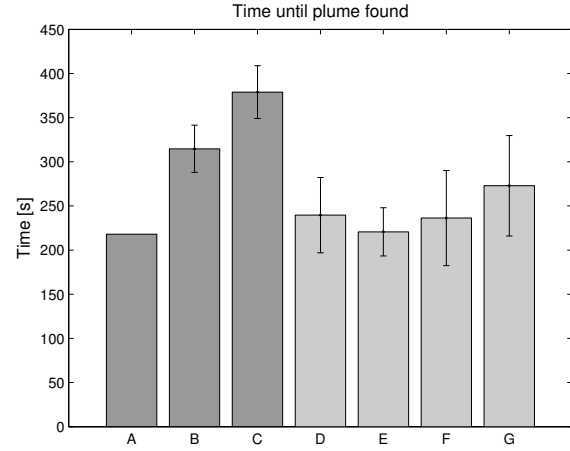


Fig. 11. Time (mean with 95 % confidence interval for normal data) from the start of the experiment until the target area was reached. Only successful runs were included in the analysis. The confidence interval for configuration A is not shown because only two runs were successful.

The spiral-surge algorithm on the other hand yields very similar success ratios, distances, and times for all chosen gap lengths. The optimal gap length here seems to be around 36 cm. While bigger gaps unnecessarily increase the spiral, smaller gaps may force the robot to make more turns until it finds the plume again. Furthermore, the spiral-surge algorithm was very robust - almost all runs were successful.

Most strikingly, though, the spiral-surge algorithm is significantly faster than the casting algorithm. The latter can only compete for very steep angles, in which case its robustness suffers. Even with only 10 runs, the difference is statistically significant: an ANOVA test yields P-value of 0.0228 between the distances of configurations B and D, and a P-value of 0.0940 between the distances of B and G.

By looking at the detailed results in Table I, a few more observations can be made:

- ▷ The low robot speed for the casting algorithm is due to the sharp turns, during which the robot did not travel any distance, but clearly spent some time.
- ▷ The distance traveled outside the plume, as well as the mean robot speed are about the same for configurations

B and G, but G reached the target area faster. This is because the robot is traveling straight upwind with the spiral-surge algorithm, instead of ascending with $\beta = 30$ degrees with casting.

- ▷ The advancement speed reveals that the mean speed towards the source is about 70 % of the robot's forward speed (7.1 cm/s) for configuration E, and goes back to 41 % for configuration C.

We furthermore observed that the spiral-surge algorithm often remained at the border of the plume. Hence, when the source was found, the robot was located a few centimeters left or right of the actual odor source outlet. The casting algorithm on the other hand stopped right in front of the outlet most of the time.

B. Discussion

Recall from Section II that both algorithms consist of two phases: upwind surge and plume reacquisition.

Plume reacquisition of the casting algorithm is extremely simple. It is based on the knowledge that the plume has just been crossed and must therefore be located behind the robot (with respect to the current wind direction) when it leaves it.

This works very well in theory, and can be proven to succeed in an ideal setting even for slowly meandering plumes. If the wind direction measurements are noisy, however, this strategy may fail, as shown in Figure 12: if the measurement error is in the order of β , the robot may try to reacquire the plume on the wrong side, even if the plume is ideal.

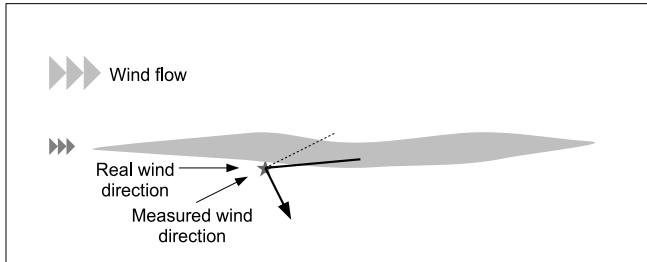


Fig. 12. A situation where the casting algorithm fails. Due to the error in the perceived wind direction, the acute angle is not on the side of the plume, and the robot continues in the wrong direction.

The spiral-surge algorithm does not suffer from this problem, as its plume reacquisition strategy does not make any assumptions about the plume location. Hence, it can afford to move straight upwind during surge, and thereby save time and distance. In addition, it is very robust against wind sensor measurement errors. In fact, even if the wind direction sensor returned a random value from time to time, the algorithm would still be able to locate the source, and just be less efficient.

VI. CONCLUSION

We carried out real-robot odor source localization experiments and compared two known algorithms: casting and spiral-surge. Both algorithms require an odor sensor and a wind sensor, and can be applied in settings with a main wind flow. The experiments were run in a wind tunnel with a low-speed, laminar wind profile.

Two main conclusions can be drawn. First, a good wind direction sensor is crucial for the casting algorithm, and presumably for all algorithms based on upwind movements. Second, the spiral-surge algorithm is faster (and more robust) as compared to casting (at least in low-speed, laminar flows) and the difference is statistically significant. This was surprising to us insofar that variants of zig-zagging have been used much more often than spiral-surge in previous studies.

Clearly, both algorithms can be enhanced in various ways, and the results presented here may not be globally applicable to all wind and odor profiles. In addition, both strategies could be combined, as silkworm moths [8] do, for example.

In future work, we will refine our analysis by observing the robot trajectories with a multi-camera system. In addition, we will study other odor source localization strategies based on the observations presented in this paper, to further decrease time and distance to find the source. Finally, the algorithms should be tested in more complex wind and odor profiles.

REFERENCES

- [1] G. S. Settles, "Sniffers: Fluid-dynamic sampling for olfactory trace detection in nature and homeland security—the 2004 freeman scholar lecture," in *Journal of Fluids Engineering*, ser. Transactions of the ASME, vol. 127, 2005, pp. 189–218.
- [2] D. W. Gage, "Many-robot MCM search systems," in *Proceedings of the Autonomous Vehicles in Mine Countermeasures Symposium*, April 1995, pp. 9.56–9.64.
- [3] M. Long, A. Gage, R. Murphy, and K. Valavanis, "Application of the distributed field robot architecture to a simulated demining task," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005)*. IEEE, April 2005, pp. 3193–3200.
- [4] Swiss foundation for mine action. [Online]. Available: <http://www.fsd.ch>
- [5] Mineseeker foundation. [Online]. Available: <http://www.mineseeker.com>
- [6] J. W. Gardner, *Electronic Noses and Sensors for the Detection of Explosives*. Springer, October 2004.
- [7] A. Rose, Z. Zhu, C. F. Madigan, T. M. Swager, and V. Bulovic, "Sensitivity gains in chemosensing by lasing action in organic polymers," *Nature*, vol. 434, pp. 876–879, April 2005.
- [8] R. A. Russell, *Odour Detection by Mobile Robots*, ser. World Scientific Series in Robotics and Intelligent Systems. World Scientific Publishing Company, 1999, vol. 22.
- [9] H. Ishida, T. Nakamoto, T. Moriizumi, T. Kikas, and J. Janata, "Plume-tracking robots: A new application of chemical sensors," *Biological Bulletin*, no. 200, pp. 222–226, April 2001.
- [10] A. T. Hayes, A. Martinoli, and R. M. Goodman, "Distributed odor source localization," *IEEE Sensors Journal*, vol. 2, no. 3, pp. 260–271, June 2002.
- [11] —, "Swarm robotic odor localization: Off-line optimization and validation with real robots," *Robotica*, vol. 21, pp. 427–441, 2003.
- [12] J. H. Berlinger and M. A. Willis, "Adaptive control of odor-guided locomotion: behavioral flexibility as an antidote to environmental unpredictability," *Adaptive Behavior*, vol. 4, no. 3–4, pp. 217–253, August 1996.
- [13] G. Ferri, E. Caselli, V. Mattoli, A. Mondini, B. Mazzolai, and P. Dario, "A biologically-inspired algorithm implemented on a new highly flexible multi-agent platform for gas source localization," in *Proceedings of the IEEE RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BIOROB 2006)*. IEEE RAS-EMBS, February 2006.
- [14] W. Li, J. A. Farrell, and R. T. Cardé, "Tracking of fluid-advected odor plumes: Strategies inspired by insect orientation to pheromone," *Adaptive Behavior*, vol. 9, no. 3–4, pp. 143–170, 2001.
- [15] L. P. S. Kuenen and H. C. Rowe, "Cowpea weevil flights to a point source of female sex pheromone: analyses of flight tracks at three wind speeds," *Physiological Entomology*, vol. 31, no. 2, p. 103, June 2006.
- [16] B. Webb, R. R. Harrison, and M. A. Willis, "Sensorimotor control of navigation in arthropod and artificial systems," *Arthropod Structure and Development*, vol. 33, pp. 301–329, May 2004.
- [17] A. J. Lilienthal, D. Reiman, and A. Zell, "Gas source tracing with a mobile robot using an adapted moth strategy," in *Autonome Mobile Systeme (AMS), 18. Fachgespräch*. Stuttgart, Germany: GDI, December 2003, pp. 150–160.
- [18] W. Li, J. A. Farrell, S. Pang, and R. M. Arrieta, "Moth-inspired chemical plume tracing on an autonomous underwater vehicle," *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 292–307, April 2006.
- [19] E. Balkovsky and B. I. Shraiman, "Olfactory search at high reynolds number," *PNAS*, vol. 99, no. 20, pp. 12 589–12 593, October 2002.
- [20] K. A. Justus and R. T. Cardé, "Flight behaviour of males of two moths, *cadra cautella* and *pectinophora gossypiella*, in homogeneous clouds of pheromone," *Physiological Entomology*, vol. 27, no. 1, pp. 67–75, March 2002.
- [21] W. Jatmiko, K. Sekiyama, and T. Fukuda, "A particle swarm-based mobile sensor network for odor source localization in a dynamic environment," in *Proceedings of the 8th Symposium on Distributed Autonomous Robotic Systems (DARS 2006)*. IEEE, 2006, pp. 71–80.
- [22] L. Marques, U. Nunes, and A. T. de Almeida, "Particle swarm-based olfactory guided search," *Autonomous Robots*, vol. 20, no. 3, pp. 183–287, June 2006.
- [23] H. Ishida, H. Tanaka, H. Taniguchi, and T. Moriizumi, "Mobile robot navigation using vision and olfaction to search for a gas/odor source," *Autonomous Robots*, vol. 20, no. 3, pp. 231–238, June 2006.