Tracking Odor Plumes in a Laminar Wind Field with Bio-Inspired Algorithms

Thomas Lochmatter and Alcherio Martinoli

Distributed Intelligent Systems and Algorithms Laboratory (DISAL)
École Polytechnique Fédérale de Lausanne (EPFL)
thomas.lochmatter@epfl.ch, alcherio.martinoli@epfl.ch

Summary. We introduce a novel bio-inspired odor source localization algorithm (surge-cast) for environments with a main wind flow and compare it to two well-known algorithms. With all three algorithms, systematic experiments with real robots are carried out in a wind tunnel under laminar flow conditions. The algorithms are compared in terms of distance overhead when tracking the plume up to the source, but a variety of other experimental results and some theoretical considerations are provided as well. We conclude that the surge-cast algorithm yields significantly better performance than the casting algorithm, and slightly better performance than the surge-spiral algorithm.

1 Introduction

With the advances in robotics and chemicals sensor research in the last decade, odor sniffing robots have become an active research area. Notably the localization of odor sources would allow for very interesting robotic applications, such as search and rescue operations, safety and control operations on airports or industrial plants, and humanitarian demining [20] [5] [16] [8]. Many of these applications are time-critical, i. e. odor sources should be found as fast as possible. But as the structure of plumes in the air is intermittent in both time and space [22], tracking plumes is a challenging problem.

In recent work [15], we have shown through experiments with real robots that the *surge-spiral* algorithm [6] [7] [2] [4] is faster and more reliable than *casting* [12] [11] [23] [14] [13] [1] [10] in laminar wind flow. This result was insofar surprising, as the casting algorithm got much more attention by the research community up to date.

In this paper, we introduce a third algorithm (referred to as the *surge-cast* algorithm) which belongs to the same category of odor source localization algorithms as the two previous algorithms (*surge-spiral* and *casting*). All three algorithms are combinations of strategies used by silkworm moths, and therefore bio-inspired. Silkworm moths use the following plume tracking behaviors [18] [19]:

- **Upwind surge**: straight upwind movement as long as the moth is in the plume;
- **Casting**: counter-turning (zig-zagging) to reacquire the plume right after loosing track of it;
- **Spiraling** [1]: an irregular, spiral-like movement to reacquire the plume if casting did not succeed.

While the *casting* algorithm is directly derived by the second behavior, the *surge-spiral* algorithm is a combination of the first and the third behavior. The new *surge-cast* algorithm is a combination of upwind surge and casting, which is exactly the behavior of a moth that does not loose the plume completely. To our knowledge, such an algorithm has never been tested on real robots before.

We carried out systematic experiments with a real robot in a wind tunnel under laminar flow conditions, with the goal to compare these algorithms in terms of plume tracking performance. In this paper, we present and discuss these results, and try to explain the differences with simple theoretical considerations.

Note that we only consider plume tracking (i. e. following the plume towards the source) and intentionally omit plume finding (i. e. randomized or systematic search until the plume is found) and source declaration (i. e. declaring that the source is in close vicinity). This allows us to make assertions about the plume tracking performances of the algorithms.

The remainder of this paper is structured as follows. In Section 2 we formally present the three algorithms discussed in this paper. The experimental setup and the robotic platform are introduced in Section 3. Finally, we discuss the results in Section 4 and conclude in Section 5.

2 Algorithms

All three algorithms discussed in this paper are bio-inspired and a combination of upwind surge, casting, and spiraling [18]. The 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, all three algorithms need a wind sensor to measure the wind direction. As molecules are mainly transported by advection, this piece of information is very valuable, and – as we will show later – as important as the odor sensor. The wind speed is ignored.

Since we are only interested in the plume tracking behavior, the robot starts in the plume, and declares failure if it gets too far away from it. This

[1] In [18] referred to as “irregular turning”.

allows us to rule out arena geometry effects, which could greatly influence the results (e. g., high variance introduced by randomized search techniques).

Similarly, source declaration is done by a supervisor (ideal source declaration) and therefore does not affect the results.[2] Experiments are considered successful if the robot has come in physical vicinity of the source.

2.1 The Surge-Cast Algorithm

The new algorithm we introduce here is a combination of upwind surge and cross-wind casting. It is similar to the surge-spiral algorithm (see below), with the spiral being replaced by cross-wind movement.

A robot in the plume moves straight upwind until it looses the plume for a distance d_{lost}. It then tries to reacquire the plume by moving cross-wind for a set distance (d_{cast}), first on one side and then on the other. To maximize the chances of hitting the plume in the first cross-wind movement, the robot measures the wind direction to estimate from which side it left the plume.

The wind direction is measured when the robot switches from upwind surge to casting and when it switches back to upwind surge, as indicated in Figure 1.

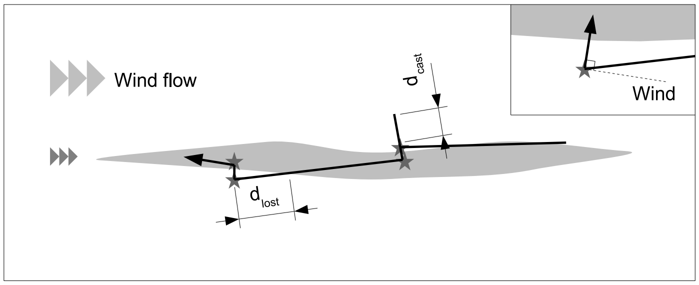


Fig. 1. Sketch of the surge-cast algorithm. The stars indicate where the wind direction is measured.

2.2 The Casting Algorithm

The casting algorithm is very similar the one described by Li et al. [12]. As shown in Figure 2, a robot in the plume moves upwind with an angle β until it is out of the plume for a certain distance, denoted d_{lost}. Once the plume is lost, the robot 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.

[2] On the real robots, this is done using IR sensors detecting a specific colored patch on the floor. See Section 3 for further details on the setup.

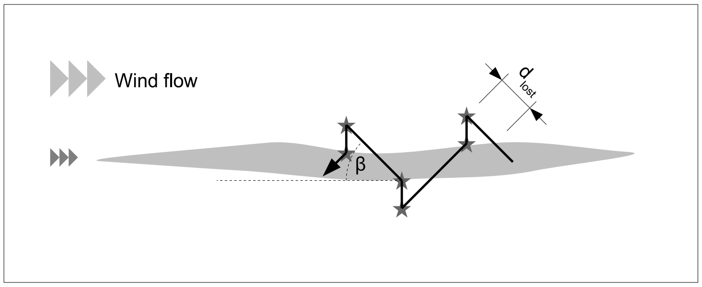


Fig. 2. Sketch of the casting algorithm. The stars indicate where the wind direction is measured.

2.3 Surge-Spiral

The surge-spiral algorithm is similar to Hayes' algorithm presented in [6], except that here we focus exclusively on its use for plume tracking. Hence, we have a single spiral gap parameter.

A robot in the plume moves straight upwind until it looses the plume for a distance d_{lost}. It then tries to reacquire the plume by moving along an Archimedes spiral with gap size d_{gap}. Unlike [6], we start our spiral in upwind direction, as drawn in Figure 3.

The wind direction is measured when the robot switches from upwind surge to spiraling, and when it switches back to upwind surge.

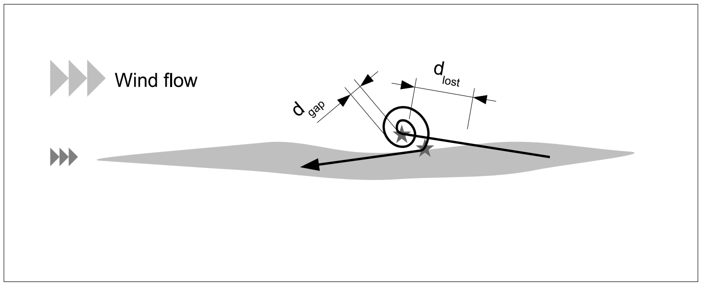


Fig. 3. Sketch of the surge-spiral algorithm. The star indicates where the wind direction is measured.

3 Real Robot Experiments

3.1 Experimental Setup

The experiments were carried out in a 16 m long and 4 m wide wind tunnel. The setup was exactly the same as described in our previous paper [15], except

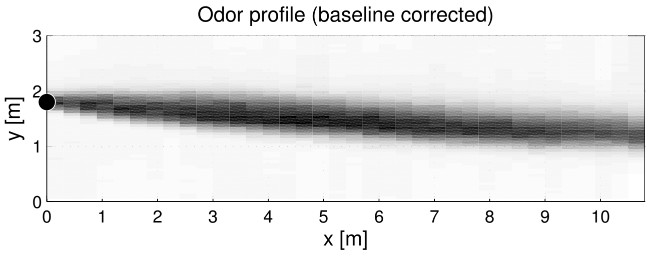


Fig. 4. Odor profile in the arena. Each measurement point is an average over about 20 seconds. The grid has a resolution of 30 cm in x-direction, and 5 cm in y-direction. The odor was measured at the height of the robot's odor sensor board using the traversing system of the wind tunnel.

that the arena was enlarged to approximately 15 m by 3.5 m. In the following paragraph, we briefly repeat the most important figures.

The wind field in the wind tunnel was laminar at roughly 1 m/s speed. The ethanol odor plume was therefore a straight line (see Figure 4), and the concentration peaks were slightly decreasing as the plume moves downwind. A constant amount a ethanol vapor was released by means of a pump. To reduce the turbulence created by the odor source, the pump was placed outside of the arena and connected with a tube to the source outlet. Nevertheless, the outlet created some turbulence right downwind the source, which sometimes disturbed the laminar wind flow in that area. The starting area was 14 meters downwind from the outlet, as depicted in Figure 5.

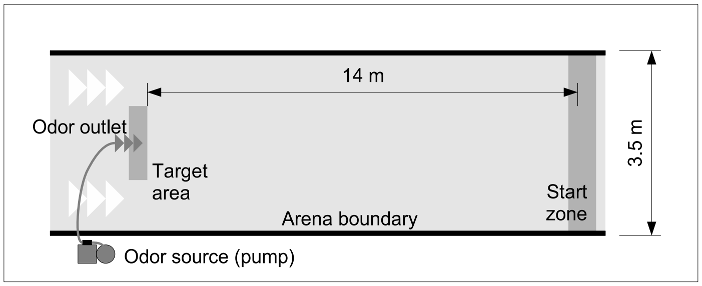


Fig. 5. Schematic drawing of our arena (not to scale).

3.2 Robotic Platform

The robot used in the experiments was a Khepera III robot (K-Team SA, Switzerland) equipped with an odor sensor and a wind sensor board, as depicted in Figure 6 (a).

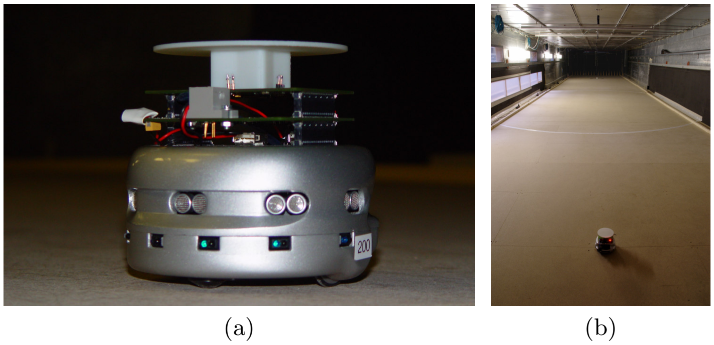


Fig. 6. (a) The Khepera III robot with the wind sensor and the odor sensor board. (b) Upwind view of the wind tunnel, with the robot in front and the odor source in the back.

The odor sensor was a MiCS-5521 volatile organic compound (VOC) sensor, which has a very fast response time (≈ 0.1 s). This sensor reacts to a wide range of organic compounds in the air, with an sensitivity to ethanol comparable to that of a human's nose (≈ 10 ppm). To take advantage of the sensors low response time, air was taken in and released with a small pump.

The wind sensor board was based on 4 thermistors placed around a star-shape obstacle. Once calibrated, a probabilistic model allowed the robot to infer the wind direction with an accuracy of roughly 10^o.

3.3 Experiments

We ran 20 experiments for each of the following configurations:

	Algorithm	**Parameter**
A	Casting	$\beta = 10^o$
B	Casting	$\beta = 20^o$
C	Casting	$\beta = 30^o$
D	Surge-spiral	$d_{\text{gap}} = 0.58$ m
E	Surge-cast	$d_{\text{cast}} = 0.72$ m
F	Surge-cast	$d_{\text{cast}} = 0.43$ m
G	Surge-cast	$d_{\text{cast}} = 0.14$ m

The forward speed of the robot (on straight lines) was approximately 10.6 cm/s and the plume lost distance was set to $d_{\text{lost}} = 40$ cm for all experiments. The plume threshold was determined before each run by measuring the response of the sensor to fresh air in the wind tunnel.

In each run, the robot was released in the odor at a position about 14.5 m downwind from the target area, and the corresponding algorithm was launched. If the robot reached the target area around the odor outlet (determined with the floor sensors), the run was considered successful. During the run, the trajectory (using odometry) and the odor concentration were

Table 1. Mean values (except for the success ratio) of all configurations. The distance overhead is the traveled distance divided by the upwind distance ($\frac{d_t}{d_u}$).

Configuration	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Success ratio	0.9	1	0.85	1	0.86364	0.9	0.4
Distance overhead [m/m]	1.1638	1.4323	1.6256	1.1429	1.1211	1.102	1.0585
Traveled distance [m]	17.08	21.08	23.90	16.65	16.36	16.08	15.33
Time to target [s]	179.9	231.2	263.3	161.2	165.4	162.1	152.0
Ratio in plume	78.8 %	63.3 %	58.1 %	82.0 %	83.4 %	84.7 %	86.9 %
Upwind speed [m/s]	0.083	0.064	0.056	0.091	0.089	0.090	0.096
Mean robot speed [m/s]	0.096	0.091	0.091	0.103	0.099	0.099	0.101

recorded. Distance and upwind distance were derived from the trajectory, and the duration of each run was measured on a host computer.

4 Results and Discussion

4.1 Experimental Results

Table 1 shows the mean values of the data recorded during the experiments. Besides the success ratio, the most interesting of these values is the ratio between the traveled distance (d_t) and the upwind distance (d_u), which is plotted in Figure 7. This value indicates what distance the robot had to drive in order to come 1 m closer to the source, and is therefore bigger or equal to 1. Furthermore, a selection of runs of all three algorithms is plotted in Figure 9.

The surprisingly good result of configuration A should be taken with a grain of salt, since the wheel diameter difference produced some bending of the trajectory (see Figure 9) which worked in favor of the algorithm. Without this effect, one would expect the success rate of this configuration to be very low [15].

At first glance, it is clear that the surge-cast algorithm introduced here outperforms casting and is at least as good as the surge-spiral algorithm. The overlapping confidence intervals do not allow us to make a statistical judgment about the configurations D, E and F, but theoretical considerations (see below) allow us to say that surge-cast should find the source in shorter distance.

The current implementation of surge-cast is less robust than surge-spiral. This is mainly the case for configuration G, in which the cross-wind distance is clearly too small. (Figure 11 shows a sample trajectory of a failing run.) However, one should bear in mind that the algorithm gives up after unsuccessful cross-wind movement, instead of switching to spiraling (as moths do) or increasing the cross-wind distance.

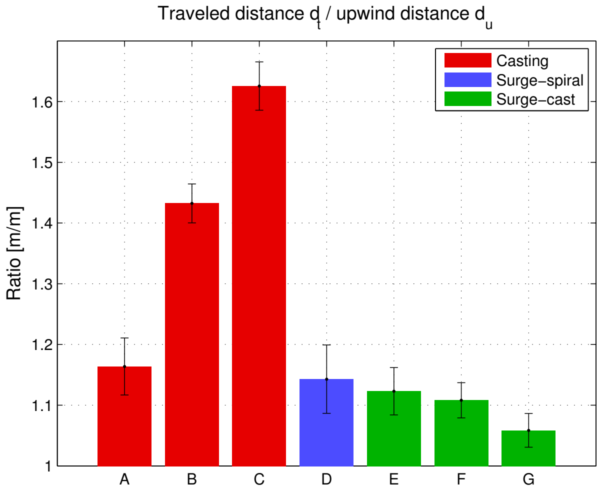


Fig. 7. Traveled distance d_t / upwind distance d_u (mean with 95 % confidence interval for normal data). Only successful runs were included in the analysis. Lower values are better.

4.2 Theoretical Considerations

Casting

Assuming that the plume is a straight line with a constant width r, an ideal casting run would yield a traveled distance / upwind distance ratio of:

$$\frac{d_t}{d_u} = \frac{\frac{1}{\sin\beta} + f\,(1+\sin\beta)}{\left(\frac{1}{\sin\beta} + f\right)\cos\beta} \geq \frac{1}{\cos\beta}$$

where

$$f = \frac{d_{\mathrm{lost}}}{r}$$

This $\frac{d_t}{d_u}$ ratio increases with increasing β or with increasing f, as drawn in Figure 8, and both parameters make a similar contribution to the bound. Optimally, one would choose a small upwind angle β and a small plume lost distance d_{lost}. This, however, decreases the robustness of the algorithm (i. e. the algorithm is more likely to fail completely) in non-ideal conditions and is therefore impractical.

Surge-Spiral and Surge-Cast

Under the same conditions, both the surge-spiral and the surge-cast algorithm are only bounded by

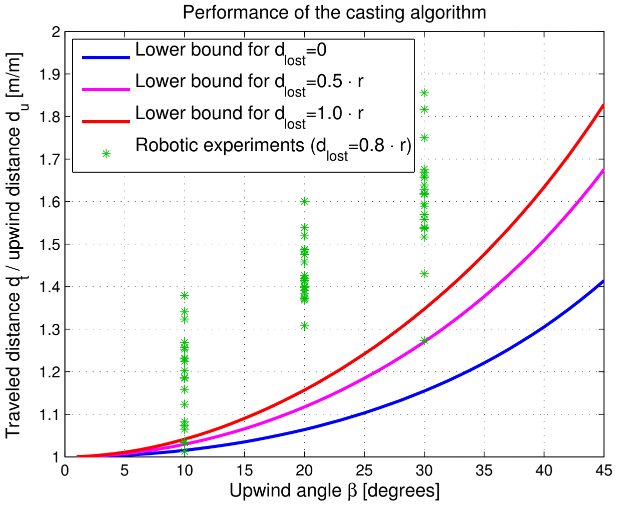


Fig. 8. Lower bounds and experimental results for pure casting. Note that "lucky runs" can beat the lower bound, as the bound is derived from a theoretical trajectory.

$$\frac{d_t}{d_u} = 1$$

which is the ideal performance a plume tracking algorithm can achieve. This value is only achieved if a robot moves in a straight line up to the source, without leaving the upwind surge mode a single time.

The two algorithms only differ in their plume reacquisition strategy, and *surge-cast* seems to be slightly more effective (distance-wise) than *surge-spiral*. Indeed, casting only requires the robot to move left and right whereas spiraling describes a circular trajectory, which is approximately π times longer to cover the same width. But since the plume is a straight line in our case, the spiral's movement in upwind-downwind direction does not provide any advantage to reacquiring the plume. This could be significantly different with other plume shapes, though.

Spiraling is in general a bit more robust than casting, as it does not require knowledge about the wind direction. With casting as plume reacquisition strategy, the probability to fail (or to produce a significant overhead) increases with the wind direction measurement error.

4.3 Differential-Drive Effects

The locomotion model has a big influence on the time to target. Since differential-drive robots cannot move sidewards, sharp corners (non-differentiable points) in the trajectory correspond to turning on the spot — an action which just takes time but does not add to the length of the trajectory.

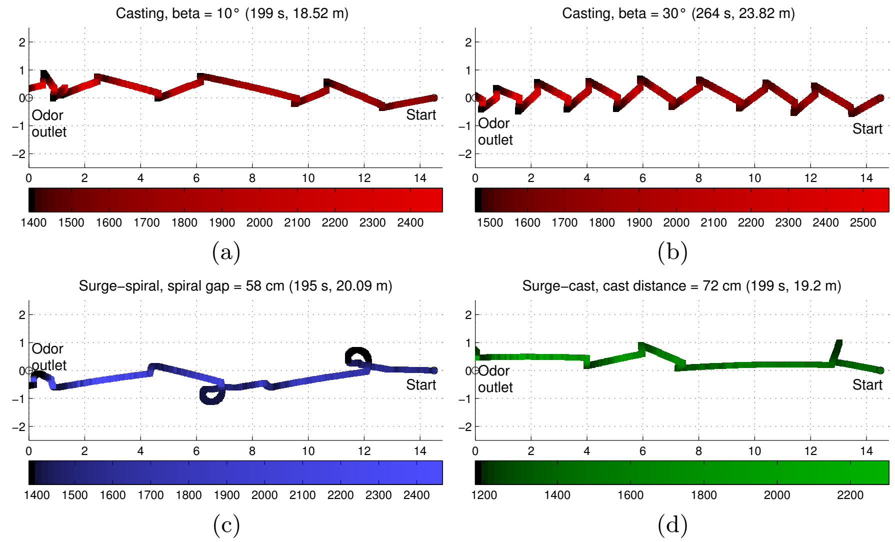


Fig. 9. Sample trajectories with odor concentration shading. The bars below the plots indicate the translation from shading to concentration (measured in arbitrary units). Note that straight trajectories are bent because of a tiny difference (0.08 mm) in wheel diameter between the left and the right wheel. The plume threshold was set to 100 units above the baseline concentration value indicated on the left side of the colored bar. (a, b) Successful runs of the casting algorithm. (c) Successful, but unlucky run of the surge-spiral algorithm. (d) Successful, but unlucky run of the surge-cast algorithm.

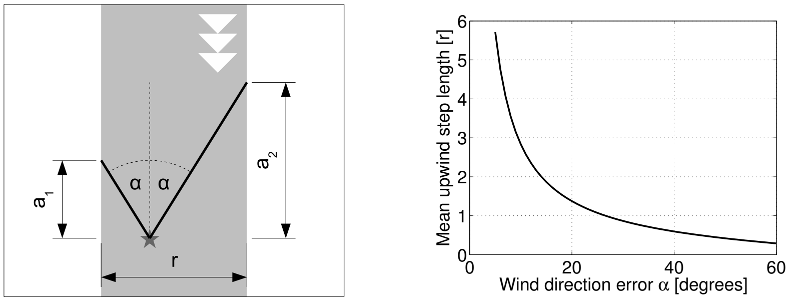
Due to this effect, surge-cast (configurations E and F) is slower (time-wise) than surge-spiral (configuration D), even though the trajectory is shorter. The surge-spiral algorithm is particularly well suited to differential-drive robots, as it makes very few sharp turns. Casting, on the other side, is very bad in that respect.

Holonomic robots [21] would not suffer from this, and their time to target would be directly proportional to the traveled distance (assuming zero weight). Other drive systems, such as Ackermann steering [21], would not even allow for sharp turns.

4.4 Wind Direction Measurement Error

An important factor is the accuracy of the wind direction sensor. Unfortunately, measuring the wind direction accurately at low wind speeds is a challenging engineering problem, and plume tracking algorithms have to be able to cope with a certain amount of error in these measurements.

In upwind surge, the average upwind distance a that robot gets closer to the source before leaving the plume (of width r) depends on the wind direction



(a) (b)

Fig. 10. (a) A simplified model for the wind direction measurement error in upwind surge. (b) The average length of upwind steps as function of the measurement error α.

measurement error α as follows:

$$E[a(\alpha)] = \frac{a_1(\alpha) + a_2(\alpha)}{2} = \frac{r}{2\tan\alpha}$$

A corresponding sketch is provided in Figure 10. Since $\tan\alpha \to 0$ for small α, it is clear that a good wind direction sensor will significantly increase the upwind step length, and therewith significantly improve the performance of algorithms based on upwind surge in laminar flow.

Pure casting is affected in a very different manner. As long as the wind direction measurement error is bigger than the casting angle, $\alpha \geq \beta$, the casting algorithm is very likely to fail. For $\alpha < \beta$, however, casting is only moderately affected. From a purely mathematical standpoint, an error $\alpha < \beta$ even increases the performance of casting (assuming a symmetric error distribution). As an example, consider a measurement error $\alpha = 15^o$ and a casting algorithm using $\beta = 20^o$. If the sign of the error decreases the effective upwind angle, the robot will move through the plume with an effective upwind angle of 5^o, and therewith make a much larger step towards the source than with 20^o. In the opposite case, the robot will cast under an angle of 35^o, which results in a just slightly smaller progress towards the source.

4.5 Odor Sensor Speed

The speed of the odor sensor comes into play each time the robot enters or leaves the plume. It essentially sets a lower bound on d_{lost}. The effective plume lost distance is

$$d_{\mathrm{lost}} = t_{\mathrm{relax}} \cdot s + d_{\mathrm{lost\_wait}}$$

where s denotes the speed of the robot.

Upon entering the plume, the sensor delay most often causes the robot to enter slightly deeper into the plume before detecting it. This slightly decreases the performance of casting and upwind surge under the theoretical model used above. Visual observations with the real robots have shown, however, that a small delay (causing the robot to penetrate 2-3 cm into the plume) rather increases the performance. This spatial hysteresis makes the plume detection mechanism more robust by avoiding transition zones with high concentration fluctuations, in the same way as the plume lost distance (d_{lost}) principle works when the plume is lost.

In any case, the sensor speed sets an upper bound on the speed of the robot. Namely, the robot needs to be still in the plume when it first perceives it. In our real-robot experiments with a plume width of 50 cm, a robot speed of 10.6 cm/s and a sensor response time of 0.1 s, we were more than an order of magnitude below that upper bound. Insects typically have very fast sensors, allowing them to move very quickly.

4.6 Odometry Error and Calibration

Most robot locomotion systems are susceptible to manufacturing inaccuracies. For differential-drive robots (such as the robot used in our experiments), for instance, a small error in wheel diameter will cause the robot to turn when its motors are turning at the same speed. This bias can be compensated for through calibration. In our setup, we did not correct for that during the experiments, but used calibration for offline processing only (to plot the trajectories). Therefore, straight lines performed by the robot appear bent on the reconstruction of the real trajectory plots in Figure 9 and Figure 11.

Visual inspection of these trajectories reveals that the locomotion error did not have a significant influence on the functioning of the algorithms. In fact, each time the robot measures the wind direction, it adjusts for the accumulated error. All three algorithms can therefore be called robust with regard to small locomotion inaccuracies. We believe, however, that large inaccuracies could significantly degrade the performance of upwind surge.

Finally, we will monitor the robotic trajectory using a multi-camera system endowed with the tracking software SwisTrack [3].

5 Conclusion

Odor source localization algorithms based on upwind surge (surge-cast or surge-spiral) are significantly faster than pure casting — at least in laminar wind flow. This is supported both by theoretical arguments and experimental results with real robots. Silkworm moths [18] and other animals use casting primarily for plume reacquisition rather than for plume tracking. (Casting as plume tracking is used by ants following a pheromone trail on the ground [18]. However, ants just need to sway their head left and right to scan the pheromone on the ground - the back part of the body goes almost straight.)

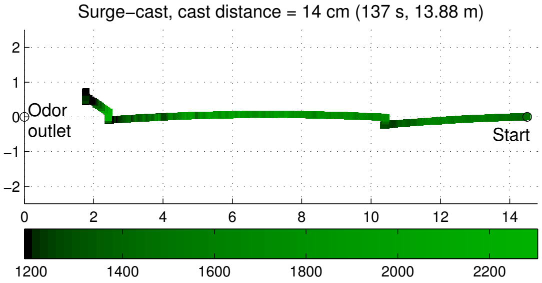


Fig. 11. Unsuccessful trajectory produced by the surge-cast algorithm. The robot lost the plume after an erroneous wind direction measurement 2 m downwind from the source.

Among the plume reacquisition strategies, casting seems to be slightly faster, but less robust than spiraling. The former, however, is not well suited for differential-drive locomotion, and probably even less suited for more complicated locomotion systems. Spiraling generates much smoother trajectories and can therefore be applied with almost any locomotion system. Even though the combination of casting and spiraling that moths are using [18] has not been tested in this paper, the available results suggest that this is a very efficient and robust strategy.

In future work, we will test the algorithms in turbulent flow and/or meandering plume conditions. In addition, we will introduce obstacles along the arena, both to generate turbulence and to hinder the robot from moving along a straight line up to the source.

Acknowledgments

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.

References

1. Eugene Balkovsky and Boris I. Shraiman. Olfactory search at high reynolds number. *PNAS*, 99(20):12589–12593, October 2002.
2. Jim H. Berlanger and Mark A. Willis. Adaptive control of odor-guided locomotion: behavioral flexibility as an antidote to environmental unpredictability. *Adaptive Behavior*, 4(3-4):217–253, August 1996.
3. Nikolaus Correll, Gregory Sempo, Yuri Lopez de Meneses, José Halloy, Jean-Louis Deneubourg, and Alcherio Martinoli. Swistrack: A tracking tool for multi-unit robotic and biological systems. In *Proceedings of the 2006 IEEE/RSJ*

International Conference on Intelligent Robots and Systems (IROS 2006), pages 2185—2191. IEEE/RSJ, 2006.

4. Gabriele Ferri, Emanuele Caselli, Virgilio Mattoli, Alessio Mondini, Barbara Mazzolai, and Paolo 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.
5. Douglas W. Gage. Many-robot MCM search systems. In *Proceedings of the Autonomous Vehicles in Mine Countermeasures Symposium,*, pages 9.56–9.64, April 1995.
6. Adam T. Hayes, Alcherio Martinoli, and Rodney M. Goodman. Distributed odor source localization. *IEEE Sensors Journal*, 2(3):260–271, June 2002.
7. Adam T. Hayes, Alcherio Martinoli, and Rodney M. Goodman. Swarm robotic odor localization: Off-line optimization and validation with real robots. *Robotica*, 21:427–441, 2003.
8. Hiroshi Ishida, Takamichi Nakamoto, Toyosaka Moriizumi, Timo Kikas, and Jiri Janata. Plume-tracking robots: A new application of chemical sensors. *Biological Bulletin*, (200):222–226, April 2001.
9. Wisnu Jatmiko, Kosuke Sekiyama, and Toshio Fukuda. A pso-based mobile robot for odor source localization in dynamic advection-diffusion with obstacles environment. *IEEE Computational Intelligence Magazine*, pages 37–51, May 2007.
10. Kristine A. Justus and Ring T. Cardé. Flight behaviour of males of two moths, cadra cautella and pectinophora gossypiella, in homogeneous clouds of pheromone. *Physiological Entomology*, 27(1):67–75, March 2002.
11. 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*, 31(2):103, June 2006.
12. Wei Li, Jay A. Farrell, and Ring T. Cardé. Tracking of fluid-advected odor plumes: Strategies inspired by insect orientation to pheromone. *Adaptive Behavior*, 9(3-4):143–170, 2001.
13. Wei Li, Jay A. Farrell, Shuo Pang, and Richard M. Arrieta. Moth-inspired chemical plume tracing on an autonomous underwater vehicle. *IEEE Transactions on Robotics*, 22(2):292–307, April 2006.
14. Achim J. Lilienthal, Denis Reiman, and Andreas Zell. Gas source tracing with a mobile robot using an adapted moth strategy. In *Autonome Mobile Systeme (AMS), 18. Fachgespräch*, pages 150–160, Stuttgart, Germany, December 2003. GDI.
15. Thomas Lochmatter, Xavier Raemy, Loïc Matthey, Saurabh Indra, and Alcherio Martinoli. A comparison of casting and spiraling algorithms for odor source localization in laminar flow. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA 2008)*, pages 1138–1143. IEEE, May 2008.
16. Matt Long, Aaron Gage, Robin Murphy, and Kimon 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)*, pages 3193–3200. IEEE, April 2005.
17. Lino Marques, Urbano Nunes, and A. T. de Almeida. Particle swarm-based olfactory guided search. *Autonomous Robots*, 20(3):277–287, June 2006.

18. R. Andrew Russell. *Odour Detection by Mobile Robots*, volume 22 of *World Scientific Series in Robotics and Intelligent Systems*. World Scientific Publishing Company, 1999.
19. Kanzaki Ryohei, Sugi Naoko, and Shibuya Tatsuaki. Self-generated zigzag turning of bombyx mori males during pheromone-mediated upwind walking. *Zoological science*, 9(3):515–527, 1992.
20. Gary 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*, volume 127 of *Transactions of the ASME*, pages 189–218, 2005.
21. Roland Siegwart and Illah R. Nourbakhsh. *Introduction to Autonomous Mobile Robots*. Intelligent Robotics and Autonomous Agents series. The MIT Press, April 2004.
22. Massimo Vergassola, Emmanuel Villermaux, and Boris I. Shraiman. 'infotaxis' as a strategy for searching without gradients. *Nature*, 445:406–409, January 2007.
23. Barbara Webb, Reid R. Harrison, and Mark A. Willis. Sensorimotor control of navigation in arthropod and artificial systems. *Arthropod Structure and Development*, 33:301–329, May 2004.