Estimation of Gaussian Plume Model Parameters Using the Simulated Annealing Algorithm

Gonçalo Cabrita and Lino Marques

Institute of Systems and Robotics,
University of Coimbra,
3030-290 Coimbra, Portugal.
{goncabrita, lino}@isr.uc.pt

Abstract. This article presents a novel cost function for estimating the parameters of the Gaussian plume model using simulated annealing. The novel cost function takes into account the meandering and intermittency phenomena found on dispersing plumes. The proposed method was validated using real gas sensor data sampled by a swarm of 5 robots performing the Decentralized Asynchronous Particle Swarm Optimization for plume tracing under a controlled environment.

Keywords: Gaussian plume model, simulated annealing, odor plume estimation, swarm robotics

1 Introduction

Robotic odor-sensing technology or simply robotic olfaction has attracted substantial interest by the research community in recent years [1] [2] [3] [4]. This interest is driven by the developments in the robotics and sensing technologies along with the vast number of areas and applications of robot olfactory systems including safety, security, and environmental inspection to name a few. Robots equipped with gas sensors can be used instead of humans in areas with odor contamination for purposes such as inspection, detection of leakages leading to the contamination source, providing continuous monitoring of the contaminated environment, for specific characterization of the odor, for building the gas distribution map of the environment and others.

One of the challenges that is at the forefront of this field of research is how to deal with multiple odor sources. We recently proposed a novel methodology which targets this problem called virtual cancelation plume [5]. In a nutshell, the goal of virtual cancelation plume is to make an odor source invisible to the robot or robots, allowing the pursuit of multiple odor sources by successively cancelling the sources already found. In order to achieve this goal once a known odor source is found a model of the plume being created by that source is generated. In order to do this the robot or robots must be able to estimate the necessary parameters. This model will then be used to affect the readings of the gas sensors equipped on the robots. Needless to say that achieving a model that represents the actual plume is of the utmost importance, thus this article tackles the problem of estimating the parameters of a plume represented by a Gaussian model, the model used in [5], using the data gathered by the robots.

Naturally this is not a new problem. In fact the process of estimating the parameters of a plume model from a sensor generated gas distribution map is known in the literature as the inverse method. In 1994 Lehning et al. used this technique to estimate gas emissions and source locations in land-fills [6]. In 2005 Flescha et al. used the same principle to achieve the same results for an agricultural scenario [7]. More recently [8] applied the inverse problem approach to locate a known gas source in a desert setting from simultaneous measurements of gas concentration and wind data. In [8] they use a simulated annealing algorithm to generate candidate distributions and present and evaluate three different cost functions with different regularisation terms. The common element in these works is the fact that these were not developed having mobile robotics in mind, in fact some of the strategies proposed to deal with the chaotic nature of odor plumes are not even applicable to a mobile robotics scenario. As the entire dataset is extracted prior to analysis a series of filtering and preprocessing is performed, in [6] unstable periods are removed from the dataset whereas in [8] the average of each location was calculated prior to parameter estimation. There are some examples of the estimation of odor plume parameters in a mobile robot context, e.g. in [9] a group of quadcopters uses a non-linear least squares approach to estimate the parameters of a Gaussian Puff model using the data gathered by their gas sensors.

In order to properly understand the challenges that we are facing when trying to estimate the parameters of a Gaussian plume model from gas sensor readings we must first take a look at the model itself and how it differs from an actual odor plume.

2 The Gaussian Plume Model

The Gaussian model is probably one of the simplest plume models commonly used. Let $\bar{c}$ be the mean chemical concentration at any given x and y given by Equation 1 where Q is the emission rate of the source, $\bar{u}$ is the mean transport velocity and σ_y and σ_z are the Gaussian plume dispersion parameters. Note that σ_y and σ_z depend on x and for non-buoyant releases can be approximated by power laws represented by equations 2 and 3. The three main requirements for a Gaussian plume model to hold are a continuous odor source, uniform wind flow and homogeneous turbulence [10].

$$\bar{c}(x,y) = \frac{Q}{\pi \sigma_y \sigma_z \bar{u}} \exp\left(-\frac{y^2}{2\sigma_y^2}\right) \tag{1}$$

$$\sigma_y = a x^p \tag{2}$$

$$\sigma_z = b x^q \tag{3}$$

Equation 1 can be derived from the assumption of a Gaussian concentration distribution in the y and z axis (considering a dominant transport velocity along the x axis) at any cross section in the plume downind of the source and the integral mass-conservation condition in Equation 4.

$$\int_0^\infty \int_{-\infty}^\infty \bar{c}\bar{u} \, \mathrm{d}y \, \mathrm{d}z = Q \tag{4}$$

A number of early diffusion experiments were conducted with the goal of characterising the values of σ_y and σ_z for different conditions. One of these experiments was carried out at the Brookhaven National Laboratory (BNL) [10]. Based on more than 15 years of diffusion data collected at the BNL site in central Long Island, New York, using nonbouyant, passive tracers, the dispersion parameters σ_y and σ_z were determined from the analysis of extensive ground-level concentration measurements for four classes of wind turbulence (Table 1). These values provide us an insight into the range of values that σ_y and σ_z can assume for an actual odor release, however these are not applicable to all locations.

Table 1. The BNL dispersion parameterization scheme [10].

Turbulence Type	σ_y	σ_z
Unstable	$0.40x^{0.91}$	$0.41x^{0.91}$
Neutral	$0.36x^{0.86}$	$0.33x^{0.86}$
Stable	$0.32x^{0.78}$	$0.22x^{0.78}$
Very Stable	$0.31x^{0.71}$	$0.06x^{0.71}$

2.1 Instantaneous vs. Average

The dispersion of gaseous and particles in the atmosphere is a complex process that is dependent on the entire spectrum of turbulent motions in the atmosphere, ranging from the micro-scale (the smallest being the random motion of molecules that cause molecular diffusion) up to the macro-scale (e.g., arising from synoptic events and zonal currents). The random nature of turbulent motions of different scales produces fluctuations of the instantaneous concentration with the probabilistic and statistical characteristics of the field determined by the turbulent motions of the boundary-layer flow [11].

Concentration fluctuations are a ubiquitous feature of dispersing plumes, and the recognition of this feature has practical importance in the estimation of the parameters of a plume model based on instantaneous gas sensor readings. This is mainly due to the fact that most plume models are in fact average models, representing a plume as if its concentration values were sampled and averaged for a large amount of time. On the other hand mobile robots instantaneously sample the plume across their trajectory, providing a picture that represents the status of the plume at multiple places in sequential times. The two main phenomena commonly observable in a dispersing plume, meandering and intermittency are discussed next.

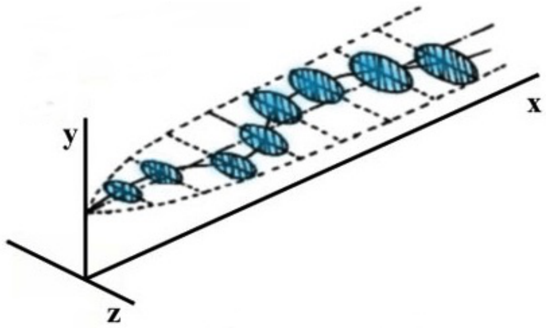
The meandering effect can be viewed in Figure 1. A meandering plume (Figure 1(a)), where the clouds both increase in size and follow a meandering path as they move away from the source can be averaged by a Gaussian plume model (Figure 1(b)). However an observer standing still inside the plume might not be able to detect odor at all times [12].

Intermittency is a characteristic of odor propagation that is also reflected in the probability of occurrence of zero concentration inside the plume. This is however a concept with which people are often more familiar with as it can be easily observed in a rising column of smoke. The smoke will not generate a smooth plume, it will in fact rise in patches, thus if an observer were to stand still inside the plume it would read intermittent values of odor. This effect is more prominent near the source, since as the odor propagates the patches tend to mix and the plume becomes more homogeneous as the distance from the source increases. There are some statistical analysis of how intermittency varies inside a plume [13], however these results do not apply to a broader scope of transport conditions.

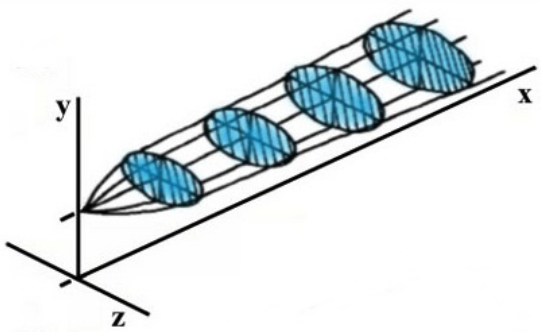
3 Simulated Annealing

Simulated annealing is a metaheuristic optimization algorithm inspired in the physical process of the annealing of a solid to low energy states. This process consist of heating a solid until it reaches its fusion temperature, so that matter shifts from the solid to the liquid state. Posteriorly the temperature is slowly lowered to avoid meta-stable states. The goal of this process is to obtain a crystal-like state, i.e. a minimum energy state. On its liquid state matter contains great amounts of energy and its particles are randomly placed. On the other hand, on its crystal solid state particles are extremely structured, resulting in minimum energy. It is extremely important that the passage from the liquid to the solid state is performed slowly so that matter does not solidify in an intermediate non-structured state.

The concept of simulated annealing was introduced as an optimization algorithm during the 80s by Kirkpatrick et al. [14] [15]. However it was in the 50s that Metropolis presented an article [16] in which the process of annealing was modelled using an algorithm based on Monte Carlo techniques [17]. Not only does the simulated annealing algorithm accepts changing state if the cost func-



(a) A meandering plume model, where the clouds both increase in size and follow a meandering path as they move away from the source.



(b) The basic Gaussian plume model, in which the spread of the cloud increases with distance from the source, estimated with a Gaussian standard deviation.

Fig. 1. Adapted from [12].

tion is minimized, it might also move to states which increase the cost function. It is this ability that allows this algorithm to escape local minima and find the global cost function minimum. As the algorithm progresses and the temperature cools down these worse states get increasingly less likely to be accepted. At the same time the step between states also decreases.

The process of estimating the Gaussian plume model parameters is treated as an optimization problem, where the goal is to minimize a cost function that depends on the model generated for each stage and the gas sensor readings provided by the robots. The simulated annealing was chosen for this purpose since metaheuristic and evolutionary algorithms have proven in the past to be able to cope with the chaotic nature of odor dispersion [18] [19] [20] [8]. The pseudocode of the simulated annealing algorithm used in this work is presented next.

Simulated Annealing

```
function simulatedAnnealing

    state = initial_state
    best_state = state

    cost = meanderingIntermittencyCostFunction(readings, state)
    best_cost = cost

    k = 0

    while k < k_max and cost > target_cost

        T = T_start*(k_max-k)/k_max

        new_state = generateState(best_state, T)
        new_cost = meanderingIntermittencyCostFunction(readings, new_state)

        if exp((cost - new_cost)/(T)) > random(0, 1)

            state = new_state
            cost = new_cost
        end

        if new_cost < best_cost

            best_state = new_state
            best_cost = new_cost
        end

        k = k+1
    end
end
```

The cost function is at the core of the simulated annealing algorithm. In the next section we describe a novel cost function designed having in mind the characteristics of gas dispersions discussed in 2.1.

3.1 The Meandering-Intermittency Cost Function

A common approach to define a cost associated with a certain state is to calculate the squared error between the generated model and the set of input data. However in this case it might not be reasonable to penalize a reading that is inside the plume and deviates from the value provided by the average model when we know *a priori* that such an occurrence is not only possible, but extremely likely to happen.

We developed a novel cost function that will degrade with the presence of chemical readings above the clean air threshold outside of the plume model. On the other hand chemical readings inside the plume model will not contribute towards the cost unless they are above a concentration value that depends on the average value as determined by the plume model. The main goal is to find the shape that can fit all the sensor readings inside while at the same time representing the decay of chemical concentration along the centerline properly. The proposed cost function is shown next where $c_t h$ is the clean air threshold and $alpha$ is a tunning value that allows to adjust the magnitude of the error.

The meandering-intermittency cost function

```
function meanderingIntermittencyCostFunction

    for each gas sensor reading

        error = 0
        c = gaussianPlumeModel(reading_x, reading_y)

        if c > c_th and reading > c
            error = error + alpha*exp(pow(reading - c, 2))
        end
    end
end
```

4 Experimental Setup

The experiments described next were performed inside a controlled environment, an arena designed specifically for odor experiments represented by the schematic in Figure 2 and shown in Figure 3(b). The $3m \times 4m \times 0.5m$ arena is an enclosed environment delimited by four walls where two extremities are made of honeycomb-like plastic, allowing for the air to circulate. It includes an array of controllable fans thus making it possible to control the airflow inside the arena. The top is covered by a large transparent acrylic cover. This setup allows to generate laminar and constant wind-flow. As a result the requirements for the Gaussian plume to be applicable are met.

Experiments were performed using a swarm of 5 miniQ robots (shown in Figure 3(a)) running the Decentralized Asynchronous Particle Swarm Optimization (DAPSO) algorithm for plume tracing [21]. The miniQs are small and cheap robots based on the popular Arduino platform. They were modified to achieve olfactory swarming mainly due to the e2v MiSC5524 gas sensors that they carry. Moreover, two LEDs (one red and one blue) were also integrated in the robots for usage with an Arecont MegaVideo IP camera for correcting the odometry of the robots and to provide global localization. Odometry correction and global localization is achieved using SwisTrack [22], a software designed for tracking robots.

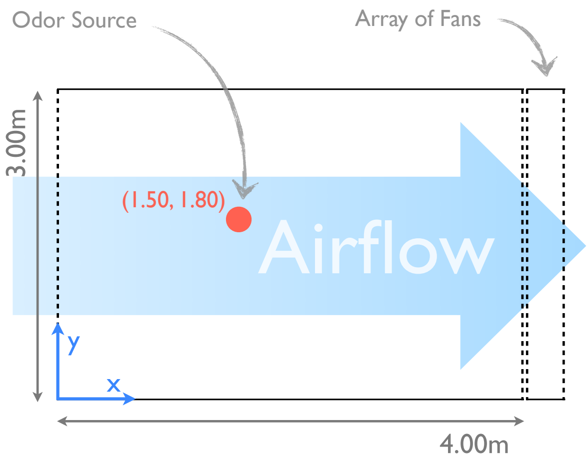


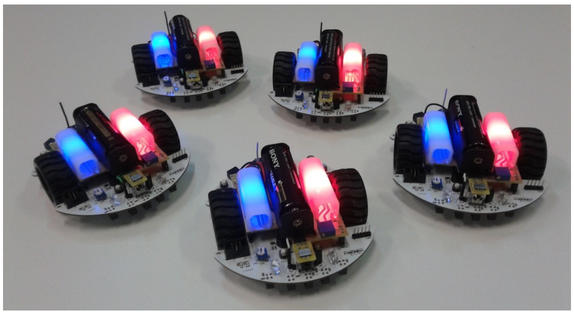
Fig. 2. The arena used for the experiments.

All experiments were performed under an average wind speed of $0.1m/s$ with a chemical release rate of approximately 1 mg/s at a height of approximately 0.07 m (about 0.04 m above the height at which the gas sensors are located). A total of 3 experiments was performed. The data collected by the robots was used to estimate the parameters of the Gaussian plume model in MATLAB where the simulated annealing algorithm using the proposed cost function was implemented. The values of c_{th} and α used were respectively 0.1 and 1.2. Notice that the chemical data collected by the robots was normalized between 0 and 1, meaning that we consider that the plume ends wherever the chemical concentration falls bellow 10% of the maximum value. We are also admitting that model-wise the concentration is allowed to peack 20% above the average value.

5 Experimental Results

The graphics in Figures 4(a), 5(a) and 6(a) contain the gas distribution map sampled by the robots during the experiments. Notice that these maps were obtained by interpolating the data sampled by the robots using Krigging [23] and are shown here for visualization purposes only. Krigging, a method commonly used in geostatistics, is based on the notion that the value at an unknown point in space should be the average of the known values at its neighbours weighted by the variogram of the distance to the unknown point. The data used by the simulated annealing algorithm to estimate the parameters of the Gaussian plume models was not pre-processed in any way.

Figures 4(b), 5(b) and 6(b) contain the Gaussian plume models after each experiment. Table 2 contains the parameters of the Gaussian plume model estimated by the simulated annealing algorithm for each experiment.



(a) The miniQs.

(b) The test arena.

Fig. 3. The miniQ robots used in the experiments and the test environment.

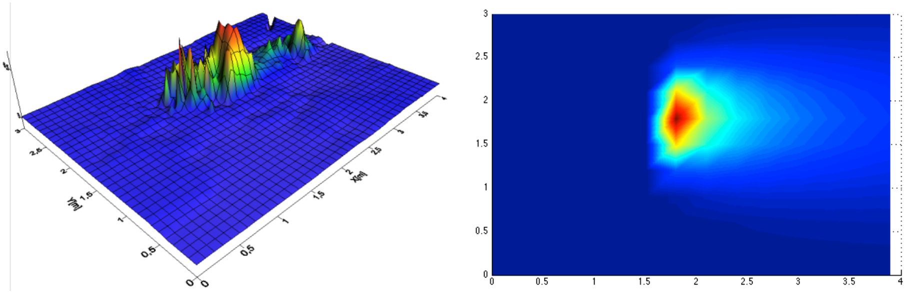
6 Discussion

Although the robots performed all 3 experiments under the same conditions it is normal that the data extracted differs from experiment to experiment due to both the chaotic nature of the odor plume and the random component present in the DAPSO algorithm. The latest results in completely different paths taken by each robot during each experiment, thus resulting in an entirely different dataset space-wise.

Nevertheless the resulting plume models resemble one another in shape. The plume model generated for experiment 2 presents the value of $\bar{u}$ closest to the

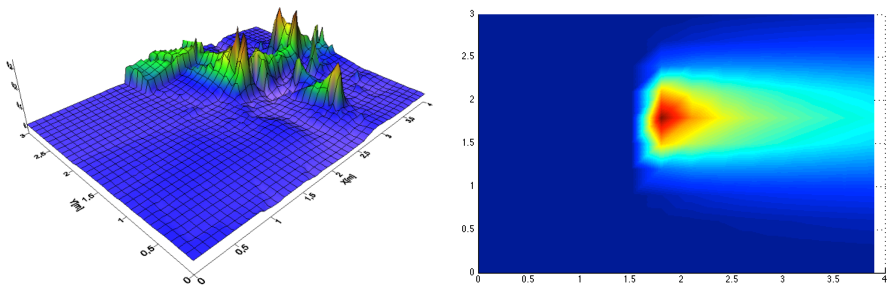
Table 2. The Gaussian plume parameters estimated for each experiment.

Experiment	Q	$\bar{u}$	σ_y	σ_z
1	0.7039	0.6762	$0.5013x^{0.3713}$	$0.3499x^{0.5774}$
2	0.3425	0.1992	$0.4602x^{0.2534}$	$0.3956x^{0.2089}$
3	1.7305	0.6511	$0.8156x^{0.7215}$	$0.3643x^{0.5381}$



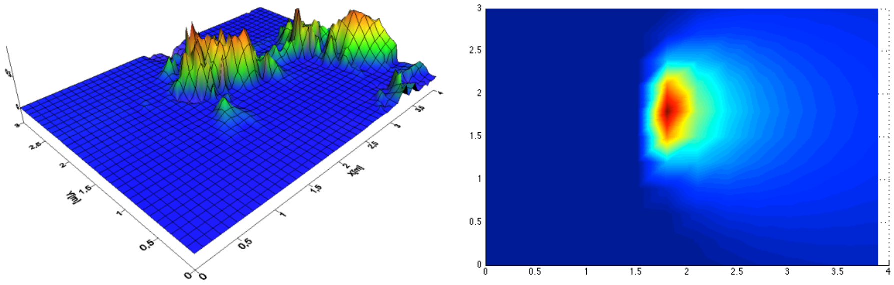
(a) Odor map (b) Estimated Gaussian plume model.

Fig. 4. Results for experiment 1.



(a) Odor map (b) Estimated Gaussian plume model.

Fig. 5. Results for experiment 2.



(a) Odor map (b) Estimated Gaussian plume model.

Fig. 6. Results for experiment 3.

measured value. The plume model generated during experiment 3 is the widest, having to accommodate the sensor readings present at $x = 4\,m$.

Comparing the estimated values of σ_y and σ_z with those in Table 1 it is possible to see that these are well inside the expected range. There is however no identifiable pattern in relation to the conditions in Table 1. In spite of this no conclusions can be drawn in this regard as the values of σ_y and σ_z in Table 1 might not necessarily be applicable for different scenarios, particularly a small-scale scenario as the experimental setup used in this work. These values (Table 1) were obtained from data collected in large field experiments, some extending up to 60 km.

7 Conclusions

The meandering-intermittency cost function allowed to successfully estimate the parameters of the Gaussian plume model for real world experiments. The resulting plume models are adequate in the contexts of the virtual cancelation plume algorithm.

Future work will focus on re-running the experiments with a sensor network in place to allow for the measurement of ground truth data and the accurate estimation of the odor plume. This procedure will allow assessing the accuracy of the proposed method.

Acknowledgments This work was partially carried-out in the framework of TIRAMISU project. This project is funded by the European Community's Seventh Framework Program (FP7/2007-2013) under grant 284747.

References

1. Ishida, H., Wada, Y., Matsukura, H.: Chemical sensing in robotic applications: A review. (2012)
2. Lochmatter, T., Göl, E.A., Navarro, I., Martinoli, A.: A plume tracking algorithm based on crosswind formations. In: Distributed Autonomous Robotic Systems. Springer (2013) 91–102
3. Neumann, P.P., Hernandez Bennetts, V., Lilienthal, A.J., Bartholmai, M., Schiller, J.H.: Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms. Advanced Robotics **27**(9) (2013) 725–738
4. Marques, L., Nunes, U., de Almeida, A.: Olfaction-based mobile robot navigation. Thin Solid Films **418**(1) (2002) 51–58 1st Int. School on Gas Sensors.
5. Cabrita, G., , Marques, L., Gazi, V.: Virtual Cancelation Plume for Multiple Odor Source Localization. In: Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2013). (2013)
6. Lehning, M., Shonnard, D.R., Chang, D.P., Bell, R.L.: An inversion algorithm for determining area-source emissions from downwind concentration measurements. Air & waste **44**(10) (1994) 1204–1213

7. Flesch, T.K., Wilson, J.D., Harper, L.A., Crenna, B.P.: Estimating gas emissions from a farm with an inverse-dispersion technique. Atmospheric Environment **39**(27) (2005) 4863–4874
8. Thomson, L.C., Hirst, B., Gibson, G., Gillespie, S., Jonathan, P., Skeldon, K.D., Padgett, M.J.: An improved algorithm for locating a gas source using inverse methods. Atmospheric Environment **41**(6) (2007) 1128–1134
9. Euler, J., Stryk, O.v.: Optimal Cooperative Control of Mobile Sensors for Dynamic Process Estimation. In: RSS213 Workshop on Robotics for Environmental Monitoring. (2013)
10. Arya, S.P.: Air Pollution Meteorology and Dispersion. Oxford University Press (1999)
11. Yee, E., Chan, R., Kosteniuk, P., Chandler, G., Biltoft, C., Bowers, J.: The vertical structure of concentration fluctuation statistics in plumes dispersing in the atmospheric surface layer. Boundary-Layer Meteorology **76**(1-2) (1995) 41–67
12. Murlis, J., Elkinton, J., Card, R.: Odor plumes and how insects use them. Annu. Rev. Entomol. **37** (1992) 505–532
13. Yee, E., Chan, R.: A simple model for the probability density function of concentration fluctuations in atmospheric plumes. Atmospheric Environment **31**(7) (1997) 991–1002
14. Kirkpatrick, S., Jr., D.G., Vecchi, M.P.: Optimization by simmulated annealing. science **220**(4598) (1983) 671–680
15. Kirkpatrick, S.: Optimization by simulated annealing: Quantitative studies. Journal of statistical physics **34**(5-6) (1984) 975–986
16. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. The journal of chemical physics **21** (1953) 1087
17. Gilks, W.R., Richardson, S., Spiegelhalter, D.J.: Markov chain Monte Carlo in practice. Volume 2. CRC press (1996)
18. Marques, L., Nunes, U., de Almeida, A.: Cooperative odour field exploration with genetic algorithms. In: Proc. 5th Portuguese Conf. on Automatic Control (CONTROLO 2002). (2002) 138–143
19. Marques, L., Nunes, U., De Almeida, A.: Odour searching with autonomous mobile robots: An evolutionary-based approach. In: Proceedings of the IEEE Int. Conf. on Advanced Robotics. (2003) 494–500
20. Marques, L., Nunes, U., De Almeida, A.: Particle swarm-based olfactory guided search. Autonomous Robots **20**(3) (2006) 277–287
21. Turduev, M., Cabrita, G., Kırtay, M., Gazi, V., Marques, L.: Experimental studies on chemical concentration map building by a multi-robot system using bio-inspired algorithms. Journal of Autonomous Agents and Multi-Agent Systems (2013)
22. Lochmatter, T., Roduit, P., Cianci, C., Correll, N., Jacot, J., Martinoli, A.: Swistrack-a flexible open source tracking software for multi-agent systems. In: Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, IEEE (2008) 4004–4010
23. Hawkins, D.M., Cressie, N.: Robust kriging - a proposal. Journal of the International Association for Mathematical Geology **16**(1) (1984) 3–18