Olfactory Search in Turbulent Flows

Antony Pearson, Jonathan Kroc, Alexia Tatem, and Andrew Burns

Introduction:

On small scales (the scale of bacteria or cells, for instance), a diffusing molecule creates a uniform gradient that can be followed by a seeker to locate the source. Such a method of locating the source of a molecule is called chemotaxis, and it simply involves the seeker moving along the path of the greatest increase in concentration. On larger scales (the scale of moths, butterflies, bees, or microrobots), the fluid environment is air rather than the liquid environment in which bacteria and cells operate. As a result, the Reynolds number becomes large. The Reynolds number is defined $Re = \rho vL/\mu$, where ρ is the density of fluid, v is the velocity of the particles in the flow, L is the characteristic length of the system, and μ is the viscosity of the fluid. In effect, this means that the trajectories of molecules carried in the flow can be perturbed and thus, chemotaxis would be an ineffective method for locating a source in such a turbulent environment. Balkovsky and Shraiman outline “a more complex strategy involving, in addition to the sense of smell, the ability to determine wind direction.” A strategy for locating the source of an odor in such turbulent flows would be useful for the design of small smelling robots used to find small gas leaks or explosives.

Methods:

Balkovsky and Shraiman proposed a discrete model of a turbulent plume. Instead of equations that describe a diffusion of particles superimposed on a wind velocity, consider an xy-plane with the source located at the origin. The mean wind velocity, V, we will consider to be in one direction (positive y direction) and constant in magnitude for a long time. Each time step, 3 things happen: (1) a new particle is released from the source, (2) each particle is advected by the mean velocity and moves 1 unit in the +y direction, and (3) each particle moves either -1, 0, or 1 in the +x direction, seen in figure one.

$$D = \frac{p_R + p_L}{2}$$

eq. 1

$$p(\mathbf{r}) = \frac{1}{\sqrt{4\pi D y}} \exp\left[-\frac{x^2}{4Dy}\right]$$

eq. 2

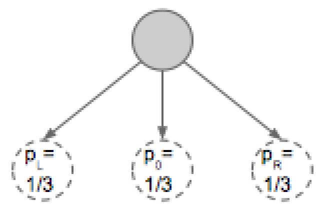


fig. 1

At scale lengths larger than L, the motion is Brownian, as the x-direction motion reflects the diffusion of particles. After this system has been allowed to fully develop (t>>1, with particles extending y>>1), the distribution of the particles through stochastic simulation closely matches equation 1 where equation 2 is the diffusivity coefficient. The probability distribution function given here is an analytical solution to the diffusion equation, suggesting that the discrete model is adequate to describe the mechanics of the turbulent flow. Notice also that in the fully developed plume (y>>1), at fixed y the probability distribution for a particle with respect to x-position is Gaussian.

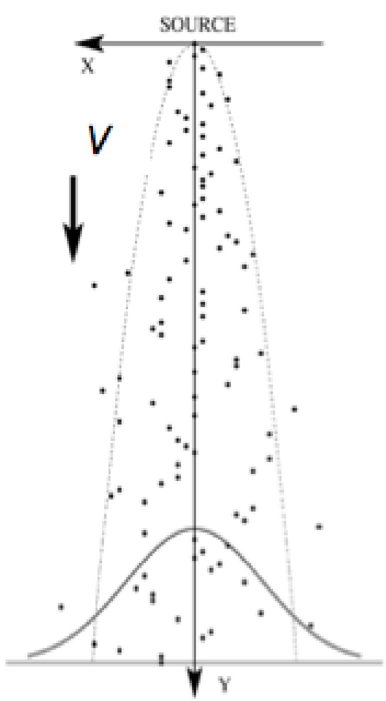


fig. 2

Consider a robot or moth located at a distance y_0 from the source. The moth can detect both the event of an odor patch arriving at its current location and the direction from which the odor patch arrived. Each time step the robot is able to move at most one lattice step along the y-axis and/or one step along the x-axis. Finally, the robot does not begin its search until it initially encounters a patch.

If (x_0, y_0) is the source of the odor patch one time step ago, the source can only be located in the interior of the cone formed by:

$$y - y_0 = \pm (x - x_0),\ y < y_0$$

This is known as the causality cone.

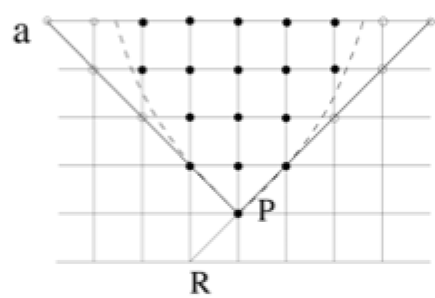


fig. 3

Multiple search algorithms are introduced by Balkovsky and Shraiman, and they are analyzed by the time it took to find the source- or the mean search time. Due to the random nature of the plume, the search time is a random quantity. Moreover, the researchers evaluated the three algorithms wand plotted the probability that the source is found during a *t*, *t*+1 interval as a function of time, ρ(t). Algorithms with means closer to zero were deemed more effective.

The first search algorithm introduced is the passive search. The robot waits at one site until it detects an odor patch. When such a patch impinges upon the robot, the robot moves along the lattice to the site from which the patch came. Over a sufficiently long period of time, this method guarantees that the robot will find the source. A major shortcoming with this method is its inefficient use of search time, a problem of particular concern in cases where the robot begins its search in regions of low odor patch density, i.e. away from the center of the plume along the y-axis. Accordingly, the authors pursued search algorithms that made better use of search time by actively seeking odor patches.

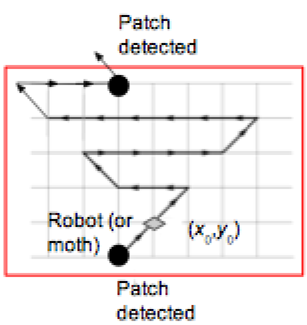


fig. 4

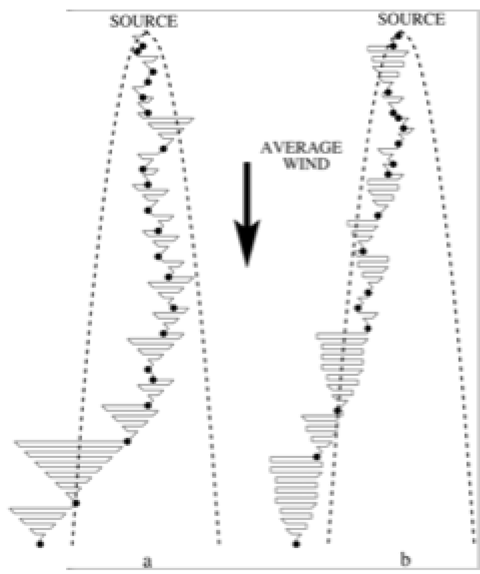


fig. 5

The first active search algorithm introduced was the conical search algorithm. With this algorithm, the robot, positioned downwind from the odor source, waits until it detects an odor patch. When an odor patch is detected, the robot moves upwind to the position the odor patch once occupied and begins its a conical search, moving along a counter-turning path, as seen in Figures 4 and 5, so as to completely cover the detected odor patch's possible previous locations. The conical search, unlike the passive search, actively seeks out odor patches. Moreover, because its path covers the odor particles causality cone, it is certain to find the odor source. Still, by covering so much area, the conical search method needlessly spends time in areas of the causality cone that are unlikely to have been the previous location of the odor particle. Accordingly, the authors next present a modified version of this conical search algorithm called the "parabolic search algorithm." Whereas the conical search algorithm visited the upper "corners" of the causality cone, this new algorithm passes them over, as shown previously in Figure 3. By eliminating locations with probabilities in the tails of the distribution, the search time is thereby greatly reduced.

Results:

To assess the algorithms, a probability distribution function of the search time is used and efficiency is defined with respect to mean search time. With respect to this definition, the parabolic search algorithm is the most effective.

The histograms illustrated below, obtained via Monte Carlo simulations, model the search time. Figure 6(a) shows the robot with an initial position at (0, 50), while in figure 6(b) it is initially at (10, 50). The mean search time of the active algorithm, represented by the broken line, is not affected by adjustment of the initial position. However, the passive algorithm, represented by the solid line, is.

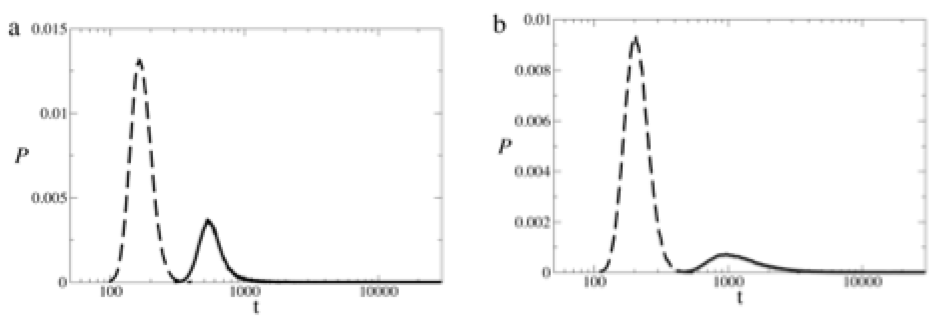


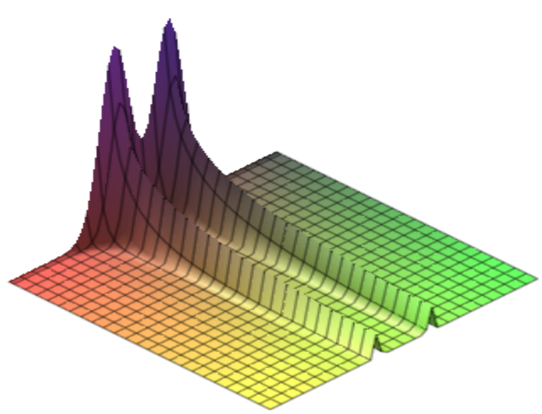
fig. 6

Future Directions:

When contemplating future work regarding this olfactory search algorithm, we considered a number of directions. One idea was to introduce a group of entities searching for a source, rather than a single seeker. Then, when one robot encounters an odor patch, the others change the angle of their search pattern. Another option considered was to create a design that altered the algorithm based upon the time between detection of the odor patches. The idea is to cover more ground where the probability of detecting an odor patch is low, and explore less area as odor patches are detected with more frequency. Our final consideration was to add a second odor source to the model, and analyze the efficiency of the parabolic search algorithm with respect to multiple sources. We will focus our research on the model containing multiple sources.

Our rationale for considering multiple sources is the observation that it is a common natural occurrence for multiple sources to be influenced in the same flow. (Consider a bee trying to land on a flower contained in a row of flowers). It seems intuitive that the addition of the probability density function for two sources will create distributions at high y-value that would appear to originate from a single source in-between the two actual sources.

The probability distribution under a two-source scenario where the sources are located at (-1,0) and (1,0) can be given by:



To evaluate the effectiveness of the parabolic search algorithm, we will make the two sources symmetric about the y-axis. The separation of the two sources, x_0, and y_0 will be fixed to as-yet undetermined values. We will then conduct Monte Carlo simulations of the parabolic search algorithm for the two-source simulation and collect search times (simulations where the robot misses all sources will be excluded from these counts) until our histogram reaches a minimum smoothness predetermined by our Python histogram package. We will remove one of the sources and repeat the simulations. We hypothesize that the typical search time for the two-source situation will be higher due to “confusion” of the robot in response to the two sources. Additionally, we expect to see a flatter histogram due to the two-source simulations increasing variability in search time. We will also compare the proportion of “misses”, i.e., the robot failing to find a source, between the single-source and two-source situations. In the natural world, complete failure to find a source could be catastrophic, so the inability of the parabolic search algorithm to adapt to multiple sources, a common real-world occurrence, would be a major limitation for the algorithm’s use in olfactory robots.

References:

1. Balkovsky E. and Shraiman, B.I. “Olfactory search at high Reynolds number”, Proc Natl Acad Sci USA.99,12589-93 (2002).

2. Torney, C., Neufeld, Z., and Couzin, I.D. Context-dependent interaction leads to emergent search behavior in social aggregates. PNAS 2009 106 (52) 22055-22060; published ahead of print December 14, 2009.