Always Follow Your (Dog's) Nose: Searching for an Optimal Strategy for Tracking a Moving Target by Airborne Odor Trail

Ian Smith
CSCI 4314
University of Colorado
Boulder, Colorado
Ian.C.Smith@colorado.edu

Curtis Lin
CSCI 4314
University of Colorado
Boulder, Colorado
Curtis.Lin@colorado.edu

***Abstract*—Strategies (in dogs) for finding a moving target using only olfactory information have not been closely examined. While previous studies have discovered optimal strategies for finding a hidden non-moving target by odor trails, we simulate an airborne odor plume of a moving target. We discover a successful intermittent strategy composed of correlated random walks, a form of momentum, and chemotaxis for finding such a target. This discovered strategy was largely successful on our modeled environments and the global behavior of the agent following this strategy was similar to the observed behavior of real-life odor-tracking species, like rats and dogs.**

***Index Terms*—dog, search, agent, odor, tracking, crw, momentum, diffusion, random walks, chemotaxis, intermittent**

I. Introduction

Dogs are very good at tracking prey over long distances (many miles) and they do so by sensing a combination of airborne and depositional odorants [1]. Airborne odorants diffuse in the air as animals move through an environment and are often modeled according to Gaussian dispersion models and are easily affected by turbulence. Depositional odorants on the other hand are left behind on the ground as prey (or search and rescue targets) travel through a landscape. These depositional odorants can be resuspended into the air and detected when disturbed (by wind or other animals). In tracking prey (like rabbits) or searching for humans (in the case of search and rescue missions), dogs rely on both of these odorants [2].

Little is known about the navigational strategies that dogs use in these moving target scenarios, so the goal of this project is to gain some insight into how dogs perform so well at this task. In this project, we want to find an optimal strategy for tracking a moving target, prey or human (to be rescued), where the only informational cues are airborne odorants. For the purpose of this paper, we will disregard depositional odorants as they are harder to simulate and are dependent on traits of the target/prey because they are subject to variations in shedding patterns and other biophysical features. The strategies considered will be motivated by what is known about dog olfactory tracking (observed in experiment and anecdotally) and our strategies will be evaluated in simulations.

II. Background and Related Work

There has been a good amount of research into the optimal search strategies of animals for finding a hidden target with an odor plume [3]. These strategies often involve two phases: a slow-moving phase where the agent sweeps a small area very intensely (and accurately) and a faster phase where the agent quickly moves to a new location.

There has also been a great amount of research into pursuit and evasion strategies in predator-prey interactions. There is for example extensive research into optimal attack trajectories of dragonflies or even Cheetah hunting strategies. Previous research has even looked at a pursuit predation technique called motion camouflage, which involves predators matching prey movement so that they appear to be non-moving (i.e. relative velocity of zero) [4], [5].

In this paper, we are interested in considering a relatively unstudied predation behavior: tracking prey by odor. This behavior is sometimes exploited in dogs for search and rescue purposes. This behavior is similar to finding a hidden target with the added twist that the target is moving. Previous research has found that during olfactory tracking, dogs have three distinct phases: (1) an initial phase of track discovery characterized by fast movement in varied directions with short sniffing periods (in which they slow down), (2) a second, slower-moving phase to determine the direction of the track, (3) a third phase of tracing the track of the scent where the dog alternates between quick movement along the track and slow-moving sniffing periods (often near footprints) to stay on track [6].

Previous research has also found that an intermittent strategy where a relatively small amount of time is given to searching compared to relocating is very effective at finding a (non-moving) hidden target [7]. Similar intermittent strategies have also more recently been investigated for the non-moving hidden prey problem. In a 2010 paper titled "*Optimal intermittent search strategies: smelling the prey*", researchers used an intermittent strategy and optimized the rate of switching between the two search states (slow and compact vs fast and large) [8].

Other research has found an optimal strategy for initially finding the odor plume in turbulent conditions called Lévy-Taxis, which combines Lévy walks with a correlated random walk [9].

In this paper, we will combine and attempt to implement a few of these techniques in building a model agent to find a moving target. The researchers implement a chemotaxis agent, a random walk agent, and an composite strategy that involves components of random walks and components of chemotaxis.

III. Methodology

In order to determine an 'optimal' strategy for finding a moving target, we decided to use a simulated model. Our model was based on some initial diffusion navigation code provided to our team by Dr. Orit Peleg. This original code was implemented in MATLAB and our team refactored this code in Python, adding some object-oriented design that would make swapping out agent strategies and target paths a little easier. We also implemented from scratch all of our strategies and evaluation methods and changed Dr. Peleg's diffusion equation to the instantaneous point source diffusion equation, seen in (1), (2), and (3).

Our model consisted of two main parts: a moving source/target (emitting some diffusing odor) in an environment and an agent/dog following one of many strategies for finding the target. Both the agent and environment were updated once per time step and the simulation ended at `endtime` or whenever the agent was within 5 units of the target.

A. Target and Environment

To keep our model environment easier to implement, we modeled diffusion using the analytic solution to an instantaneous point source release in two-dimensions, which we derive from Frick's Law and a differential equation for a point source diffusion [10]. We then have that the concentration, C, at any point (x, y) in our environment at time t after release, where the point mass of airborne odorant was released at $(0, 0)$ and $t = 0$ is given by

$$C(x, y, t) = \frac{A}{4\pi t D} \exp\left(-\frac{x^2 + y^2}{4Dt}\right) \tag{1}$$

for initial concentration of odorant A and diffusion rate D. We assume that diffusion rate in x and y directions is the same.

We model our environment by a 2-D grid of coordinates, where our target/source is tracing out a curve over some finite number of time steps. For our simulation, the target/source's path is defined by a finite number of sources, n. The path of the target is defined by n equally spaced point sources along a curve in the 2-D grid. The i^{th} point source, located at (x_i, y_i) $(i \in \{1, \ldots, n\})$, is released/activated at time t_i. At each time step, we compute the concentration of odorant at every square on our 2-D grid **for each** activated point source (this is computing partial concentrations). Then, we sum the contributions of each of the activated point sources to determine the total concentration at all points on the grid at a given time. At the i^{th} point source's origin, we also have some amount of wind in the x and y directions, $v_{i,x}$ and $v_{i,y}$. Thus, for $1 \leq k \leq n$ the concentration at time $t = t_k \leq t_n$ at point (x, y) is given by

$$C(x, y, t) = \sum_{i=1}^{k-1} \frac{A}{4\pi(t_k - t_i)D} \exp\left(-\frac{\tilde{x}^2 + \tilde{y}^2}{4D(t_k - t_i)}\right) \tag{2}$$

with $\tilde{x} = x - x_i - v_{i,x}(t_k - t_i)$ and $\tilde{y} = y - y_i - v_{i,y}(t_k - t_i)$, where t_i is when the i^{th} source was activated, t_k is the current time step and (x_i, y_i) is the location of the i^{th} source. When $t > t_n$, there are no newly activated sources, but the odorants continue to diffuse through the environment and we compute the concentration at (x, y) at time t with

$$C(x, y, t) = \sum_{i=1}^{n} \frac{A}{4\pi(t - t_i)D} \exp\left(-\frac{\tilde{x}^2 + \tilde{y}^2}{4D(t - t_i)}\right) \tag{3}$$

where $\tilde{x}$ and $\tilde{y}$ have the same meanings as (2).

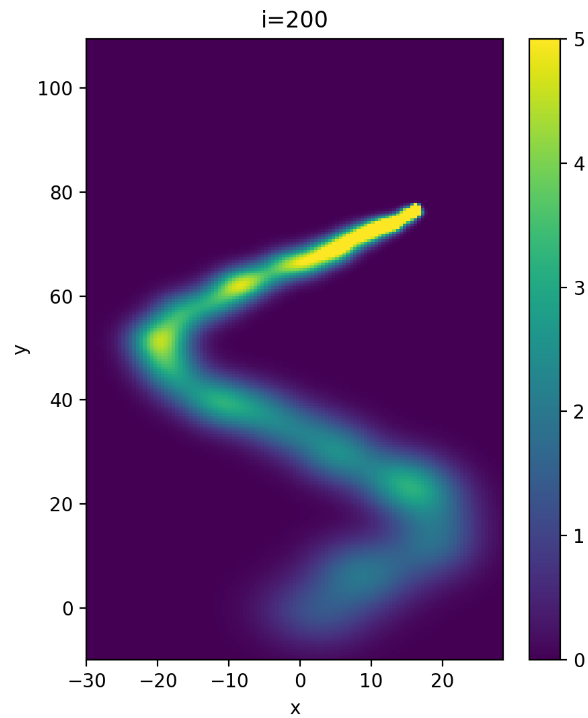

Fig. 1. Example of our diffusion model for target with sinusoidal path. In this frame of the simulation, 200 of the 250 point sources have been activated at the end of 200 time steps. This plot shows the concentration of odorant throughout our 2-D grid environment. We can see pockets where the odorant is pooled; this is due to the effects of wind in the simulation.

The movement of the target is defined by the placement of the n point sources. For the purposes of this paper we set $n = 250$ and we arrange the point sources along a sinusoidal curve and the target traces out the shape of this curve over $n = 250$ time steps. We also used $A = 15$ and $D = 0.1$ for the initial concentration at instantaneous release and diffusion rate, respectively. The resulting simulation is exemplified in Fig. 1. Once the final source is activated, the simulation continues and odorant diffuses throughout the rest of the grid but no new sources are activated and no new odorant is released into the environment.

B. Agent and Strategies

In our simulation, we also had to model our dog/agent. The goal of the agent is to “find” the target. In our model, the target was considered found and the simulation ended whenever the agent was 4 or less units away from the target. On a 30x160 grid, this meant that the “endzone” for the agent was only 0.5% of the area of the entire board. The agent is introduced to the environment 40 time steps after the target is moving. The agent starts at the base of the target’s path. At each time step, the agent employs one of four strategies: correlated random walk, biased random walk, chemotaxis, or our custom “chemomentum” strategy, which adds a form of momentum to the chemotaxis strategy.

1) Correlated Random Walk (CRW): In this strategy, the agent determined its next move by choosing an angle from the uniform distribution $[-\sigma, \sigma]$ and turning that angle (relative to the previous direction of the agent). The agent then took a step of size v_{crw} in this new direction. For our paper, we kept $\sigma = \pi/16$ constant.

2) Bias Random Walk (BRW): In this strategy, the agent determined its next move in a similar fashion to the CRW strategy, the only difference was that the new angle was measured relative to a fixed direction $\theta_{biased} = \pi/2$. The same constant $\sigma = \pi/16$ was used as in the CRW strategy. This strategy was not based in actual behavior of dogs (as dogs would not have a sense of which biased direction to use), but was employed in the exploratory phase of our project.

3) Chemotaxis: For the chemotaxis strategy, the concentration gradient at all points on the grid is computed using Numpy’s second-order accurate methods. The agent then takes a step at speed $v_{chemotaxis}$ in the direction of the gradient. Essentially, the agent using this strategy will move up the concentration gradient at every step.

4) Chemomentum: In this custom strategy, designed after implementation of the model (not ahead of time, like the other three strategies), the agent uses chemotaxis to determine the direction of increasing concentration, but instead of moving in the direction of increasing concentration, the agent takes a weighted average of this gradient direction and the agent’s previous moves. The agent has a memory of size m (an integer). The agent “remembers” the x and y directions of its last m moves in arrays $\texttt{mem}_x$ and $\texttt{mem}_y$, where the most recent directions are appended to the front of the arrays and the oldest directions are popped off the end of the arrays as the memory limit m is reached. After the current direction of increasing concentration is added to the agent’s memory, the direction of the agent’s next step is determined by

$$\vec{v}_{new} = \begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix} = \frac{1}{m} \sum_{i=0}^{m} \begin{bmatrix} \alpha^i \cdot \texttt{mem}_x[i] \\ \alpha^i \cdot \texttt{mem}_y[i] \end{bmatrix} \tag{4}$$

where α is the discount rate, so that previous directions further in the agent’s past contribute less to the agent’s current decision. For the purpose of this project we leave $\alpha = 0.9$ constant, which was decided after quick paramter sweeps

This strategy combines chemotaxis (in determining directions), but also adds a form of momentum because the agent will not ‘turn on a dime’. Instead, the agent will tend to continue moving in the directions it has been moving. The idea is that the agent will not stop local maxima and continue along the target’s path.

The simulation also included an optional parameter to add make the agent “turnaround” if the concentration dropped below a certain threshold. With this turnaround feature, the agent changes its direction by 180 degrees and performs a step of correlated random walk if the odor concentration drops below a certain threshold (in our paper, this threshold was 0.01). We can motivate this theory by observing that real dogs that lose the trail (odor concentration below there ability to smell) will turn around in an attempt to find the trail again.

C. Evaluating Our Agent’s Strategies

In order to compare strategies for the agent, we computed the success rate of different strategies. The success rate, in this context, is defined as the percent of runs where the agent found the target.

Each strategy was tested with 25 trials of 800 timesteps. In most cases, we did not vary the path of the target, but the diffusion simulation would change based on randomness in the wind. In all model runs, the agent’s strategy consisted of a combination of two of the four strategies. Combined strategies meant that there was a probability p of using the first strategy and probability $1 - p$ of using the second strategy. In our project, optimal parameters were determined through a parameter sweep over this p value, for two-strategy combinations of (CRW, Chemotaxis), (BRW, Chemotaxis) and (CRW, Chemomentum).

IV. Results and Analysis

With our model implemented in Python, we were able to begin the next phase of the project: determining an optimal strategy for the agent to find the target. We quickly found that in our model (before wind was added), the chemotaxis strategy alone was sufficient for the agent to consistently find the target (see Supplementary Video 1). After adding some wind the the model (more realistic), we found that chemotaxis alone would quickly lead to the agent getting stuck at local maxima for concentration (see Fig. 2 and Supplementary Video 2).

We then decided that an intermittent strategy (inspired by strategies that worked for a non-moving hidden target) that switches between some form of random walk and chemotaxis would be a start to finding an optimal search strategy.

We begin with a two-phase strategy of CRW and chemotaxis. We perform a paramter sweep over p(crw) = 1 - p(chemotaxis), to determine the optimal ratio for an agent using these two strategies. For this paper, we assume that the agent can move 150% faster when performing a CRW step compared to a step of chemotaxis, this is motivated by experimental observations where dogs move slower in

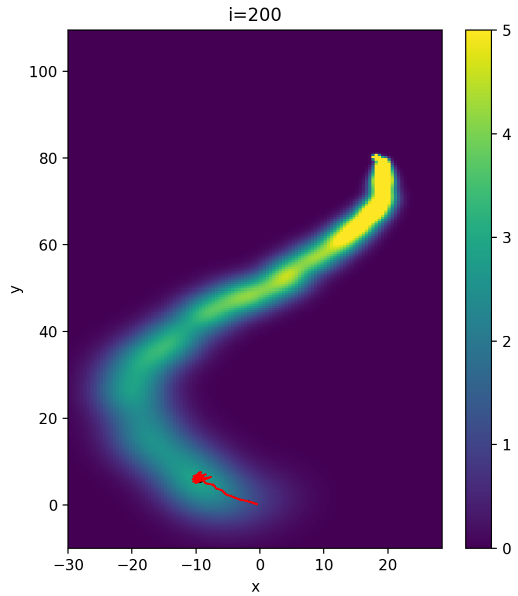

Fig. 2. An example of our agent getting stuck with a chemotaxis/crw strategy with no momentum (p(crw) = 0.1 and p(chemotaxis) = 0.9). The agent finds a local maxima, but due to wind, it will not progress any further and will never find the target.

the intense search phases [6]. Fig. 3 shows that there is no convergence for an intermittent strategy of a CRW combined with chemotaxis assuming speed ratio of 1.5:1. However, we do note that the agent has a 20% chance of randomly finding the target when moving close to 100% CRW, which is due the nature of our agent initially having the same general direction as the sources end location. If it was tasked with finding sources that had end locations that had no relation to the initial positioning of the agent, the success rate would be much lower.

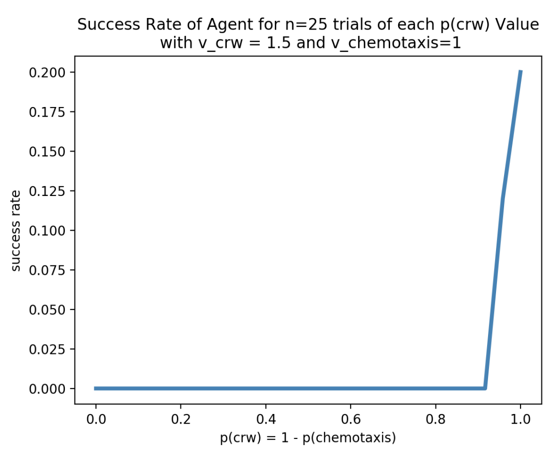

Fig. 3. Parameter sweep over p(crw) = 1 - p(chemotaxis). We see that combinations of these two strategies do not work (for speed ratios of 1.5:1 nor any speed ratios that we tried) and we only have small (random) success when the agent is moving completely randomly. Trials end after the target is found or after 800 timesteps, whichever comes first.

After failure of CRW combined with chemotaxis, we decided to test another two-phase strategy that employed a BRW combined with chemotaxis. We found that this random walk strategy did work, with our biased direction set to North and a BRW:chemotaxis speed ratio of 6:1 and p(brw) = 1-p(chemotaxis) = 0.1 (see Supplementary Video 3). However, we were unable to motivate this method with actual dog behavior (they would not have a sense of where the target will end up), but it provided a strategy for escaping local maxima that inspired our custom strategy of chemomentum (III-B4).

When our 'chemomentum' strategy was first implemented, we began with a 'memory' for the agent of 10 steps. This means that the agent took into account its previous 10 steps when determining its next move in a step of 'chemomentum'. This memory size led to a successful find by the agent when performing chemotaxis 60% of the time and CRW 40% (see Supplementary Video 4). However, this choice of memory size was arbitrary and we decided to perform a paramter sweep over memory size. Fig. 4 shows our test of the success rate of the agent in relation to the number of previous steps it was able to "remember" (with a discount of 0.9 at each step). The agent generally had a higher success rate when it was able to "remember" more steps, and in our final agent strategy we ended up using all of the agents previous moves.

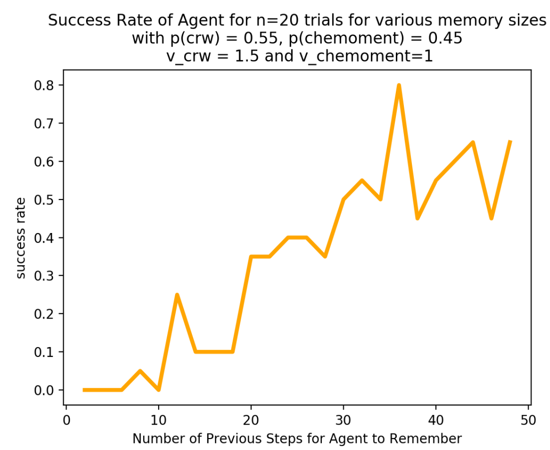

Fig. 4. Agent using 'chemomentum' tends to do better when it considers its more of its previous moves (discounted by 0.9 at each step) – for our model we ended up using all of agents previous moves (i.e. remember everything, with 0.9 discount rate).

Finally discovering a strategy that was consistently finding the target (30% of the time in Supplementary Video 4), we decided to optimize the proportion of time that the agent in our two-phase strategy spent in a CRW compared to perfomring 'chemomentum'.

Fig. 5 and Fig. 6 show parameter sweeps over p(crw) = 1 - p(chemomentum). Fig. 5 shows a sweep over p(crw) $\in [0, 1]$ (probability of the agent using a CRW) in an intermittent strategy with momentum-driven chemotaxis, chemomentum. After narrowing down the optimal range, Fig. 6 shows a sweep for p(crw) $\in [0.35, 0.65]$. The success rate of our agent peaks for a strategy that is 55% CRW and 45% chemomentum. The sweep over $p(crw)$ yields a max 85% success rate, justifying our choice of this parameter.

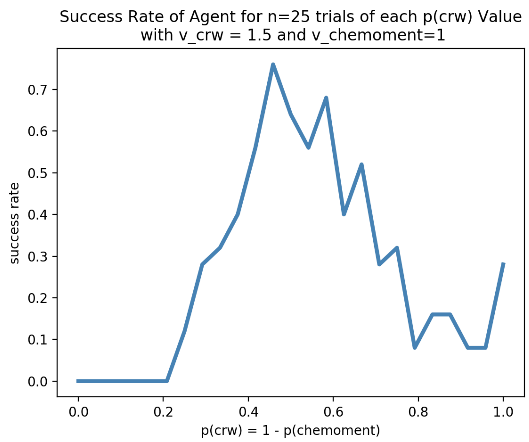

Fig. 5. Parameter sweep for two-phase strategy of CRW and Chemomentum that searches for optimal ratio between these two phases. We see that when CRW is 100% of the time, we get 30% chance of success, this is just due to nature of of CRW having a chance to wander to the target after 800 time steps.

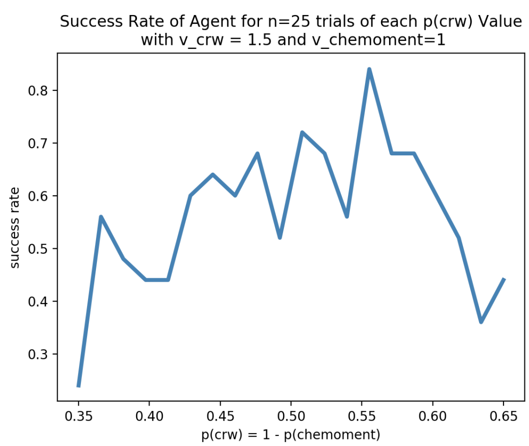

Fig. 6. Parameter sweep for two-phase strategy of CRW and Chemomentum that searches for optimal ratio between these two phases finds optimal probabilities of strategies: p(crw) = 1-p(chemomentum) = 0.55.

In observing some of the failed searches for our optimized CRW-Chemomentum strategy, we notice that the agent will often fail if it wanders very far from the trail in a run of CRW steps. To mitigate this, we add the 'turnaround' strategy, described in (III-B4), that forces the agent to turn $180°$ and take a single step of CRW in the event that concentration of odor drops deblow 0.01. This turnaround feature even allowed the agent to navigate a more complicated sinusoidal path, which completes an extra half period (see Supplementary Video 5). Fig. 7 shows a paramater sweep over p(crw) = 1 - p(chemomentum) with this added turnaround feature, and we find the optimal p(crw) = 1 - p(chemomentum) is again 0.55, but we now achieve a success rate of 93%.

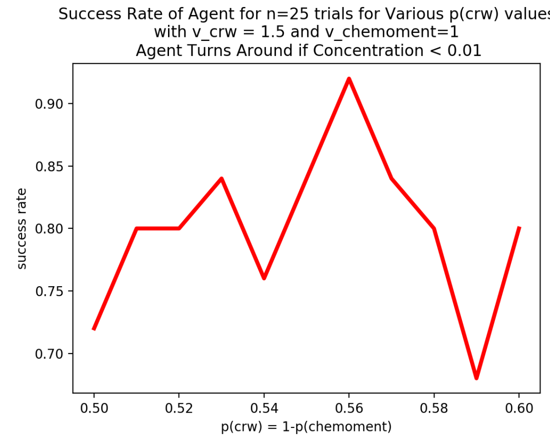

Fig. 7. Parameter sweep for two-phase strategy of CRW and Chemomentum WITH 'turnaround' strategy added. Finds optimal probabilities of strategies: p(crw) = 1-p(chemomentum) = 0.55.

The final figure (Fig. 8) demonstrates our agent finding the source target after 318 time steps with our intermittent strategy of p(crw)=0.55 combined with the momentum-based chemotaxis strategy. We found a casting behavior, which is also seen in rats and humans [11], [12]. This behavior is found throughout nature, bees moths, cockroaches. While this behavior might be a behavior that animals use to increase their chances of re-encountering a lost trail, the fact that a similar behavior emerged from a momentum, chemotaxis, CRW strategy suggests that animals could be using some form of momentum (either memory or physical momentum) in order to not get stuck in these local maxima when following odor that is subject to wind turbulence.

V. Challenges

In an environment with no wind, our agent could use chemotaxis 100% of the time with a high success rate, but the strategy would fail with the introduction of wind. In situations where our agent found a local maxima, our agent would get "stuck" and stay on top of the local maxima. To get "unstuck" we introduced a correlated random walk into the agents strategy. Our intermittent strategy initially involved a randomly distributed combination of CRW and chemotaxis, with varying percentages of each strategy. Unfortunately, we were never able to find a combination of CRW and chemotaxis strategies that would lead our agent to the source.

In another attempt to help our agent maintain its path, we tried using a biased random walk with a very large step size (step size of 6) to get the agent to "jump" out of the local maxima. While we found that an agent using this strategy would find its target, we found it hard to justify using this strategy because of the many assumptions that were needed. First, using a biased random walk would require us to have some knowledge about the end location/general direction of the agent, which is something that is not always known in many scenarios. Although one of our references suggested making "short local searches" and to "relocate keeping a main

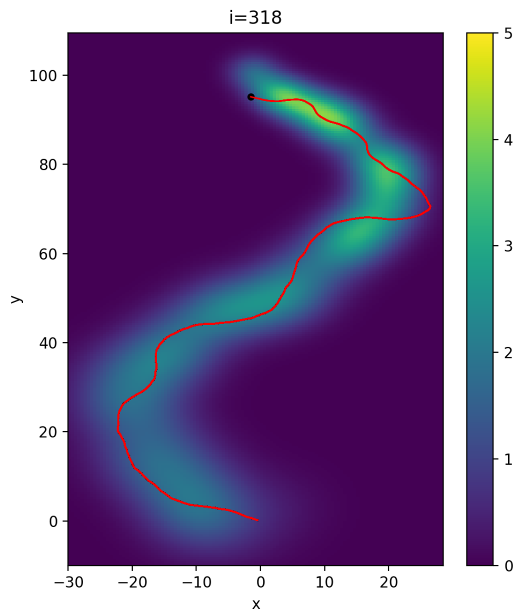

Fig. 8. Agent with two-phase CRW-Chemomentum Strategy with Turnaround finding the target. We can observe a casting behavior that is typical of animals like rats in similar search problems.

direction" with an intermittent strategy, it was hard to motivate this type of behavior in a dog [7]. Secondly, a dog would not necessarily be able to "know" this bias without some form of additional guidance, such as a human companion with knowledge of the target's location. We concluded that the addition of a biased random walk was not in the scope of finding a target with olfactory information and therefore wouldn't be suitable for our project's goal.

Even our final momentum strategy can still fail if the target is upwind and its odor is blown away (and pools at a global maxima away from the target). This is probably results from the target/source no longer emitting odor after it reaches the endpoint, when in reality wind would probably continue to blow some odor deposition from the target to this global pool (maybe leaving a trail for the agent to follow).

Original odor diffusion model, given in our starter code, was not based in any easily interpretable equations, so we had to refactor the diffusion model to the instantaneous point source equations from methodology (see link to GitHub repo in Supplementary Materials). In refactoring for instantaneous point source model, we had to run many simulations experimenting with A (initial concentration), D (diffusion rate), dt (time step), and n (# of instantaneous point sources along path) to get a diffusion model that looked good (the odor hung out long enough for it to at least be possible for dog to find it, given its walking speed). In refactoring to Python, `numpy` often took (y,x) as opposed to MATLAB equivalents taking (y,x) as parameter orders, which led to some simulation that were hard to debug. Also, the original code took 5 minutes for each run of the simulation and so we spent some time refactoring the code that we could quickly run multiple agent simulations (15 seconds per run) on the same odor-diffusion simulation (which take 5 minutes to run each time), this allowed for us to perform parameter sweeps over 100s of trials in a few minutes as opposed to hours.

Another, limitation of our paper is that our simulation had many parameters in both the source propagation portion of the simulation related to diffusion and in the agent strategy portion related to speeds of strategies. In our paper, we fixed speeds of the strategies except in very early manual testing and we kept the diffusion model parameters fixed.

VI. Future Directions

The next step of research for this topic would be to test our agent using our model of CRW and momentum-driven chemotaxis on different source paths as we only tested two source paths in our project. In all the tests, our source agent moved in a sinusoidal manner with a fixed amount of wind. Our agent was also given favorable start conditions by being placed directly on the path of the source agent. Future work could test the agent against different source paths, vary the amount of wind used in the stimulation, and also start the search agent in different locations away from the path of the source agent. Our two-dimensional model also does not account for the odorant behavior that occurs in three-dimensions. More advanced models could take into account that the odorants can move above the tracking agent due to the wind, and varying odorant heights could affect the ability of the dog to use chemotaxis.

A different direction for future work could be the implementation of multiple tracking agents working in collaboration to find a source target. This would be similar to the real-world behavior of wolves hunting prey, as well as search and rescue teams using multiple dogs in their missions [13]. Multiple collaborative agents, with varying degrees of informed individuals and loose flocking behavior, might gauge odor from multiple locations and angles to perform the search task better than a single agent.

VII. Project Contributions

Ian refactored Dr. Peleg's existing simulation code to Python, added some object-oriented coding principles that made the diffusion model easier to modify and agent strategies easier to swap out. Ian implemented the chemomentum strategy, which was designed by both Ian and Curtis. He designed and performed the parameter sweeps and all plots and videos related to the simulation and parameter sweeps. In the report, Ian wrote the results section and analysis.

Curtis helped to create the correlated random walk strategy and the biased random walk strategy based on Orit Peleg's code. Curtis also helped Ian in designing the project. In the report, he wrote the discussions and challenges section and helped Ian in various other parts of the paper.

Both Ian and Curtis designed the structure and scope of the project. For the presentation, Ian and Curtis worked side-by-side to write and practice the presentation. Both worked on the design of our custom chemomentum strategy. In the writeup, they both wrote the abstract, background and introduction.

Supplementary Materials

The following YouTube Videos are visualizations from our project:

1) Successful Agent (Chemotaxis with No Wind): https://youtu.be/KQjxEz7OIS8
2) Agent Stuck in Local Maxima (Pure Chemotaxis in Wind): https://youtu.be/K9MXzjsMs0o
3) Successful Agent (BRW + Chemotaxis): https://youtu.be/z6Jiny5wdag
4) Example of 'Chemomentum' Allowing Agent to Escape Local Maxima https://youtu.be/TsQtSzmRMHk
5) Example of Agent Tracking More Complex Target Path. Agent turnsaround when concentration drops below 0.01. https://youtu.be/fI6zQX5HDT4
6) Successful Run of Final Model with 'Chemomentum' Strategy (Chemomentum + CRW): https://youtu.be/b6kb-MZIhnY

The code for this project is available at https://github.com/reusablebuffalo/csci_4314_project.

References

[1] M. R. Conover, "Detecting and locating prey through depositional odor trails," in *Predator-Prey Dynamics : The Role of Olfaction*, CRC Press, Mar. 30, 2007, pp. 23–34, ISBN: 978-1-4200-0912-5.

[2] ——, "Using airborne odorants to detect the presence of prey," in *Predator-Prey Dynamics : The Role of Olfaction*, CRC Press, Mar. 30, 2007, pp. 35–54, ISBN: 978-1-4200-0912-5.

[3] O. Bénichou, M. Coppey, M. Moreau, P.-H. Suet, and R. Voituriez, "Optimal search strategies for hidden targets," *Physical Review Letters*, vol. 94, no. 19, May 16, 2005.

[4] A. M. Wilson, J. C. Lowe, K. Roskilly, P. E. Hudson, K. A. Golabek, and J. W. McNutt, "Locomotion dynamics of hunting in wild cheetahs," *Nature*, vol. 498, no. 7453, pp. 185–189, Jun. 2013, ISSN: 1476-4687. DOI: 10.1038/nature12295. [Online]. Available: https://www.nature.com/articles/nature12295 (visited on 03/20/2019).

[5] *Motion camouflage*, in *Wikipedia*, Page Version ID: 884529022, Feb. 22, 2019. (visited on 03/20/2019).

[6] A. Thesen, J. B. Steen, and K. B. Doving, "Behaviour of dogs during olfactory tracking," *Journal of Experimental Biology*, vol. 180, no. 1, pp. 247–251, Jul. 1, 1993.

[7] O. Bénichou, M. Coppey, M. Moreau, and R. Voituriez, "Intermittent search strategies: When losing time becomes efficient," *EPL (Europhysics Letters)*, vol. 75, no. 2, Jun. 16, 2006.

[8] J. A. Revelli, F. Rojo, C. E. Budde, and H. S. Wio, "Optimal intermittent search strategies: Smelling the prey," *Journal of Physics A: Mathematical and Theoretical*, vol. 43, no. 19, p. 195 001, Apr. 2010. DOI: 10.1088/1751-8113/43/19/195001. [Online]. Available: https://doi.org/10.1088%2F1751-8113%2F43%2F19%2F195001 (visited on 03/20/2019).

[9] Z. Pasternak, F. Bartumeus, and F. W. Grasso, "Lévy-taxis: A novel search strategy for finding odor plumes in turbulent flow-dominated environments," *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 43, Oct. 2009.

[10] H. Nepf, "Lecture 3: Diffusion of an instantaneous point source release," in *Transport Processes in the Environment — MIT Course No. 1.061*, MIT OpenCourseWare, Cambridge, MA, 2008. [Online]. Available: https://ocw.mit.edu/courses/civil-and-environmental-engineering/1-061-transport-processes-in-the-environment-fall-2008/index.htm.

[11] A. G. Khan, M. Sarangi, and U. S. Bhalla, "Rats track odour trails accurately using a multi-layered strategy with near-optimal sampling," en, *Nature Communications*, vol. 3, p. 703, Feb. 2012, ISSN: 2041-1723. DOI: 10.1038/ncomms1712. [Online]. Available: https://www.nature.com/articles/ncomms1712 (visited on 05/02/2019).

[12] J. Porter, B. Craven, R. M. Khan, S.-J. Chang, I. Kang, B. Judkewitz, J. Volpe, G. Settles, and N. Sobel, "Mechanisms of scent-tracking in humans," en, *Nature Neuroscience*, vol. 10, no. 1, pp. 27–29, Jan. 2007, ISSN: 1546-1726. DOI: 10.1038/nn1819. [Online]. Available: https://www.nature.com/articles/nn1819 (visited on 05/02/2019).

[13] M. R. Conover, *Predator-prey dynamics : The role of olfaction*. CRC Press, Mar. 30, 2007, ISBN: 978-1-4200-0912-5.