

Q-learning with temporal memory to navigate turbulence

Marco Rando , Martin James, Alessandro Verri, Lorenzo Rosasco, Agnese Seminara 

MaLga, Department of Computer Science, Bioengineering, Robotics and Systems Engineering, University of Genova, Genova, Italy • MaLga, Department of Civil, Chemical and Environmental Engineering, University of Genoa, Genoa, Italy

 https://en.wikipedia.org/wiki/Open_access

 Copyright information

Reviewed Preprint

v2 • March 5, 2025

Revised by authors

Reviewed Preprint

v1 • December 16, 2024

eLife Assessment

This **important** study uses reinforcement learning to study how turbulent odor stimuli should be processed to yield successful navigation. The authors find that there is an optimal memory length over which an agent should ignore blanks in the odor to discriminate whether the agent is still inside the plume or outside of it, complementing recent studies using recurrent neural networks and finite state controllers to identify optimal strategies for navigating a turbulent plume. The strength of evidence is **compelling**, presenting a novel approach to understanding optimal representations for navigation in stochastic sensory environments.

<https://doi.org/10.7554/eLife.102906.2.sa3>

Abstract

We consider the problem of olfactory searches in a turbulent environment. We focus on agents that respond solely to odor stimuli, with no access to spatial perception nor prior information about the odor. We ask whether navigation to a target can be learned robustly within a sequential decision making framework. We develop a reinforcement learning algorithm using a small set of interpretable olfactory states and train it with realistic turbulent odor cues. By introducing a temporal memory, we demonstrate that two salient features of odor traces, discretized in few olfactory states, are sufficient to learn navigation in a realistic odor plume. Performance is dictated by the sparse nature of turbulent odors. An optimal memory exists which ignores blanks within the plume and activates a recovery strategy outside the plume. We obtain the best performance by letting agents learn their recovery strategy and show that it is mostly casting cross wind, similar to behavior observed in flying insects. The optimal strategy is robust to substantial changes in the odor plumes, suggesting minor parameter tuning may be sufficient to adapt to different environments.

Introduction

Bacterial cells localize the source of an attractive chemical even if they hold no spatial perception. They respond solely to temporal changes in chemical concentration and the result of their response is that they move toward attractive stimuli by climbing concentration gradients [Berg \(1975\)](#) [DOI](#). Larger organisms also routinely sense chemicals in their environment to localize or escape targets, but cannot follow chemical gradients since turbulence breaks odors into sparse pockets and gradients lose information [Murlis et al. \(1992\)](#) [DOI](#); [Vergassola et al. \(2007\)](#) [DOI](#); [Shraiman and Siggia \(2000\)](#) [DOI](#); [Balkovsky and I. \(2002\)](#) [DOI](#); [Reddy et al. \(2022\)](#) [DOI](#). The question of which features of turbulent odor traces are used by animals for navigation is natural, but not well understood. Beyond olfaction, some animals could use also prior spatial information to navigate [Cardé \(2021\)](#) [DOI](#); [Schal \(1982\)](#) [DOI](#); [Gire et al. \(2016\)](#) [DOI](#); [Baker et al. \(2018\)](#) [DOI](#), but if and how chemosensation and spatial perception are coupled is still not clear.

An algorithmic perspective to olfactory navigation in turbulence can shed light on some of these questions. Without aiming at an exhaustive taxonomy, see e.g. [Celani and Panizon \(2024\)](#) [DOI](#) for a recent review, we recall some approaches relevant to put our contribution in context. One class of methods are biomimetic algorithms, where explicit navigation rules are crafted taking inspiration from animal behavior. An advantage of these methods is interpretability, in the sense that they provide insights into features that effectively achieve turbulent navigation, for example: odor presence/absence [Baker \(1990\)](#) [DOI](#); [Kramer \(1997\)](#) [DOI](#); [Belanger and Willis \(1988\)](#) [DOI](#); [Balkovsky and I. \(2002\)](#) [DOI](#); odor slope at onset of detection [Atema \(1996\)](#) [DOI](#); number of detections in a given interval of time [Michaelis et al. \(2020\)](#) [DOI](#) and the time of odor onset [Demir et al. \(2020a\)](#) [DOI](#). On the flip side, in biomimetic algorithms behaviors are hardwired and typically reactive, not relying on any optimality criterion.

Away to tackle this shortcoming is to cast olfactory navigation within a sequential decision making framework [Sutton and Barto \(1998\)](#) [DOI](#). In this context, navigation is formalized as a task with a reward for success; by maximizing reward, optimal strategies can be sought to efficiently reach the target. A byproduct is that most algorithmic choices can often be done in a principled way. Within this framework, some approaches make explicit use of spatial information. Bayesian algorithms use a spatial map to guess the target location and use odor to refine this guess or “belief”. A prominent algorithm for olfactory navigation based on the concept of belief is the information-seeking algorithm [Vergassola et al. \(2007\)](#) [DOI](#) akin to exploration heuristics widely used in robotics [Cassandra et al. \(1996\)](#) [DOI](#); [LaValle \(2006\)](#) [DOI](#) (see e.g. [Loisy and Eloy \(2022\)](#) [DOI](#); [Ishida \(2020\)](#) [DOI](#)). Using Bayesian sequential decision making and the notion of beliefs, navigation can be formalized as a Partially Observable Markov Decision Process (POMDP) [Krishnamurthy \(2016\)](#) [DOI](#); [Hauskrecht \(2000\)](#) [DOI](#); [Shani et al. \(2013\)](#) [DOI](#), that colorblack can be approximatively solved [Rigolli et al. \(2022b\)](#) [DOI](#); [Heinonen et al. \(2023\)](#) [DOI](#); [Loisy and Heinonen \(2023b\)](#) [DOI](#). POMDP approaches are appealing since beliefs are a sufficient statistics for the entire history of odor detections. However, they are computationally cumbersome. Further, they leave the question open of whether navigation as sequential decision making can be performed using solely olfactory information.

Recently, two algorithms studied navigation as a response to olfactory input alone [Singh et al. \(2023\)](#) [DOI](#); [Verano et al. \(2023\)](#) [DOI](#). In [Singh et al. \(2023\)](#) [DOI](#) artificial neural networks were shown to learn near optimal strategies as a response to odor and instantaneous flow direction, although they were trained on odor cues with limited sparsity, and training with sparse odor cues typical of turbulence remains to be tested. In [Verano et al. \(2023\)](#) [DOI](#) an approach based on finite state controllers was proposed. Here, optimization was done assuming fixed known mean flow direction, and using a model-based technique, relying on prior knowledge of the likelihood to detect the odor in space, hence still using spatial information. A different model free optimization could also be considered avoiding spatial information but this latter approach also remains to be

tested. More generally, all the above approaches manipulate internally the previous history (memory) of odor detections. In this sense they are less interpretable, since the features of odor traces that drive navigation do not emerge explicitly.

In this paper, we propose a reinforcement learning (RL) approach to navigation in turbulence based on a set of interpretable olfactory features, with no spatial information other than the ability to orient relative to the mean flow, and highlight the role played by memory within this context. More precisely, we learn optimal strategies from data by training tabular Q learning Sutton and Barto (1998) [with realistic odor cues obtained from state-of-the-art Direct Numerical Simulations of turbulence](#). From the odor cues, we define features as moving averages of odor intensity and sparsity: the moving window is the temporal memory and naturally connects to the physics of turbulent odors. States are then obtained discretizing such features. Due to sparsity, agents may detect no odor within the moving window. We show there is an optimal memory minimizing the occurrence of this “void state”. The optimal memory scales with the blank time dictated by turbulence as it emerges from a trade off requiring that: (i) short blanks-typical of turbulent plumes are ignored by responding to detections further in the past, and (ii) long blanks promptly trigger a recovery strategy to make contact with the plume again. We leverage these observations to tune the memory adaptively, by setting it equal to the previous blank experienced along an agent’s path. With this choice, the algorithm tests successfully in distinct environments, suggesting that tuning can be made robustly to enable generalization. The agent learns to surge upwind in most non-void states and to recover by casting crosswind in the absence of detections. Optimal agents limit encounters with the void state to a narrow band right at the edge of the plume. This suggests that the temporal odor features we considered effectively predict when the agent is exiting the plume and point to an intimate connection between temporal predictions and spatial navigation.

Results

Background

Given a source of odor placed in an unknown position of a two-dimensional space, we consider the problem of learning to reach the source, **Figure 1A** [We formulate the problem as a discrete Markov Decision Process by discretizing space in tiles, also called “gridworld” in the reinforcement learning literature Sutton and Barto \(1998\)](#) [In this problem, an agent is in a given state \$s\$ which is one of a discrete set of \$n\$ tiles: \$s \in S := \{s_1, \dots, s_n\}\$. At each time step it chooses an action \$a\$ which is a step in any of the coordinate directions \$a \in \{\text{upwind}, \text{downwind}, \text{crosswind-left}, \text{crosswind-right}\}\$. Directions are labeled relative to the mean wind, which is assumed known. In our figures the flow always goes from left to right, hence the actions upwind, downwind, crosswind-right and crosswind-left correspond in the figures to a step left, right, up and down respectively. The goal is to find sequences of actions that lead to the source as fast as possible and is formalized with the notion of policy and reward which we will introduce later. If agents have perfect knowledge of their own location and of the location of the source in space, the problem reduces to finding the shortest path.](#)

Using time vs space to address partial observability

In our problem however, the agent does not know where the source is, hence its position s relative to the source, is unknown or “partially observed”. Instead, it can sense odor released by the target. In the language of RL, odor is an “observation” - but does it hold information about the position s ? The answer is yes: several properties of odor stimuli depend on the distance from the source Boie et al. (2018) [Ackels et al. \(2021b\)](#) [Nag and van Breugel \(2024\)](#) [However in the presence of turbulence, information lies in the statistics of the odor stimulus. Indeed, when odor is carried by a turbulent flow, it develops into a dramatically stochastic plume, i.e. a complex and convoluted region of space where the fluid is rich in odor molecules. Turbulent plumes break into structures](#)

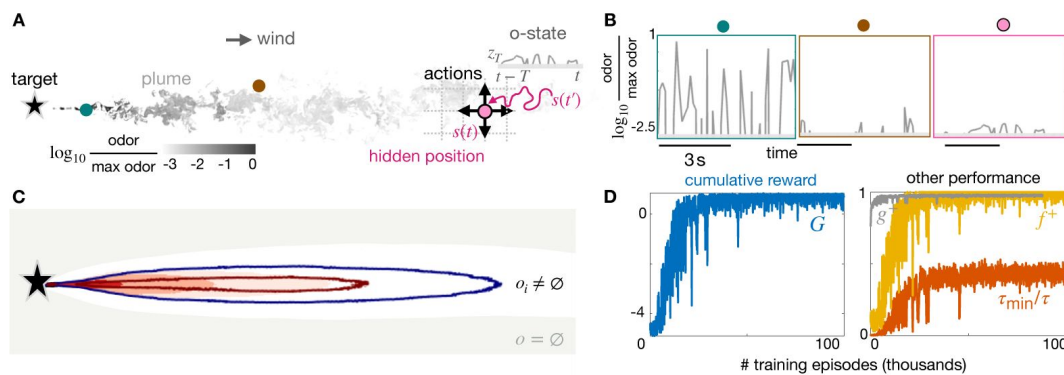


Figure 1.

Learning a stimulus-response strategy for turbulent navigation.

(A) Representation of the search problem with turbulent odor cues obtained from Direct Numerical Simulations of fluid turbulence (grey scale, odor snapshot from the simulations). The discrete position s is hidden; the odor concentration $z_T = z(s(t), t) | t - T \leq t' \leq t$ is observed along the trajectory $s(t')$, where T is the sensing memory. (B) Odor traces from direct numerical simulations at different (fixed) points within the plume. Odor is noisy and sparse, information about the source is hidden in the temporal dynamics. (C) Contour maps of olfactory states with nearly infinite memory ($T = 2598$): on average olfactory states map to different locations within the plume and the void state is outside the plume. Intermittency is discretized in three bins defined by two thresholds 66% (red line) and 33% (blue line). Intensity is discretized in 5 bins (dark red shade to white shade) defined by four thresholds (percentiles 99%, 80%, 50%, 25%). (D) Performance of stimulus-response strategies obtained during training, averaged over 500 episodes. We train using realistic turbulent data with memory $T = 20$ and backtracking recovery.

that distort and expand while they travel away from their source and become more and more diluted Falkovich et al. (2001) [↗](#); Shraiman and Siggia (2000) [↗](#); Celani et al. (2014) [↗](#); Reddy et al. (2022) [↗](#), see **Figure 1A** [↗](#). As a consequence, an agent within the plume experiences intermittent odor traces that endlessly switch on (whiff) and off (blank) **Figure 1B** [↗](#). The intensity of odor whiffs and how they are interleaved with blanks depends on distance from release, as dictated by physics Celani et al. (2014) [↗](#). Thus the upshot of turbulent transport is that the statistical properties of odor traces depend intricately on the position of the agent relative to the source. In other words, information about the state s is hidden within the observed odor traces.

This positional information can be leveraged with a Bayesian approach that relies on guessing s , i.e. defining the probability distribution of the position, also called belief. This is the approach that has been more commonly adopted in the literature until now Rigolli et al. (2022b) [↗](#); Heinonen et al. (2023) [↗](#); Loisy and Heinonen (2023b) [↗](#). Note that because of the complexity of these algorithms, only relatively simple measures of the odor are computationally feasible, e.g. instantaneous presence/absence. Here we take a different model-free and map-free approach. Instead of guessing the current state s , we ignore the spatial position and respond directly to the temporal traces of the odor cues. Two other algorithms have been proposed to solve partial observability by responding solely to odor traces with recurrent neural networks Singh et al. (2023) [↗](#) and finite state controllers Verano et al. (2023) [↗](#) that manipulate implicitly the odor traces. Here instead we manipulate odor traces explicitly, by defining memory as a moving window and by crafting a small number of features of odor traces.

Features of odor cues: definition of discrete olfactory states and sensing memory

To learn a response to odor traces, we set out to craft a finite set of *olfactory states*, $o \in O$, so that they bear information about the location s . Defining the olfactory states is a challenge due to the dramatic fluctuations and irregularity of turbulent odor traces. To construct a fully interpretable low dimensional state space, we aim at a small number of olfactory states that bear robust information about s , i.e. for all values of s . We previously found that pairing features of sparsity as well as intensity of turbulent odor traces predicts robustly the location of the source for all s Rigolli et al. (2022a) [↗](#). Guided by these results, we use these two features extracted from the temporal history of odor detections to define a small set of olfactory states.

We proceed to define a function that takes as input the history of odor detections along an agent's path and returns its current olfactory state. We indicate with $s(t)$ the (unknown) path of an agent, and with $z(s(t), t)$ the observations i.e. odor concentration along its path. First, we define a sensing memory T and we consider a short excerpt of the history of odor detections z_T of duration T prior to the current time t . Formally, $z_T(t) := \{z(s(t'), t') \mid t - T \leq t' \leq t\}$. Second, we measure the average intensity c (moving average of odor intensity over the time window T , conditioned to times when odor is above threshold), and intermittency i (the fraction of time the odor is above threshold during the sensing window T). Both features c and i are described by continuous, positive real numbers. Third, we define 15 olfactory states by discretizing i and c in 3 and 5 bins respectively. Intermittency i is bounded between 0 and 1 and we discretize it in 3 bins by defining two thresholds (33% and 66%). The average concentration, c , is bounded between 0 and the odor concentration at the source, hence prior information on the source is needed to discretize c using set thresholds. To avoid relying on prior information, we define thresholds of intensity as percentiles, based on a histogram that is populated online, along each agent's path (see Materials and Methods). The special case where no odor is detected over T deserves attention, hence we include it as an additional state named “void state” and indicate it with $o \equiv \varnothing$. When T is sufficiently long, the resulting olfactory states map to different spatial locations (**Figure 1C** [↗](#), with T equal to the simulation time). Hence this definition of olfactory states can potentially mitigate the problem of partial observability using temporal traces, rather than spatial maps. But will these olfactory states with finite memory T guide agents to the source?

Q learning: a map-less and model-free navigation to odor sources

To answer this question, we trained tabular episodic Q learning [Sutton and Barto \(1998\)](#). In a nutshell, we use a simulator to place an agent at a random location in space at the beginning of each episode. The agent is not aware of its location in space, but it senses odor provided by the fluid dynamics simulator and thus can compute its olfactory state o , based on odor detected along its path in the previous T sensing window. It then makes a move according to a set policy of actions $a \sim \pi_0(a|o)$. After the move, the simulator displaces the agent to its new location and relays the agent a penalty $R = -\pi$ with $\pi = 0.001$ if it is not at the source and a reward $R = 1$ if it reaches the source. The goal of RL is to find a policy of actions that maximizes the expected cumulative future reward $G = E_\pi(\sum_{t=0}^{\infty} \gamma^t R_{t+1})$ where the expectation is over the ensemble of trajectories and rewards generated by the policy from any initial condition. Because reward is only positive at the source, the optimal policy is the one that reaches the source as fast as possible. To further encourage the agent to reach the source quickly, we introduce a discount factor $\gamma < 1$.

Episodes where the agent does not reach the source are ended after $H_{\max} = 5000$ with no positive reward. As it tries actions and receives rewards, the agent learns how good the actions are. This is accomplished by estimating the quality matrix $Q(o, a)$, i.e. the maximum expected cumulative reward conditioned to being in o and choosing action a at the present time: $Q(o, a) = \max_\pi E_\pi(\sum_{t=0}^{\infty} \gamma^t R_{t+1} | o_t = o, a_t = a)$. At each step, the agent improves its policy by choosing more frequently putatively good actions. Once the agent has a good approximation of the quality matrix, the optimal policy corresponds to the simple readout: $\pi^*(a|o) = \delta(a - a^*(o))$ where $a^*(o) = \arg \max_a Q(o, a)$, for non-void states $o \neq \emptyset$.

Recovery strategy

To fully describe the behavior of our Q-learning agents, we have to prescribe their policy from the void state $o \equiv \emptyset$. This is problematic because turbulent plumes are full of holes thus the void state can occur anywhere both within and outside the plume, **Figure 1A**. As a consequence, the optimal action $a^*(\emptyset)$ from the void state is ill defined. We address this issue by using a separate policy called “recovery strategy”. Inspired by path integration as defined in biology [Etienne and Jeffery \(2004\)](#); [Etienne et al. \(1996\)](#); [Heinze et al. \(2018\)](#), we propose the backtracking strategy consisting in retracing the last T_a steps after the agent lost track of the odor. If at the end of backtracking the agent is still in the void state, it activates Brownian motion. Backtracking requires that we introduce memory of the past T_a actions. This timescale T_a for activating recovery is conceptually distinct from the duration of the sensing memory - however here we set $T_a = T$ for simplicity. Backtracking was observed in ants displaced in unfamiliar environments [Wystrach et al. \(2013\)](#), tsetse flies executing reverse turns bringing them back towards the location where they last detected odor [Torr \(1988\)](#); [Gibson and Brady \(1985\)](#) and cockroaches retracing their steps downwind, sometimes walking all the way back to the release point upon plume loss [Willis et al. \(2008\)](#); it was also previously used in computational models [Park et al. \(2016\)](#).

We find that Q-learning agents successfully learn to navigate to the odor source by responding solely to their olfactory state, with no sense of space nor models of the odor cues. Learning can be quantified by monitoring the cumulative reward which continuously improves with further training episodes (**Figure 1D**, left). Improved reward corresponds to agents learning how to reach the source more quickly and reliably with training. Indeed, it is easy to show that the expected cumulative reward $G = \langle e^{-\lambda\tau} - \sigma(1 - e^{-\lambda\tau})/(1 - \gamma) \rangle$, where τ is a random variable corresponding to time to reach the source and $\gamma = e^{-\lambda\Delta t}$ is the discount factor, with the time step $\Delta t = 1$ (see Materials and Methods). Large rewards arise when (i) a large fraction f^* of agents successfully reaches the source and (ii) the agents reach the source quickly, which maximizes $g^+ = \langle e^{-\lambda\tau} | \text{success} \rangle$. Indeed $G = f^* G^+ + (1 - f^*) G^-$, where $G^+ = g^+ - \sigma(1 - g^+)/(1 - \gamma)$ and $G^- = -\sigma(1 - e^{-\lambda H_{\max}})/(1 - \gamma)$. $H_{\max}$ is the horizon of the agent i.e. the maximum time the agent is allowed to

search, and after which the search is considered failed. Note that agents starting closer to the target receive larger rewards purely because of their initial position. To monitor performance independently on the starting location, we introduce the inverse time to reach the source relative to the shortest-path time from the same initial location, which goes from 0 for failing agents to 1 for ideal agents ($\tau_{\min}/\tau$), independently on their starting location. Note that this is not the quantity that is optimized for. One may specifically target this performance metrics, which is agnostic on the duration of an agent's path, by discounting proportionally to $t/\tau_{\min}$.

All four measures of performance plateau to a maximum, suggesting learning has achieved a nearly optimal policy (**Figure 1D**). Once training is completed, we simulate the trajectory of test-agents starting from any of the about 43 000 admissible locations within the plume and moving according to the optimal policy. Admissible locations are defined as any location where the odor is non-zero at least once within the entire simulation. We will recapitulate performance with the cumulative reward G averaged over the test-agents and dissect it into speed g^+ , convergence f^+ and relative time $\langle t_{\min}/\tau \rangle$.

Optimal memory

By repeating training using different values of T we find that performance depends on memory and an optimal memory T^* exists (**Figure 2A**). Why is there an optimal memory? The shortest memory $T = 1$ corresponds to instantaneous olfactory states: the instantaneous contour maps of the olfactory states are convoluted and the void state is pervasive (**Figure 2C**, top). As a consequence, agents often activate recovery even when they are within the plume, the policy almost always leads to the source ($f^+ = 79\% \pm 13\%$) but follows lengthy convoluted paths ($\tau_{\min}/\tau = 0.14 \pm 0.05$, **Figure 2C**, bottom). As memory increases, the olfactory states become smoother and agents encounter less voids (**Figure 2C**, center), perform straighter trajectories ($\tau_{\min}/\tau = 0.5 \pm 0.3$) and reach the source reliably ($\tau^+ = 95\% \pm 8\%$), **Figure 2A**, bottom. Further increasing memory leads to even less voids within the plume and even smoother olfactory states. However - perhaps surprisingly - performance does not further improve but slightly decreases (at $T = 50$, $f^+ = 94\% \pm 8\%$ and $\tau_{\min}/\tau = 0.38 \pm 0.36$). A long memory is deleterious because it delays recovery from accidentally exiting the plume, thus increases the number of voids *outside* of the plume (**Figure 2C**, bottom). Indeed, agents often leave the plume accidentally as they measure their olfactory state *while they move*. They receive no warning, but realize their mistake after T steps, when they enter the void state and activate recovery to re-enter the plume. The delay is linear with memory when agents recover by backtracking, but it depends on the recovery strategy (see Materials and Methods and **Figure 2-figure supplement 1**). Thus short memories increase time in void *within* the plume, whereas long memories increase time in void *outside* the plume: the optimal memory minimizes the overall chances to experience the void (**Figure 2B**). Intuitively, T^* should match the typical duration τ_b of blanks encountered within the plume, so that voids within the plume are effectively ignored without delaying recovery unnecessarily. Consistently, $\langle \tau_b \rangle$ averaged across all locations and times within the plume is $\langle \tau_b \rangle = 9.97 \pm 41.16$, comparable with the optimal memory T^* (**Figure 2A**).

Adaptive memory

There is no way to select the optimal memory T^* without comparing several agents or relying on prior information on the blank durations. In order to avoid prior information, we venture to define memory adaptively along each agent's path, using the intuition outlined above. We define a buffer memory T_b , and let the agent respond to a sensing window $T < T_b$. Ideally we would like to set $T \sim \langle \tau_b \rangle$. With this choice, blanks shorter than the average blank are ignored, as they are expected within the plume, whereas blanks longer than average initiate recovery, as they signal that the agent exited the plume. However, agents do not have access to $\langle \tau_b \rangle$ hence we set $T = \tau_b^-$, where τ_b^- is the most recent blank experienced by the agent. With this choice, the sensing memory T fluctuates considerably along an agent's path, due to turbulence (Celani et al. (2014) and **Figure 3A-B**). Note that blanks are estimated along paths, thus the statistics of T only

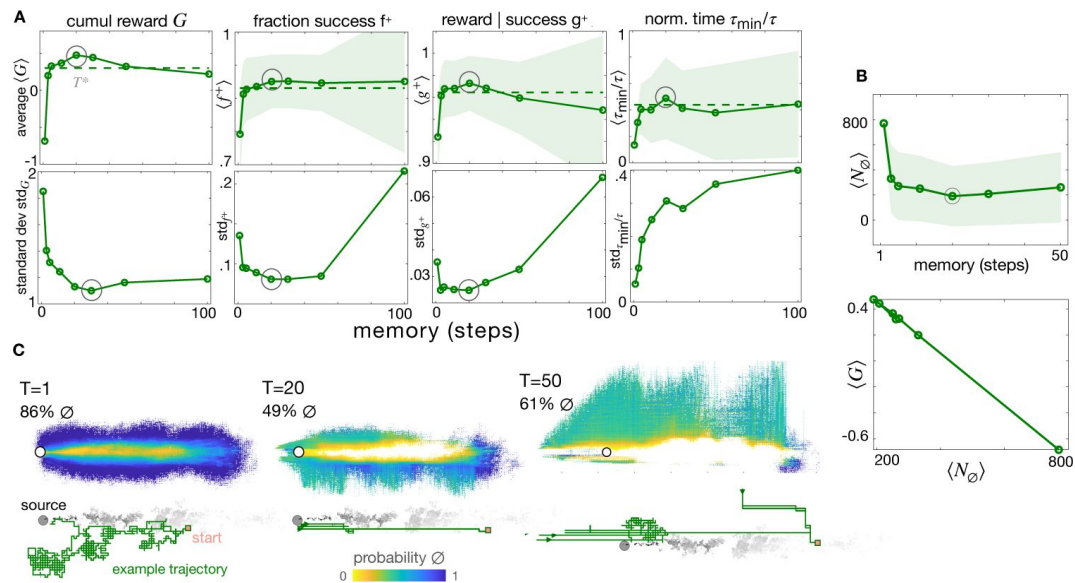


Figure 2.

The optimal memory T^* .

(A) Four measures of performance as a function of memory with backtracking recovery (solid line) show that the optimal memory $T^* = 20$ maximizes average performance and minimizes standard deviation, except for the normalized time. Top: Averages computed over 10 realizations of test trajectories starting from 43000 initial positions (dash: results with adaptive memory). Bottom: standard deviation of the mean performance metrics for each initial condition (see Materials and Methods). (B) Average number of times agents encounter the void state along their path, $\langle N_{\emptyset} \rangle$, as a function of memory (top); cumulative average reward $\langle G \rangle$ is inversely correlated to $\langle N_{\emptyset} \rangle$ (bottom), hence the optimal memory minimizes encounters with the void. (C) Colormaps: Probability that agents at different spatial locations are in the void state at any point in time, starting the search from anywhere in the plume and representative trajectory of a successful searcher (green solid line) with memory $T=1$, $T=20$, $T=50$ (left to right). At the optimal memory agents in the void state are concentrated near the edge of the plume. Agents with shorter memories encounter voids throughout the plume; agents with longer memories encounter more voids outside of the plume as they delay recovery. In all panels, shades are $\pm$ standard deviation.

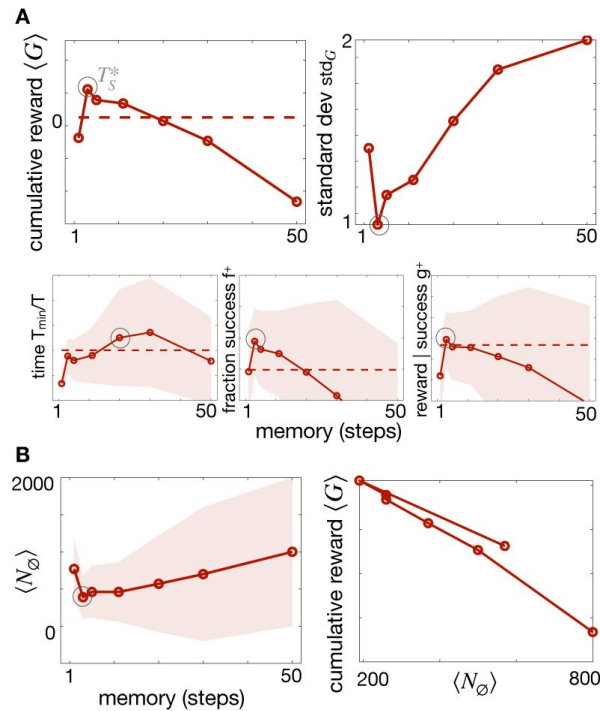


Figure 2—figure supplement 1.

The role of temporal memory with Brownian recovery strategy (same as main Figure 2A).

(A) Total cumulative reward (top left) and standard deviation (top right) as a function of memory showing an optimal memory $T^* = 3$ for the Brownian agent. Other measures of performance with their standard deviations show the same optimal memory (bottom). The trade-off between long and short memories discussed in the main text holds, but here exiting the plume is much more detrimental because regaining position within the plume by Brownian motion is much lengthier. (B) As for the Backtracking agent, the optimal memory corresponds minimizes the number of times the agent encounters the void state.

qualitatively matches the Eulerian statistics of τ_b . Despite the fluctuations, performance using the adaptive memory nears performance with the optimal memory (**Figure 3C**). This result confirms our intuition that memory should match the blank time. The advantage of the adaptive memory is that it relies solely on experience, with no prior information whatsoever. This is unlike T^* which can only be selected using prior information, with no guarantee of generalization to other plumes.

Learning to recover

So far, our agents combine a learned policy from non-void states to a heuristics from the void state, which we called the recovery strategy. We have considered a biologically-inspired heuristics where searchers make it back to locations within the plume by retracing their path backward. To further strip the algorithm of heuristics, we ask whether the recovery strategy may be learned, rather than fixed a priori. To this end, we split the void state in many states, labeled with the time elapsed since first entering the void. We pick 50 void states, as less than 50 void states results in no convergence, and states above 50 are useless because rarely visited. The counter is reset to 0 whenever the searcher detects the odor. The definition of the 15 non-void states $o_1, \dots, o_{15}$ remains unaltered. Interestingly, with this added degree of freedom, the agent learns an even better recovery strategy as reflected by all our measures of performance **Figure 3D**. Note that the learned recovery strategy resembles the casting behavior observed in flying insects [David et al. \(1983\)](#), as discussed below. In fact, insects deploy a range of recovery strategies depending on locomotor mode and environment. To corroborate these results, we compare performance using two additional biologically-inspired recovery strategies, i.e. circling (observed in windless environments [Stupski and van Breugel \(2024\)](#)), and cast & surge [David et al. \(1983\)](#) as well as a Brownian recovery which does not have a direct biological relevance but represents a simple computational benchmark. The learned recovery outperforms all heuristic recoveries, as seen by the cumulative reward G (**Figure 3D**). Circling is the second best recovery and shortly follows the learned recovery. Circling achieves nearly optimal performance by further decreasing failures (metrics f^+), but slowing down (metrics g^+ and τ_{min}/τ).

Characterization of the optimal policies

To understand how different recoveries affect the agent's behavior, we characterize the optimal policies obtained using the three recovery strategies. We visualize the probability to encounter each of the 16 olfactory states, or occupancy (circles in **Figure 4**), and the spatial distribution of the olfactory states. In the void state, the agent activates the recovery strategy. Recovery from the void state affects non-void olfactory states as well: their occupancy, their spatial distribution, and the action they elicit (**Figure 4** and **Figure 4-figure supplement 3**). This is because the agent computes its olfactory state online, according to its prior history which is affected by encounters with the void state. However, for all recoveries, non-void states are mostly encountered within the plume and largely elicit upwind motion (**Figure 4**, top, center). Thus macroscopically, all agents learn to surge upwind when they detect any odor within their memory, and to recover when their memory is empty. This suggests a considerable level of redundancy which may be leveraged to reduce the number of olfactory states, thus the computational cost. Reducing the number of non-empty olfactory states drastically to just 1 does indeed show degraded performance (see **Figure 5 - figure supplement 5**). A systematic optimization of odor representation requires a considerable reformulation of the algorithm which is beyond the scope of the current work. Note that, exclusively for the learned recovery, the optimal policy is enriched in actions downwind to avoid overshooting the source. Indeed, from positions beyond the source, the learned strategy is unable to recover the plume as it mostly casts sideways, with little to no downwind action. Intuitively, the precise locations where agents move downwind may be crucial to efficiently avoid overshooting. Thus the policy may depend on specific details of the odor plume, consistent with poorer generalization of the learned recovery (discussed next). We expect that in conditions where

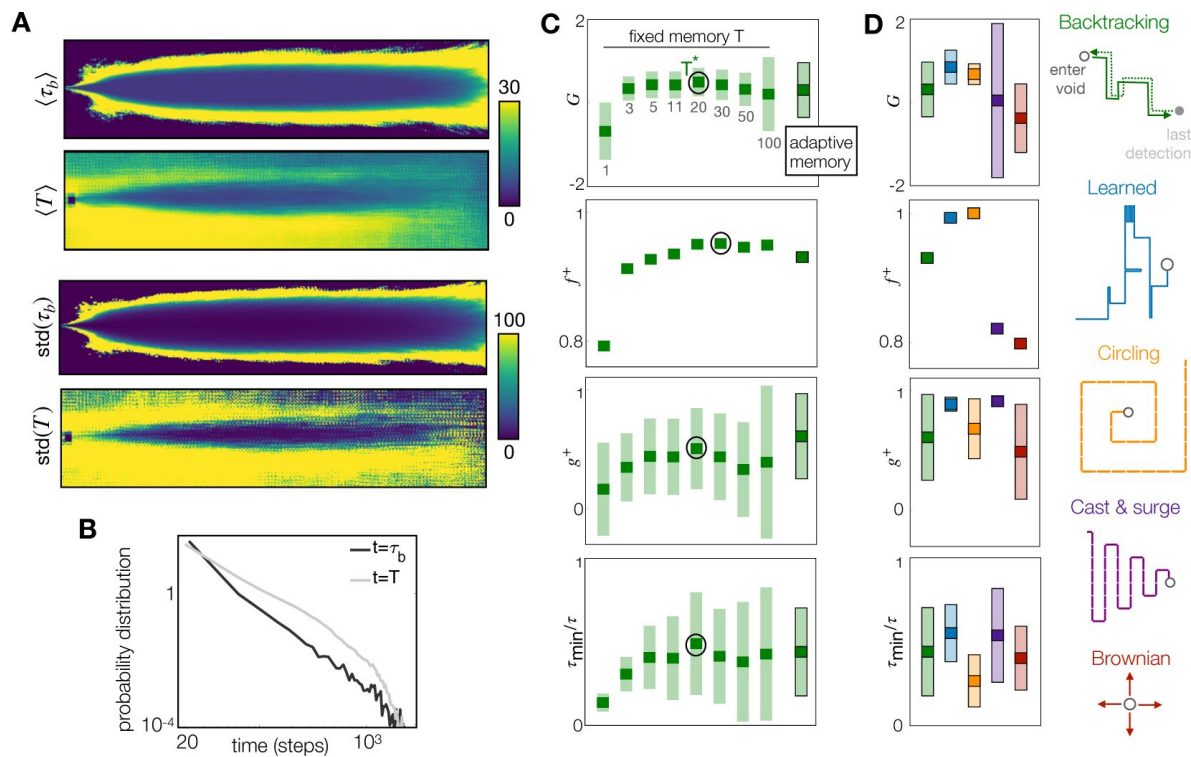


Figure 3.

The adaptive memory approximates the duration of the blank dictated by physics and it is an efficient heuristics, especially when coupled with a learned recovery strategy.

(A) Top to bottom: Colormaps of the Eulerian average blank time τ_b ; average sensing memory T ; standard deviation of Eulerian blank time and of sensing memory. The sensing memory statistics is computed over all agents that are located at each discrete cell, at any point in time. (B) Probability distribution of τ_b across all spatial locations and times (black) and of T across all agents at all times (gray). (C) Performance with the adaptive memory nears performance of the optimal fixed memory, here shown for backtracking; similar results apply to the Brownian recovery (**Figure 3-figure supplement 2**). (D) Comparison of five recovery strategies with adaptive memory: The learned recovery with adaptive memory outperforms all fixed and adaptive memory agents. In (C) and (D) dark squares mark the mean, and light rectangles mark $\pm$ standard deviation. f^* is defined as the fraction of agents that reach the target at test, hence has no standard deviation.

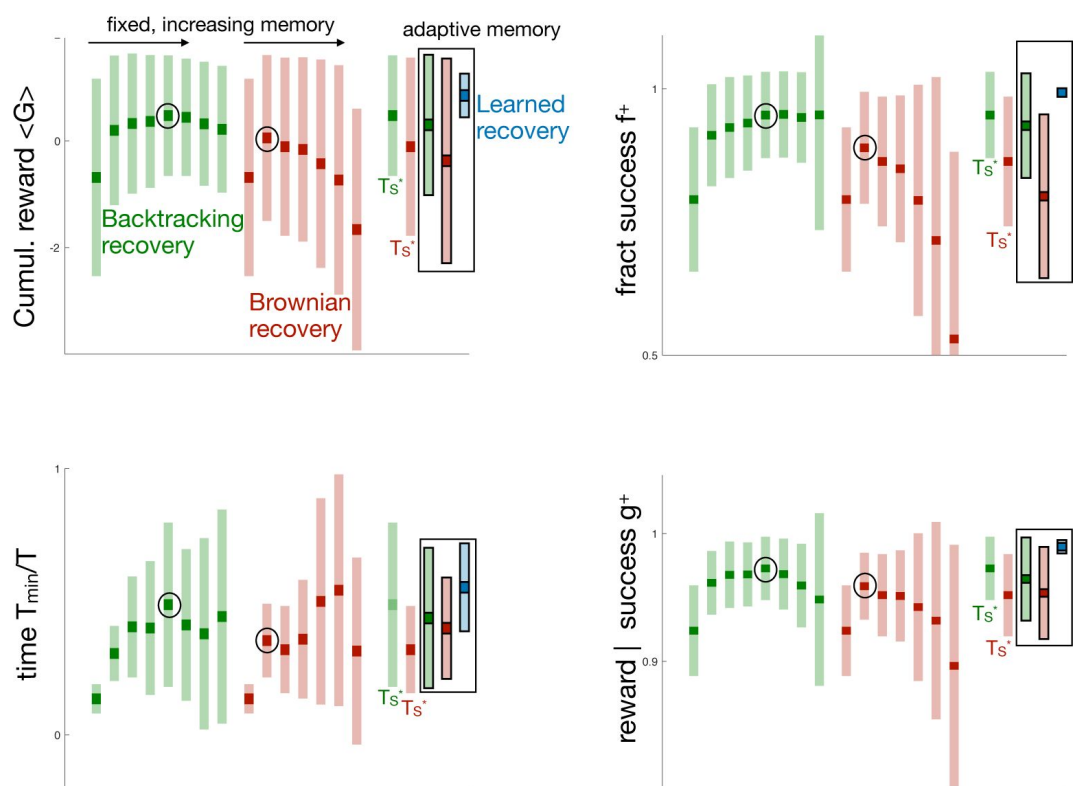


Figure 3—figure supplement 2.

All four measures of performance across agents with fixed memory and Backtracking vs Brownian recovery (green and red respectively, unframed boxes) and with adaptive memory for Backtracking, Brownian and Learned recovery (green, red and blue respectively, framed boxes)

Figure 4.

Optimal policies with adaptive memory for different recovery strategies: backtracking (green), Brownian (red) and learned (blue).

For each recovery, we show the spatial distribution of the olfactory states (top); the policy (center) and the state occupancy (bottom) for non-void states (left) *vs* the void state $\pi^*(a|\emptyset)$ (right). Spatial distribution: probability that an agent at a given position is in any non-void olfactory state (left) or in the void state (right), color-coded from yellow to blue. Policy: actions learned in the non-void states $\sum_{o \neq \emptyset} n_o \pi^*(a|o)$, weighted on their occupancy n_o (left, arrows proportional to the frequency of the corresponding action) and schematic view of recovery policy in the void state (right). State occupancy: fraction of agents that is in any of the 15 non-void states (left) or in the void state (right) at any point in space and time. Occupancy is proportional to the radius of the corresponding circle. The position of the circle identifies the olfactory state (rows and columns indicate the discrete intensity and intermittency respectively). All statistics is computed over 43000 trajectories, starting from any location within the plume.

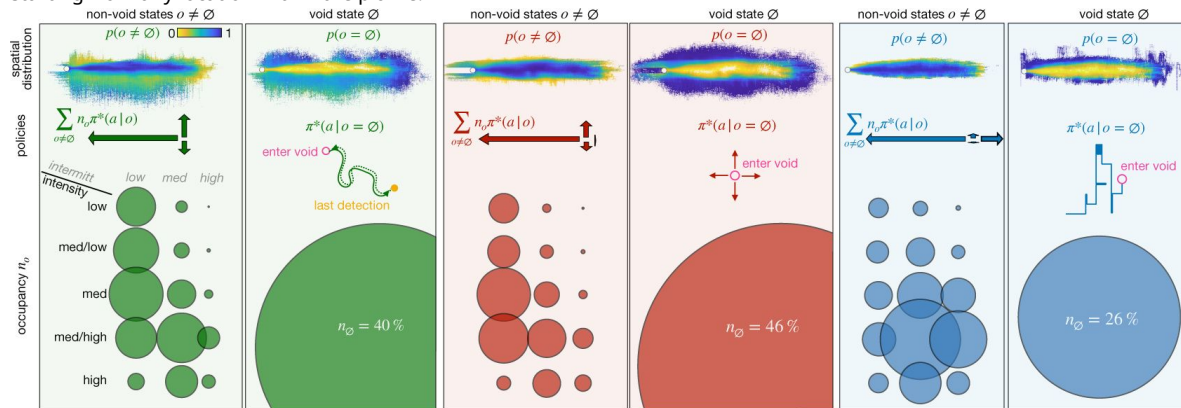
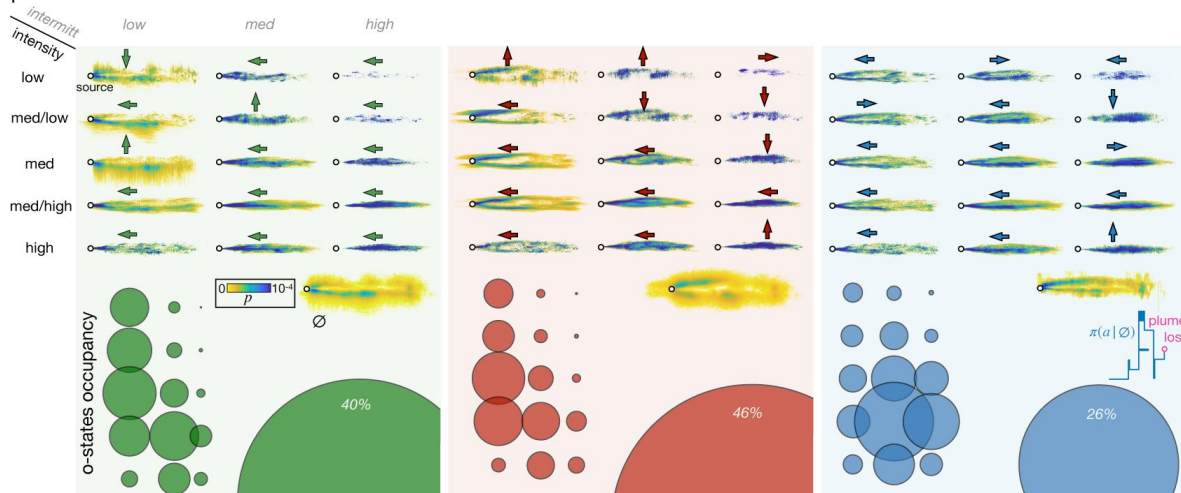


Figure 4—figure supplement 3.

Optimal policies for different recovery strategies and adaptive memory. From left to right: results for backtracking (green), Brownian (red) and learned (blue) recovery strategies. Top: probability that an agent in a given olfactory state is at a specific spatial location color-coded from yellow to blue. Rows and columns indicate the olfactory state; the void state is in the lower right corner. Arrows indicate the optimal action from that state. Bottom: Circles represent occupancy of each state, olfactory states are arranged as in the top panel. All statistics is computed over 43000 trajectories, starting from any location within the plume.



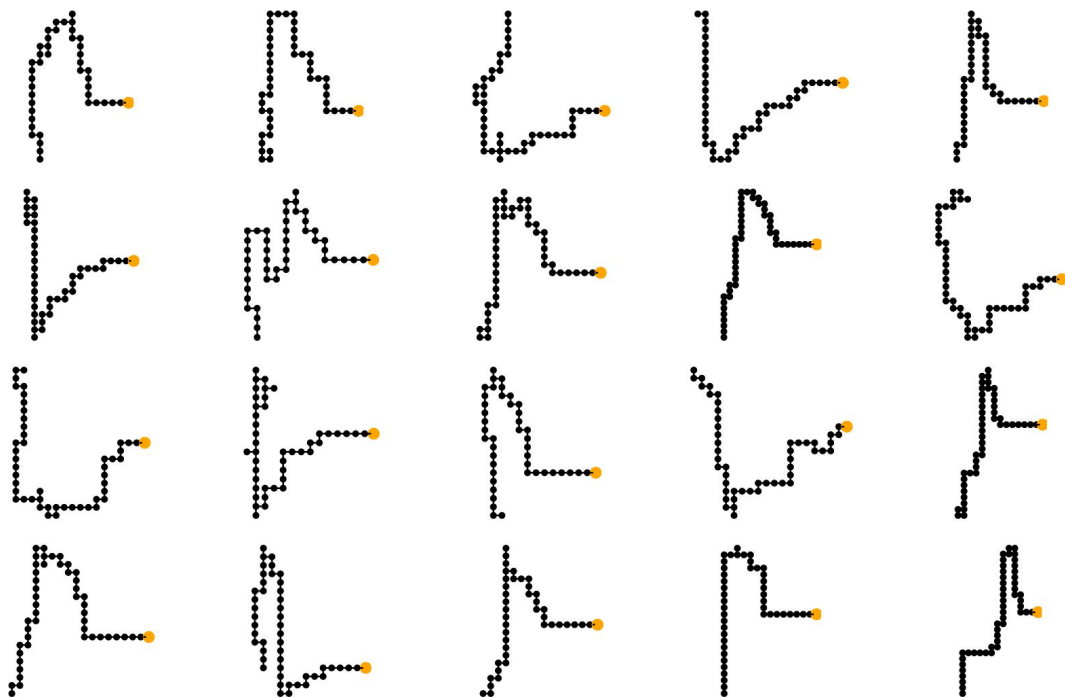


Figure 4—figure supplement 4.

The learned recovery resembles the cast and surge observed in animals, with an initial surge of 5 ± 2 steps and a subsequent motion crosswind starting from either side of the centerline and overshooting to the other side.

overshooting the source is more prominent, downwind motion may emerge as an effective component of the recovery strategy, similar to observations in insects (e.g. [Wolf and Wehner \(2000\)](#) [↗](#); [Alvarez-Salvado et al. \(2018\)](#) [↗](#)).

The void state shows the most relevant differences: for both heuristic recoveries, 40% or more of the agents are in the void state and they are spatially spread out. In contrast, in the case of learned recovery, the optimal policy limits occurrence of the void state to 26% of the agents, confined to a narrow band near the edge of the plume. From these locations, the agents quickly recover the plume, explaining the boost in performance discussed above. In all distinct trainings of the agents with learned-recovery, we observed that the trajectories in the void start with an initial surge of 6 ± 2 steps; continue with either crosswind direction and then switch to the other side, with little to no downwind actions (see **Figure 4 - Figure Supplement 4** [↗](#) and **Table 1** [↗](#)). This recovery thus mixes aspects of exploitation (surge) to aspects of exploration (cast): we defer a more in-depth analysis that disentangles these two aspects elsewhere.

Tuning for adaptation to different environments

Finally, we test performance of the trained agents on six environments, characterized by distinct fluid flows and odor plumes (**Figure 5** [↗](#) and Materials and Methods). Environment 1 is the native environment, where the agents were originally trained; Environment 2 is obtained by increasing the threshold of detection, which makes the signals considerably more sparse with longer blanks. Environments 3 and 4 are closer to the lower surface of the simulated domain, where the plume is smaller and fluctuates less. Environment 5 is a similar geometry, but obtained for a smaller Reynolds number and a different way to generate turbulence. Finally Environment 6 has an even larger Reynolds number, a longer domain and a smaller source, which creates an even more dramatically sparse signal. Environments 1, 2, 5 and 6 are representative of conditions encountered far from a substrate, e.g. by flying or swimming organisms. Environments 3 and 4 represent odor near surfaces relevant to terrestrial or benthic navigation (but not directly applicable to trail tracking, where odor traces *on* the substrate is tracked). Note that we consider a Schmidt number $Sc = 1$ appropriate for odors in air but not in water. However, we expect a weak dependence on the Schmidt number as the Batchelor and Kolmogorov scales are below the size of the source and we are interested in the large scale statistics [Falkovich et al. \(2001\)](#) [↗](#); [Celani et al. \(2014\)](#) [↗](#); [Duplat et al. \(2010\)](#) [↗](#).

We consider agents with adaptive memory and compare the five recovery strategies discussed above - backtracking, learned, circling, cast & surge and Brownian, see **Figure 5B** [↗](#). Comparing performance across environments we find that: (i) although performance is degraded when testing in non native environments, backtracking, learned and circling recoveries with adaptive memory are still extremely likely to find the source. The upshot of generalization is that agents may navigate distinct turbulent plumes using a baseline strategy learned in a specific plume. Importantly, as most of these agents still do reach the source, fine-tuning may enable efficient adaptation to different environments. Further work is needed to establish how much fine-tuning is needed to fully adapt to different environments. (ii) Brownian and cast& surge recoveries have lowest performance and generalization across all environments. Cast & surge is often used as a comparison [Verano et al. \(2023\)](#) [↗](#) and can be extremely effective: our results do not contradict the literature, but simply showcase that cast width and surge length need to be carefully defined. (iii) The cumulative reward G shows that the learned recovery is the best at generalizing to environments 2, 5 and 6. Particularly, in the most intermittent Environment 6 a striking 91% of agents succeeds in finding the source, with trajectories less the twice as long as the shortest path to the source. (iv) Circling is the best at generalizing to environments 3 and 4, representative of less intermittent regions near the substrate. As observed for the native environment, circling favors success rate (metrics f^*) against speed (metrics g^* and $\tau_{\min}/\tau$).

Table 1.

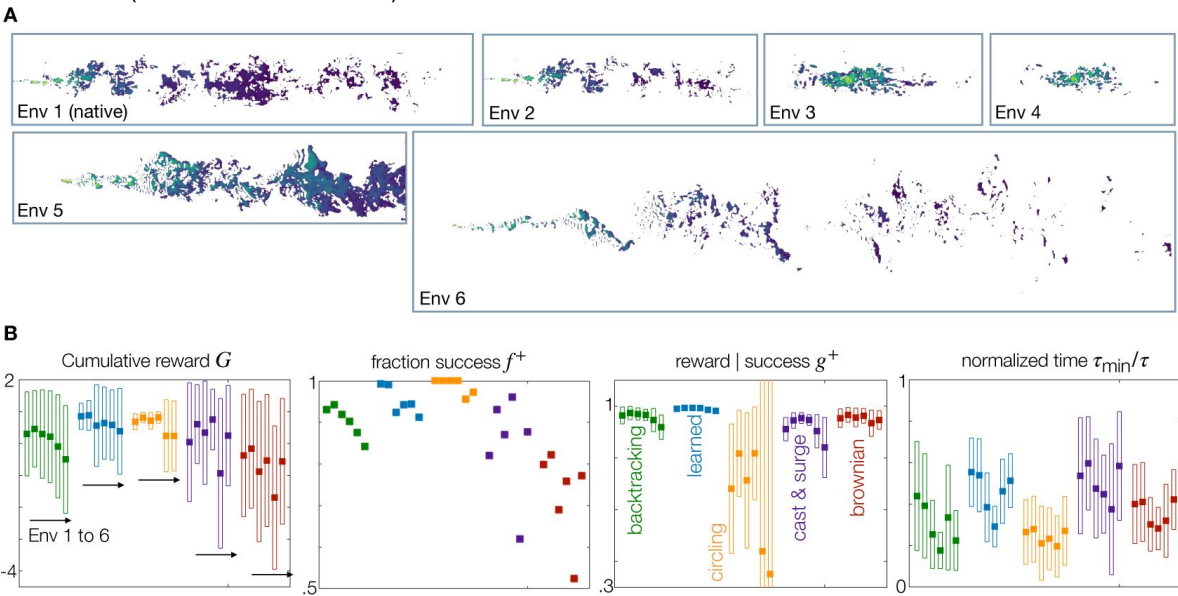
Parameters of the learned recovery, statistics over 20 independent trainings.

Initial surge upwind	6 ± 2
Total steps upwind	15 ± 2
Total steps downwind	1.3 ± 1.4
Total steps to the right	15 ± 3
Total steps to the left	18 ± 6

Figure 5.

Generalization to statistically different environments.

(A) Snapshots of odor concentration normalized with concentration at the source, colorcoded from blue (0) to yellow (1) for environment 1 to 6 as labeled. Environment 1* is the native environment where all agents are trained. (B) Performance for the five recovery strategies brownian (red), backtracking (green), learned (blue), circling (orange) and zigzag (purple) with adaptive memory, trained on the native environment and tested across all environments 1 to 6. Four measures of performance defined in the main text are shown. Dark squares mark the mean, and empty rectangles $\pm$ standard deviation. No standard deviation is shown for the f^+ measure for the learned, circling and zigzag recoveries as these strategies are deterministic (see Materials and Methods).



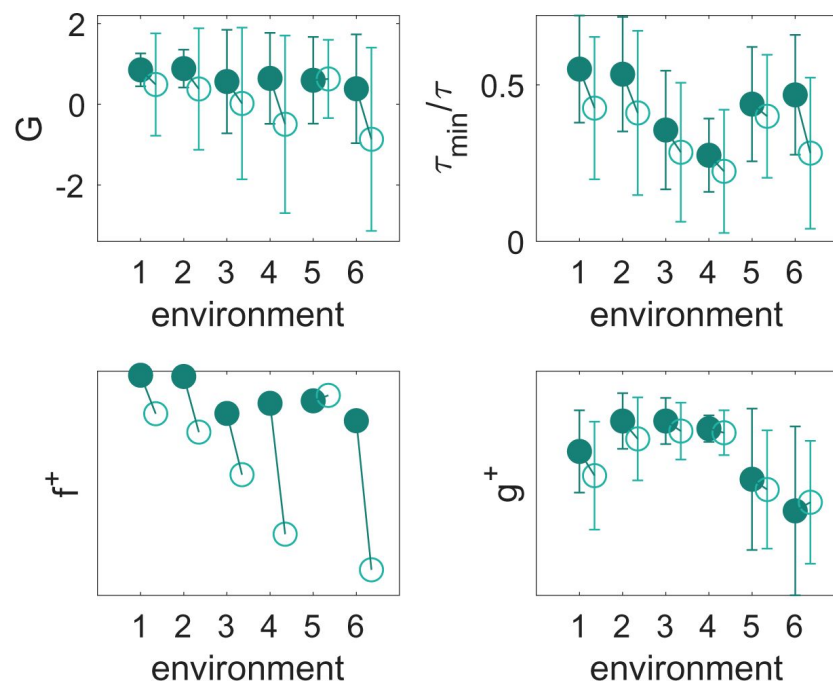


Figure 5—figure supplement 5.

The learned recovery with adaptive memory and a single nonempty olfactory state (empty circles) displays degraded performance with respect to the full model (full circles).

Discussion

In this work, we showed that agents exposed to a turbulent plume learn to associate salient features of the odor time trace - the olfactory state - to an optimal move that guides them to the odor source. The upshot of responding solely to odor is that the agent does not navigate based on *where* it believes the target is and thus needs no map of space nor prior information about the odor plume, which avoids considerable computational burden. The only spatial awareness needed to implement this algorithm is the ability to orient motion relative to the mean flow which is assumed known. In reality animals cannot measure the mean flow but rely on local measures of flow speed, using e.g. antennas for insects [Reynolds et al. \(2010\)](#), whiskers for rodents [Yu et al. \(2016\)](#) or the lateral line for marine organisms [Liao \(2006\)](#). Further work is needed to bridge the gap with our simplified setting. On the flip side, in our stimulus-response algorithm, agents need to start from within the plume, however sparse and fragmented. Indeed, far enough from the source, Q-learning agents are mostly in the void state and they can only recover the plume if they have previously detected the odor or are right outside the plume. In contrast, agents using a map of space can navigate from larger distances than are reachable by responding directly to odor cues. Indeed, in the map-based POMDP setting, absence of odor detection is still informative and it enables agents to first find the plume, and then refine the search to localize the target within the plume [Rigolli et al. \(2022b\)](#); [Loisy and Heinonen \(2023a\)](#).

We show that because the odor signal within a turbulent plume constantly switches on and off, navigation must handle both absence and presence of odor stimuli. We address this fundamental issue by alternating between two distinct strategies: (i) Prolonged absence of odor prompts entry in the void state and triggers a recovery strategy to make contact with the plume again. We explored four heuristic recoveries and found that spiralling around the location where the agent last detected odor is the most efficient heuristics, which privileges reliable success rather than speed. An even more efficient recovery can be learned that resembles cross-wind casting and limits the void state to a narrow region right outside of the plume. Casting is a well-studied computational strategy [Baker \(1990\)](#); [Balkovsky and I. \(2002\)](#) also observed in animal behavior, most famously in flying insects [David et al. \(1983\)](#). Intriguingly, cast and surge also emerges in algorithms making use of a model of the odor, whether for Bayesian updates or for policy optimization [Vergassola et al. \(2007\)](#); [Rigolli et al. \(2022b\)](#); [Verano et al. \(2023\)](#). Whether natural casting behavior is learned, as in Q-learning, or is hard-wired in a model of the odor plume remains a fascinating question for further research. Clearly, the width of the casts and length of the surges are of crucial importance: a hard-wired cast and surge recovery with arbitrary parameters shows poor performance. (ii) Odor detections prompt entry in non-void olfactory states, which predominantly elicit upwind surge. Blanks shorter than the sensing memory are ignored, i.e. agents do not enact recovery but respond to stimuli experienced prior to the short blank. The non-void olfactory states are crafted based on biologically-plausible features which have been shown to harness positional information. Further work may optimize these non-void olfactory states by feature engineering, e.g. testing different discretizations to reduce redundancy. A drastic reduction to a single non-void olfactory state degrades performance, suggesting screening a large library of features using supervised learning as in [Rigolli et al. \(2022a\)](#) may be used to potentially improve performance. This approach will provide a systematic ranking of the most efficient temporal features of odor time traces for navigation; however it will have to test different memories, discretizations and regression algorithms as well, making it cumbersome. Alternatively, feature engineering may be bypassed altogether by the use of recurrent neural networks (RNNs) [Singh et al. \(2023\)](#) or finite state controllers [Verano et al. \(2023\)](#). These algorithms are appealing in that they bypass entirely the need to hand-craft explicit features of odor time traces. On the flip side, they provide no explicit handle at the odor features that drive behavior nor at the specific duration of the temporal memory and how it is related to the physics of the odor cues. Thus to extract these information extra work is needed to

interrogate these algorithms. For example, principal component analysis in [Singh et al. \(2023\)](#) suggests the hidden state of trained agents correlates with biologically relevant variables, including head direction, odor concentration and time since last detection. Finally, a systematic comparison using a common dataset is needed to elucidate how other heuristic and normative model-free algorithms handle odor presence vs odor absence.

To switch between the odor-driven strategy and the recovery strategy, we introduce a timescale T , which is an explicit form of temporal memory. T delimits a sensing window extending in the recent past, prior to the present time. All odor stimuli experienced within the sensing window affect the current response. By using fixed memories of different duration, we demonstrate that an optimal memory exists and that the optimal memory minimizes the occurrence of the void state. On the one hand, long memories are detrimental because they delay recovery from accidentally exiting the plume. On the other hand, short memories are detrimental because they trigger recovery unnecessarily, i.e. even for blanks typically experienced within the turbulent plume. The optimal memory thus matches the typical duration of the blanks. To avoid using prior information on the statistics of the odor, we propose a simple heuristics setting memory adaptively equal to the most recent blank experienced along the path. The adaptive memory nears optimal performance despite dramatic fluctuations dictated by turbulence. Success of the heuristics suggests that a more accurate estimate of the future blank time may enable an even better adaptive memory; further work is needed to corroborate this idea.

Thus in Q-learning, memory is a temporal window matching odor blanks and distinguishing whether agents are in or out of the plume. The role of memory for olfactory search has been recently discussed in ref. [Verano et al. \(2023\)](#). In POMDPs, memory is stored in a detailed belief of agent position relative to the source. In finite state controllers, memory denotes an internal state of the agent and was linked to a coarse grained belief of the searcher being within or outside of the plume, similar to our findings. In recurrent neural networks memory is stored in the learned weights. A quantitative relationship between these different forms of memory and their connection to spatial perception remains to be understood.

We conclude by listing a series of experiments to test these ideas in living systems. First, olfactory search in living systems displays memory ([Verano et al. \(2023\)](#); [Baker et al. \(2018\)](#) and refs. therein). In insects, temporal scales can be measured associated to memory. Indeed, for flying insects loss of contact with a pheromone plume triggers crosswind casting and sometimes even downwind displacement [Cardé \(2021\)](#); [Kuenen and Cardé \(1994\)](#). Interestingly, the onset of casting is delayed with respect to loss of contact with the plume [Kuenen and Cardé \(1994\)](#); [van Breugel F and MH \(2014\)](#), but this delay is not understood. Similarly, an odor detection elicits upwind surges that can last several seconds [Kathman et al. \(2024\)](#); [Alvarez-Salvado et al. \(2018\)](#). In walking flies, the timing of previous odor encounters biases navigation [Demir et al. \(2020b\)](#). (How) do these temporal timescales depend on the waiting times between previous detections? Using optogenetics [Gepner et al. \(2015\)](#); [Hernandez-Nunez et al. \(2015\)](#); [Matheson et al. \(2022\)](#); [David Stupski and van Breugel \(2024\)](#) or olfactory virtual reality with controlled odor delivery [Radvansky and Dombeck \(2018\)](#) experiments may measure memory as a function of the full history of odor traces. For insects, one may monitor memory by tracking the onset of cross wind casting with respect to the loss of the plume. More in general, a temporal memory may be defined by monitoring how far back in the past should two odor traces be identical in order to elicit the same repertoire of motor controls. Second, our algorithm learns a stimulus-response strategy that relies solely on odor cues. The price to pay is that the agent must follow the ups and downs of the odor trace in order to compute averages and recognize blanks. A systematic study may use our algorithm to test the requirements of fidelity of this temporal representation, and how it depends on turbulence. How does turbulence affect the fidelity of odor temporal representation in living systems? Crustaceans provide excellent model system to ask this question, as they are known to use bursting olfactory receptor neurons to encode temporal information from olfactory scenes [Bobkov and Ache \(2007\)](#); [Ache et al. \(2016\)](#). Temporal information is

also encoded in the olfactory bulb of mammals [Carey et al. \(2009\)](#) ; [Ackels et al. \(2021a\)](#) . Organisms with chemo-tactile systems like the octopus [Allard et al. \(2023\)](#)  may serve as a comparative model, to ask whether touch-chemosensation displays a sloppier temporal response, reflecting that surface-bound stimuli are not intermittent.

Third, our Q-learning algorithm requires the agent to receive olfactory information, thus start near or within the odor plume. In contrast, algorithms making use of a spatial map and prior information on the odor plume may first search for the plume (in conditions of near zero information) and then search the target within the plume [Rigolli et al. \(2022b\)](#) ; [Loisy and Heinonen \(2023a\)](#) ; [Vergassola et al. \(2007\)](#) . Animals are known to use prior information to home into regions of space where the target is more likely to be found; but they can switch to navigation in response to odor (see e.g. [Cardé \(2021\)](#) ; [Schal \(1982\)](#) ; [Gire et al. \(2016\)](#) ; [Baker et al. \(2018\)](#) ). What triggers the switch from navigation driven by spatial perception to navigation driven by odor? For mice, the need for spatial perception may be tested indirectly by comparing paths in light vs dark, noting that neuronal place fields, that mediate spatial perception, are better stabilized by vision than olfaction [Save et al. \(2000\)](#) ; [Zhang and D \(2015\)](#) . Thus in the light, animals have the ability to implement both map-less and map-based algorithms, whereas in the dark they are expected to more heavily rely on mapless algorithms. To make sure animals start searching for the odor target even before sensing odor, operant conditioning can be deployed so that animals associate an external cue (e.g. a sound) to the beginning of the task. Note that distinct species control locomotion differently and as a result trajectories are usually far more complex than a sequence of discrete steps on a checker board. Thus to compare algorithms to animal behavior a more detailed model of the specific motor controls is to be developed.

The reinforcement learning view of olfactory navigation offers an exciting opportunity to probe how living systems interact with the environment to accomplish complex real-world tasks affected by uncertainty. Coupling time varying odor stimuli with spatial perception is an instance of the broader question asking how animals combine prior knowledge regarding the environment with reaction to sensory stimuli. We hope that our work will spark further progress into connecting these broader questions to the physics of fluids.

Methods and materials

Data description

The data we used to train the agents (Environment 1) is a set of 2598 matrices $\{D_t\}_{t=0}^{2597}$. Every matrix $D_t \in \mathbb{R}^{1225 \times 280}$ contains the odor intensity in every position (i, j) i.e. $(D_t)_{i,j}$ represents the odor intensity in position (i, j) at time t . The source of odor is in position $(20, 142)$ and, in order to simplify the training, we considered as terminal states every position in a circle centered in the source position and with radius 10 called *source region*. Data are obtained from a direct numerical simulation of the Navier-Stokes equations and the equations of transport of the odor. Environments 1 to 4 are derived from Simulation 1, a direct numerical simulation of a channel flow described in [Rigolli et al. \(2022a\)](#)  and used to develop a POMDP algorithm in [Rigolli et al. \(2022b\)](#) ; dataset available from [Rigolli et al. \(2022c\)](#) . The simulation represents a boundary layer matching the lowest ~1m of the atmosphere and with horizontal dimensions ~1.9×9.5 m and Reynolds number $Re_\nu \sim 1400$ (on the low side of atmospheric Reynolds typically ranging from 1000 to 10000 [Gulitski et al. \(2007\)](#) ). Odor snapshots are extracted at a height ~.5 m from the ground for Environments 1 and 2 and ~.01 m for Environments 3 and 4 respectively. We preprocess the data to zero every entry of these matrices when they are smaller than a *noise level* $n_{|v|} := 3 \times 10^{-6}$ (or 0.13 % relative to concentration at the source). The noise level is increased to 0.22% in Environments 2 and 4. Data information are summarized in [Table 2](#)  and [3](#) . Levels of intermittency in [Figure 1](#)  show that only a thin core region has intermittency larger than 66%,

	Simulation 1	Simulation 2	Simulation 3
2D simulation grid	1225×280	1024×256	2000×500
# time stamps	2598	5000	5000
# decisions per time stamp	1	1	1
Speed (grid points / time stamp)	10	10	10
n_{IV}	3×10^{-6}	3×10^{-6}	10^{-4}
Source location	(20, 142)	(128, 128)	(150, 250)

Table 2.

Gridworld geometry. From Top: 2D size of the simulation, agents that leave the simulation box continue to receive negative reward and no odor; number of time stamps in the simulation, beyond which simulations are looped; number of actions per time stamp; speed of the agent; noise level below which odor is not detected; location of the source on the grid. See **Table 3** for the values of the grid size Δx and time stamps at which odor snapshots are saved.

whereas the most challenging regions at the edge of the plume have intermittency under 33%. For reference, experimental values of 25% to 20% were reported for a surrogate odor in the atmospheric boundary layer, along the centerline at 2 to 15 m from the source [Murlis and Jones \(1981\)](#).

Environments 5 and 6 correspond to horizontal slices at mid height extracted from two additional simulations we performed to corroborate the results (Simulation 2 and 3). In Simulation 2, the odor is advected by a turbulent open channel flow, with three hemispherical obstacles placed on the ground close to the inlet to generate turbulence. The Navier-Stokes equations (1) and advection-diffusion equation for odor transport (2) are solved using a central second order finite difference scheme. The convective terms are discretized in time using an explicit Adams-Bashforth method; and the viscous and diffusion terms using an implicit Crank-Nicolson method [Viola et al. \(2020\)](#). The code is written in Fortran and is GPU parallelized. The channel is divided into $1024 \times 256 \times 128$ grid points along streamwise, spanwise and wall-normal directions respectively. The corresponding average spatial resolutions are $\Delta x = 5\eta$, $\Delta y = 5\eta$, $\Delta z = 4\eta$, where η is the Kolmogorov length scale. Three hemispheres of radius 100η are placed at a distance of 250η from the inlet on the ground, equally spaced along the spanwise direction. The channel is forced using a constant pressure gradient. For the velocity field, we impose a no-slip boundary condition at the ground and on the obstacles ($\mathbf{u} = 0$) and a free-slip boundary on top ($u_z = 0$, $\partial_z u_x = \partial_z u_y = 0$). The velocity field is periodic along the streamwise and spanwise directions. The bulk Reynolds number is 7800. For the odor field, we impose Dirichlet condition ($c = 0$) at the ground, on the obstacles and inlet, no-flux ($\partial_z c = 0$) on top, and outflow along other directions. Similar to the native environment, we choose the Schmidt number to be 1. The odor source is located downstream of the obstacle and centered at $[640\eta, 640\eta, 256\eta]$ along streamwise, spanwise and wall-normal directions respectively. The odor source has a Gaussian profile with a standard deviation of 8η .

Simulation 3 is similar to Simulation 2, albeit with a higher bulk Reynolds number of 17500. Here, the channel is divided into $2000 \times 500 \times 200$ grid points and has an average spatial resolution of $\Delta x = \Delta y = \Delta z = 5.5\eta$. The odor source has a Gaussian profile centered at $[825\eta, 1375\eta, 550\eta]$ with a standard deviation of 3η . For Environments 5 and 6, noise level is 0.01% relative to concentration at the source. See Table for a summary of parameters and how they match the physical dimensions of the domain.

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla P + \mu \nabla^2 \mathbf{u} + \mathbf{f}; \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0.$$

$$\frac{\partial z}{\partial t} + \mathbf{u} \cdot \nabla z = D \nabla^2 z + s. \quad (2)$$

Olfactory states, Features & Discretization

Each agent stores the odor concentrations detected in the previous T time steps in a vector $\mathbf{M} = (z(s(t-T), t-T), \dots, z(s(t), t))$. We introduce an adaptive sensitivity threshold function $s_{\text{thr}}(\cdot)$ defined

$$s_{\text{thr}}(T) := \max \left\{ \frac{C_{\text{thr}}}{T} \sum_{i=1}^T M_i, n_{\text{thr}} \right\}, \quad (3)$$

Sim ID	L	W	H	U_b	η	Δx	τ_η	ϵ	y^+	Re_b	$\frac{u'}{U_b}$	T	$\omega_{\text{save}} \times \tau_\eta$
1	40	8	4	23	0.006	0.025	0.01	39	0.0035	11500	15%	$64\tau_\eta$	1
air	9.50 m	1.90 m	0.96 m	$36 \frac{\text{cm}}{\text{s}}$	0.15 cm	0.6 cm	0.15 s	$6.3\text{e-}4 \frac{\text{m}^2}{\text{s}^3}$	0.09 cm				
water	2.66 m	0.53 m	0.27 m	$8.6 \frac{\text{cm}}{\text{s}}$	0.04 cm	0.2 cm	0.18 s	$3\text{e-}5 \frac{\text{m}^2}{\text{s}^3}$	0.02 cm				
2	20	5	2	14	0.004	0.02	0.005	163	0.0038	7830	15%	$95\tau_\eta$	5
air	7.50 m	1.875 m	0.75 m	$17.5 \frac{\text{cm}}{\text{s}}$	0.15 cm	0.75 cm	0.15 s	$6.3\text{e-}4 \frac{\text{m}^2}{\text{s}^3}$	0.142 cm				
water	2.00 m	0.50 m	0.20 m	$3.9 \frac{\text{cm}}{\text{s}}$	0.04 cm	0.2 cm	0.18 s	$3\text{e-}5 \frac{\text{m}^2}{\text{s}^3}$	0.038 cm				
3	20	5	2	22	0.0018	0.01	0.0025	204	0.0012	17500	13%	$141\tau_\eta$	2.5
air	16.7 m	4.18 m	1.67 m	$30.6 \frac{\text{cm}}{\text{s}}$	0.15 cm	0.83 cm	0.15 s	$6.3\text{e-}4 \frac{\text{m}^2}{\text{s}^3}$	0.1 cm				
water	4.44 m	1.11 m	0.44 m	$6.8 \frac{\text{cm}}{\text{s}}$	0.04 cm	0.22 cm	0.18 s	$3\text{e-}5 \frac{\text{m}^2}{\text{s}^3}$	0.03 cm				

Table 3.

Parameters of the simulations. From Left to Right: Simulation ID (1,2, 3); Length L , width W , height H of the computational domain; mean horizontal speed $U_b = \langle u \rangle$; Kolmogorov length scale $\eta = (\nu^3/\epsilon)^{1/4}$ where ν is the kinematic viscosity and ϵ is the energy dissipation rate; mean size of gridcell Δx ; Kolmogorov timescale $\tau_\eta = \eta^2/\nu$; energy dissipation rate $\epsilon = \nu/2(\langle \partial u_i/\partial x_j + \partial u_j/\partial x_i \rangle^2)$; wall unit $y^+ = \nu/u_\tau$ where u_τ is the friction velocity; bulk Reynolds number $Re_b = U_b(H/2)/\nu$ based on the bulk speed U_b and half height; magnitude of velocity fluctuations u' relative to the bulk speed; large eddy turnover time $T = H/2u'$; ω_{save} frequency at which odor snapshots are saved $\times \tau_\eta$. For each simulation, first row reports results in non dimensional units. Second and third rows provide an idea of how non-dimensional parameters match dimensional parameters in real flows in air and water, assuming the Kolmogorov length is 1.5 mm in air and 0.4 mm in water.

where M_i denotes the i -th element of M and $C_{\text{thr}} > 0$ is a scaling constant (in our experiments we set it as 0.5). T denotes the cardinality of M . Given a memory M , we can define the filtered memory Δ^M as the set which contains every element of the memory M that is higher than the sensitivity threshold $s_{\text{thr}}(M)$ i.e.

$$\Delta^M := \{z \in M \mid z > s_{\text{thr}}(M)\}. \quad (4)$$

Then at timestep t , given the agent memory M_t , we define the average intensity $c(M_t)$ and the intermittency $i(M_t)$ as:

$$c(M_t) := \begin{cases} \frac{1}{|\Delta^{M_t}|} \sum_{i=1}^{|\Delta^{M_t}|} (\Delta^{M_t})_i, & |\Delta^{M_t}| > 0 \\ 0 & \end{cases}, \quad (5)$$

$$i(M_t) := \frac{|\Delta^{M_t}|}{|M_t|}.$$

Note that the average intensity is defined on the filtered memory Δ^M , i.e. conditioned to detecting odors above threshold. Since the features defined in (5) returns real numbers, in order to use (tabular) q-learning, we need to discretize them. We denote with $\bar{i}(M_t)$ the discretized intermittency. This is defined as follow

$$\bar{i}(M_t) := \begin{cases} 0, & \text{if } i(M_t) \leq 0.33 \\ 1, & \text{if } 0.33 < i(M_t) \leq 0.66 \\ 2, & \text{if } i(M_t) > 0.66 \end{cases}. \quad (6)$$

The average intensity is bounded between zero and the maximum concentration of odor at the source. To avoid prior information on the source, we use a more structured procedure to discretize the average intensity online, based on the agent's experience only. At every timestep t , the average intensity $c(M_t)$ is computed and collected in a dataset X_t i.e.

$$X_t := \{c(M_0), \dots, c(M_t)\}.$$

Then, its discretized value is obtained by the following rule:

$$\bar{c}(M_t, X_t) := \begin{cases} 0, & c(M_t) \leq p(X_t, 25) \\ 1, & p(X_t, 25) < c(M_t) \leq p(X_t, 50) \\ 2, & p(X_t, 50) < c(M_t) \leq p(X_t, 80) \\ 3, & p(X_t, 80) < c(M_t) \leq p(X_t, 99) \\ 4, & c(M_t) > p(X_t, 99) \end{cases}, \quad (7)$$

where $p(X_t, n)$ denotes the n -th percentile of X_t . Finally, we can define the feature map ϕ_t as a function of the memory M_t and the dataset of average intensities X_t at timestep t

$$\phi_t(M_t, X_t) := [\bar{i}(M_t), \bar{c}(M_t, X_t)].$$

This defines the current olfactory state s_t i.e. at timestep t , the agent is in the olfactory state $o_t := \varphi_t(M_t, X_t)$. The case where the agent has no odor detections above threshold in its current memory, i.e. $|\Delta(M_t)| = 0$ corresponds to an additional state called void state ($\emptyset$) in the main text.

Agent Behavior and Policies

Now, we describe how the agent interacts with the environment to solve the navigation problem. At every time step $t \in \mathbb{N}$, the agent observes an odor point z_t and updates its memory including the new observation and removing the oldest i.e. it defines a memory M_t with the following rule

$$M_t := \left[\left(M_{t-1} \right)_2, \dots, \left(M_{t-1} \right)_{|M_{t-1}|}, o_t \right]. \quad (8)$$

Then, it updates the dataset of average intensities i.e. $X_t := X_{t-1} \cup \{c(M_t)\}$ and it computes the olfactory state o_t . According to o_t , the agent chooses an action a_t , using a policy. As indicated in the main text, actions are the coordinate directions i.e. we define an action set $\mathbf{A}$ as follow

$$\mathcal{A} := \{e_1, e_2, -e_1, -e_2\},$$

where e_i denotes the i -th canonical base. Actions are steps in any of the 4 directions, labeled relative to the mean flow which assumed fixed and known. The gridworld is infinite, in that agents can leave indefinitely. If they exit the simulation box, they continue to receive zero signal and negative reward -0.001. As explained in the main text, actions are selected using one of two policies according to the current olfactory state o_t . More precisely, if the olfactory state o_t is not the void state, then the (ϵ -greedy) Q-learning policy is used. Formally, let Q be the Q matrix of the agent and let $o_t \neq \emptyset$, then the agent plays the action a_t such that

$$a_t = \begin{cases} a \in \arg \max_{a \in \mathcal{A}} Q(o_t, a) & \text{with probability } 1 - \epsilon \\ a \sim \mathcal{U}(\mathcal{A}) & \text{with probability } \epsilon \end{cases}, \quad (9)$$

where, with $a \sim \mathbf{V}(\mathbf{A})$, we indicate an action a uniformly sampled from $\mathbf{A}$. At test phase, the exploration-exploitation parameter ϵ is set to 0 and, thus, in an olfactory state $o_t \neq \emptyset$ the policy is deterministic. While training phase behavior is described in next paragraphs. In the void state $o_t = \emptyset$, the agent chooses the action $a_t \in \mathbf{A}$ according to a separated policy called *recovery strategy*. In our experiments, we defined and compared three different recovery strategies: Brownian, Backtracking and Learned.

Brownian recovery

It is the simplest strategy we consider, consisting of playing random actions in the void state. Suppose that at time step t , the agent is in the void olfactory state, i.e., $o_t = \emptyset$, then a_t is sampled uniformly from the action set $\mathbf{A}$. However, it is important to note for long memories agents start to recover when they are already far from the plume, and hitting the plume by random walk is prohibitively long. To avoid wandering away from the plume, the memory is constrained to be shorter, consistent with the observation that the optimal memory is $T^* = 3$ to 5, much shorter than for backtracking. At this memory, several blanks within the plume will cause the agent to recover, hence the lower performance of the Brownian recovery.

Backtracking Recovery

In order to accelerate recovery from accidentally exiting the plume, we let the agents backtrack to the position where they last detected the odor. To this end, we first enumerate the actions with numbers from one to four. Then we introduce a new memory called *action memory* A . For

simplicity, we consider the setting in which $|A| = |M|$. At time-step $t = 0$, this memory is initialized as a vector of zeros indicating that the action memory is empty i.e. we define $A_0 \in \mathbb{N}^{|M|}$ such that for every $i = 1, \dots, |A|$

$$A_i = 0.$$

For every timestep $t > 0$, the agent observes an odor point z_t and updates the memory through (8). Moreover, the action memory is updated according to the status of the memory. If the last observation is smaller than the sensitivity threshold i.e. $z_t < s_{\text{thr}}(M_t)$, the action previously played a_{t-1} (represented by a natural number in $[1, 4]$) is stored in the action memory i.e. for some $\Delta > 0$, let

$$A_{t-1} = [a_{t-\Delta}, \dots, a_{t-2}, 0, \dots, 0].$$

Then

$$A_t = [a_{t-\Delta}, \dots, a_{t-2}, a_{t-1}, \dots, 0].$$

If at time-step t , the observation z_t is larger than the sensitivity threshold then the action memory is reset i.e. $A_t \in \mathbb{N}^{|M|}$ with $(A_t)_i = 0$ for every i . If at timestep t , the memory is empty i.e. $c(M_t) = 0$, then the backtracking procedure is executed: the last non-zero element of the action memory is extracted and the inverse action is played i.e. For some $\Delta > 0$, let

$$A_{t-1} = [a_{t-\Delta}, \dots, a_{t-2}].$$

Then, it plays the action a_{t-2} and updates the action memory as follow

$$A_t = [a_{t-\Delta}, \dots, a_{t-3}, 0].$$

This procedure is repeated until either an observation larger than the sensitivity threshold is obtained or the action memory becomes empty. In the former case, the action memory is cleared and the action is chosen according to the Q-learning policy ((9)). In the latter case, a random action is played.

Note that this strategy only provides exploration after the backtracking fails to recover detections. Also, if agents start with no detection at time 0, the procedure is equivalent to Brownian motion.

Circling recovery

In this case, the recovery strategy consists of adopting a circling behavior [Stupski and van Breugel \(2024\)](#) [\[24\]](#) when the agent is in a void state. The agent keeps in memory two counters, t_{void} and T_{change} , as well as an action $a_{\text{void}} \in \mathbf{A}$, initialized to 0, 1, and e_1 , respectively. The first counter represents the consecutive number of void observations, a_{void} is the action to play when the void state is reached, and T_{change} is a time threshold that indicates when to switch the action a_{void} . When the agent reaches a void state, it plays the action a_{void} and increments the counter t_{void} by one. If $t_{\text{void}} = T_{\text{change}}$, then the agent resets t_{void} to zero, increases T_{change} by one, and updates the action a_{void} as follows

$$a_{\text{void}} \leftarrow \begin{cases} e_2 & \text{if } a_{\text{void}} = e_1 \\ -e_1 & \text{if } a_{\text{void}} = e_2 \\ -e_2 & \text{if } a_{\text{void}} = -e_1 \\ e_1 & \text{if } a_{\text{void}} = -e_2 \end{cases}$$

When the agent receives a non-void observation, it resets t_{void} to 0, T_{change} to 1, and a_{void} to e_1 .

Cast-Surge recovery

In this case, the agent plays a cast-surge behavior [Baker \(1990\)](#) when reaching the void state. As in the circling recovery strategy, the agent keeps two counters, t_{void} and T_{change} , as well as an action $a_{\text{void}} \in \mathbf{A}$, initialized to 0, 1, and e_2 , respectively. At every step in the void state, the agent plays the action a_{void} and increments the void counter t_{void} by one. If $t_{\text{void}} = T_{\text{change}}$, the agent takes an upwind step, doubles the value of T_{change} , updates a_{void} by setting it to $-a_{\text{void}}$, and resets t_{void} to zero. When the agent receives a non-void observation, it resets t_{void} to 0, T_{change} to 1, and a_{void} to e_2 .

Learned recovery

In this case recovery policy is learned by splitting the void state in several states labeled by the time since entry in the void state. In our experiments, we split the void state in 30 states. Actions are then learned as in all other non-void states and the optimal action is always chosen with (9).

Training

An agent start at a random location within the odor plume at time 0. Its memory is initialized with the prior $|M_0|$ odor detections at its initial location $M_0 = [z_{-|M|}, \dots, z_0]$, obtained from the fluid dynamics simulation. The Q-function Q_0 is initialized with 0.6 for all actions and olfactory states. The first dataset of average intensities contains the first value $X_0 = \{c(M_0)\}$. At every times step $t > 0$, the agent gets an odor observation z_t from its new position and updates its memory including the new observation and removing the oldest and the olfactory state o_t is computed (as described in previous paragraphs). The dataset of average intensities is updated: $X_t = X_{t-1} \cup \{c(M_t)\}$. Exploration-exploitation parameter ϵ_k is scheduled as follow

$$\epsilon_k = \eta_{\text{init}} \exp(-\eta_{\text{decay}} k),$$

where, in our experiments, $\eta_{\text{init}} = 0.99$ and $\eta_{\text{decay}} = 0.0001$. At every episode k , the Q-function is updated at every time step t as

$$Q_{k+1}(s_t, a_t) := (1 - \alpha_k)Q_k(s_t, a_t) + \alpha_k(r_t + \gamma \max_{a'} Q_k(s_{t+1}, a')),$$

where R_t is the immediate reward received playing the action a_t . o_t and o_{t+1} are the current and the next olfactory states and α_k is the learning rate at episode k . This is scheduled as

$$\alpha_k = \alpha_{\text{init}} \exp(-\alpha_{\text{decay}} k),$$

where, in our experiments, $\alpha_{\text{init}} = 0.25$ and $\alpha_{\text{decay}} = 0.001$. For the experiments, agents are trained in 100000 episodes and an horizon of 5000 steps. The agent velocity is set to 10 and the discount factor is $\gamma = 0.9999$.

Agents Evaluation

To evaluate the performance of the different agents, we consider four metrics: the cumulative reward G (which is the actual quantity that the algorithm optimizes for); normalized time (defined below); the fraction of success f^+ and the value conditioned on success g^+ . For a fixed position (i, j) , we denote with $\tau_{\text{min}}(i, j)$ the minimum number of steps required to reach the source region from (i, j) i.e. the length of the shortest path.

We define D_{init} the set of points in which the first observation is above the sensitivity threshold (valid points). For each initial position $(i, j) \in D_{\text{init}}$, let $\tau(i, j)$ be the duration of the path obtained by an agent to reach the source. Note that $\tau(i, j)$ is a random variable for the stochastic backtracking and Brownian recoveries, but it is deterministic for the learned strategy that has no random components. For each admissible location (i, j) , we define four performance metrics:

$$\begin{aligned} G(i, j) &= \langle e^{-\lambda\tau(i,j)} - \frac{\sigma}{1-\gamma}(1 - e^{-\lambda\tau(i,j)}) \rangle \\ f^+(i, j) &= \frac{n_{\text{success}}(i,j)}{n_{\text{reps}}} \\ g^+(i, j) &= \langle e^{-\lambda\tau(i,j)} | \text{success} \rangle \\ \frac{\tau_{\min}}{\tau}(i, j) &= \langle \frac{\tau_{\min}(i,j)}{\tau(i,j)} \rangle \end{aligned}$$

where n_{reps} is the number of test trajectories from each admissible location, and we use $n_{\text{reps}} = 10$. We then compute statistics of the performance metrics over the D_{init} initial positions and report the average ($\langle \cdot \rangle$) and standard deviation (std). Note that for the learned strategy, $\tau(i, j)$ is deterministic, hence $f^+(i, j)$ is 0 or 1 and therefore we omit its standard deviation.

Data availability

The datasets and code used to perform the experiments are available at the following links

- code: <https://github.com/Akatsuki96/qNav>
- datasets: <https://doi.org/10.5281/zenodo.14655992>

Acknowledgements

This research was supported by grants to AS from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 101002724 RIDING), the Air Force Office of Scientific Research under award number FA8655-20-1-7028, and the National Institutes of Health (NIH) under award number R01DC018789. L. R. and M.R. acknowledge the financial support of the European Research Council (grant SLING 819789), the European Commission (Horizon Europe grant ELIAS 101120237), the US Air Force Office of Scientific Research (FA8655-22-1-7034), the Ministry of Education, University and Research (FARE grant ML4IP R205T7J2KP; grant BAC FAIR PE00000013 funded by the EU - NGEU) and the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216. M.R. is a member of the Gruppo Nazionale per l'Analisi Matematica, la Probabilità e le loro Applicazioni (GNAMPA) of the Istituto Nazionale di Alta Matematica (INdAM). This work represents only the view of the authors. The European Commission and the other organizations are not responsible for any use that may be made of the information it contains. We thank Francesco Viola for support and discussions regarding computational fluid dynamics as well as Antonio Celani, Venkatesh Murthy, Yujia Qi, Francesco Boccardo, Luca Gagliardi, Francesco Marcolli and Arnaud Ruymaekers for comments on the manuscript.

References

- Ache BW, Hein AM, Bobkov YV, Principe JC 2016) **Smelling Time: A Neural Basis for Olfactory Scene Analysis** *Trends Neurosci* **39**:649–655
- Ackels T, Erskine A, Dasgupta Dea 2021a) **Fast odour dynamics are encoded in the olfactory system and guide behaviour** *Nature* **593**:558
- Ackels T, Erskine A, Dasgupta D, Marin AC, Warner TPA, Tootoonian S, Fukunaga I, Harris JJ, Schaefer AT 2021b) **Fast odour dynamics are encoded in the olfactory system and guide behaviour** *Nature* **593**:558–563
- Allard C, Kang G, Kim J, Valencia-Montoya W, Hibbs R, Bellono N 2023) **Structural basis of sensory receptor evolution in octopus** *Nature* **616**:373
- Alvarez-Salvado E, Licata AM, Connor EG, McHugh MK, King BM, Stavropoulos N, Victor JD, Crimaldi JP, Nagel KI 2018) **Elementary sensory-motor transformations underlying olfactory navigation in walking fruit-flies** *eLife* **7**:e37815 <https://doi.org/10.7554/eLife.37815>
- Atema J 1996) **Eddy chemotaxis and odor landscapes: exploration of nature with animal sensors** *Biol Bull* **191**:129
- Baker KL, Dickinson M, Findley TM, Gire DH, Louis M, Suver MP, Verhagen JV, Nagel KI, Smear MC 2018) **Algorithms for Olfactory Search across Species** *Journal of Neuroscience* **38**:9383
- Baker TC 1990) **Upwind flight and casting flight: complementary and tonic systems used for location of sex pheromone sources by male moths** *Proc 10th Intl Symposium on Olfaction and Taste* :18
- Balkovsky E, Shraiman BI 2002) **Olfactory search at high Reynolds number** *Proc Nat Acad Sci* **99**:12589–93
- Belanger JH, Willis MA 1988) **Biologically-inspired search algorithms for locating unseen odor sources** *Proc IEEE Symp Intell Control (ISIC '98) and IEEE Symp Comp Intell Robot Autom (CIRA '98)* :265
- Berg HC 1975) **Chemotaxis in Bacteria** *Annual Review of Biophysics and Bioengineering* **4**:119–136
- Bobkov YV, Ache B 2007) **Intrinsically bursting olfactory receptor neurons** *J Neurophysiol* **97**:1052
- Boie SD, Connor EG, McHugh M, Nagel KI, Ermentrout GB, Crimaldi JP, Victor JD 2018) **Information-theoretic analysis of realistic odor plumes: What cues are useful for determining location?** *PLoS Comp Bio* **14**:e1006275
- van Breugel F, Dickinson MH 2014) **Plume-tracking behavior of flying *Drosophila* emerges from a set of distinct sensorymotor reflexes** *Curr Biol* **24**:274

- Cardé RT 2021) **Navigation along windborne plumes of pheromone and resource-linked odors** *Annual Review of Entomology* **66**:317
- Carey RM, Verhagen JV, Wesson DW, Pirez N, Wachowiak M 2009) **Temporal Structure of Receptor Neuron Input to the Olfactory Bulb Imaged in Behaving Rats** *Journal of Neurophysiology* **101**:1073–1088
- Cassandra AR, Kaelbling LP, Kurien JA 1996) **Acting under uncertainty: Discrete Bayesian models for mobile-robot navigation** *Proc IEEE/RSJ Internl Conf Intelligent Robots and Systems IROS '96* **2**:963
- Celani A, Panizon E 2024) **Olfactory search. in review**
- Celani A, Villermanx E, Vergassola M 2014) **Odor Landscapes in Turbulent Environments** *Phys Rev X* **4**:041015
- David CT, Kennedy JS, Ludlow AR 1983) **Finding of a sex pheromone source by gypsy moths released in the field** *Nature* **303**:804–806
- David Stupski SD, van Breugel F 2024) **Wind Gates Search States in Free Flight** *bioRxiv* <https://doi.org/10.1101/2023.11.30.569086>:1
- Demir M, Kadakia N, Anderson HD, Clark DA, Emonet T 2020a) **Walking Drosophila navigate complex plumes using stochastic decisions biased by the timing of odor encounters** *eLife* **9**:e57524
- Demir M, Kadakia N, Anderson HD, Clark DA, Emonet T 2020b) **Walking Drosophila navigate complex plumes using stochastic decisions biased by the timing of odor encounters** *eLife* **9**:e57524 <https://doi.org/10.7554/eLife.57524>
- Duplat J, Jouary A, Villermanx E 2010) **Entanglement rules for random mixtures** *Phys Rev Lett* **105**:034504
- Etienne AS, Jeffery KJ 2004) **Path integration in mammals** *Hippocampus* **14**:180–192 <https://doi.org/10.1002/hipo.10173>
- Etienne AS, Maurer R, Seguinot V 1996) **Path Integration in Mammals and its Interaction With Visual Landmarks** *Journal of Experimental Biology* **199**:201–209 <https://doi.org/10.1242/jeb.199.1.201>
- Falkovich G, Gawedzki K, Vergassola M 2001) **Particles and Fields in Fluid Turbulence** *Rev Mod Phys* **73**:913
- Gepner R, Mihovilovic Skanata M, Bernat NM, Kaplow M, Gershow M 2015) **Computations underlying Drosophila photo-taxis, odor-taxis, and multi-sensory integration** *eLife* **4**:e06229 <https://doi.org/10.7554/eLife.06229>
- Gibson G, Brady J 1985) **'Anemotactic' flight paths of tsetse flies in relation to host odour: a preliminary video study in nature of the response to loss of odour** *Physiological Entomology* **10**:395–406
- Gire D, Kapoor V, Arrighi-Allisan A, Seminara A, Murthy V 2016) **Mice develop efficient strategies for foraging and navigation using complex natural stimuli** *Curr Biol* **26**:1261

- Gulitski G, Kholmyansky M, Kinzelbach W, Luthi B, Tsinober A, Yorish S 2007) **Velocity and temperature derivatives in high-Reynolds-number turbulent flows in the atmospheric surface layer. Part 1. Facilities, methods and some general results** *Journal of Fluid Mechanics* **589**:57–81
- Hauskrecht M 2000) **Value-function approximations for partially observable Markov decision processes** *J Artif Intell Res* **13**:33
- Heinonen RA, Biferale L, Celani A, Vergassola M 2023) **Optimal policies for Bayesian olfactory search in turbulent flows** *Phys Rev E* **107**:055105 <https://doi.org/10.1103/PhysRevE.107.055105>
- Heinze S, Narendra A, Cheung A 2018) **Principles of Insect Path Integration** *Current Biology* **28**:R1043
- Hernandez-Nunez L, Belina J, Klein M, Si G, Claus L, Carlson JR, Samuel AD 2015) **Reverse-correlation analysis of navigation dynamics in Drosophila larva using optogenetics** *eLife* **4**:e06225 <https://doi.org/10.7554/eLife.06225>
- Ishida H 2020) **Chemical sensing in robotic applications: a review** *IEEE Sensors, J* **12**:3163
- Kathman ND, Lanz AJ, Freed JD, Nagel KI 2024) **Neural dynamics for working memory and evidence integration during olfactory navigation in Drosophila** *biorXiv*
- Kramer E 1997) **A tentative intercausal nexus and its computer model on insect orientation in windborne pheromone plumes** *Insect Pher Res, New Dir* :232
- Krishnamurthy V 2016) **Partially Observed Markov Decision Processes** Cambridge University Press
- Kuenen LPS, Cardé RT 1994) **Strategies for recontacting a los pheromone plume: casting and upwind flight in the male gypsy moth** *Physiological Entomology* **15**:317
- LaValle SM 2006) **Planning algorithms** Cambridge University Press
- Liao JC 2006) **The role of the lateral line and vision on body kinematics and hydrodynamic preference of rainbow trout in turbulent flow** *Journal of Experimental Biology* **209**:4077–4090 <https://doi.org/10.1242/jeb.02487>
- Loisy A, Eloy C 2022) **Searching for a source without gradients, how good is infotaxis and how to beat it** *Proc R Soc A* **478**:20220118
- Loisy A, Heinonen RA 2023a) **Deep reinforcement learning for the olfactory search POMDP: a quantitative benchmark** *European Physical Journal E* **46**:17
- Loisy A, Heinonen RA 2023b) **Deep reinforcement learning for the olfactory search POMDP: a quantitative benchmark** *Cereb CortexThe European Physical Journal E* **46**:17
- Matheson AMM, Lanz AJ, Medina AM, Licata AM, Currier TA, Syed MH, Nagel KI 2022) **A neural circuit for wind-guided olfactory navigation** *Nature Communications* **13**:4613
- Michaelis B, Leathers K, Bobkov Y, Ache B, Principe J, Baharloo R, Park I, Reidenbach M 2020) **Odor tracking in aquatic organisms: the importance of temporal and spatial intermittency of the turbulent plume** *Sci Rep* **10**:7961

Murlis J, Elkinton JS, Cardé RT 1992) **Odor plumes and how insects use them** *Annual Review of Entomology* **37**:505

Murlis J, Jones CD 1981) **Fine-scale structure of odour plumes in relation to insect orientation to distant pheromone and other attractant sources** *Physiological Entomology* **6**:71–86

Nag A, van Breugel F 2024) **Odour source distance is predictable from a time history of odour statistics for large scale outdoor plumes** *J R Soc Interface* **21**:2120240169

Park IJ, Hein AM, Bobkov YV, Reidenbach MA, Ache BW, Principe JC 2016) **Neurally Encoding Time for Olfactory Navigation** *Plos Comput Biol* **12**:e1004682

Radvansky BA, Dombeck DA 2018) **An olfactory virtual reality system for mice** *Nature Communications* **9**:839

Reddy G, Murthy VN, Vergassola M 2022) **Olfactory Sensing and Navigation in Turbulent Environments** *Annual Review of Condensed Matter Physics* **13**:191–213 <https://doi.org/10.1146/annurev-conmatphys-031720-032754>

Reynolds AM, Reynolds DR, Smith AD, Chapman JW 2010) **Orientation Cues for High-Flying Nocturnal Insect Migrants: Do Turbulence-Induced Temperature and Velocity Fluctuations Indicate the Mean Wind Flow?** *Plos ONE* **5**:e15758

Rigolli N, Magnoli N, Rosasco L, Seminara A 2022a) **Learning to predict target location with turbulent odor plumes** *eLife* **11**:e72196 <https://doi.org/10.7554/eLife.72196>

Rigolli N, Reddy G, Seminara A, Vergassola M 2022b) **Alternation emerges as a multi-modal strategy for turbulent odor navigation** *eLife* **11**:e76989

Rigolli N, Reddy G, Seminara A, Vergassola M 2022c) **Alternation emerges as a multi-modal strategy for turbulent odor navigation - Dataset** *Zenodo* <https://doi.org/10.5281/zenodo.6538177>

Save E, Nerad L, Poucet B 2000) **Contribution of multiple sensory information to place field stability in hippocampal place cells** *Hippocampus* **10**:64

Schal C 1982) **Intraspecific vertical stratification as a mate-finding mechanism in tropical cockroaches** *Science* **215**:1505

Shani G, Pineau J, Kaplow R 2013) **A survey of point-based pomdp solvers** *Autonomous Agents and MultiAgent Systems* **27**:1–51

Shraiman B, Siggia E 2000) **Scalar Turbulence** *Nature* **405**:639

Singh SH, van Breugel F, Rao RPN, Brunton BW 2023) **Emergent behaviour and neural dynamics in artificial agents tracking odour plumes** *Nature Machine Intelligence* **5**:58–70

Stupski SD, van Breugel F 2024) **Wind gates olfaction-driven search states in free flight** *Current Biology* **34**:4397–4411

Sutton R, Barto A 1998) **Reinforcement Learning: An Introduction** MIT Press

- Torr SJ 1988) **Behaviour of tsetse flies (*Glossina*) in host odour plumes in the field** *Physiological Entomology* **13**:467–478
- Verano KVB, Panizon E, Celani A 2023) **Olfactory search with finite-state controllers** *Proc Nat Acad Sci* **120**:e2304230120
- Vergassola M, Villerman E, Shraiman BI 2007) **‘Infotaxis’ as a strategy for searching without gradients** *Nature* **445**:406
- Viola F, Meschini V, Verzicco R 2020) **Fluid-structure-electrophysiology interaction (FSEI) in the left-heart: a multi-way coupled computational model** *European Journal of Mechanics-B/Fluids* **79**:212–232
- Willis MA, Avondet JL, Finnell AS 2008) **Effects of altering flow and odor information on plume tracking behavior in walking cockroaches, *Periplaneta americana* (L.)** *The Journal of Experimental Biology* **211**:2317
- Wolf H, Wehner R 2000) **Pinpointing Food Sources: Olfactory and Anemotactic Orientation in Desert Ants, *Cataglyphis Fortis*** *Journal of Experimental Biology* **203**:857–868
- Wystrach A, Schwarz S, Baniel A, Cheng K 2013) **Backtracking behaviour in lost ants: an additional strategy in their navigational toolkit** *Proc Roy Soc B* **280**:20131677
- Yu YSW, Graff MM, Bresee CS, Man YB, Hartmann MJZ 2016) **Whiskers aid anemotaxis in rats** *Science Advances* **2**:e1600716 <https://doi.org/10.1126/sciadv.1600716>
- Zhang S, Manahan-Vaughan D 2015) **Spatial olfactory learning contributes to place field formation in the hippocampus** *Cereb Cortex* **25**:423–32
- Rando M., James M., Verri A., Rosasco L., Seminara A 2025) **Q-learning with temporal memory to navigate turbulence** *Zenodo* <https://doi.org/10.5281/zenodo.14655992>

Author information

Marco Rando

MaLGA, Department of Computer Science, Bioengineering, Robotics and Systems Engineering, University of Genova, Genova, Italy

For correspondence: rando@edu.unige.it

Martin James

MaLGA, Department of Civil, Chemical and Environmental Engineering, University of Genoa, Genoa, Italy

Alessandro Verri

MaLGA, Department of Computer Science, Bioengineering, Robotics and Systems Engineering, University of Genova, Genova, Italy

Lorenzo Rosasco

MaLGA, Department of Computer Science, Bioengineering, Robotics and Systems Engineering,
University of Genova, Genova, Italy

Agnese Seminara

MaLGA, Department of Civil, Chemical and Environmental Engineering, University of Genoa,
Genoa, Italy

For correspondence: agnese.seminara@unige.it

Editors

Reviewing Editor

Gordon Berman

Emory University, Atlanta, United States of America

Senior Editor

Aleksandra Walczak

École Normale Supérieure - PSL, Paris, France

Reviewer #1 (Public review):

Overall I found the approach taken by the authors to be clear and convincing. It is striking that the conclusions are similar to those obtained in a recent study using a different computational approach (finite state controllers), and lends confidence to the conclusions about the existence of an optimal memory duration. There are a few questions that could be expanded on in future studies:

(1) Spatial encoding requirements

The manuscript contrasts the approach taken here (reinforcement learning in a gridworld) with strategies that involve a "spatial map" such as infotaxis. However, the gridworld navigation algorithm has an implicit allocentric representation, since movement can be in one of four allocentric directions (up, down, left, right), and wind direction is defined in these coordinates. Future studies might ask if an agent can learn the strategy without a known wind direction if it can only go left/right/forward/back/turn (in egocentric coordinates). In discussing possible algorithms, and the features of this one, it might be helpful to distinguish (1) those that rely only on egocentric computations (run and tumble), (2) those that rely on a single direction cue such as wind direction, (3) those that rely on allocentric representations of direction, and (4) those that rely on a full spatial map of the environment.

(2) Recovery strategy on losing the plume

The authors explore several recovery strategies upon losing the plume, including backtracking, circling, and learned strategies, finding that a learned strategy is optimal. As insects show a variety of recovery strategies that can depend on the model of locomotion, it would be interesting in the future to explore under which conditions various recovery strategies are optimal and whether they can predict the strategies of real animals in different environments.

(3) Is there a minimal representation of odor for efficient navigation?

The authors suggest that the number of olfactory states could potentially be reduced to reduce computational cost. They show that reducing the number of olfactory states to 1

dramatically reduces performance. In the future it would be interesting to identify optimal internal representations of odor for navigation and to compare these to those found in real olfactory systems. Does the optimal number of odor and void states depend on the spatial structure of the turbulence as explored in Figure 5?

<https://doi.org/10.7554/eLife.102906.2.sa2>

Reviewer #2 (Public review):

Summary:

The authors investigate the problem of olfactory search in turbulent environments using artificial agents trained using tabular Q-learning, a simple and interpretable reinforcement learning (RL) algorithm. The agents are trained solely on odor stimuli, without access to spatial information or prior knowledge about the odor plume's shape. This approach makes the emergent control strategy more biologically plausible for animals navigating exclusively using olfactory signals. The learned strategies show parallels to observed animal behaviors, such as upwind surging and crosswind casting. The approach generalizes well to different environments and effectively handles the intermittency of turbulent odors.

Strengths:

- * The use of numerical simulations to generate realistic turbulent fluid dynamics sets this paper apart from studies that rely on idealized or static plumes.
- * A key innovation is the introduction of a small set of interpretable olfactory states based on moving averages of odor intensity and sparsity, coupled with an adaptive temporal memory.
- * The paper provides a thorough analysis of different recovery strategies when an agent loses the odor trail, offering insights into the trade-offs between various approaches.
- * The authors provide a comprehensive performance analysis of their algorithm across a range of environments and recovery strategies, demonstrating the versatility of the approach.
- * Finally, the authors list an interesting set of real-world experiments based on their findings, that might invite interest from experimentalists across multiple species.

Weaknesses:

- * Using tabular Q-learning is both a strength and a limitation. It's simple and interpretable, making it easier to analyze the learned strategies, but the discrete action space seems somewhat unnatural. In real-world biological systems, actions (like movement) are continuous rather than discrete. Additionally, the ground-frame actions may not map naturally to how animals navigate odor plumes (e.g. insects often navigate based on their own egocentric frame).

<https://doi.org/10.7554/eLife.102906.2.sa1>

Author response:

The following is the authors' response to the original reviews.

Reviewer #1 (Public review):

Overall I found the approach taken by the authors to be clear and convincing. It is striking that the conclusions are similar to those obtained in a recent study using a different computational approach (finite state controllers), and lend confidence to the conclusions about the existence of an optimal memory duration. There are a few points or questions that could be addressed in greater detail in a revision:

(1) Discussion of spatial encoding

The manuscript contrasts the approach taken here (reinforcement learning in a grid world) with strategies that involve a "spatial map" such as infotaxis. The authors note that their algorithm contains "no spatial information." However, I wonder if further degrees of spatial encoding might be delineated to better facilitate comparisons with biological navigation algorithms. For example, the gridworld navigation algorithm seems to have an implicit allocentric representation, since movement can be in one of four allocentric directions (up, down, left, right). I assume this is how the agent learns to move upwind in the absence of an explicit wind direction signal. However, not all biological organisms likely have this allocentric representation. Can the agent learn the strategy without wind direction if it can only go left/right/forward/back/turn (in egocentric coordinates)? In discussing possible algorithms, and the features of this one, it might be helpful to distinguish

- (1) those that rely only on egocentric computations (run and tumble),*
- (2) those that rely on a single direction cue such as wind direction,*
- (3) those that rely on allocentric representations of direction, and*
- (4) those that rely on a full spatial map of the environment.*

As Referee 1 points out, even if the algorithm does not require a map of space, the agent is still required to tell apart directions relative to the wind direction which is assumed known. Indeed, although in the manuscript we labeled actions allocentrically as “up down left and right”, the source is always placed in the same location, hence “left” corresponds to upwind; “right” to downwind and “up” and “down” to crosswind right and left. Thus in fact directions are relative to the mean wind, which is therefore assumed known. We have better clarified the spatial encoding required to implement these strategies, and re-labeled the directions as upwind, downwind, crosswind-right and crosswind-left.

In reality, animals cannot measure the mean flow, but rather the local flow speed e.g. with antennas for insects, with whiskers for rodents and with the lateral line for marine organisms. Further work is needed to address how local flow measures enable navigation using Q learning.

(2) Recovery strategy on losing the plume

While the approach to encoding odor dynamics seems highly principled and reaches appealingly intuitive conclusions, the approach to modeling the recovery strategy seems to be more ad hoc. Early in the paper, the recovery strategy is defined to be path integration back to the point at which odor was lost, while later in the paper, the authors explore Brownian motion and a learned recovery based on multiple "void" states. Since the learned strategy works best, why not first consider learned strategies, and explore how lack of odor must be encoded or whether there is an optimal division of void states that leads to the best recovery strategies? Also, although the authors state that the learned recovery strategies resemble casting, only minimal data are shown to support this. A deeper statistical analysis of the learned recovery strategies would facilitate comparison to those observed in biology.

We thank Referee 1 for their remarks and suggestion to give the learned recovery a more prominent role and better characterize it. We agree that what is done in the void state is definitely key to turbulent navigation. In the revised manuscript, we have further substantiated the statistics of the learned recovery by repeating training 20 times and comparing the trajectories in the void (Figure 3 figure supplement 3, new Table 1). We believe however that starting with the heuristic recovery is clearer because it allows to introduce the concept of recovery more clearly. Indeed, the learned “recovery” is so flexible that it ends up

mixing recovery (crosswind motion) to aspects of exploitation (surge): we defer a more in-depth analysis that disentangles these two aspects elsewhere. Also, we added a whole new comparison with other biologically inspired recoveries both in the native environment and for generalization (Figure 3 and 5).

(3) Is there a minimal representation of odor for efficient navigation?

The authors suggest (line 280) that the number of olfactory states could potentially be reduced to reduce computational cost. This raises the question of whether there is a maximally efficient representation of odors and blanks sufficient for effective navigation. The authors choose to represent odor by 15 states that allow the agent to discriminate different spatial regimes of the stimulus, and later introduce additional void states that allow the agent to learn a recovery strategy. Can the number of states be reduced or does this lead to loss of performance? Does the optimal number of odor and void states depend on the spatial structure of the turbulence as explored in Figure 5?

We thank the referee for their comment. Q learning defines the olfactory states prior to training and does not allow a systematic optimization of odor representation for the task. We can however compare different definitions of the olfactory states, for example based on the same features but different discretizations. We added a comparison with a drastically reduced number of non-empty olfactory states to just 1, i.e. if the odor is above threshold at any time within the memory, the agent is in the non-void olfactory state, otherwise it is in the void state. This drastic reduction in the number of olfactory states results in less positional information and degrades performance (Figure 5 figure supplement 5).

The number of void states is already minimal: we chose 50 void states because this matches the time agents typically remain in the void (less than 50 void states results in no convergence and more than 50 introduces states that are rarely visited).

One may instead resort to deep Q-learning or to recurrent neural networks, which however do not provide answers as for what are the features or olfactory states that drive behavior (see discussion in manuscript and questions below).

Reviewer #2 (Public review):

Summary:

The authors investigate the problem of olfactory search in turbulent environments using artificial agents trained using tabular Q-learning, a simple and interpretable reinforcement learning (RL) algorithm. The agents are trained solely on odor stimuli, without access to spatial information or prior knowledge about the odor plume's shape. This approach makes the emergent control strategy more biologically plausible for animals navigating exclusively using olfactory signals. The learned strategies show parallels to observed animal behaviors, such as upwind surging and crosswind casting. The approach generalizes well to different environments and effectively handles the intermittency of turbulent odors.

Strengths:

(1) The use of numerical simulations to generate realistic turbulent fluid dynamics sets this paper apart from studies that rely on idealized or static plumes.

(2) A key innovation is the introduction of a small set of interpretable olfactory states based on moving averages of odor intensity and sparsity, coupled with an adaptive temporal memory.

(3) The paper provides a thorough analysis of different recovery strategies when an agent loses the odor trail, offering insights into the trade-offs between various approaches.

(4) The authors provide a comprehensive performance analysis of their algorithm across a range of environments and recovery strategies, demonstrating the versatility of the approach.

(5) Finally, the authors list an interesting set of real-world experiments based on their findings, that might invite interest from experimentalists across multiple species.

Weaknesses:

(1) The inclusion of Brownian motion as a recovery strategy, seems odd since it doesn't closely match natural animal behavior, where circling (e.g. flies) or zigzagging (ants' "sector search") could have been more realistic.

We agree that Brownian motion may not be biologically plausible -- we used it as a simple benchmark. We clarified this point, and re-trained our algorithm with adaptive memory using circling and zigzagging (cast and surge) recoveries. The learned recovery outperforms all heuristic recoveries (Figure 3D, metrics G). Circling ranks second, and achieves these good results by further decreasing the probability of failure and paying slightly in speed. When tested in the non-native environments 2 to 6, the learned recovery performs best in environments 2, 5 and 6 i.e. from long range more relevant to flying insects; whereas circling generalizes best in odor rich environments 3 and 4, representative of closer range and close to the substrate (Figure 5B, metrics G). In the new environments, similar to the native environment, circling favors convergence (Figure 5B, metrics f^+) over speed (Figure 5B, metrics g^+ and $\tau_{\min}/\tau$), which is particularly deleterious at large distance.

(2) Using tabular Q-learning is both a strength and a limitation. It's simple and interpretable, making it easier to analyze the learned strategies, but the discrete action space seems somewhat unnatural. In real-world biological systems, actions (like movement) are continuous rather than discrete. Additionally, the ground-frame actions may not map naturally to how animals navigate odor plumes (e.g. insects often navigate based on their own egocentric frame).

We agree with the reviewer that animal locomotion does not look like a series of discrete displacements on a checkerboard. However, to overcome this limitation, one has to first focus on a specific system to define actions in a way that best adheres to a species' motor controls. Moreover, these actions are likely continuous, which makes reinforcement learning notoriously more complex. While we agree that more realistic models are definitely needed for a comparison with real systems, this remains outside the scope of the current work. We have added a remark to clarify this limitation.

(3) The lack of accompanying code is a major drawback since nowadays open access to data and code is becoming a standard in computational research. Given that the turbulent fluid simulation is a key element that differentiates this paper, the absence of simulation and analysis code limits the study's reproducibility.

We have published the code and the datasets at

- code: <https://github.com/Akatsuki96/qNav>

- datasets: <https://zenodo.org/records/14655992>

Recommendations for the authors:

Reviewer #1 (Recommendations for the authors):

(1) Line 59-69: In comparing the results here to other approaches (especially the Verano and Singh papers), it would also be helpful to clarify which of these include an explicit representation of the wind direction. My understanding is that both the Singh and Verano approaches include an explicit representation of wind direction. In Singh wind direction is one of the observations that inputs to the agent, while in Verano, the actions are defined relative to the wind direction. In the current paper, my understanding is that there is no explicitly defined wind direction, but because movement directions are encoded allocentrically, the agent is able to learn the upwind direction from the structure of the plume- is this correct? I think this information would be helpful to spell out and also to address whether an agent without any allocentric direction sense can learn the task.

Thank you for the comment. In our algorithm the directions are defined relative to the mean wind, which is assumed known, as in Verano et al. As far as we understand, Singh et al provide the instantaneous, egocentric wind velocities as part of the input.

(1) Line 105: "several properties of odor stimuli depend on the distance from the source" might cite Boie...Victor 2018, Ackles...Schaefer, 2021, Nag...van Breugel 2024.

Thank you for the suggestions - we have added these references

(2) Line 130: "we first define a finite set of olfactory states" might be helpful to the reader to state what you chose in this paragraph rather than further down.

We have slightly modified the incipit of the paragraph. We first declare we are setting out to craft the olfactory states, then define the challenges, finally we define the olfactory states.

(3) Line 267: "Note that the learned recovery strategy resembles casting behavior observed in flying insects" Might note that insects seem to deploy a range of recovery strategies depending on locomotor mode and environment. For example, flying flies circle and sink when odor is lost in windless environments (Stupski and van Breugel 2024).

Thank you for your comment. We have included the reference and we now added comparisons to results using circling and cast & surge recovery strategies.

(4) Line 289: "from positions beyond the source, the learned strategy is unable to recover the plume as it mostly casts sideways, with little to no downwind action" This is curious as many insects show a downwind bias in the absence of odor that helps them locate the plumes in the first place (e.g. Wolf and Wehner, 2000, Alvarez-Salvado et al. 2018). Is it possible that the agent could learn a downwind bias in the absence of odor if given larger environments or a longer time to learn?

The reviewer is absolutely correct – Downwind motion is not observed in the recovery simply because the agent rarely overshoots the source. Hence overall optimization for that condition is washed out by the statistics. We believe downwind motion will emerge if an agent needs to avoid overshooting the source – we do not have conclusive results yet but are planning to introduce such flexibility in a further work. We added this remark and refs.

(5) Line 377-391: *testing these ideas in living systems. Interestingly, Kathman..Nagel 2024 (bioRxiv) shows exactly the property predicted here and in Verano in fruit flies- an odor memory that outlasts the stimulus by a duration of several seconds, appropriate for filling in "blanks." Relatedly, Alvarez-Salvado et al. 2018 showed that fly upwind running reflected a temporal integration of odor information over ~10s, sufficient to avoid responding to blanks as loss of odor.*

Indeed, we believe this is the most direct connection between algorithms and experiments. We are excited to discuss with our colleagues and pursue a more direct comparison with animal behavior. We were aware of the references and forgot to cite them, thank you for your careful reading of our work !

Reviewer #2 (Recommendations for the authors):

Suggestions

(1) *The paper does not clearly specify which type of animals (e.g., flying insects, terrestrial mammals) the model is meant to approximate or not approximate. The authors should consider clarifying how these simulations are suited to be a general model across varied olfactory navigators. Further, it isn't clear how low/high the intermittency studied in this model is compared to what different animals actually encounter. (Minor: The Figure 4 occupancy circles visualization could be simplified).*

Environment 1 represents the lower layers of a moderately turbulent boundary layer. Search occurs on a horizontal plane ~half meter from the ground. The agent is trained at distances of about 10 meters and also tested on longer distances ~ 17 meters (environment 6), lower heights ~1cm from the ground (environments 3-4), lower Reynolds number (environment 5) and higher threshold of detection (environment 2 and 4). Thus Environments 1,2,5 and 6 are representative of conditions encountered by flying organisms (or pelagic in water), and Environments 3 and 4 of searches near the substrate, potentially involved in terrestrial navigation (benthic in water). Even near the substrate, we use odor dispersed in the fluid, and not odor attached to the substrate (relevant to trail tracking).

Also note that we pick Schmidt number $Sc = 1$ and this is appropriate for odors in air but not in water. However, we expect a weak dependence on the Schmidt number as the Batchelor and Kolmogorov scales are below the size of the source and we are interested in the large scale statistics Falkovich et al., 2001; Celani et al., 2014; Duplat et al., 2010.

Intermittency contours are shown in Fig 1C, they are highest along the centerline, and decay away from the centerline, so that even within the plume detecting odor is relatively rare. Only a thin region near the centerline has intermittency larger than 66%; the outer and most critical bin of the plume has intermittency under 33%; in the furthest point on the centerline intermittency is <10%. For reference, experimental values in the atmospheric boundary layer report intermittency 25% to 20% at 2 to 15m from the source along the centerline (Murlis and Jones, 1981).

We have more clearly labeled the contours in Fig 1C and added these remarks.

We included these remarks and added a whole table with matching to real conditions within the different environments.

(2) *Could some biological examples and references be added to support that backtracking is a biologically plausible mechanism?*

Backtracking was observed e.g. in ants displaced in unfamiliar environments (Wystrach et al, P Roy Soc B, 280, 2013), in tsetse flies executing reverse turns uncorrelated to wind, which bring them back towards the location where they last detected odor (Torr, Phys Entom, 13, 1988, Gibson & Brady Phys Entom 10, 1985) and in cockroaches upon loss of contact with the plume (Willis et al, J. Exp. Biol. 211, 2008). It is also used in computational models of olfactory navigation (Park et al, Plos Comput Biol, 12:e1004682, 2016).

(3) Hand-crafted features can be both a strength and a limitation. On the one hand, they offer interpretability, which is crucial when trying to model biological systems. On the other hand, they may limit the generality of the model. A more thorough discussion of this paper's limitations should address this.

(4) The authors mention the possibility of feature engineering or using recurrent neural networks, but a more concrete discussion of these alternatives and their potential advantages/disadvantages would be beneficial. It should be noted that the hand-engineered features in this manuscript are quite similar to what the model of Singh et al suggests emerges in their trained RNNs.

Merged answer to points 3 and 4.

We agree with the reviewer that hand-crafted features are both a strength and a limitation in terms of performance and generality. This was a deliberate choice aimed at stripping the algorithm bare of implicit components, both in terms of features and in terms of memory. Even with these simple features, our model performs well in navigating across different signals, consistent with our previous results showing that these features are a “good” surrogate for positional information.

To search for the most effective temporal features, one may consider a more systematic hand crafting, scaling up our approach. In this case one would first define many features of the odor trace; rank groups of features for their accuracy in regression against distance; train Q learning with the most promising group of features and rank again. Note however that this approach will be cumbersome because multiple factors will have to be systematically varied: the regression algorithm; the discretization of the features and the memory.

Alternatively, to eliminate hand crafting altogether and seek better performance or generalization, one may consider replacing these hand-crafted features and the tabular Q-learning approach with recurrent neural networks or with finite state controllers. On the flip side, neither of these algorithms will directly provide the most effective features or the best memory, because these properties are hidden within the parameters that are optimized for. So extra work is needed to interrogate the algorithms and extract these information. For example, in Singh et al, the principal components of the hidden states in trained agents correlate with head direction, odor concentration and time since last odor encounter. More work is needed to move beyond correlations and establish more systematically what are the features that drive behavior in the RNN.

We have added these points to the discussion.

(5) Minor: the title of the paper doesn't immediately signal its focus on recovery strategies and their interplay with memory in the context of olfactory navigation. Given the many other papers using a similar RL approach, this might help the authors position this paper better.

We agree with the referee and have modified the title to reflect this.

(6) Minor: L 331: *"because turbulent odor plumes constantly switch on and off" -- the signal received rather than the plume itself is switching on and off.*

Thank you for the suggestion, we implemented it.

<https://doi.org/10.7554/eLife.102906.2.sa0>