Olfactory Search with Finite State Controllers

Learning to Navigate in Complex Odor Landscapes

Kyrell Vann B. Verano
Emanuele Panizon
Antonio Celani

UNIVERSITÀ DEGLI STUDI DI TRIESTE

The Abdus Salam International Centre for Theoretical Physics

CPerMix
European ITN project

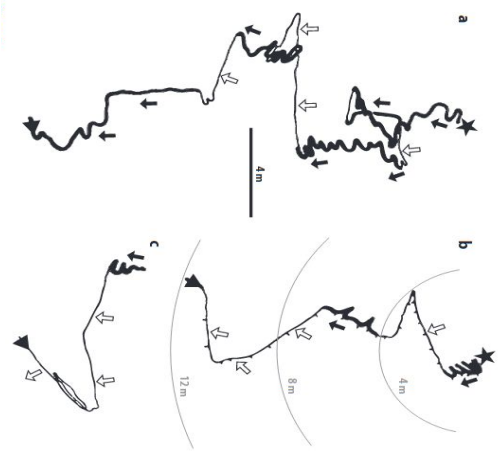
This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement N°956457.

Searching in Turbulent Environments

Turbulent transport leads to a dynamic, complex, and sparse odor landscape

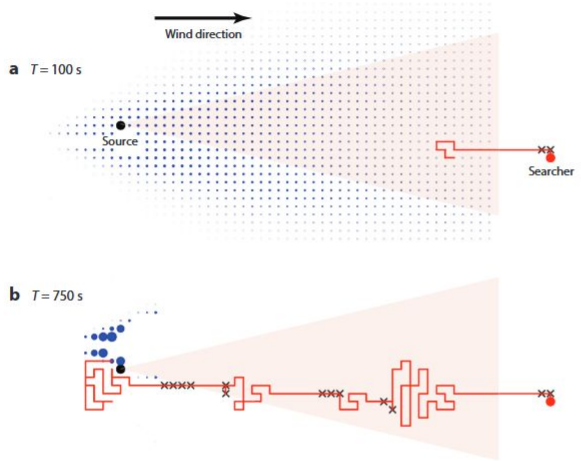
“A nonsequential turbulent mixing process” by Duplat J., Innocenti C., Villermaux E., 2010. Phys. Fluids 22:035104, https://doi.org/10.1063/1.3319821

Searchers are able to develop a strategy to cope with the fluctuations, e.g. moths “cast” and “surge”, to locate the source



“Finding of a sex pheromone source by gypsy moths released in the field” by David C, Kennedy J, Ludlow A. 1983. Nature 303:804–6

Using machine learning techniques, we study how the searcher develop such strategies to navigate through turbulent environment.



“**Olfactory Sensing and Navigation in Turbulent Environments”** by Reddy G, Murthy V, Vergassola M, Annu. Rev. Condens. Matter Phys. 2022. 13:191–213

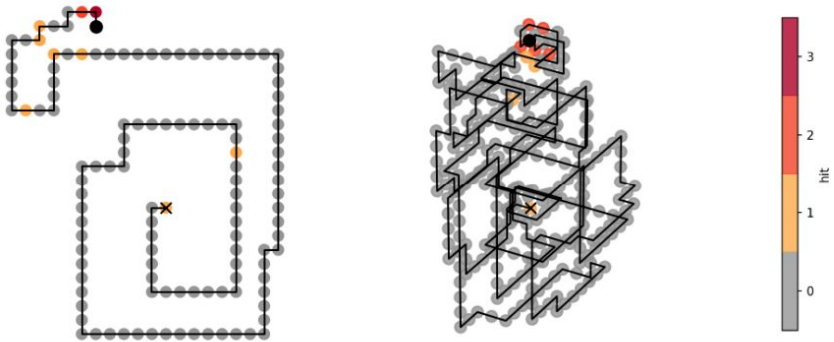
Review of Algorithmic Approaches for source-tracking tasks

model-based

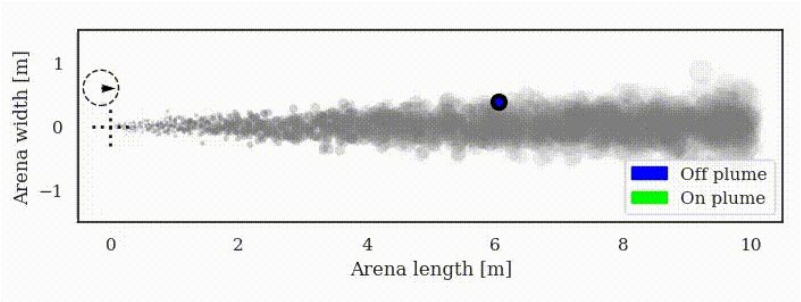
Strategy: Solve optimal strategy that maps belief state to actions
Challenge: a precise description of the environment is needed (which is often difficult to obtain for some complex landscapes) and a large memory is required to form the spatial map

model-free

Strategy: Deep Reinforcement Learning as a way to keep track of histories.
Challenge: It may need a large volume of data to train the network and the memory component may be high dimensional or difficult to interpret.

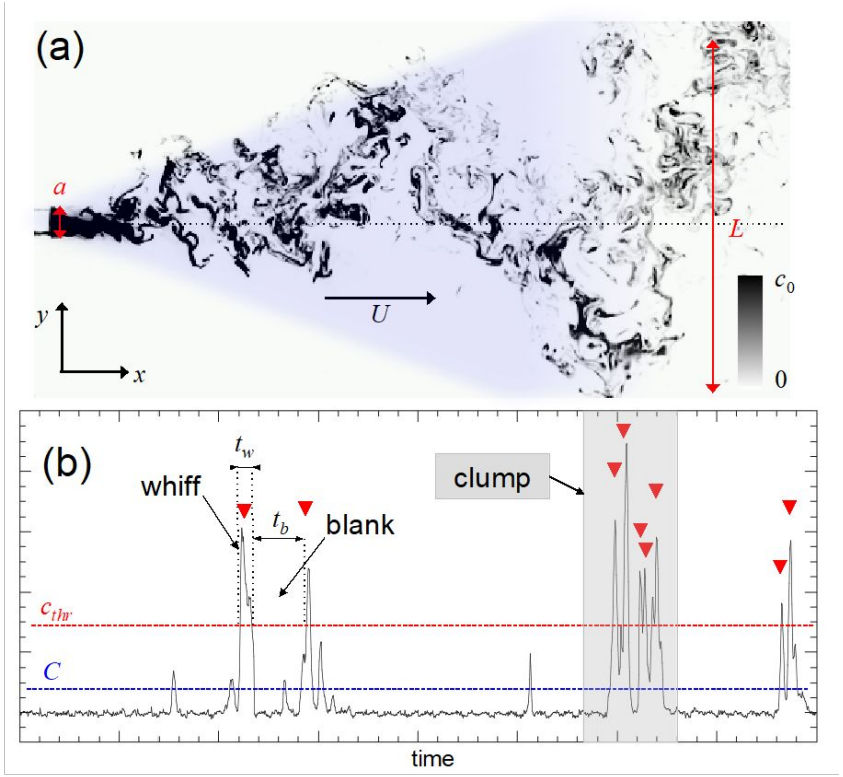


Searching for a source without gradients: how good is infotaxis and how to beat it by Loisy A and Eloy C. 2021. https://arxiv.org/abs/2112.10861v1



Emergent behavior and neural dynamics in artificial agents tracking turbulent plumes by Singh S, et.al. 2021. https://arxiv.org/abs/2109.12434v2

Memory is needed when signals are intermittent



Odor Landscapes in Turbulent Environments by A. Celani, E. Villermaux, M. Vergassola 2014, Physical Review X 4, DOI:10.1103/PhysRevX.4.041015

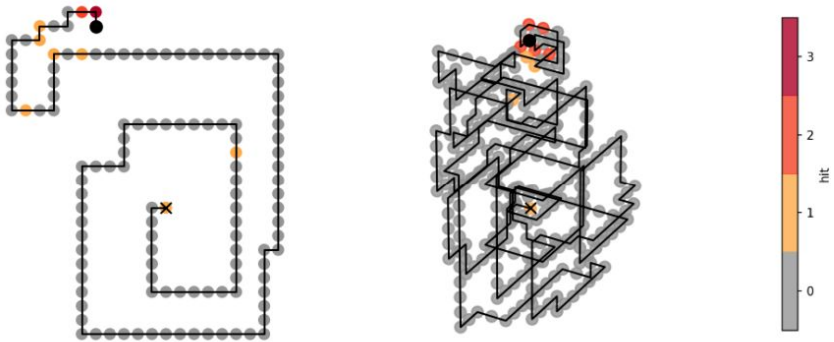
Due to the intermittent nature of the signals, searching for the source can be difficult: it needs to store the information that it received a signal some time ago.

For both settings, we want to propose an approach using a low dimensional memory in solving the the optimal strategy while navigating in a complex landscape (when information is sparsely available).
We call them as FINITE STATE CONTROLLERS or MEMORY AUGMENTED ALGORITHMS.

model-based

Strategy: Solve optimal behavior that maps belief state to actions
Cons:a precise description of the environment is needed (which is often difficult to obtain for some complex landscapes) and a large memory is required to form the spatial map

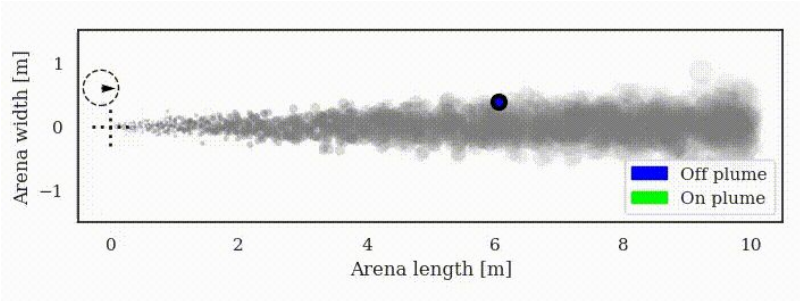
An agent tracks the source according to the signals it receive



Searching for a source without gradients: how good is infotaxis and how to beat it by Loisy A and Eloy C. 2021. https://arxiv.org/abs/2112.10861v1

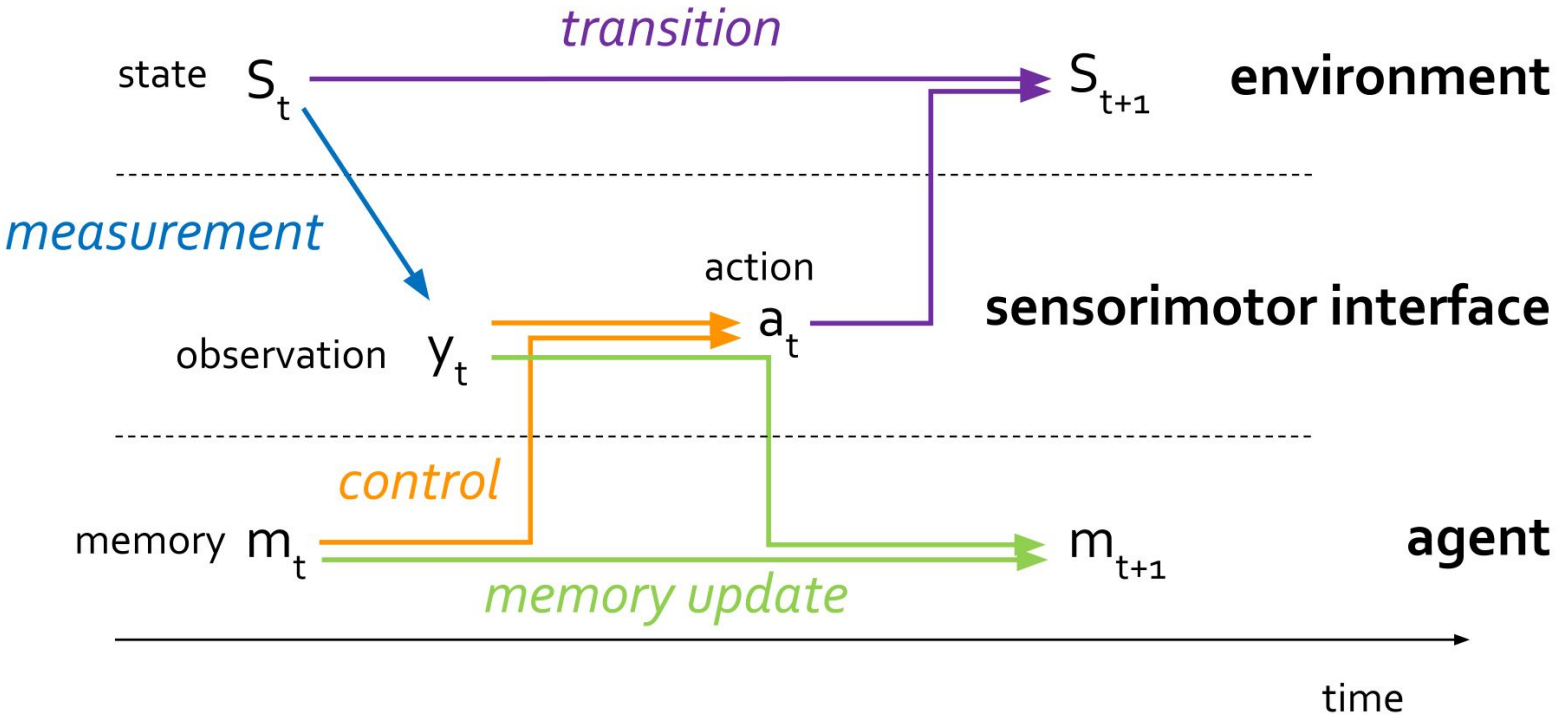
model-free

Strategy: Deep Reinforcement Learning as a way to keep track of histories.
Cons: It is mostly difficult to train

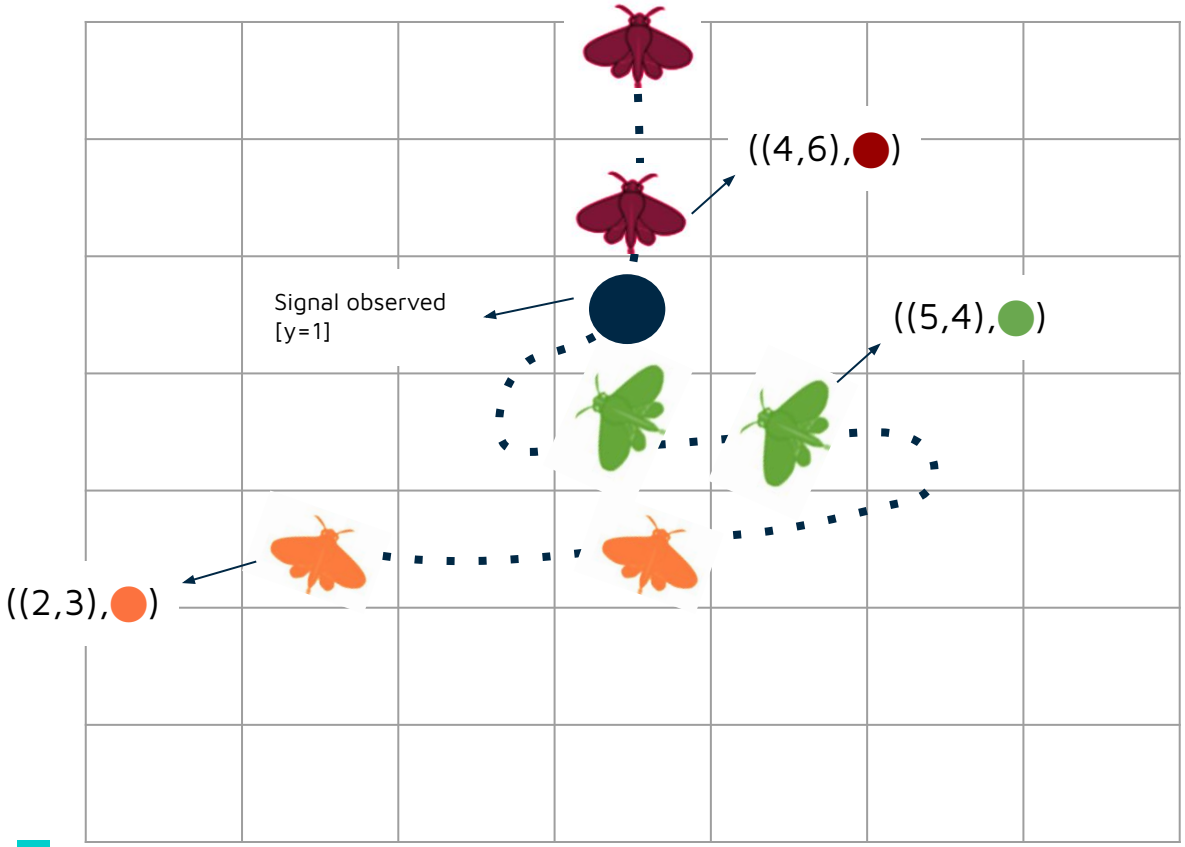


Emergent behavior and neural dynamics in artificial agents tracking turbulent plumes by Singh S, et.al. 2021. https://arxiv.org/abs/2109.12434v2

POMDP framework



Set-up



In a sample 7x7 grid, we can represent the space-memory state as ((x,y),m')

the color of the agent represents which memory state it is in

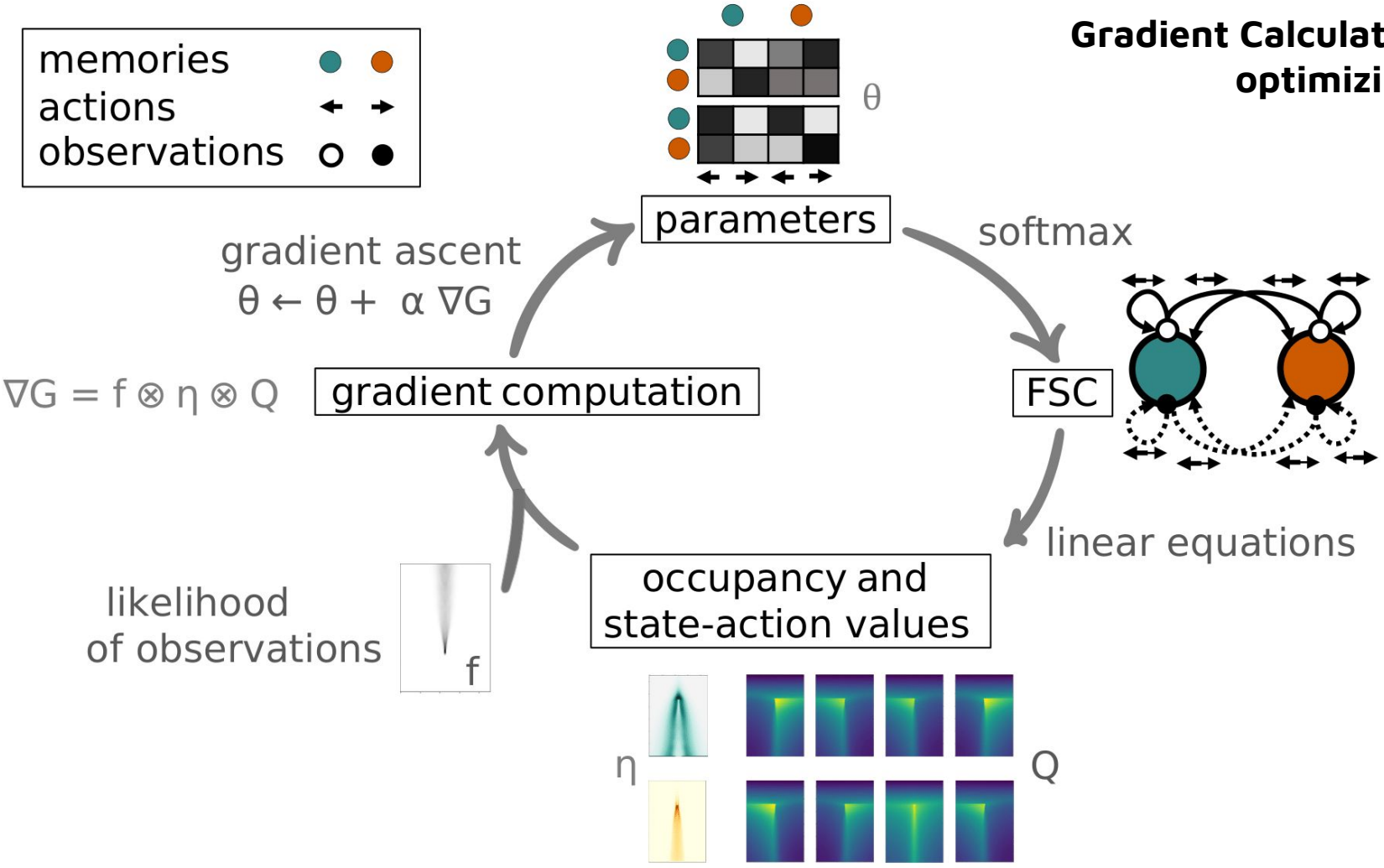
Goal: Find a policy P((a,m')|m,y) to minimize the search time

a ∈ {↑,→,↓,←} actions

m∈ {●,●,●} memory state

m'∈ {●,●,●} memory update

Gradient Calculation for optimizing FSC



Source of our data for the plume [model of detection]

Mahmut Demir, Nirag Kadakia, Hope D Anderson, Damon A Clark, Thierry Emonet (2020) **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

Smoke in Air

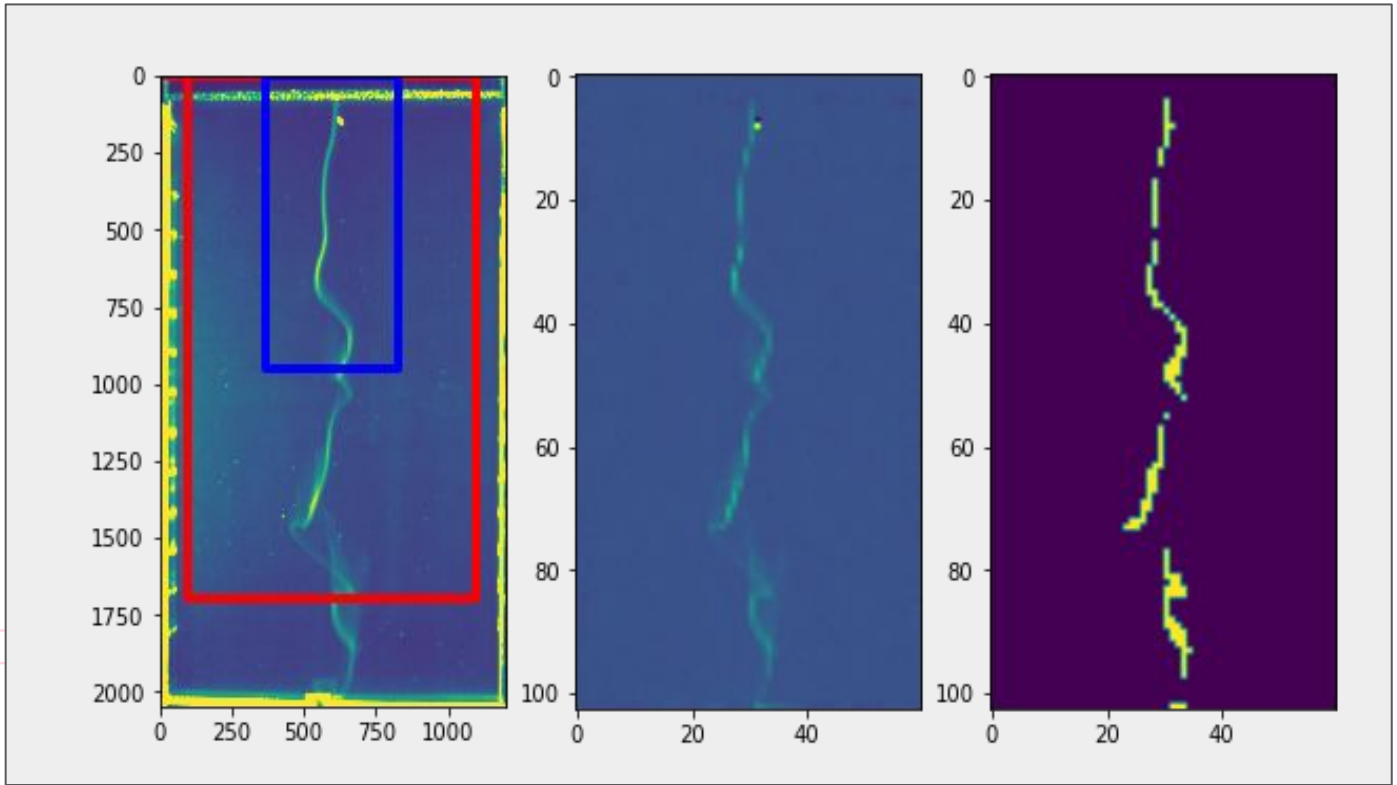
Another source for model of detection

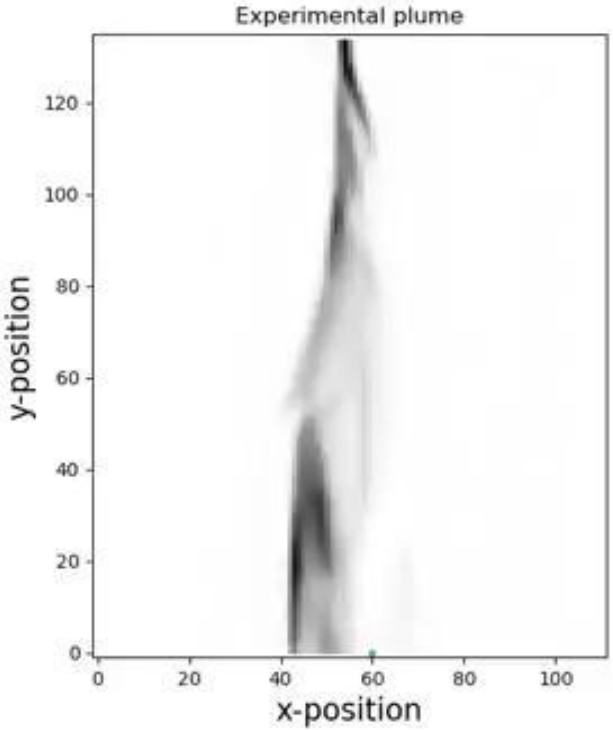
“A nonsequential turbulent mixing process” by Duplat J., Innocenti C., Villermaux E., 2010. Phys. Fluids 22:035104, https://doi.org/10.1063/1.3319821

Dye in Water

Setting up the environment: from data

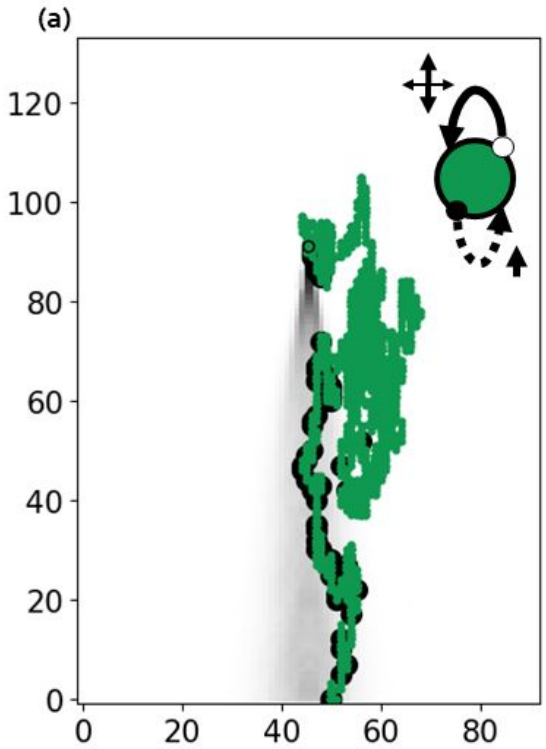
Plume pre-processing: I) de-noise, ii) reducing, iii) threshold



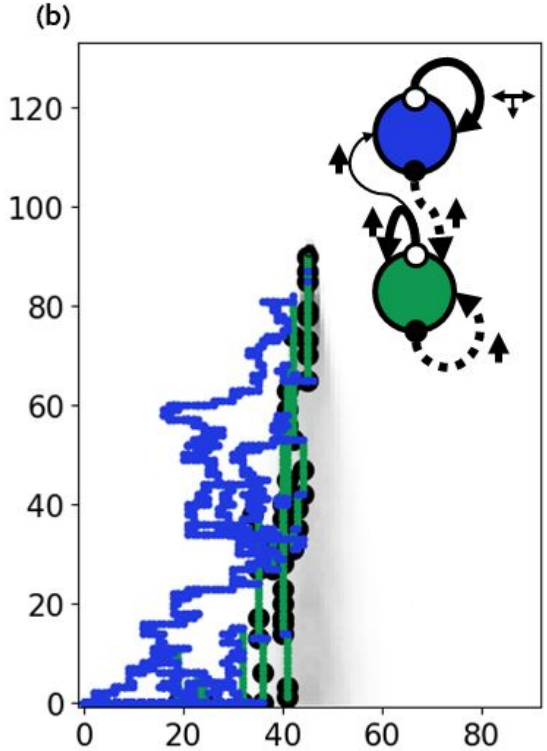


Optimal behavior involves strategic memory state transitions as agents navigate to find the source

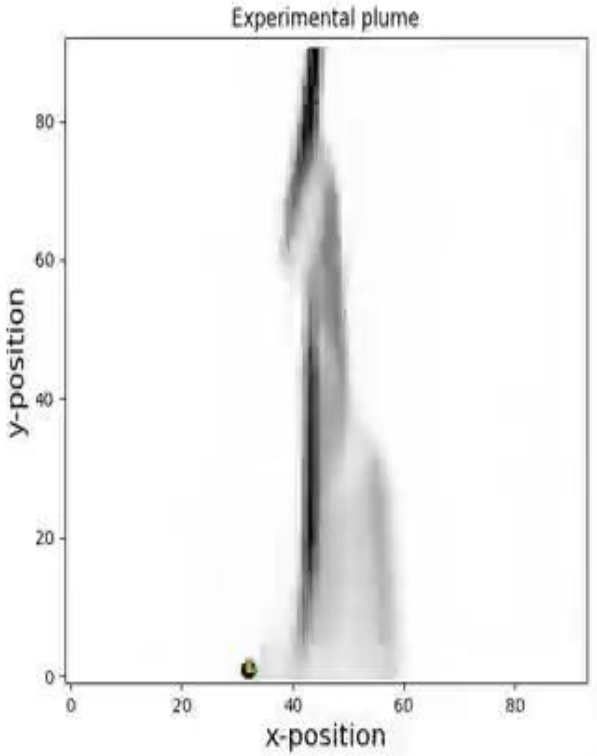
Best Reactive strategy [1 Memory State]

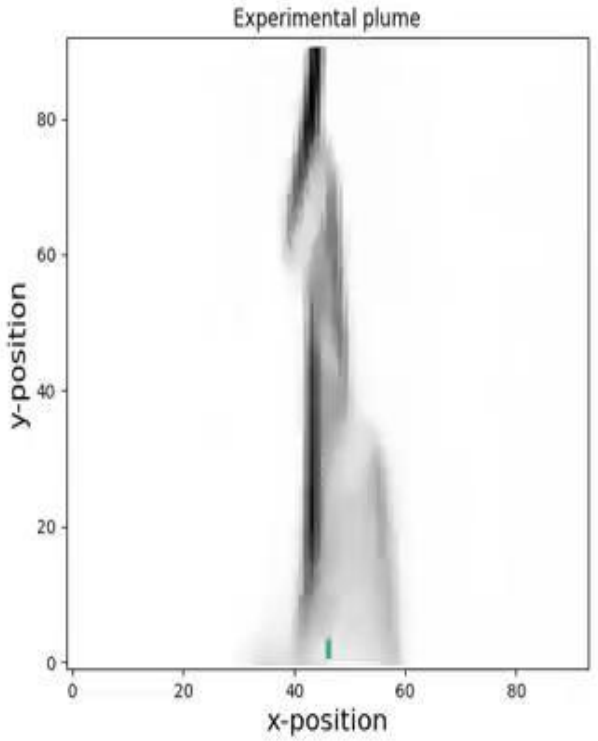


Surge and backward random walk [2 Memory States]

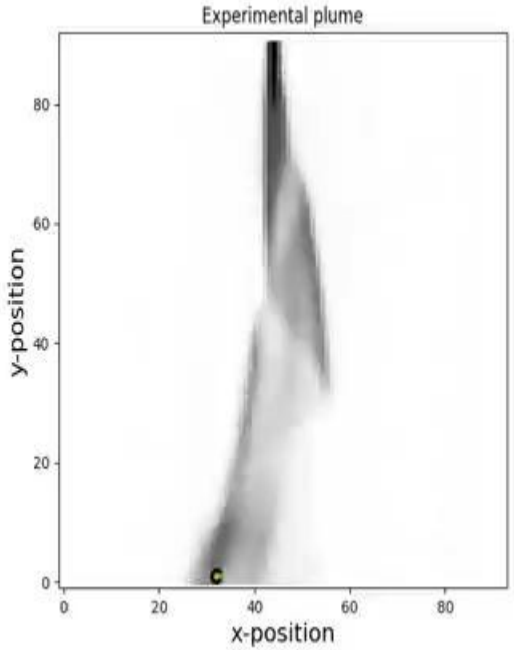


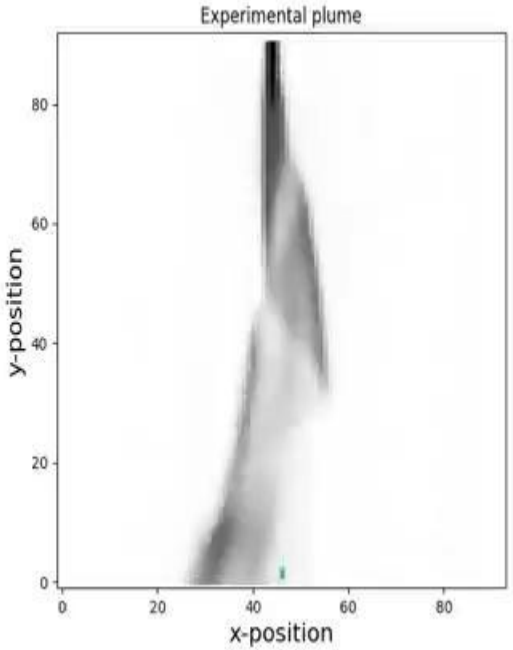
3 memory states



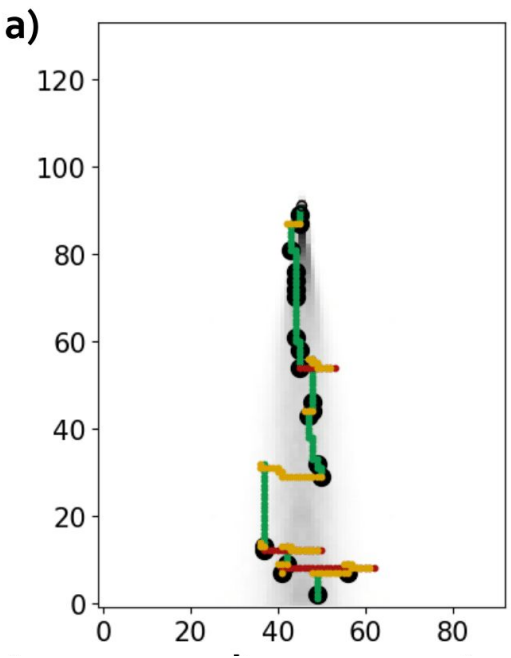


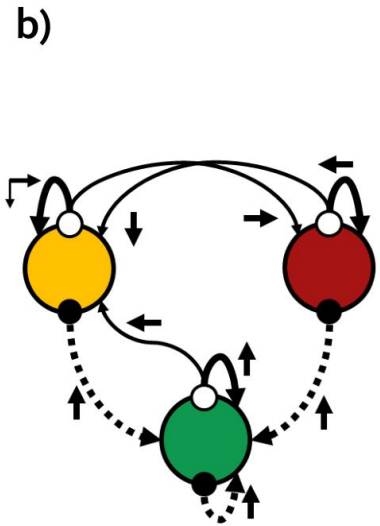
3 Memory States

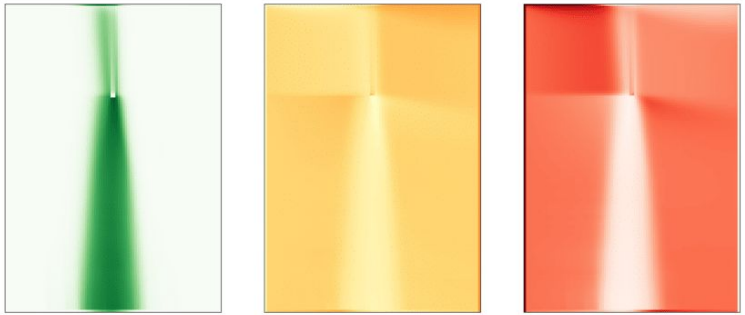




Surge upwind and casting (left/right)
[3 Memory states]



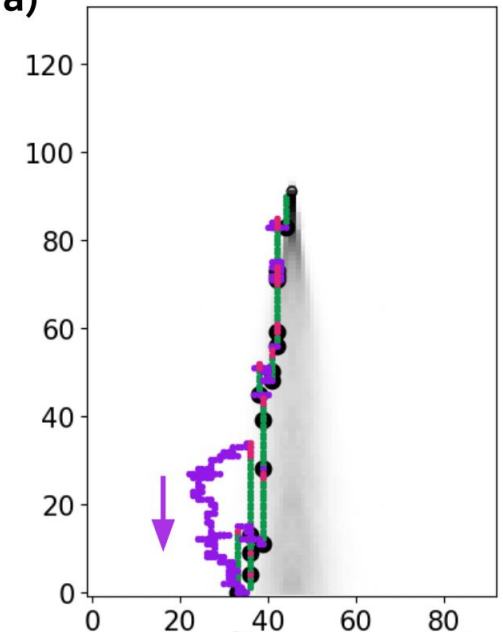




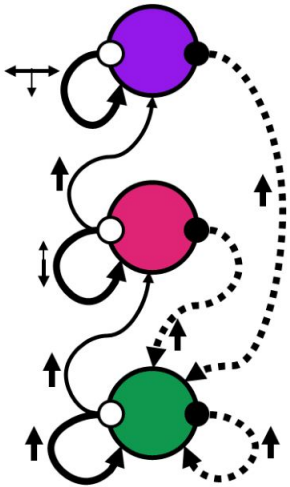
Heatmaps for the spatial memory state occupation **Prob(m|x)**

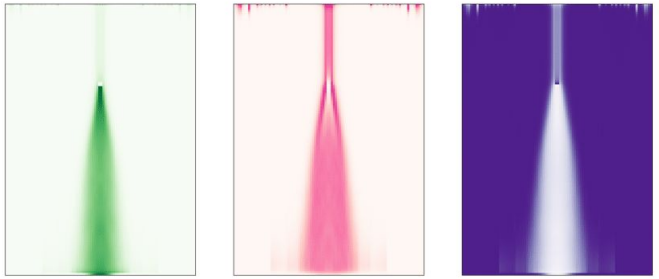
Surge upwind and biased random walk
[3 Memory states]

a)

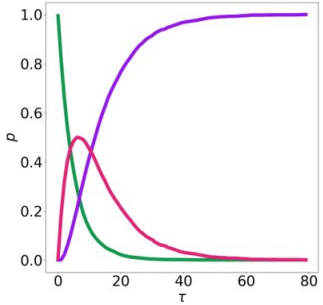


b)



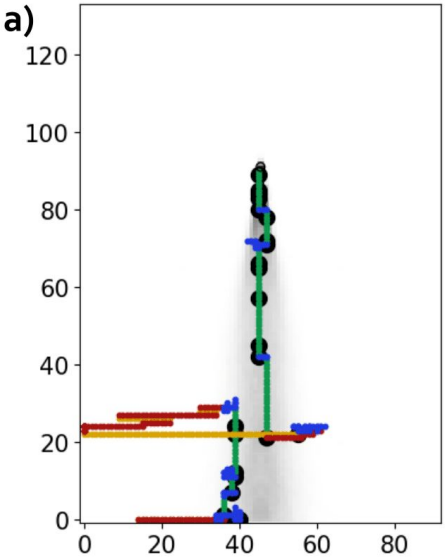


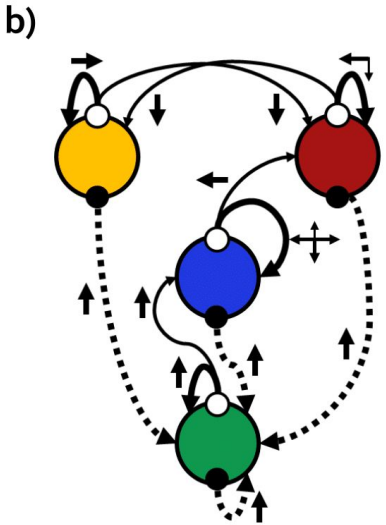
Heatmaps for the spatial memory state occupation **Prob(m|x)**

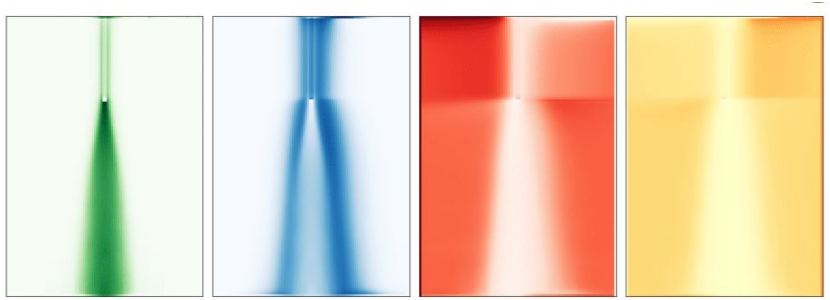


Occupation of memories given time since last detection τ
Prob(m|τ)

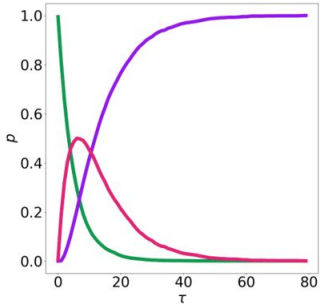
Surge upwind, biased random walk, and casting (left/right)
[3 Memory states]





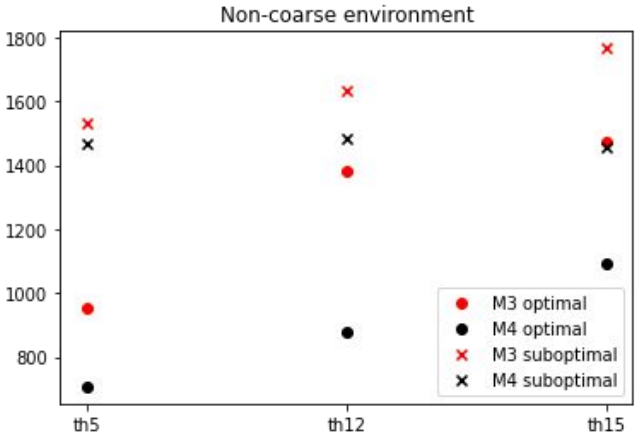


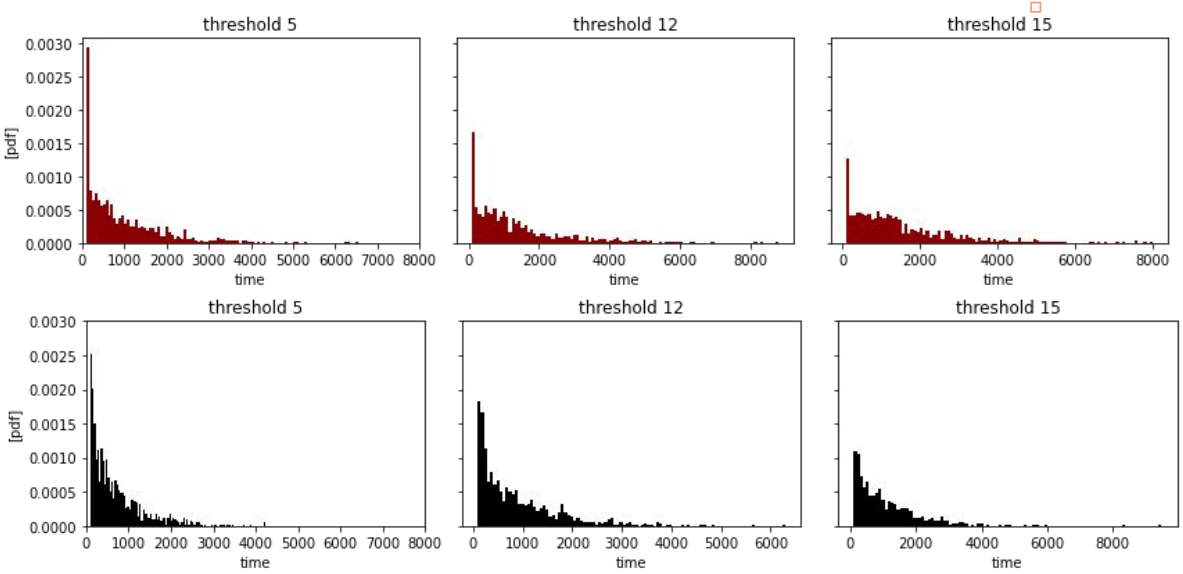
Heatmaps for the spatial memory state occupation **Prob(m|x)**



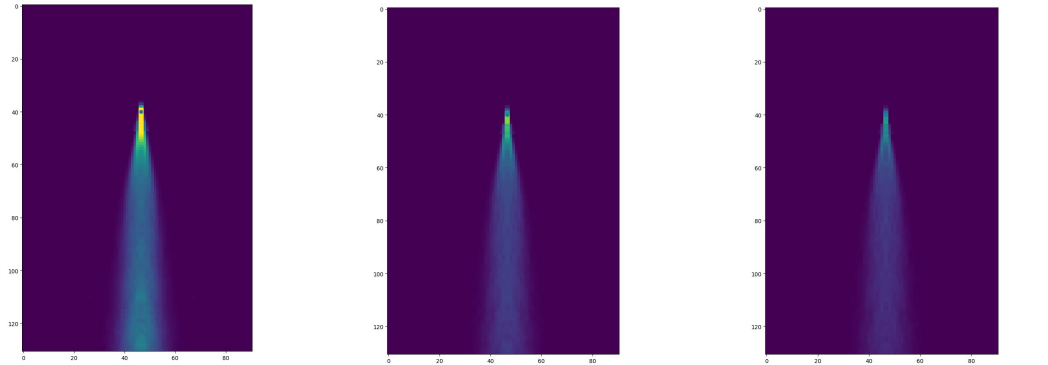
Occupation of memories given time since last detection τ
Prob(m|τ)

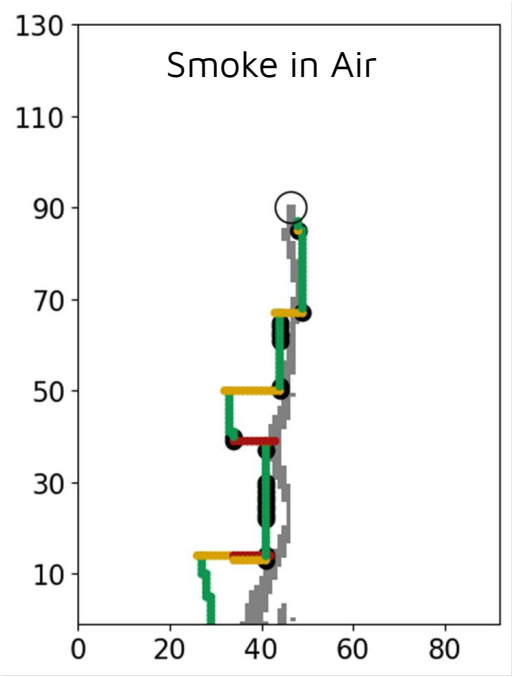
Increasing difficulty by decreasing detection thresholds:

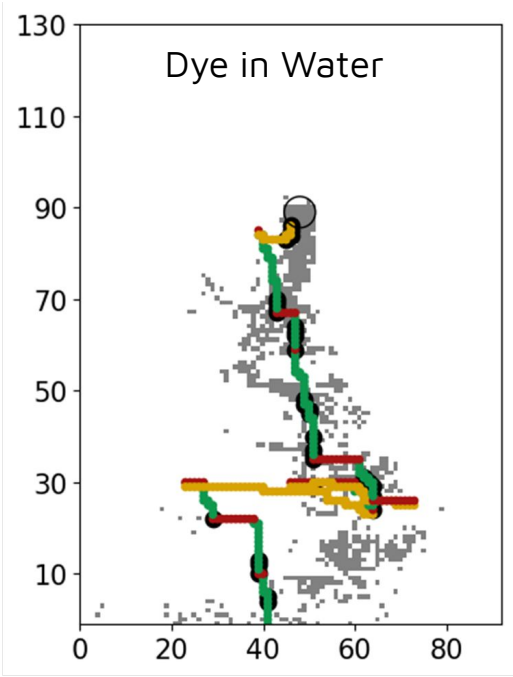


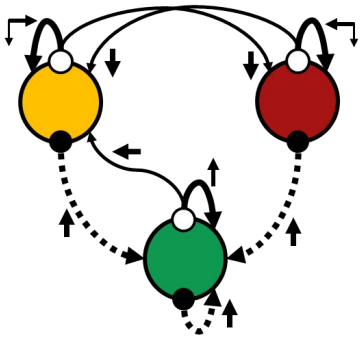


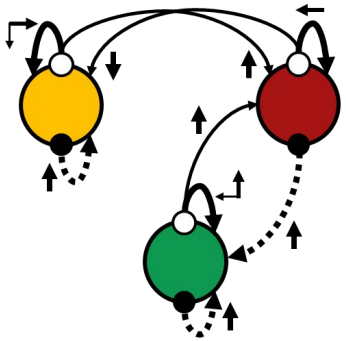
Mean T increase as difficulty increases











Different temporal and spatial distribution of the odor signal leads to similar cast and surge strategy, but with difference in time spent in each memory state and resetting rule.

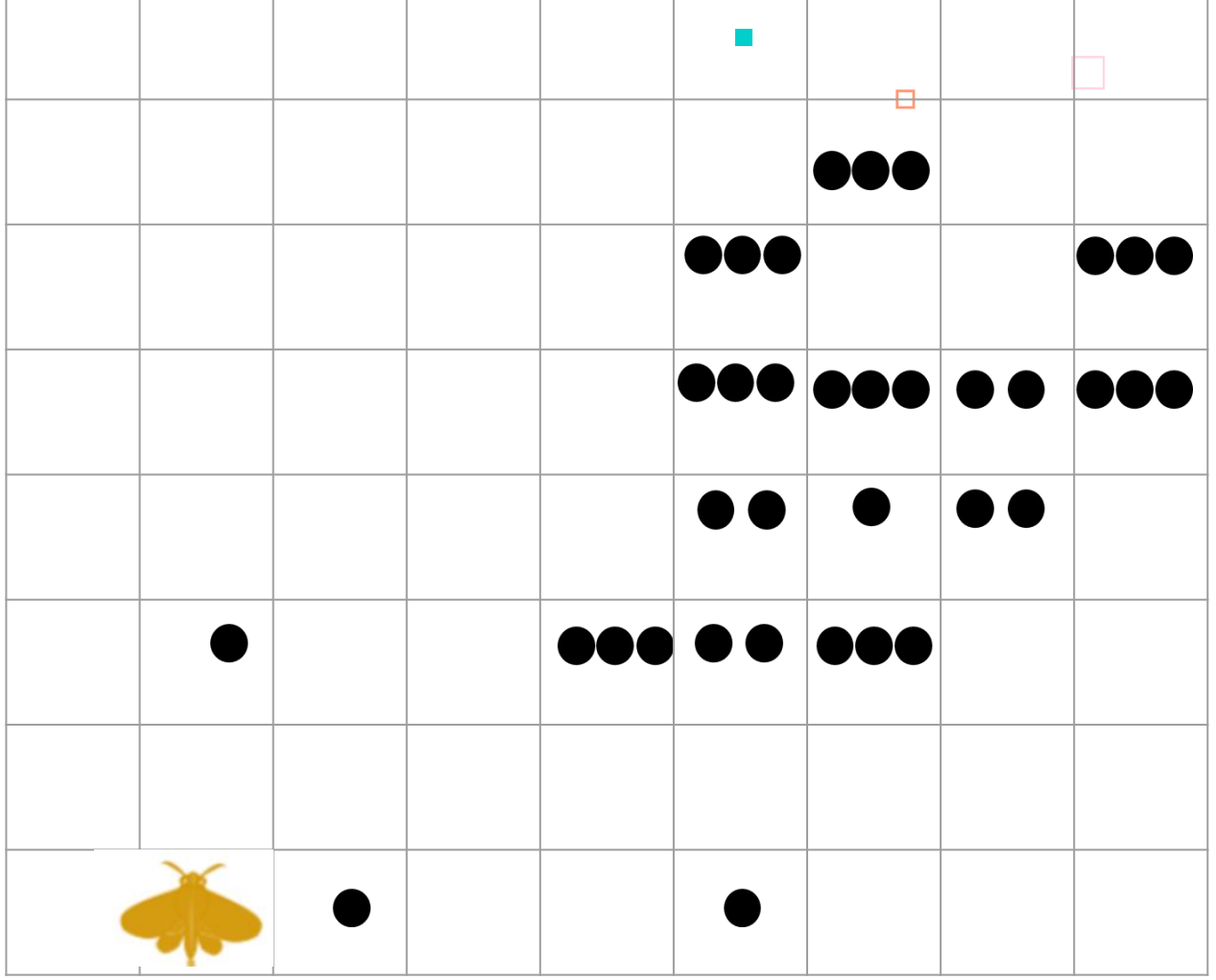
	Algorithm	Success rate*	Average time**
Weak	1M-FSC	72.3%	2190 ± 70
	2M-FSC	99.5%	1075 ± 54
	3M-FSC	100%	495 ± 15
	4M-FSC	100%	440 ± 11
	Cast-and-surge	100%	350 ± 2
	Infotaxis	100%	209 ± 3
Strong	1M-FSC	82.2%	1982 ± 73
	2M-FSC	99.6%	1052 ± 54
	3M-FSC	100%	390 ± 11
	4M-FSC	100%	336 ± 8
	Cast-and-surge	100%	273 ± 1
	Infotaxis	99.9%	197 ± 1

Success rates and average search time in a dynamic environment

Conclusions and Perspectives

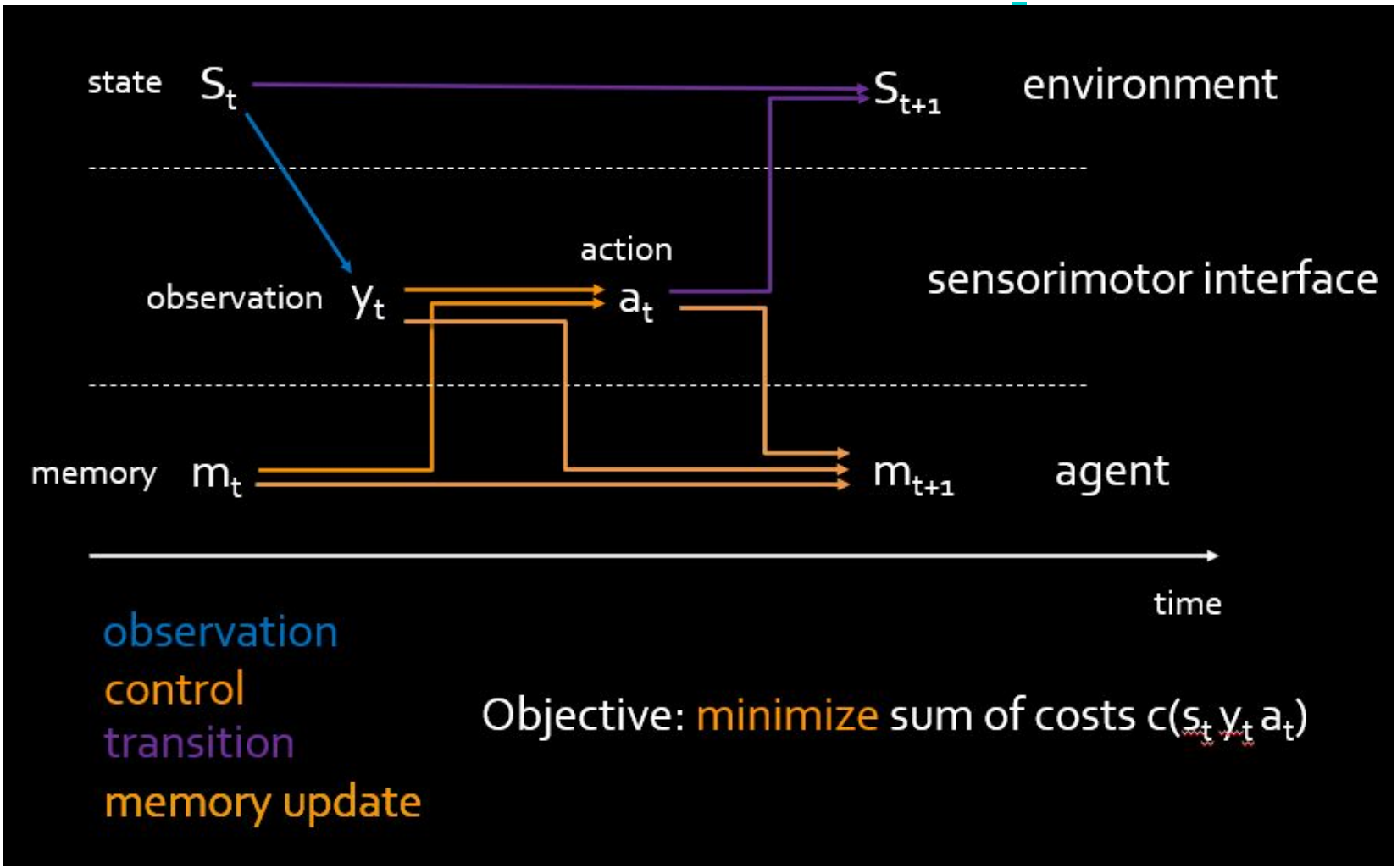
- Simple finite state controllers can be optimized to discover effective strategies for olfactory search
- The memory states appear to encode both temporal (clock) and spatial (map) information.
- FSCs can also be optimized in a model-free approach (stochastic gradient methods)
- The topology of the controller may have a more universal structure related to the olfactory task: perhaps the transition rates between the memory states adapts to a specific context by some biological process occurring on developmental or even faster time scales

Thank you for your attention!



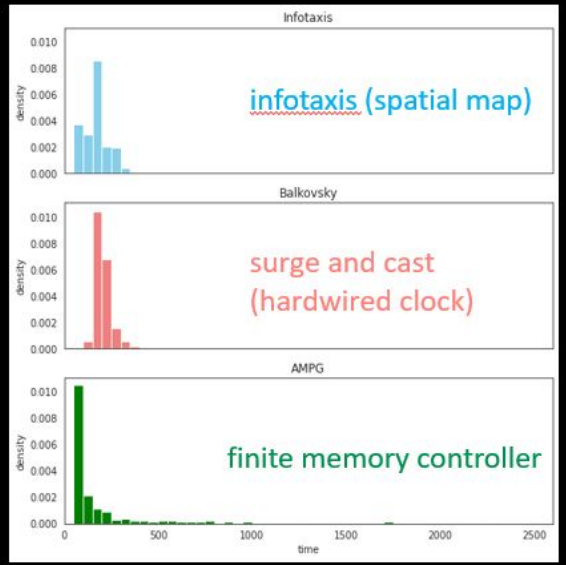
Simulation set-up

Grid = 131 x 92
Reward of finding = 0
Cost per move = 0.00025
[In this way, maximizing reward means minimizing the cost or equivalently, minimizing the time it takes to find the source]
Actions = 4 [left, right, up, down] *We tried to run 5 actions (with option to stay)

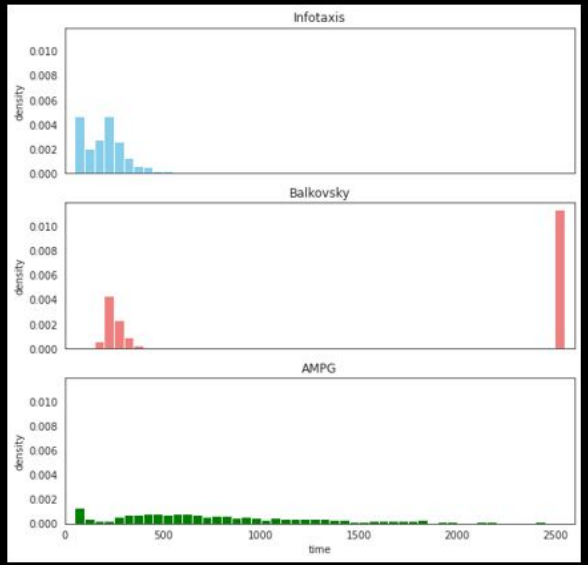

state S_t
S_{t+1}
environment
action
observation y_t
a_t
sensorimotor interface
memory m_t
m_{t+1}
agent
time
observation
control
transition
memory update
Objective: minimize sum of costs $c(s_t, y_t, a_t)$

Performance comparisons

strong wind
large emission rate

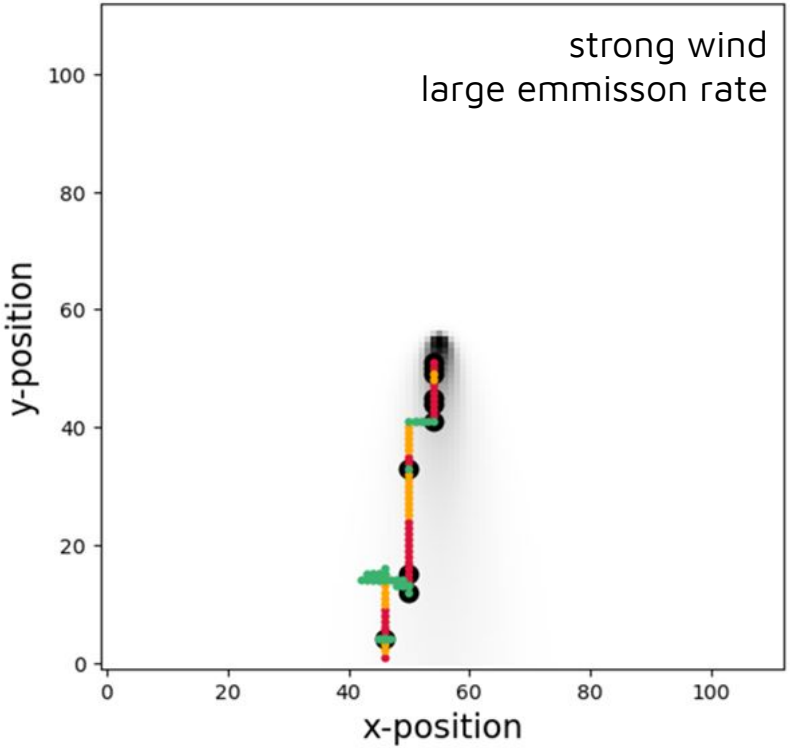


weak wind
low emission rate



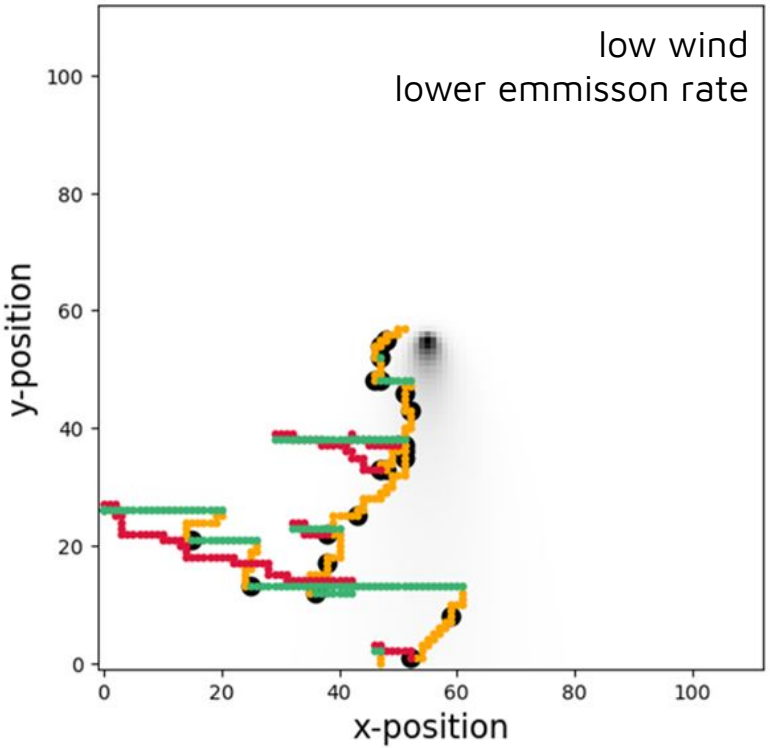
Memory is used as

an approximate clock



surge and biased random walk

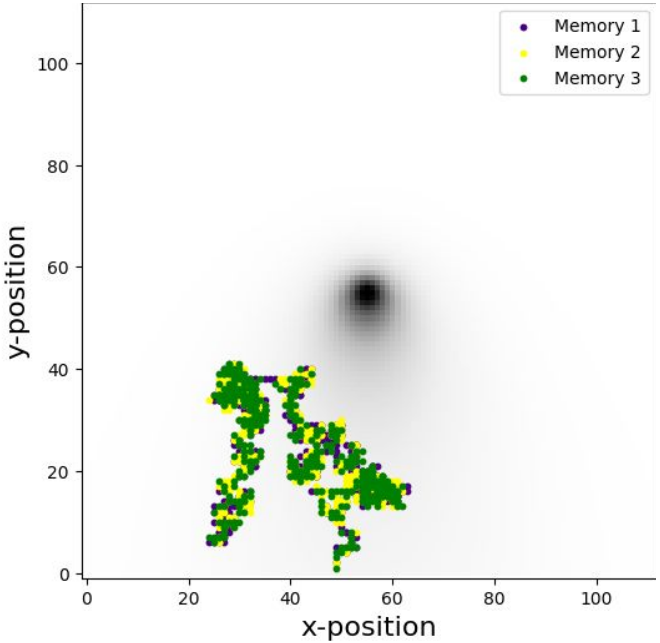
or an approximate map



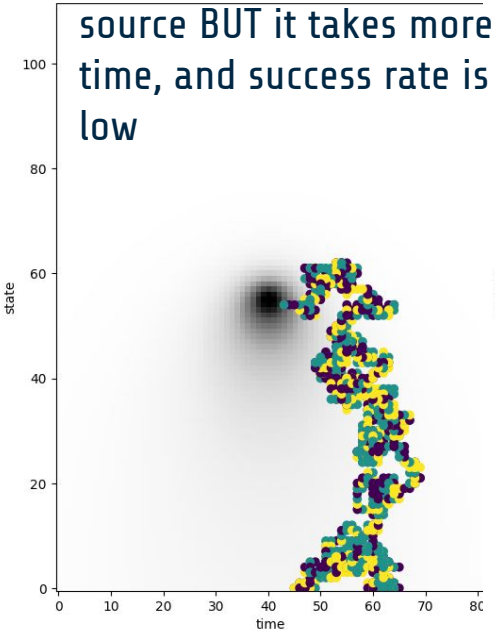
slanted surge and cast

During the Optimization Process

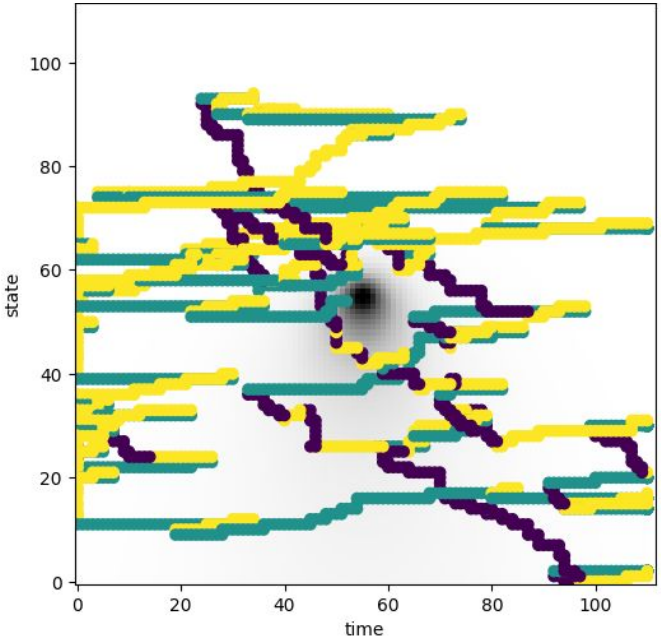
agents start with a random walk



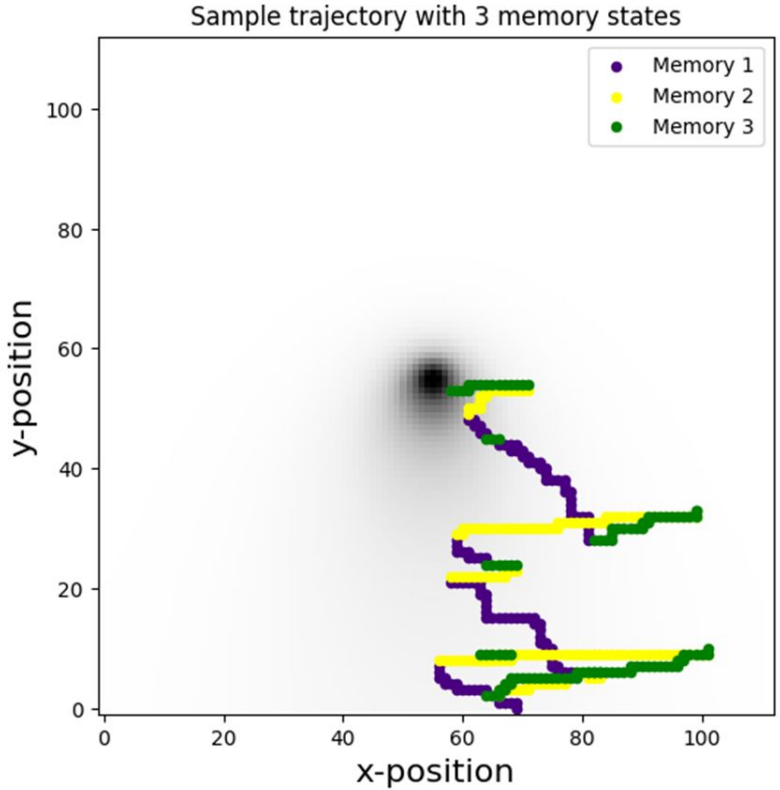
agents learn a random walk which can find the source BUT it takes more time, and success rate is low



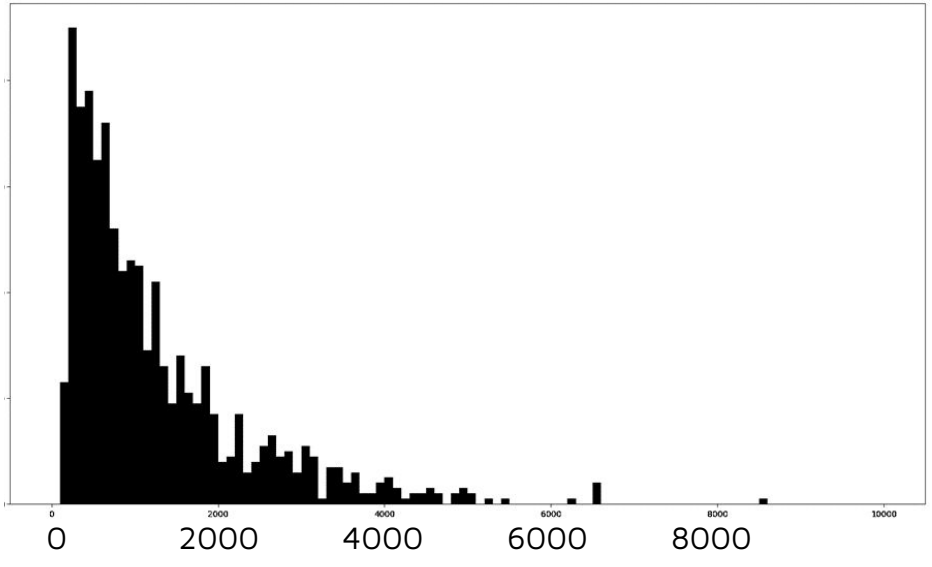
agents develop a strategy to use the memory states until the average reward is optimized



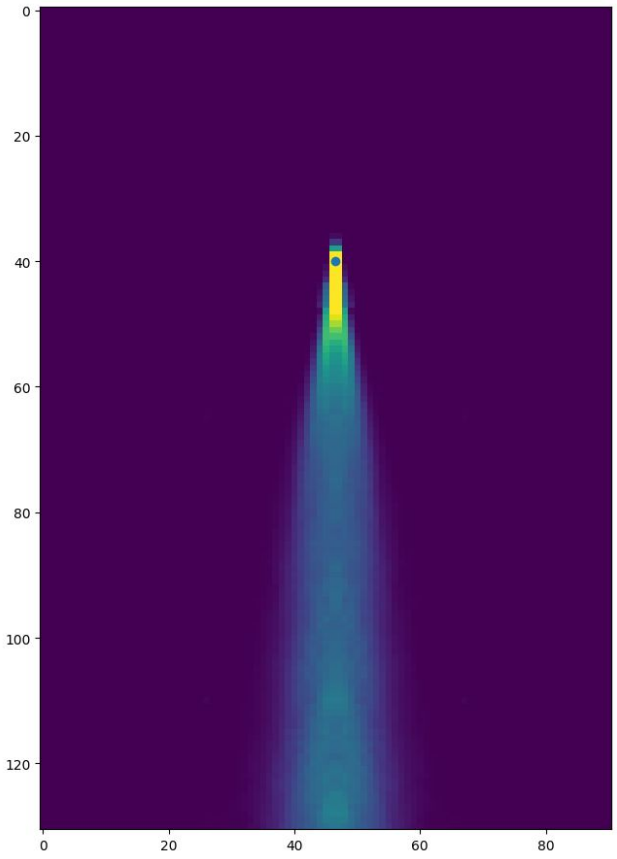
End of training

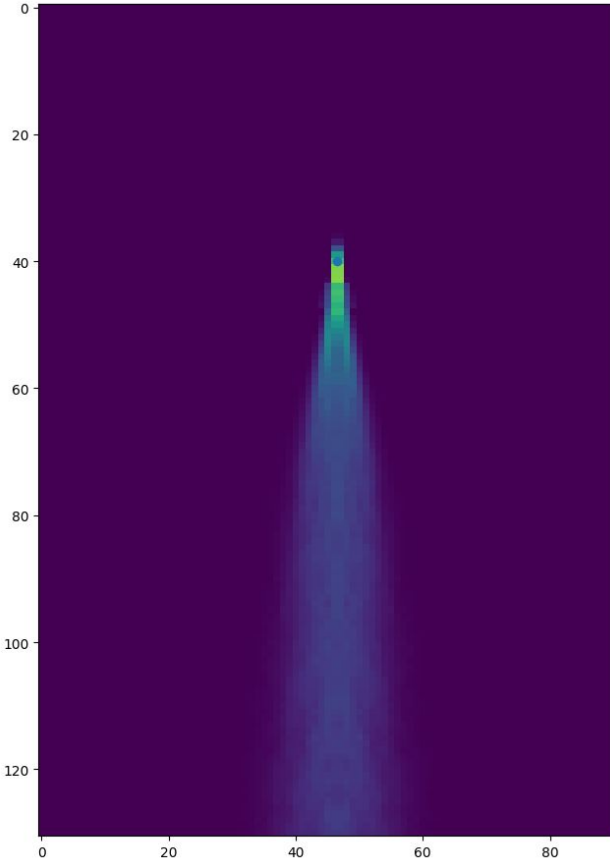


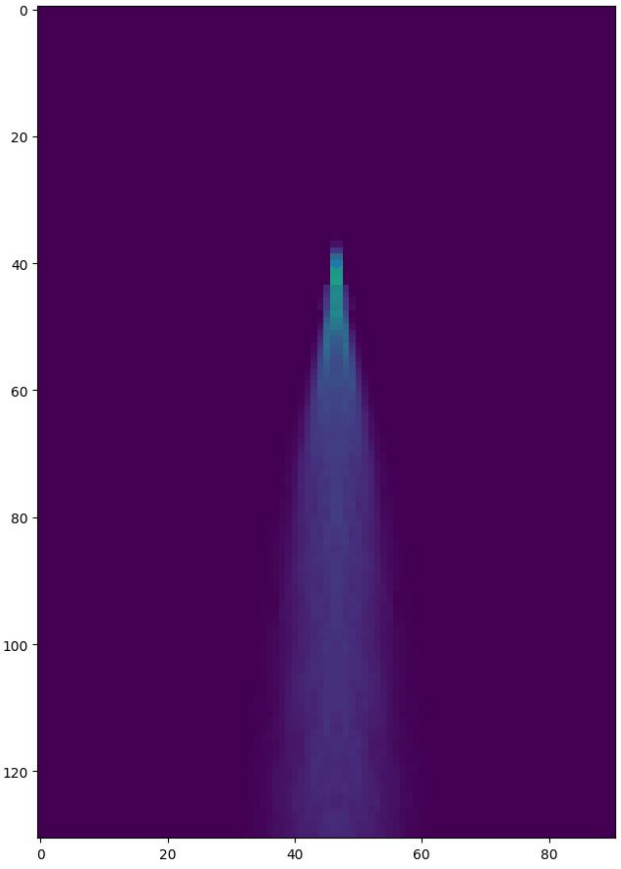
Distribution of time at which the source is found



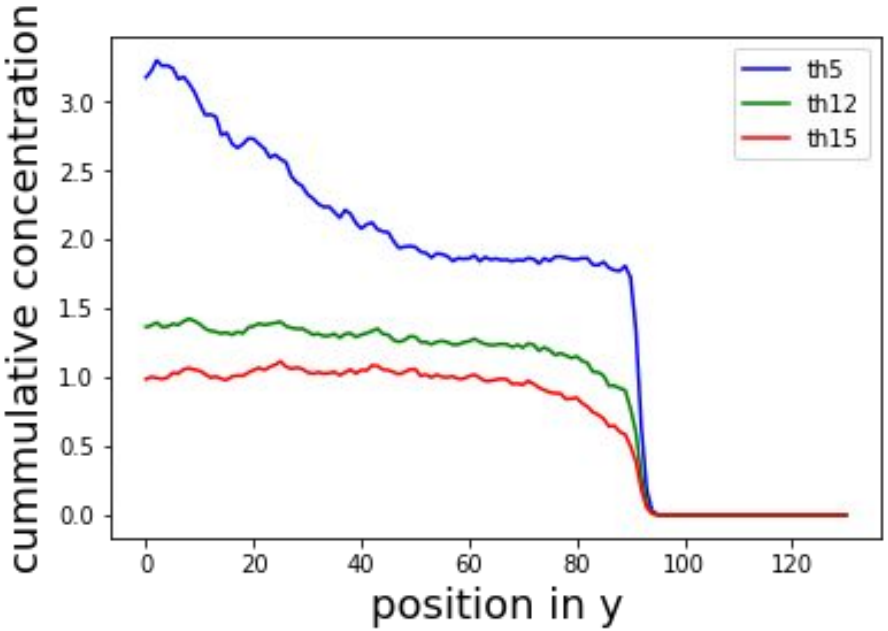
Threshold 5, 12, 15



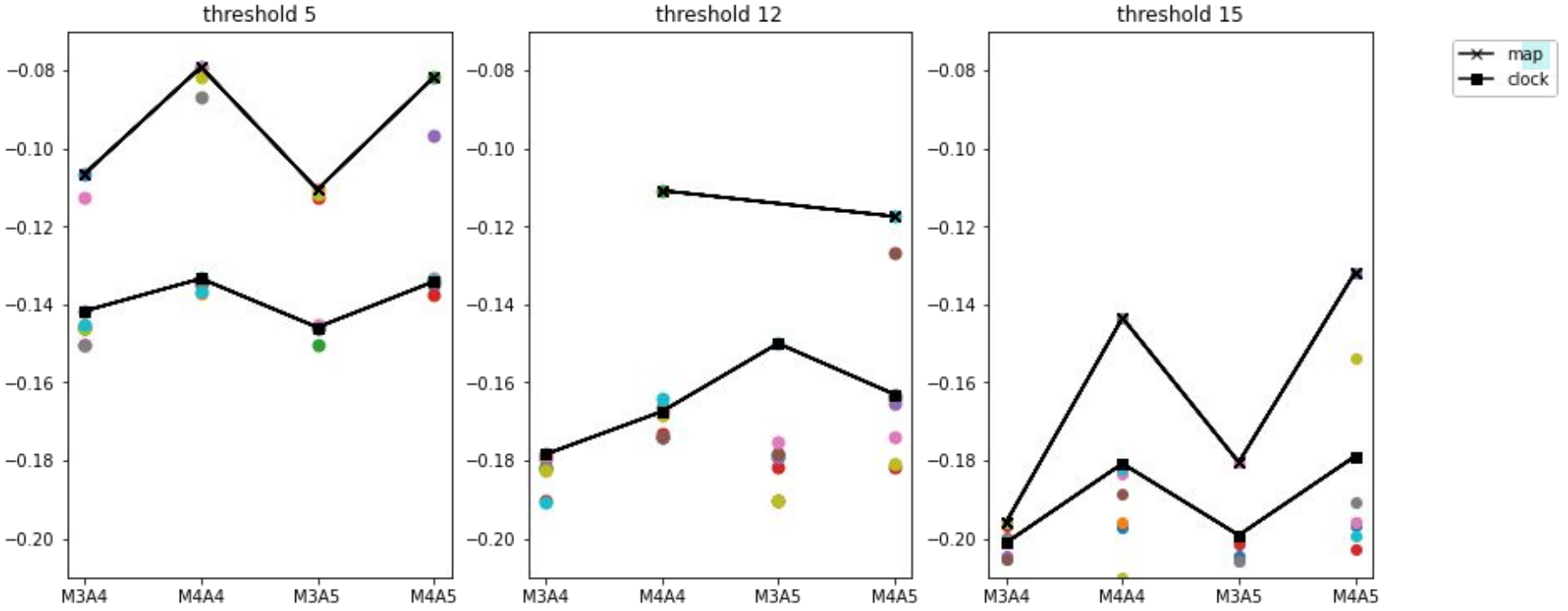




Cumulative signal changes when you change the threshold

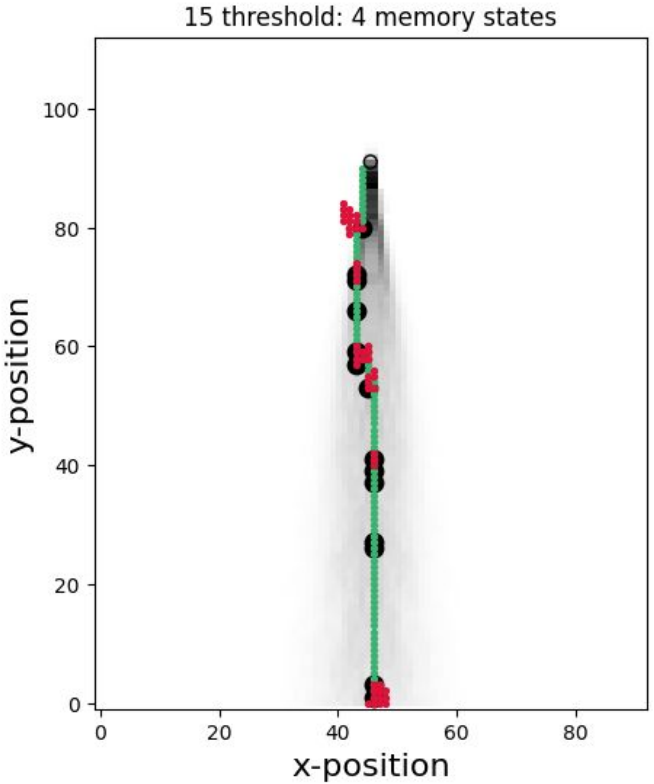


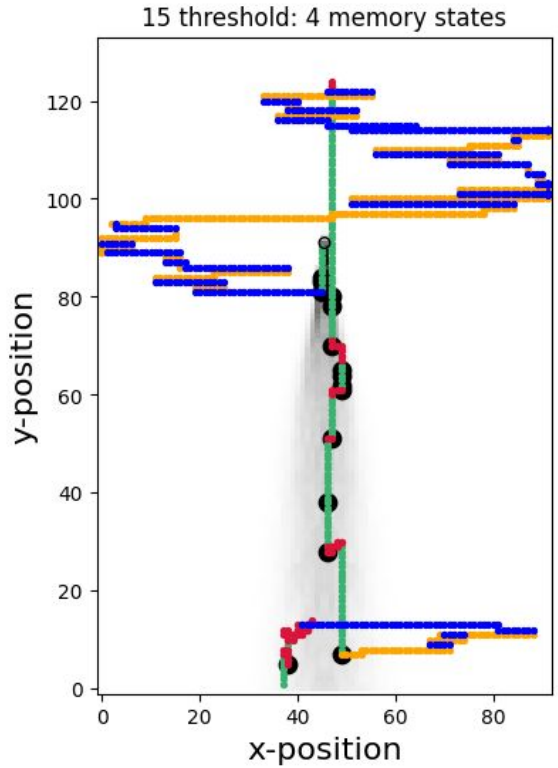
Different thresholds results



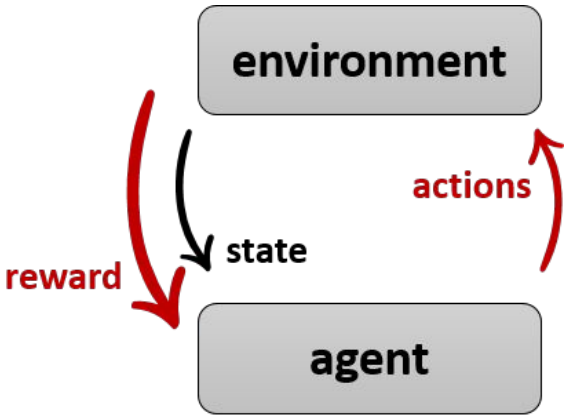
The different colored dots are different replicas. We have done 12 replicas so far, in each case.

The two types of searching





Agents learn by experience



Reinforcement Learning

Main Idea: Learning an optimal strategy by interacting with the environment

Strategies to be learned are termed “policies”, the simplest is to define as probability of doing a certain action in a given state (or given an observation, if state is unknown to the agent).

Goal: We find the policy (or strategy) that maximizes the agent’s reward