

Learning with a Mole: Transferable latent spatial representations for navigation without reconstruction

NAVER LABS
Europe

ICLR
Europe

Vienna, 2024



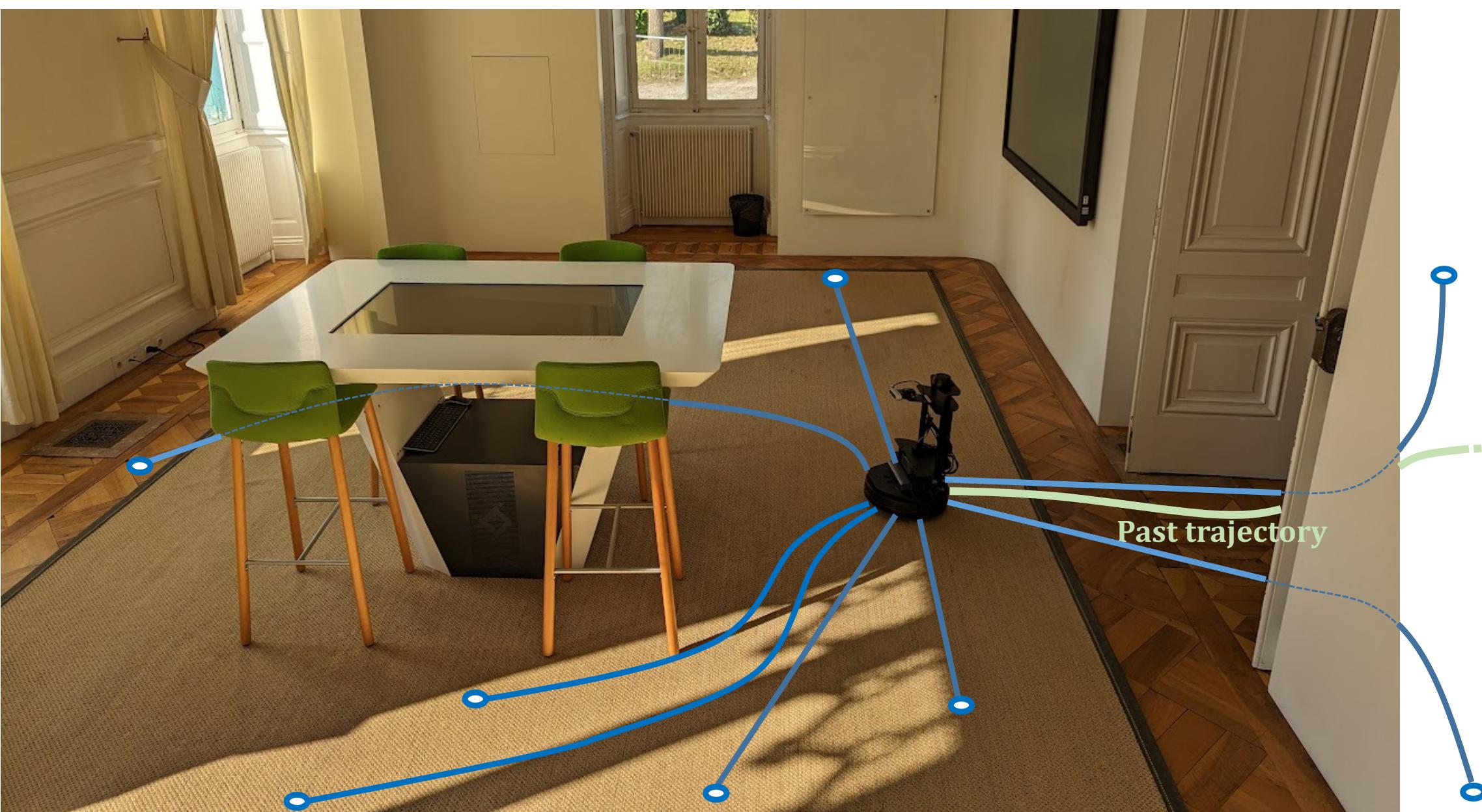
Guillaume Bono Leonid Antsfeld Assem Sadek Gianluca Monaci Christian Wolf

europe.naverlabs.com

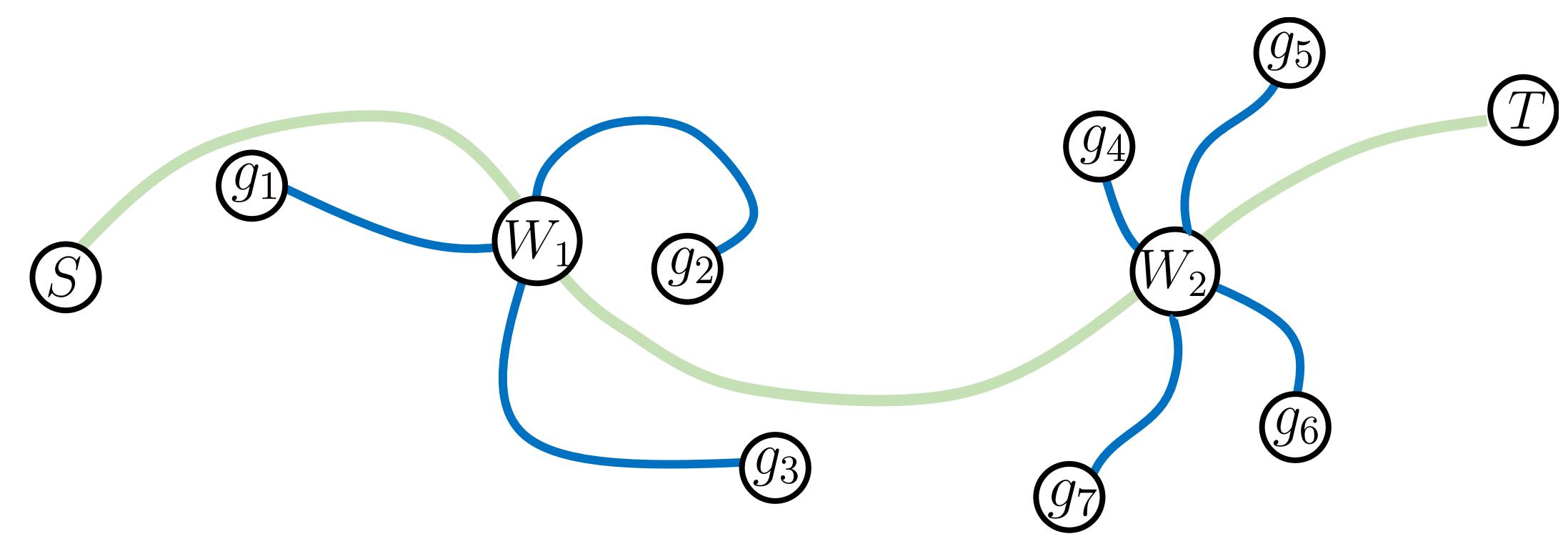
Motivation

- Learn **useful** and **transferable** representations for navigation.
- Provide a richer learning signal than RL or IL.
- Avoid supervising reconstruction (avoid learning irrelevant details, which hurts sim2real transfer)

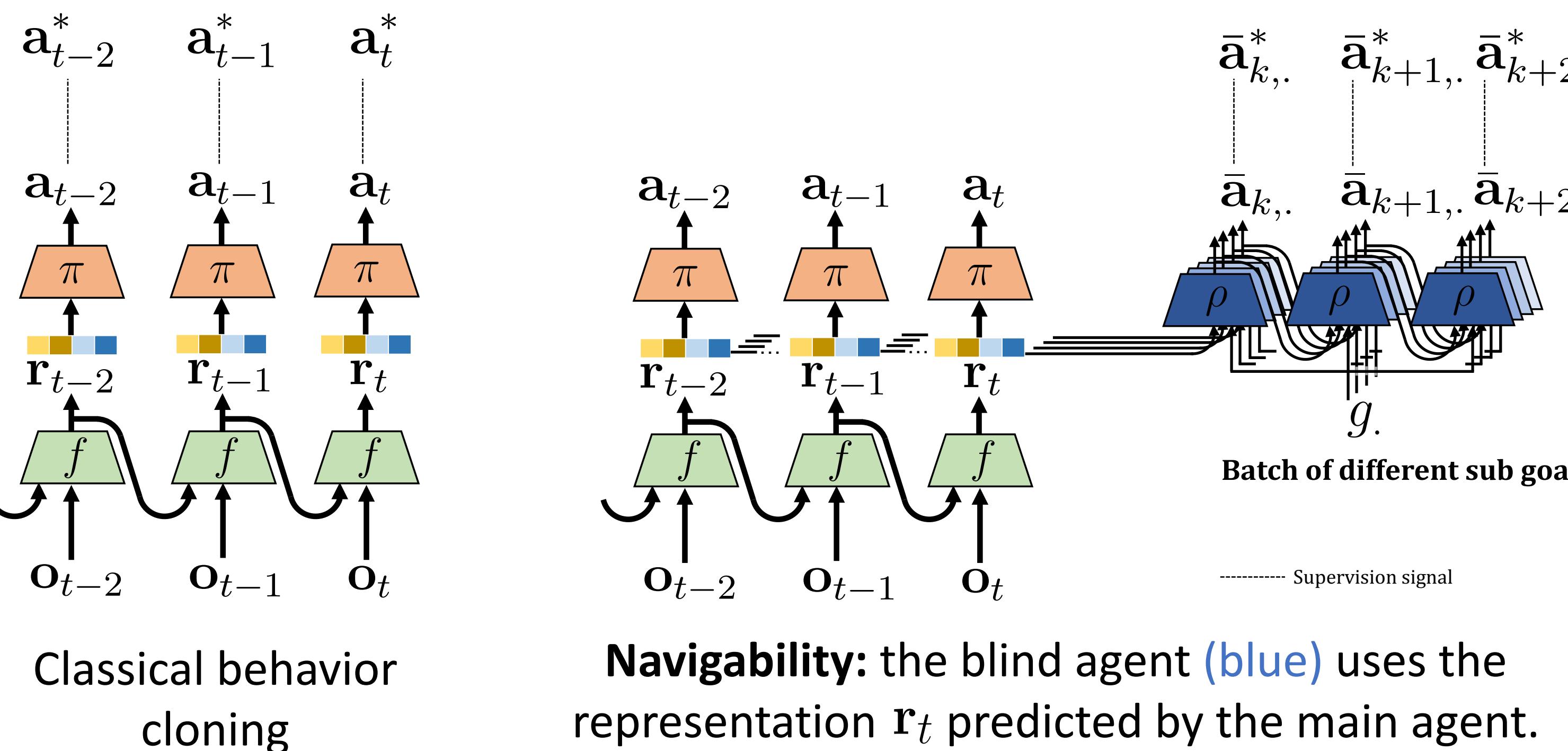
Overview of method



- Representations r_t are collected from visual sensors with a (main) agent (green) using a recurrent network.
- At periodic waypoints W_i , the representation r_t is given to a batch of multiple otherwise blind auxiliary agents, (blue) which navigate to a batch of sub goals.
- The blind aux agents are trained with behavior cloning, the loss is backpropagated over the representation module of the main agent.

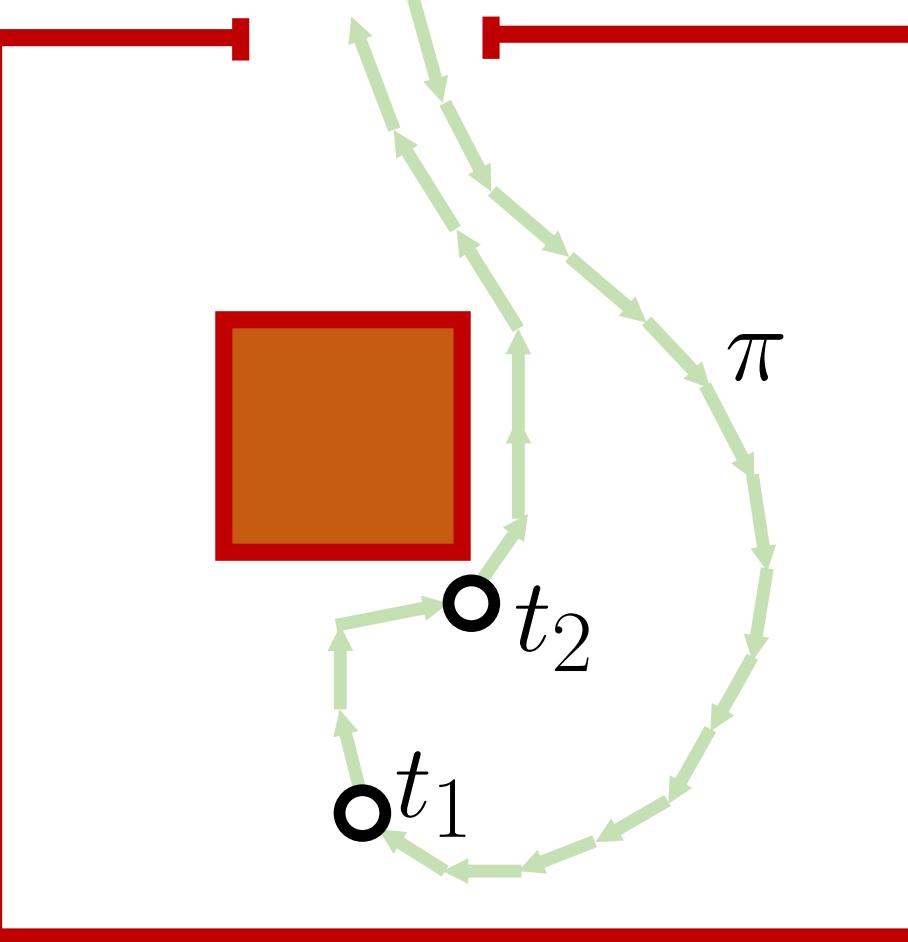


Model: Behavior cloning vs. “Navigability” = “Mole”



On the importance of the blindness property

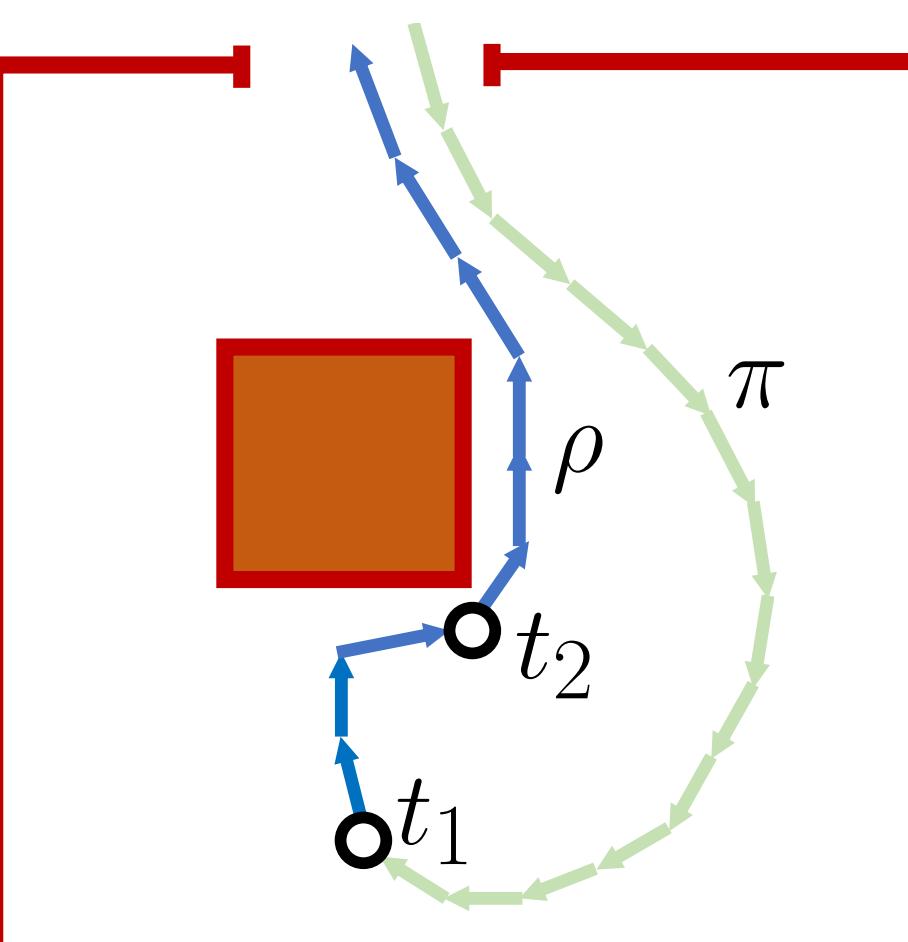
An agent (green) enters, makes a circular motion, observes an obstacle.
Goal: maximize the amount of information on obstacles and navigable space extracted by the agent through its training objective.



Classical training with behavior cloning

Example: the training signal at $t=t_2$ corrects the decision “*do not crash into the obstacle*”. Two possible updates of the agent’s reasoning mode:
(r1) avoid obstacles visible in cur observation (no memory required).
(r2) avoid obstacles detected in internal latent map (integrated over history of observations).

Compression mechanisms in training recurrent networks will lead to favoring reasoning (r1), reactive behavior (undesired).



Training with our navigability loss

Example: the main agent (green) generates a representation at $t=t_1$. The blind agent (blue) uses it, follows sub-trajectory, gets corrected at $t=t_2$ “*do not crash into the obstacle*”.

Reactive behavior (r1) cannot be learned (the agent is blind). Sequential integration of observations into r_{t_1} is forced.

Results: comparison of representations (sim)

Step 1 (50M env steps): representation training (“Navigability”)

Step 2 (50M env steps): reinitialize main policy from scratch, RL/PPO training of main agent only

Method	Nr. Env steps	Nr. visual obs seen	Do \mathbb{S}	Nr. Envs $\pi:\text{PPO}$	Nr. BC Envs $\pi:\text{I} \pi:\mathbb{S} \rho:\mathbb{S}$	Eval Sim Success SPL	Eval NoisySim Success SPL
(a) Pure PPO	50M	50M	—	12	— — —	89.6 71.7	74.6 55.8
(b) Pure BC I	50M	50M	—	12	— — —	92.0 79.6	76.0 61.7
(c) BC I \mathbb{S} (data augm.)	50M	50M	✓	—	12 12 —	94.2 80.1	89.6 74.0
(d) Navig.	50M	2.5M	✓	—	— 12	92.9 77.3	86.8 68.8
(e) Navig. + BC I	50M	25+1.25M	✓	—	12 12	95.5 80.3	90.9 73.3
(f) PPO + Navig.	50M	25+1.25M	✓	6	— — 6	91.5 72.6	85.2 63.8
(g) PPO + Navig. + BC \mathbb{S}	50M	50M	✓	6	6 — 6	90.3 73.7	83.9 66.0
(h) AUX reconstr. + BC I	50M	50M	—	—	12 — —	94.9 80.4	76.7 61.2

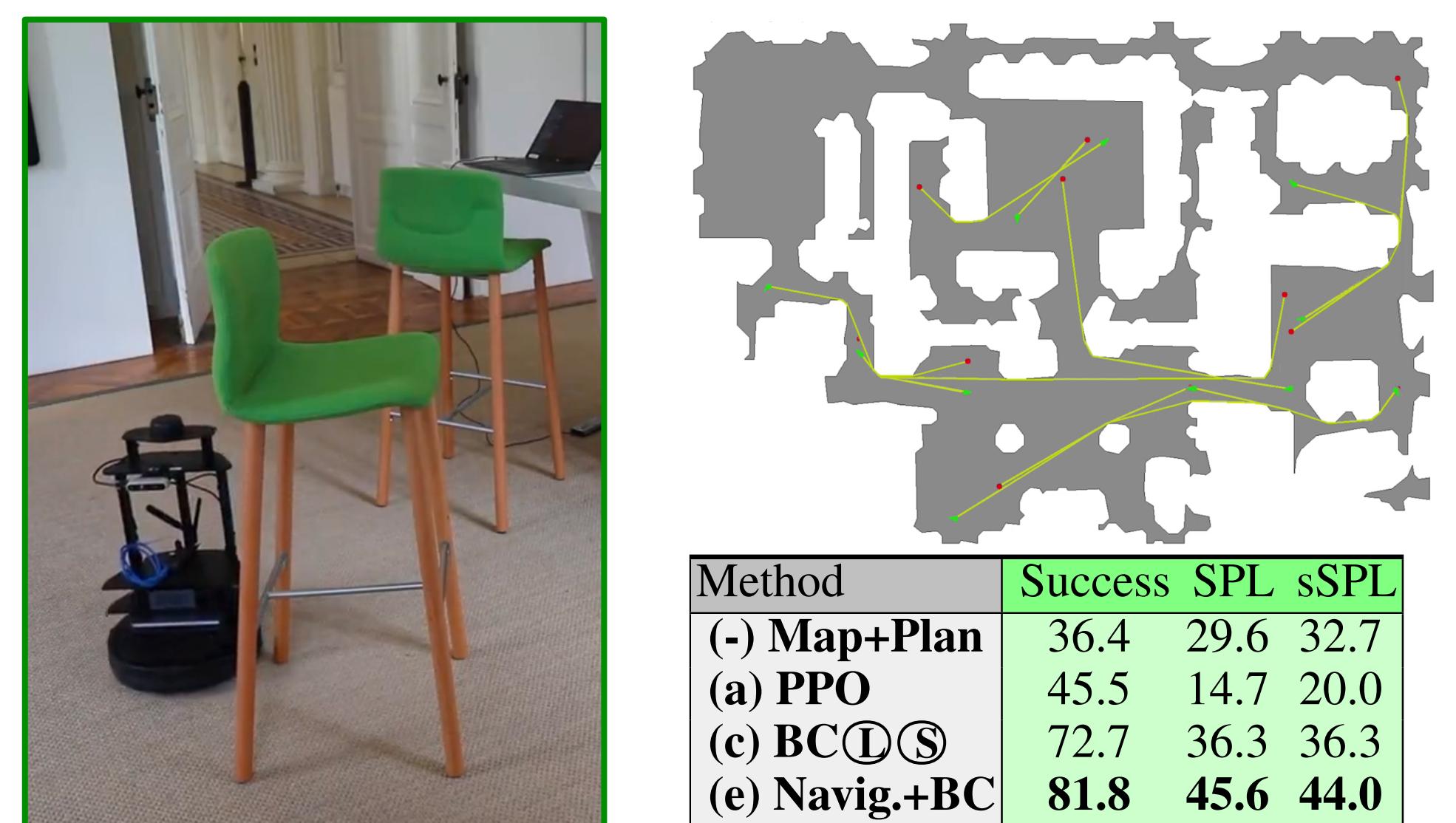
Communicating with the Mole (sim)

The learned representation r_t is passed to the blind agent as:

- “*observation*” (favors allocentric representation)
- through init of its own state (favors egocentric r)

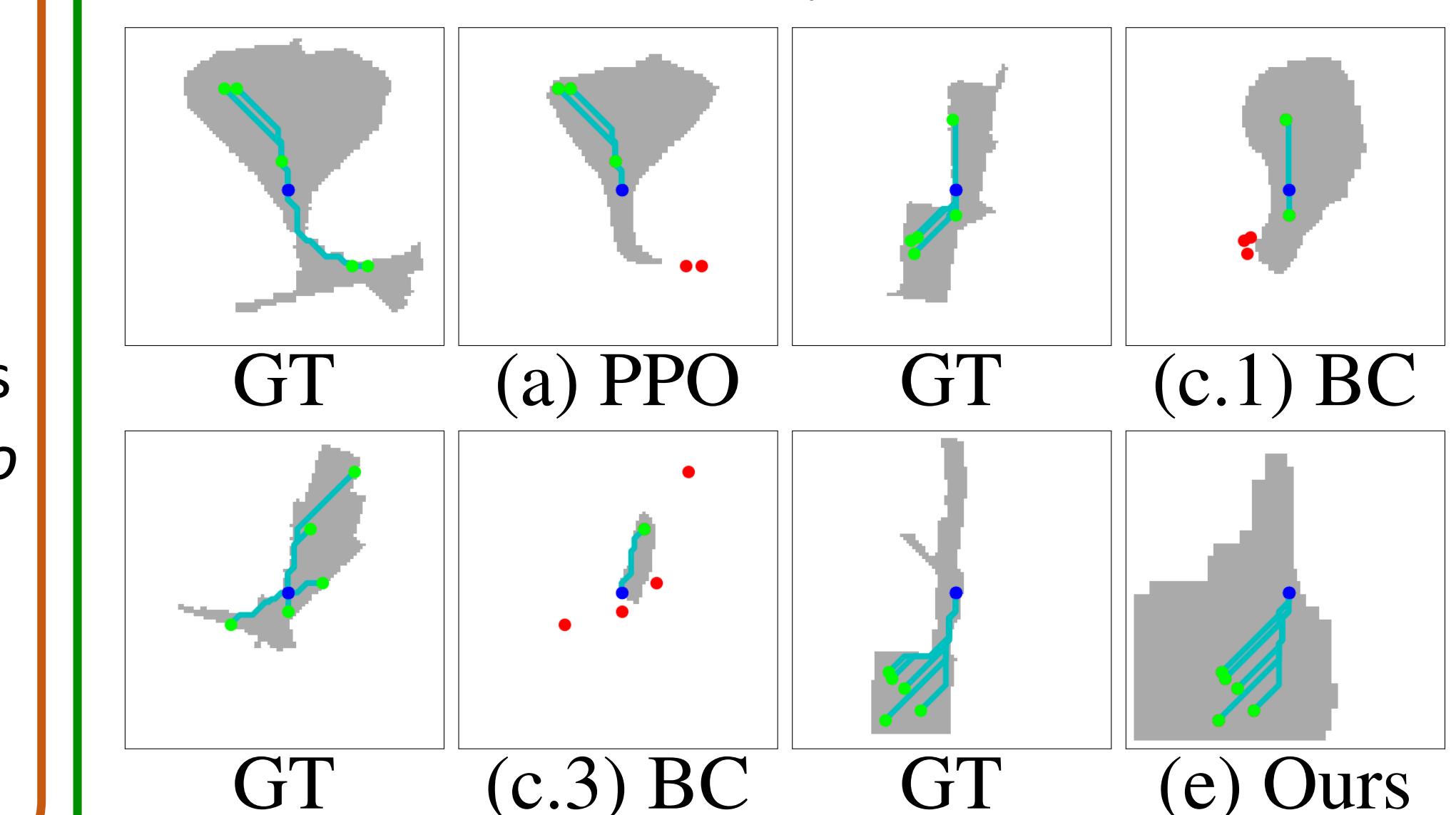
Method	h	Eval Sim Success SPL	Eval NoisySim Success SPL
(e.1) As observation	512	92.8 77.4	90.9 73.3
(e.2) Copy	512	95.5 80.3	80.7 62.5
(e.3) Copy+extend	640	90.4 76.9	83.3 66.6

Results: real robot



Probing experiments

Non-linear probing of occupancy maps describing the seen scene (excluding the “fog-of-war”) from the learned representation r_t .



Method	Reconstr. IoU	Sim		NoisySim	
		2D Nav. Succ. Sym-SPL	2D Nav. Reconstr. IoU	2D Nav. Succ. Sym-SPL	NoisySim
(a) PPO	31.0	37.9	10.1	15.8	18.4 6.1
(c.1) BC I \mathbb{S}	25.5	45.2	11.0	11.1	16.2 4.6
(c.3) BC I \mathbb{S}	11.9	13.2	3.5	1.1	0.6 0.3
(e) Navig.+BC	19.8	66.5	14.7	18.0	64.8 13.2

Viewpoints on figures on the left are not comparable as different agents perform different trajectories. The agent is the blue dot in the center and 5 trajectories out of 10 used for Sym-SPL are plotted. Goal points are green when reachable (always in GT) and red otherwise.