Hot–Cold Navigation with Sign Feedback: PPO under Partial Observability

Husain Mistry

Abstract- We study a deterministic 2D grid navigation task with "hot–cold" sign feedback. The agent observes whether its last move decreased or increased the Manhattan distance to a hidden goal. The observation is non-Markov. A worst-case optimal strategy reaches the goal in at most $D_0 + 6$ steps, where D_0 is the start–goal Manhattan distance. We evaluate Proximal Policy Optimization with an LSTM backbone (Recurrent PPO). The learned policy approaches the theoretical bound on many episodes but shows gaps due to axis misidentification and turn dithering. The task provides a minimal, interpretable benchmark for reinforcement learning under partial observability.

I.INTRODUCTION

Many navigation settings are partially observable. In hot–cold search, the agent receives a binary sign after each action: $+1$ if the last move reduced the distance to the goal and -1 otherwise. The goal location is hidden. Despite the sparse signal, there exists a simple worst-case optimal strategy: probe both axes, then march to the goal, which takes at most $D_0 + 6$ steps.

Feed-forward PPO struggles because the state is not Markov. The agent must retain a memory of actions and signs to infer hidden goal directions. We deploy Recurrent PPO (PPO–LSTM) and measure how closely it matches the theoretical upper bound.

II.PROBLEM FORMULATION

Grid and goal. The environment is a $W \times H$ grid. At episode start, the agent is at $s = (x_s, y_s)$; the goal is at $g = (x_g, y_g)$. Define Manhattan distance

$$D_0 = |x_s - x_g| + |y_s - y_g|.$$

Actions and dynamics. The action set is $\{up, right, down, left\}$. Transitions are deterministic with clamping at borders. The episode ends on reaching the goal or on a step limit.

Observation. At time t the agent receives

$$o_t = [\, sign_t,\ \hat{x}_t,\ \hat{y}_t,\ onehot(a_{t-1}) \,],$$

where $sign_t \in \{-1, +1\}$ reports whether the previous move reduced the true Manhattan distance, $(\hat{x}_t, \hat{y}_t) \in [-1,1]^2$ are normalized coordinates, and $onehot(a_{t-1}) \in \{0,1\}^4$ encodes the previous action.

Reward. The per-step reward equals $sign_t$. A small terminal bonus is added on success. During training, episodes may be truncated after a fixed number of consecutive non-improving steps to avoid endless wandering. Evaluation uses deterministic policies without shaping.

Baseline policy. A worst-case optimal deterministic strategy: (i) probe one axis; if sign is negative, reverse once; (ii) march along that axis until overshoot, then correct by one step; (iii) repeat for the remaining axis. If both axes are nonzero, the worst-case steps are $D_0 + 6$; if only one axis is nonzero, $D_0 + 2$.

III.REINFORCEMENT LEARNING SETUP

We employ Recurrent PPO with an LSTM backbone (MlpLstmPolicy) to allow memory over action–feedback sequences. Policy and value networks consist of two fully connected layers (64 units each), followed by a single LSTM layer with hidden size 128. Eight parallel environments stabilize updates. Key hyperparameters include learning rate 3×10^{-4}, $n\_steps = 512$, batch size 2048, clip range 0.15, $\gamma = 0.99$, $\lambda = 0.95$, target KL 0.03, entropy coefficient 0.01, and value coefficient 0.3.

PPO training statistics (good runs).

Metric	Typical Range
Approx. KL	0.015–0.025
Clip fraction	0.10–0.20
Entropy loss	-1.3 → -0.6
Explained variance	0.6–0.8

IV.RESULTS

The recurrent policy approaches the theoretical bound in many episodes. Success rates exceed 90%; median gaps to the bound are ≈ 4 steps. Failure cases include axis misidentification, dithering near turns, and rare loops.

Example evaluation episodes. Ideal = $D_0 + 6$ for 2D cases.

Ep	Start	Goal	D_0	Ideal	Steps	Gap
165	(12,3)	(3,9)	15	21	25	+4
166	(13,16)	(9,20)	8	14	28	+14
169	(4,4)	(6,7)	5	11	21	+10
170	(18,16)	(10,12)	12	18	500	Timeout
174	(16,19)	(7,5)	23	29	25	-4

V.ANALYSIS

The observation is not Markov; single sign feedback conflates multiple hidden goal positions. Recurrent PPO learns to integrate sequences, but failures reveal where this integration is incomplete. Dithering dominates the gap. Early truncation reduces loops but not dithering.

VI.RELATED WORK

Hot–cold navigation tasks highlight memory and inference from minimal feedback. POMDPs require history integration; recurrent networks such as DRQN and PPO–LSTM address this. Our environment isolates partial observability without perceptual confounds.

VII.CONCLUSION

We presented a deterministic hot–cold navigation task with a known optimal bound. PPO–LSTM learns policies approaching the bound but exhibits characteristic failures. The environment provides a clean benchmark for studying partial observability in reinforcement learning, focusing on memory and reasoning rather than perception. Future work includes richer memory architectures and hierarchical option discovery.

REFERENCE

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.

[2] M. Hausknecht and P. Stone. Deep recurrent q-learning for partially observable MDPs. *arXiv:1507.06527*, 2015.