Probabilistic Recurrent Intention Switching Model

Wenyuan Sheng[1,*], Hao Zhu[1,2,*], and Joschka Boedecker[1,2]

[1]Department of Computer Science, University of Freiburg
[2]IMBIT//BrainLinks-BrainTools
[*]These authors contributed equally.

Abstract

Inverse reinforcement learning (IRL) recovers reward functions from observed behavior, yet traditional methods assume a single stationary reward that cannot capture goal switching within an episode. Recent multi-intention IRL methods address this by segmenting trajectories, but model intention transitions as either a memoryless Markov chain or via manual state augmentation with a fixed history window. We propose the Probabilistic Recurrent Intention Switching Model (PRISM), which replaces both mechanisms with a lightweight recurrent network that maps observation history to a per-step intention distribution. We prove that the resulting EM objective decomposes exactly into independent per-intention reward subproblems, each solvable in closed form, yielding an $\mathcal{O}(nK)$ E-step with no variational approximation. We evaluate PRISM on a non-Markovian gridworld, a mouse labyrinth, and BridgeData V2 robotic manipulation, the first large-scale robotic application of multi-intention IRL. Across all settings PRISM achieves the highest held-out log-likelihood while recovering nameable, temporally coherent intentions from unlabeled demonstrations, suggesting that discrete goal switching is present in both biological and artificial agents.

1 Introduction

Demonstrations from any goal-directed agent, whether biological or artificial, contain latent goal switches. A mouse navigating a labyrinth alternates between seeking water and returning home; a teleoperated robot arm switches between reaching, grasping, and placing within a single demonstration; even human drivers shift between lane-keeping, overtaking, and parking. Recovering the reward functions that drive each goal, the moments at which the agent switches between them, and the aspects of the history that trigger these switches is essential for understanding complex sequential behavior. Yet standard inverse reinforcement learning (IRL) assumes a single stationary reward, conflating multiple objectives into one function that cannot distinguish between them.

Three lines of work have attempted to lift this stationarity assumption. Dynamic IRL (DIRL) [Ashwood et al., 2022a] models the reward as a smoothly time-varying combination of spatial goal maps, but its continuity assumption conflicts with evidence that animals switch between *discrete* strategies [Ashwood et al., 2022b], and its inference requires T separate Bellman solves, one per timestep. Hierarchical Inverse Q-Learning (HIQL) [Zhu et al., 2024] replaces smooth variation with a first-order Markov chain over intentions, but the memoryless transition model cannot capture switching driven by cumulative experience such as fatigue or satiation, and its E-step requires the Baum–Welch forward–backward algorithm at $\mathcal{O}(nK^2)$ cost per trajectory. SWIRL [Ke et al., 2025] adds state-dependent transition kernels and history-augmented rewards, but represents history via state augmentation: with history length L on $|\mathcal{S}|$ states the augmented state space grows as $|\mathcal{S}|^L$ (e.g., L=2 on 127 states yields 127^2=16,129 augmented states), limiting L to small values in practice. None of these approaches scale naturally beyond small tabular environments.

Preprint.

We propose the *Probabilistic Recurrent Intention Switching Model* (PRISM), a framework that absorbs all temporal complexity into a single lightweight recurrent neural network. At each timestep the network reads an observation from the environment, updates a hidden state that summarizes the trajectory so far, and outputs a soft assignment over a finite set of intentions. This assignment is combined with per-intention Boltzmann policy likelihoods to form a per-step posterior responsibility over intentions. We prove that the resulting expectation-maximization (EM) objective decomposes exactly into independent per-intention reward subproblems, each solvable in closed form via inverse action-value iteration (IAVI) [Kalweit et al., 2020], requiring no variational approximation. The posterior factorizes into independent per-step terms, yielding an $\mathcal{O}(nK)$ E-step. With approximately 50K trainable parameters in a single-layer RNN, PRISM trains in minutes on a laptop GPU and produces human-interpretable reward maps without manual specification of the temporal horizon.

We evaluate PRISM on three domains of increasing complexity, each testing a different property of the framework. Our central application is the *127-node mouse labyrinth* [Rosenberg et al., 2021], where PRISM recovers three intentions (water-seeking, homing, exploration) aligned with known biological drives, matching the modes identified by both DIRL [Ashwood et al., 2022a] and SWIRL [Ke et al., 2025] on the same dataset while achieving higher held-out log-likelihood. A *frustration gridworld* with a hidden counter provides controlled validation that PRISM captures provably non-Markovian switching, which is essential for modeling history-dependent behavioral states such as satiation and fatigue. The *BridgeData V2 robotic manipulation dataset* [Walke et al., 2023] tests whether the method generalizes beyond neuroscience: without any supervision, PRISM discovers four temporally coherent manipulation phases (approach, grasp, carry, idle) from human-teleoperated demonstrations. Together, these experiments suggest that discrete intention switching is present in both biological and artificial agents, and that the reward maps PRISM recovers provide a useful lens for interpreting the latent goals behind complex sequential behavior.

In summary, our contributions are: (i) the PRISM framework with a proven EM decomposition and closed-form reward recovery; (ii) an $\mathcal{O}(nK)$ E-step via posterior factorization, compared to the $\mathcal{O}(nK^2)$ forward–backward pass required by Markov-chain-based alternatives; (iii) to our knowledge, the first application of multi-intention IRL to a large-scale robotic manipulation dataset; (iv) recovery of nameable, interpretable intentions across three domains without supervision.

2 Related Work

Inverse reinforcement learning. IRL recovers a reward function from demonstrations, assuming the expert maximizes long-term return. The problem was formalized by Ng et al. [2000], who showed it is inherently ill-posed. Maximum entropy IRL [Ziebart et al., 2008] and its causal variant [Ziebart et al., 2010] resolve this ambiguity via entropy regularization. Kalweit et al. [2020] derived a closed-form reward solution via inverse action-value iteration (IAVI), which we use as the inner-loop solver in our EM framework. For a comprehensive survey, see Arora and Doshi [2021].

Multi-intention and dynamic IRL. Standard IRL assumes a single fixed reward. In practice, agents switch between distinct strategies within an episode [Ashwood et al., 2022b]. Babes et al. [2011] cluster entire trajectories by intention but do not allow within-episode switching. Bayesian nonparametric methods [Dimitrakakis and Rothkopf, 2011, Surana and Srivastava, 2014] avoid fixing the number of intentions but scale poorly. DIRL [Ashwood et al., 2022a] models reward as a continuously time-varying combination of goal maps; HIQL [Zhu et al., 2024] uses a first-order Markov chain; SWIRL [Ke et al., 2025] adds state-dependent transitions with fixed-window history augmentation. PRISM combines the strengths of all three: discrete switching like HIQL but with memory; history-dependent like SWIRL but learned end-to-end; data-driven like DIRL but with discrete intentions and closed-form reward recovery.

Hierarchical imitation learning and option discovery. The options framework [Sutton et al., 1999] decomposes policies into temporally extended sub-policies. CompILE [Kipf et al., 2019] segments demonstrations into composable latent skills, and play-data approaches [Lynch et al., 2020] discover reusable motor primitives from unstructured teleoperation. These methods recover *policies* or *skills*; PRISM instead recovers per-intention *reward functions*, which are more compact, transferable across dynamics, and directly interpretable. PRISM can be seen as performing "inverse option discovery," recovering the reward structure that generates the behavioral options.

Imitation learning at scale. Behavioral cloning [Ross et al., 2011], GAIL [Ho and Ermon, 2016], and recent generalist robot policies such as RT-1 [Brohan et al., 2022] learn policies directly from demonstrations without recovering reward structure. PRISM complements these approaches: the intention segmentation it produces could serve as a pre-processing step for skill-conditioned imitation learning.

Interpretable reward and decision models. Reward decomposition [Juozapaitis et al., 2019] and programmatic RL [Verma et al., 2018] seek to make RL decisions transparent. PRISM contributes to this space by recovering discrete, nameable intentions and per-intention reward maps from unlabeled demonstrations.

Positioning. PRISM is a table-based, model-based (requires known or estimated transition model P), offline multi-intention IRL method. It operates on fully observed demonstration trajectories: all states, actions, and observations are available before inference begins. Its EM algorithm alternates between closed-form reward recovery via IAVI and gradient-based updates to the recurrent intention network, without end-to-end differentiation through the Bellman equation.

3 Background

Notation. We consider an MDP $\langle \mathcal{S}, \mathcal{A}, P, \gamma \rangle$ with finite state space $\mathcal{S}$, finite action space $\mathcal{A}$, transition function $P(s' \mid s, a)$, and discount factor $\gamma \in [0, 1)$. A policy $\pi(a \mid s)$ specifies a distribution over actions in each state.

Hidden-intention MDP. A Hidden-Intention MDP (HI-MDP) extends the standard MDP with a latent intention space $\mathcal{Z}$ and is denoted $\langle \mathcal{Z}, \mathcal{S}, \mathcal{A}, P, \{r_z\}_{z \in \mathcal{Z}}, \gamma \rangle$. At each timestep t, the expert operates under a latent intention $z_t \in \mathcal{Z}$ and receives reward according to $r_{z_t} \colon \mathcal{S} \times \mathcal{A} \to \mathbf{R}$. The expert also perceives an observation $\varphi_t \in \mathbf{R}^m$ from the environment; the observation need not be in one-to-one correspondence with the state. For each trajectory $\xi = \{(s_1, a_1), \ldots, (s_n, a_n)\}$ there is a corresponding observation sequence $\psi = \{\varphi_1, \ldots, \varphi_n\}$, and we denote the set of all such sequences by $\mathcal{O}$. The mechanism by which observations influence intention assignments is left unspecified in the HI-MDP; PRISM instantiates this mechanism via a learned gating function f_θ (§4.1).

Inverse action-value iteration. Given demonstrations $\mathcal{D}$ in an MDP with known P, IAVI [Kalweit et al., 2020] formulates IRL as maximum likelihood estimation under a Boltzmann policy and solves for the reward in closed form via least squares. We use IAVI as the inner-loop solver in our EM framework; the full optimization program is given in §A.1.

4 Methods

4.1 Probabilistic Recurrent Intention Switching Model

We formulate the multi-intention IRL problem under three assumptions.

Assumption 1. *Each expert demonstration step is generated according to a Boltzmann-optimal policy under one of the reward functions in a K-dimensional finite set $\mathcal{R} = \{r_1, \ldots, r_K\}$.*

Assumption 2. *Each trajectory ξ in the demonstration set $\mathcal{D}$ is accompanied by an observation sequence $\psi = \{\varphi_1, \ldots, \varphi_n\} \in \mathcal{O}$, as defined in §3. The observation φ_i may encode any information available at step i, including the state, the action, or features derived from either.*

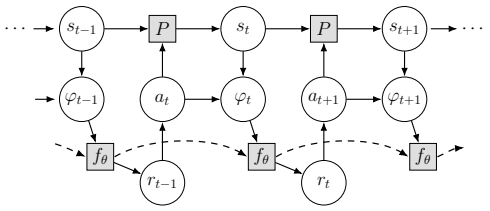

Figure 1: Probabilistic graphical model of the expert's decision process. Dashed lines represent the recurrent connection of f_θ.

Assumption 3. *The soft intention assignment at each demonstration step is produced by a parametric function $f_\theta \colon \mathbf{R}^m \to \Delta^K$, where $f_\theta(\varphi_i)_k$ denotes the weight assigned to intention k given observation φ_i, for $k = 1, \ldots, K$. The function f_θ may maintain internal state across steps within a trajectory, as realized by recurrent neural networks.*

Under these assumptions the expert's decision process is fully specified by $\Theta = \{\theta, \mathcal{R}\}$, and the resulting decision network can be represented with a probabilistic graphical model (Figure 1). We refer to the resulting framework as the *Probabilistic Recurrent Intention Switching Model* (PRISM). The PRISM inference problem consists of determining (1) a set of reward functions and (2) the intention index for each demonstration step that best jointly explain the observed expert behavior. An expectation-maximization (EM) algorithm can be devised to iteratively learn Θ. For convenience, we introduce $\eta = \{z_1, \ldots, z_n\}$ as the predicted sequence of intention indices for trajectory ξ and corresponding observations ψ. Then each iteration of the EM process maximizes the auxiliary function:

$$\text{maximize} \quad J(\Theta^+ \mid \Theta) = \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O}),\,\eta} \log \mathbf{P}(\xi, \eta \mid \psi, \Theta^+), \tag{1}$$

where Θ^+ is the optimization variable and $(\mathcal{D}, \mathcal{O}), \Theta$ are the problem data.

Theorem 1. *Solving problem* (1) *is equivalent to solving a sequence of independent optimization problems: first maximize over the intention network parameters θ^+,*

$$\text{maximize (over } \theta^+) \quad \mathbf{E}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\textstyle\sum_{k=1}^{K} \sum_{i=1}^{n} \mathbf{P}(z_i{=}k \mid \xi, \psi, \Theta) \log f_{\theta^+}(\varphi_i)_k \right), \tag{2}$$

and then maximize independently over each reward $r_k^+ \in \mathcal{R}^+$,

$$\begin{array}{ll} \text{maximize (over } r_k^+) & \mathbf{E}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\textstyle\sum_{i=1}^{n} \mathbf{P}(z_i{=}k \mid \xi, \psi, \Theta) \log \pi_{r_k^+}(a_i \mid s_i) \right) \\ \text{subject to} & \pi_{r_k^+}(s,a) = \exp\big(Q(s,a) - \log \textstyle\sum_{a' \in \mathcal{A}} \exp Q(s,a')\big) \\ & Q(s,a) = r_k^+(s,a) + \gamma \textstyle\sum_{s'} P(s' \mid s,a) \max_{a' \in \mathcal{A}} Q(s',a') \\ & s \in \mathcal{S}, \quad a \in \mathcal{A}. \end{array} \tag{3}$$

This decomposition is exact, requiring no variational approximation. Each reward subproblem (3) reduces to a weighted IAVI problem solvable in closed form. Detailed proof is given in § A.2.

Posterior factorization and complexity. A key structural consequence is that, conditioned on (ξ, ψ, Θ), the intention variables $\{z_1, \ldots, z_n\}$ are mutually independent. The posterior decomposes as $\mathbf{P}(\eta \mid \xi, \psi, \Theta) = \prod_{i=1}^{n} \mathbf{P}(z_i \mid \xi, \psi, \Theta)$, where each per-step responsibility is the normalized product of the gating output and the per-intention policy:

$$\mathbf{P}(z_i{=}k \mid \xi, \psi, \Theta) = \frac{f_\theta(\varphi_i)_k \, \pi_{r_k}(a_i \mid s_i)}{\sum_{j=1}^{K} f_\theta(\varphi_i)_j \, \pi_{r_j}(a_i \mid s_i)}, \tag{4}$$

where f_θ assigns soft responsibility over intentions given the observed context φ_i, and π_{r_k} scores how well the observed action a_i is explained under intention k. Since the intention variables decouple across time steps, the E-step requires only $\mathcal{O}(nK)$ evaluations per trajectory.

4.2 Training Objective

The training objective for f_θ combines the negative log-likelihood from the M-step with temporal smoothness penalties. Writing $w_{i,k}$ for the per-step responsibility in Eq. (4):

$$\mathcal{L} \;=\; \mathcal{L}_{\mathrm{NLL}} \;+\; \lambda_{\ell_1} \mathcal{L}_{\ell_1} \;+\; \lambda_{\mathrm{kl}} \mathcal{L}_{\mathrm{kl}}, \tag{5}$$

where

$$\begin{aligned} \mathcal{L}_{\mathrm{NLL}} &= - \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \sum_{i=1}^{n} \sum_{k=1}^{K} w_{i,k} \log f_\theta(\varphi_i)_k, \\ \mathcal{L}_{\ell_1} &= \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \sum_{i=2}^{n} \sum_{k=1}^{K} w_{i,k} \big| f_\theta(\varphi_i)_k - f_\theta(\varphi_{i-1})_k \big|, \\ \mathcal{L}_{\mathrm{kl}} &= \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \sum_{i=2}^{n} D_{\mathrm{kl}}\big(f_\theta(\varphi_{i-1}) \,\|\, f_\theta(\varphi_i)\big). \end{aligned}$$

where the smoothness terms are computed over consecutive time steps within each trajectory. The ℓ_1-penalty suppresses rapid intention switches; the KL-divergence term penalizes distributional

Algorithm 1 PRISM: Probabilistic Recurrent Intention Switching Model

Require: Expert demonstrations and observations $(\mathcal{D}, \mathcal{O})$, intention network f_θ, reward set dimension K.

initialize θ, $r_1, \ldots, r_K$.
repeat
  Compute posterior $\mathbf{P}(z_i{=}k \mid \xi, \psi, \Theta)$ for each demonstration step via Eq. (4).
  Update θ by applying SGD on $(\mathcal{D}, \mathcal{O})$ with loss $\mathcal{L}$ (Eq. (5)): $\theta := \operatorname{argmin}_\theta \mathcal{L}(\theta)$.
  for $k = 1, \ldots, K$ **do**
    Update r_k by solving problem (3) via IAVI with each demonstration weighted by $\mathbf{P}(z_i{=}k \mid \xi, \psi, \Theta)$.
  end for
until stopping criterion is satisfied.

shift more gently, preserving distributional shape. The two are complementary: ℓ_1 provides sparse switching while KL preserves smooth transitions. Both weights $\lambda_{\ell_1}, \lambda_{\text{kl}} \geq 0$ are set in §5.2. We note that these penalties act as discriminative regularizers rather than a probabilistic prior over intention sequences. The RNN hidden state provides implicit temporal modeling, while the regularizers encourage the output distribution to vary smoothly.

4.3 Intention Network Architecture

The intention network maps a sequence of observations $(\varphi_1, \ldots, \varphi_n)$ to a per-step distribution over K intentions. Each observation φ_t is embedded into a d-dimensional vector; the sequence of embeddings is processed by a recurrent encoder (RNN, LSTM, or Transformer); and each encoder output is projected to K logits followed by a softmax to produce $f_\theta(\varphi_t) \in \Delta^K$. In the tabular experiments we set $\varphi_t = (s_t, a_t)$ and use learned state and action embedding matrices; in visual domains φ_t may be the output of a pretrained encoder. A single-layer IntentionRNN with $d{=}128$ and $K{=}4$ has approximately 50K trainable parameters.

5 Experiments

5.1 Frustration Gridworld

This experiment validates PRISM on a controlled environment where the ground-truth intention-switching mechanism is known and provably non-Markovian.

Setup. A $5{\times}5$ gridworld with five actions (up, down, left, right, stay) and stochastic dynamics (90% success rate). The expert has two policies: π_{goal} toward $(4, 4)$ and π_{abandon} toward the origin. A hidden frustration counter c, initially zero, increments at each barrier encounter; the switching probability is $\min\{0.15c, 0.9\}$, making intention transitions strictly non-Markovian. The counter resets upon switching. We generate 1024 trajectories and evaluate with 5-fold cross-validation.

Results. We compare PRISM ($K{=}2$) against HIQL [Zhu et al., 2024], IAVI [Kalweit et al., 2020], maximum causal entropy IRL [Ziebart et al., 2010], and maximum entropy IRL [Ziebart et al., 2008]. PRISM achieves the highest test log-likelihood with the smallest variance across folds (Figure 2a) and the lowest expected value difference under both intentions (Table B.1). HIQL follows closely in log-likelihood but shows higher EVD under the abandon intention, attributable to its Markov model being unable to capture the cumulative counter. Figure 2b shows that PRISM's recovered state-value heatmaps closely match the ground truth under both intentions; full comparisons with all baselines are in §B. Figure 2c displays the temporal intention posterior for a representative trajectory overlaid with the hidden frustration counter: PRISM's posterior tracks the accumulating frustration, switching sharply after repeated barrier encounters.

5.2 Mouse Labyrinth Navigation

Moving from simulation to real biological data, we test whether PRISM recovers interpretable intentions aligned with known biological drives.

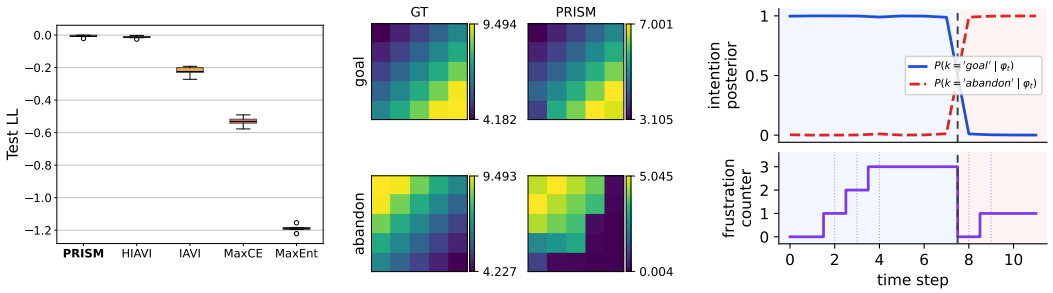

Figure 2: Frustration gridworld. **(a)** Test log-likelihood: PRISM achieves the highest score with the smallest variance. **(b)** State-value heatmaps (GT vs PRISM) under the goal and abandon intentions. **(c)** Temporal intention posterior (top) overlaid with the hidden frustration counter (bottom) for a representative trajectory; PRISM's posterior tracks the accumulating frustration and switches sharply after repeated barrier encounters. Full comparisons with all baselines are in §B.

Dataset and benchmarks. We use the freely moving mouse navigation dataset from Rosenberg et al. [2021], the same benchmark used by DIRL [Ashwood et al., 2022a], HIQL [Zhu et al., 2024], and SWIRL [Ke et al., 2025]. Ten water-restricted mice explored a 127-node labyrinth for 7 hours in darkness; water was available at a designated end node but collectible at most once per 90 seconds, creating a non-Markovian reward structure. Following Ke et al. [2025], we segment raw sequences into 238 trajectories of 500 steps each. Models are evaluated via 5-fold cross-validation. We compare PRISM (K=3, IntentionRNN) against IAVI, SWIRL (S-2), and maximum causal entropy IRL. Smoothness penalties are set to λ_{ℓ_1}=2.22, λ_{kl}=1.48; in all other experiments both are zero.

Behavior prediction. PRISM achieves the highest test log-likelihood (-0.65), outperforming SWIRL S-2 (-0.73) on the same data (Figure 3a). The improvement is notable because both methods model history-dependent intention dynamics: PRISM obtains a better fit while learning the relevant history horizon end-to-end, without manual state augmentation. Figure 3b shows that all three architectures (RNN, LSTM, Transformer) improve monotonically with K; we adopt the vanilla RNN as default for its simpler dynamics and stable performance. We select K=3 because the labyrinth task contains only two explicit behavioral events (water-seeking and homing), and the plateau in test log-likelihood beginning at K=4 (Figure 3b) indicates that three intentions already capture the dominant behavioral structure, with the third corresponding to exploration. This setting also allows direct comparison with SWIRL [Ke et al., 2025], which reports its primary results at K=3.

Recovered reward maps and segmentation. Figure 3c shows the greedy policy and per-state action confidence for each intention, where color depth indicates how decisively the greedy action dominates over alternatives. Under *water*, the greedy policy traces a clear path from the entrance toward the water port, with high confidence along the main corridor. Under *home*, the major branching nodes consistently point toward the entrance, forming convergent flow inward. Under *explore*, actions are distributed across peripheral nodes with no dominant target and lower overall confidence, reflecting the diffuse nature of exploratory behavior. These three modes are consistent with those identified by SWIRL [Ke et al., 2025]. Figure 3d shows that hybrid regularization produces temporally coherent segments whose boundaries align with behavioral events (water-port visits, home visits). The four configurations shown in Figure 3d span the regularization spectrum from unconstrained to fully penalized.

Table 1 reports wall-clock timing. PRISM converges in $\sim$140 EM iterations on the labyrinth ($\sim$5 min total) on a NVIDIA RTX 3060 mobile GPU, and $\sim$80 iterations on Bridge V2 ($\sim$12 min), on a NVIDIA RTX 5060Ti GPU.

5.3 BridgeData V2: Robotic Manipulation

We push PRISM to a qualitatively new regime with high-dimensional visual observations and no prior knowledge of intentions, which constitutes, to our knowledge, the first application of multi-intention IRL to a large-scale robotic manipulation dataset.

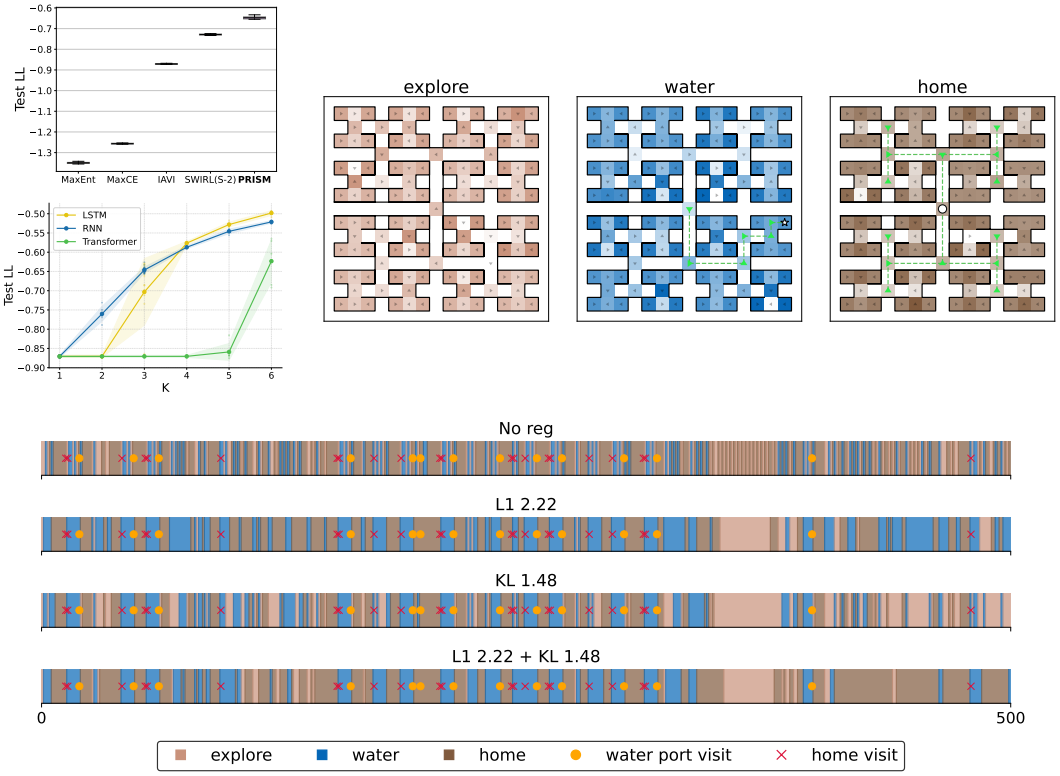

Figure 3: Labyrinth results (PRISM, K=3, IntentionRNN, hybrid regularization, 238 mouse trajectories). **(a)** Test log-likelihood: PRISM outperforms all baselines. **(b)** Test log-likelihood vs K for three architectures. **(c)** Recovered reward maps and greedy policy flow for the three inferred intentions (star: water port; circle: entrance). **(d)** Temporal intention segmentation under four regularization configurations. Orange dots: water-port visits; red crosses: home visits.

Dataset	**E-step**	**M-step (per latent)**		**Total / iter**	**Conv. iters**
	(posterior)	IAVI	Intention net		
Labyrinth	0.03 s	0.60 s	0.09 s	1.92 s	∼140
Bridge V2	4.10 s	1.53 s	2.90 s	8.54 s	∼80

Table 1: Wall-clock time per EM iteration.

Motivation. In the labyrinth, recovered intentions align with known biological drives. BridgeData V2 contains no such prior: each trajectory is a human-teleoperated manipulation sequence. If PRISM nonetheless discovers temporally coherent segments corresponding to recognizable manipulation phases, it provides evidence that intention structure is a general property of goal-directed behavior rather than an artifact specific to biological agents, and that the recovered reward maps can help explain the purpose behind human-operated demonstrations.

Continuous-to-discrete pipeline. Each 256×256 RGB frame is encoded by a frozen pretrained visual encoder; 7D continuous actions are retained raw. k-means clustering produces $|\mathcal{S}|$ state tokens and $|\mathcal{A}|$=32 action tokens. We compare DINOv2 [Oquab et al., 2023] (self-supervised; ViT-S/B/L) and SigLIP2 [Tschannen et al., 2025] (vision-language; ViT-B). Table 2 reports discretization statistics across encoders and granularities. Discretization quality is governed by encoder architecture rather than scale: DINOv2-S, -B, and -L produce nearly identical statistics, so we adopt the smallest variant to minimize compute. The critical gap is between DINOv2 and SigLIP2: at $|\mathcal{S}|$=2048, SigLIP2-B collapses visually distinct frames into shared tokens (11.8 average revisits vs 7.1 for DINOv2-S), reducing the action-level state discrimination that IRL requires. At our default granularity ($|\mathcal{S}|$=2048, DINOv2-S), approximately 75% of visited states receive multiple observations, providing adequate

Table 2: Effect of k-means state granularity on trajectory discretization statistics (mean±std across trajectories).

Encoder	Metric	Number of states $\lvert\mathcal{S}\rvert$			
		1024	2048	3072	4096
DINOv2-S	Coverage (%)	$0.64_{\pm 0.40}$	$0.37_{\pm 0.23}$	$0.26_{\pm 0.17}$	$0.21_{\pm 0.13}$
	Avg. revisits	$8.0_{\pm 6.1}$	$7.1_{\pm 5.7}$	$6.8_{\pm 5.6}$	$6.5_{\pm 5.6}$
	Singleton (%)	$21.7_{\pm 19.1}$	$25.5_{\pm 19.9}$	$27.6_{\pm 20.5}$	$28.9_{\pm 20.9}$
DINOv2-B	Coverage (%)	$0.69_{\pm 0.42}$	$0.40_{\pm 0.25}$	$0.28_{\pm 0.18}$	$0.22_{\pm 0.14}$
	Avg. revisits	$7.6_{\pm 5.9}$	$6.6_{\pm 5.4}$	$6.3_{\pm 5.3}$	$6.0_{\pm 5.1}$
	Singleton (%)	$23.4_{\pm 19.3}$	$27.4_{\pm 20.2}$	$29.6_{\pm 20.7}$	$30.9_{\pm 21.0}$
DINOv2-L	Coverage (%)	$0.69_{\pm 0.42}$	$0.40_{\pm 0.25}$	$0.29_{\pm 0.18}$	$0.23_{\pm 0.14}$
	Avg. revisits	$7.4_{\pm 5.9}$	$6.5_{\pm 5.4}$	$6.2_{\pm 5.3}$	$5.9_{\pm 5.2}$
	Singleton (%)	$23.4_{\pm 19.1}$	$27.1_{\pm 20.0}$	$29.2_{\pm 20.4}$	$30.8_{\pm 20.7}$
SigLIP2-B	Coverage (%)	$0.38_{\pm 0.25}$	$0.22_{\pm 0.15}$	$0.16_{\pm 0.11}$	$0.13_{\pm 0.09}$
	Avg. revisits	$13.3_{\pm 9.6}$	$11.8_{\pm 8.9}$	$11.0_{\pm 8.4}$	$10.5_{\pm 8.1}$
	Singleton (%)	$14.7_{\pm 18.5}$	$17.5_{\pm 19.2}$	$19.0_{\pm 19.6}$	$20.0_{\pm 19.9}$

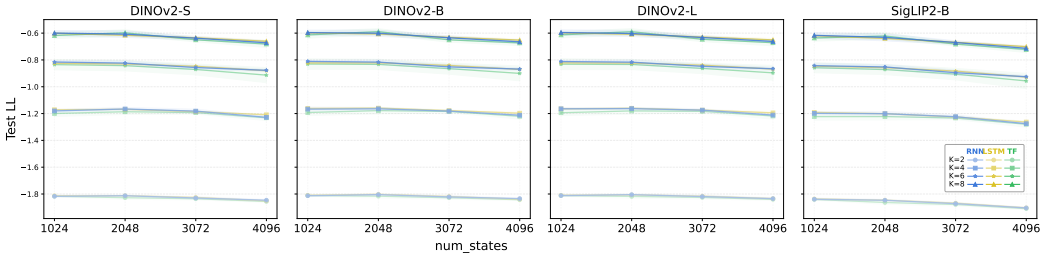

Figure 4: BridgeData V2: test log-likelihood by encoder, intention network, and number of latents K.

support for transition estimation. This encoder advantage is reflected in test log-likelihood: DINOv2 consistently outperforms SigLIP2 across all configurations (Figure 4).

Results. Test log-likelihood improves monotonically with K (Figure 4): from -1.81 at $K{=}2$ to -0.82 at $K{=}6$ with no saturation, unlike the labyrinth where gains plateau by $K{=}4$. IntentionRNN and IntentionLSTM perform similarly; IntentionTransformer lags at small K but narrows the gap as K increases, reproducing the same ranking observed on the labyrinth and supporting IntentionRNN as a robust default. We report visualizations at $K{=}4$ because Bridge V2 trajectories are relatively short; at $K{=}5$ and beyond, segments become excessively brief and lose interpretability, with new classes capturing transient gripper adjustments rather than distinct behavioral modes.

Figure 5 visualizes per-timestep intention assignments ($K{=}4$, DINOv2-S). Without any supervision, PRISM recovers four classes: APPROACH/DEPART (arm moving toward or away from target), GRASP (gripper closing), CARRY (transporting object), and IDLE (minor adjustments). The boundaries are sharp and consistent across trajectories with different objects and environments, confirming temporally coherent segmentation. Additional examples are provided in §E.

6 Conclusion and Discussion

We presented PRISM, an EM-based framework for multi-intention IRL that parameterizes intention dynamics through a lightweight recurrent neural network. By replacing the Markov chain of HIQL and the fixed-window augmentation of SWIRL with an end-to-end learned recurrent mapping, PRISM adapts to the temporal structure of each dataset without manual design choices, while retaining closed-form reward recovery. The proven EM decomposition separates the reward and gating subproblems exactly, and the $\mathcal{O}(nK)$ E-step scales gracefully. Experiments on three progressively complex domains, from an abstract gridworld validating non-Markovian theory, through real mouse behavior confirming biological interpretability, to large-scale robotic manipulation demonstrating

Figure 5: Per-timestep intention assignments on BridgeData V2 trajectories. Frame borders are color-coded by predicted intention: orange (CARRY), blue (IDLE), green (GRASP), magenta (APPROACH/DEPART).

cross-domain generality, show that PRISM consistently achieves the highest held-out likelihood while recovering nameable intentions from unlabeled data. The recovered reward maps provide a structured characterization of the latent goals behind observed behavior, revealing not just *what* an agent did but *which objective* it was pursuing at each moment.

Limitations and future work. (a) *Tabular assumption.* PRISM currently requires a discrete MDP. Our k-means pipeline bridges this gap for visual domains but introduces quantization error. Learned codebooks such as VQ-BeT [Lee et al., 2024] or SAQ [Luo et al., 2023] could adapt cluster boundaries to maximize reward-recovery quality. However, discretization has fundamental limits: even cutting-edge vision-language-action models such as $\pi_{0.5}$ [Physical Intelligence et al., 2025] employ action tokenizers like FAST for pre-training their language-model backbone, yet such tokenization serves the model architecture rather than enabling fine-grained dexterous control directly. The continuous structure that dexterous manipulation demands is inherently lost through any discretization, which motivates moving beyond tabular methods entirely. (b) *Model-free extension.* The natural path forward is to replace IAVI with model-free inverse Q-learning [Kalweit et al., 2020], which operates directly in continuous state–action spaces via function approximation and does not require an explicit transition matrix. This would remove both the discretization bottleneck of (a) and the model-based dependency, enabling PRISM to scale to dexterous robotic domains where tabular representations are infeasible. (c) *Online extension.* PRISM operates offline over fully observed trajectories, which suffices for post-hoc behavioral analysis but precludes real-time intention monitoring. Introducing a probabilistic transition prior $P(z_t \mid z_{t-1}, \varphi_{t-1})$ would enable sequential filtering for online applications. In preliminary experiments, such a Bayesian variant achieved comparable log-likelihood but produced less interpretable reward maps, suggesting that the prior structure requires careful design to preserve reward quality. (d) *Model selection.* Test log-likelihood improves monotonically with K, but beyond the true number of behavioral modes the additional intentions fragment trajectories into unnameable segments, particularly when trajectories are short (§5.3). A single-layer RNN with 128 hidden units already saturates performance on both datasets (Table D.2), suggesting that the switching dynamics in these domains are low-dimensional. Longer, more complex demonstrations would be needed to justify both larger K and more expressive sequence models.

Acknowledgments

This work has been funded as part of BrainLinks-BrainTools, which is funded by the Federal Ministry of Economics, Science and Arts of Baden-Württemberg within the sustainability program for projects of the Excellence Initiative II, and CRC/TRR 384 "IN-CODE".

References

Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, 2021.

Zoe Ashwood, Aditi Jha, and Jonathan W Pillow. Dynamic inverse reinforcement learning for characterizing animal behavior. *Advances in neural information processing systems*, 35:29663–29676, 2022a.

Zoe C Ashwood, Nicholas A Roy, Iris R Stone, International Brain Laboratory, Anne E Urai, Anne K Churchland, Alexandre Pouget, and Jonathan W Pillow. Mice alternate between discrete strategies during perceptual decision-making. *Nature Neuroscience*, 25(2):201–212, 2022b.

Monica Babes, Vukosi Marivate, Kaushik Subramanian, and Michael L Littman. Apprenticeship learning about multiple intentions. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 897–904, 2011.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Christos Dimitrakakis and Constantin A Rothkopf. Bayesian multitask inverse reinforcement learning. In *European workshop on reinforcement learning*, pages 273–284. Springer, 2011.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.

Zoe Juozapaitis, Anurag Koul, Alan Fern, Martin Erwig, and Finale Doshi-Velez. Explainable reinforcement learning via reward decomposition. In *IJCAI/ECAI Workshop on explainable artificial intelligence*, 2019.

Gabriel Kalweit, Maria Huegle, Moritz Werling, and Joschka Boedecker. Deep inverse q-learning with constraints. *Advances in neural information processing systems*, 33:14291–14302, 2020.

Jingyang Ke, Feiyang Wu, Jiyi Wang, Jeffrey Markowitz, and Anqi Wu. Inverse reinforcement learning with switching rewards and history dependency for characterizing animal behaviors. *arXiv preprint arXiv:2501.12633*, 2025.

Thomas Kipf, Yujia Li, Hanjun Dai, Vinicius Zambaldi, Alvaro Sanchez-Gonzalez, Edward Grefenstette, Pushmeet Kohli, and Peter Battaglia. Compile: Compositional imitation learning and execution. In *International Conference on Machine Learning*, pages 3418–3428. PMLR, 2019.

Seungjae Lee, Yibin Wang, Haritheja Etukuru, H Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior generation with latent actions. *arXiv preprint arXiv:2403.03181*, 2024.

Jianlan Luo, Perry Dong, Jeffrey Wu, Aviral Kumar, Xinyang Geng, and Sergey Levine. Action-quantized offline reinforcement learning for robotic skill learning. In *Conference on Robot Learning*, pages 1348–1361. PMLR, 2023.

Corey Lynch, Mohi Khansari, Ted Xiao, Vikash Kumar, Jonathan Tompson, Sergey Levine, and Pierre Sermanet. Learning latent plans from play. In *Conference on robot learning*, pages 1113–1132. Pmlr, 2020.

Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, page 2, 2000.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Physical Intelligence, Benjamin Freed, Antoine Dedieu, Clement Gehring, Nikolaos Gkanatsios, Kristian Hartikainen, Nikhil Joshi, Karl Labat, Haotian Li, Jianlan Luo, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Matthew Rosenberg, Tony Zhang, Pietro Perona, and Markus Meister. Mice in a labyrinth show rapid learning, sudden insight, and efficient exploration. *Elife*, 10:e66175, 2021.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

Amit Surana and Kunal Srivastava. Bayesian nonparametric inverse reinforcement learning for switched markov decision processes. In *2014 13th International Conference on Machine Learning and Applications*, pages 47–54. IEEE, 2014.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Abhinav Verma, Vijayaraghavan Murali, Rishabh Singh, Pushmeet Kohli, and Swarat Chaudhuri. Programmatically interpretable reinforcement learning. In *International conference on machine learning*, pages 5045–5054. PMLR, 2018.

Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023.

Hao Zhu, Brice De La Crompe, Gabriel Kalweit, Artur Schneider, Maria Kalweit, Ilka Diester, and Joschka Boedecker. Multi-intention inverse q-learning for interpretable behavior representation. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL `https://openreview.net/forum?id=hrKHkmLUFk`.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. Modeling interaction via the principle of maximum causal entropy. 2010.

A Theoretical and Technical Details

A.1 IAVI Formulation

Given expert demonstrations $\mathcal{D}$, the IRL problem under a Boltzmann policy is formulated as:

$$\begin{array}{ll} \text{maximize} & \mathbf{E}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \log \mathbf{P}\left(\xi \mid \pi_r\right) \\ \text{subject to} & \pi_r(a \mid s) = \exp\left(Q(s,a) - \log \sum_{a' \in \mathcal{A}} \exp Q(s,a')\right) \\ & Q(s,a) = r(s,a) + \gamma \sum_{s'} P(s' \mid s,a) \max_{a' \in \mathcal{A}} Q(s',a') \\ & s \in \mathcal{S}, \quad a \in \mathcal{A} \end{array} \tag{A.1}$$

where r is the optimization variable. When P is known, this can be solved in closed form via least squares, yielding IAVI [Kalweit et al., 2020].

A.2 Proof of Theorem 1

Proof. The objective function $J(\Theta^+ \mid \Theta)$ from problem (1) can be written as:

$$\begin{aligned} & J(\Theta^+ \mid \Theta) \qquad\qquad (\text{A.2}) \\ = & \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O}),\,\eta} \log \mathbf{P}(\xi, \eta \mid \psi, \Theta^+) \\ = & \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\sum_{\eta} \mathbf{P}(\eta \mid \xi, \psi, \Theta)\ \log \mathbf{P}(\xi, \eta \mid \psi, \Theta^+) \right) \\ = & \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\sum_{\eta} \mathbf{P}(\eta \mid \xi, \psi, \Theta) \sum_{i=1}^{n} \log \mathbf{P}(z_i^+ {=} z_i \mid \varphi_i, \theta^+)\ \mathbf{P}\big((s_i, a_i) \mid r_{z_i}^+\big) \right) \\ = & \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\sum_{\eta} \mathbf{P}(\eta \mid \xi, \psi, \Theta) \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbb{I}_k(z_i)\ \log \mathbf{P}(z_i^+ {=} k \mid \varphi_i, \theta^+)\ \mathbf{P}\big((s_i, a_i) \mid r_k^+\big) \right) \\ = & \mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\sum_{k=1}^{K} \sum_{i=1}^{n} \underbrace{\sum_{\eta} \mathbf{P}(\eta \mid \xi, \psi, \Theta)\, \mathbb{I}_k(z_i)}_{= \mathbf{P}(z_i = k \mid \xi, \psi, \Theta)} \log \mathbf{P}(z_i^+ {=} k \mid \varphi_i, \theta^+)\ \mathbf{P}\big((s_i, a_i) \mid r_k^+\big) \right) \\ = & \underbrace{\mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\sum_{k=1}^{K} \sum_{i=1}^{n} \mathbf{P}(z_i {=} k \mid \xi, \psi, \Theta)\ \log \mathbf{P}(z_i^+ {=} k \mid \varphi_i, \theta^+) \right)}_{\text{(I): depends only on } \theta^+} \qquad (\text{A.3}) \\ & + \underbrace{\mathop{\mathbf{E}}_{(\xi,\psi)\sim(\mathcal{D},\mathcal{O})} \left(\sum_{k=1}^{K} \sum_{i=1}^{n} \mathbf{P}(z_i {=} k \mid \xi, \psi, \Theta)\ \log \mathbf{P}\big((s_i, a_i) \mid r_k^+\big) \right)}_{\text{(II): depends only on } \mathcal{R}^+}, \qquad (\text{A.4}) \end{aligned}$$

where $\mathbb{I}_k$ is the indicator function with $\mathbb{I}_k(x) = 1$ for $x = k$ and 0 otherwise. Thus maximizing $J(\Theta^+ \mid \Theta)$ over Θ^+ is equivalent to separately maximizing (A.3) over θ^+ and (A.4) over $\mathcal{R}^+ = \{r_1^+, \ldots, r_K^+\}$.

By Assumption 3, $f_\theta(\varphi)_k = \mathbf{P}(r_k \mid \varphi)$, so the first optimization problem becomes (2). In (A.4), distinct k share no parameters, so the maximization decomposes into K independent subproblems. Each has the same structure as the single-intention IRL problem (A.1), with the i-th demonstration weighted by $\mathbf{P}(z_i = k \mid \xi, \psi, \Theta)$, yielding (3). The Boltzmann-policy constraints are introduced to make each subproblem tractable via IAVI. □

A.3 Intention Network Architecture Details

Each step (s_t, a_t) is encoded by two independent learned embedding matrices $E_s \in \mathbf{R}^{|\mathcal{S}| \times d}$ and $E_a \in \mathbf{R}^{|\mathcal{A}| \times d}$, producing input $x_t = E_s(s_t) + E_a(a_t)$. Batches of variable-length trajectories are

Method	EVD (MAE)		EVD (s_0)	
	goal	abandon	goal	abandon
PRISM (K=2)	$\mathbf{2.40 \pm 0.37}$	$\mathbf{5.60 \pm 0.45}$	$\mathbf{-1.74 \pm 0.59}$	$\mathbf{-6.02 \pm 0.85}$
HIQL (K=2)	2.51 ± 0.13	6.12 ± 0.38	-1.95 ± 0.54	-6.80 ± 1.25
IAVI (K=1)	2.06 ± 0.02	6.74 ± 0.00	-1.41 ± 0.01	-9.20 ± 0.00
MaxCausalEnt (K=1)	5.32 ± 0.00	6.67 ± 0.00	-3.32 ± 0.00	-9.12 ± 0.00
MaxEnt (K=1)	6.61 ± 0.00	6.76 ± 0.00	-4.16 ± 0.00	-9.10 ± 0.00

Table B.1: Expected value difference on the frustration gridworld (5-fold CV). Closer to zero is better.

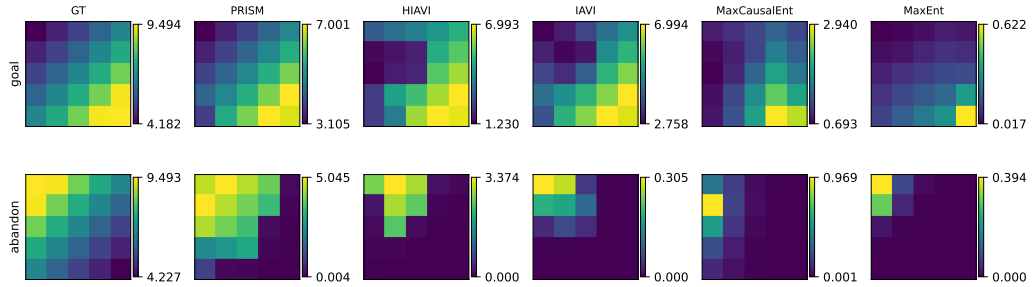

Figure B.1: Full state-value heatmaps on the frustration gridworld for all methods.

zero-padded with a binary mask. The recurrent variants (RNN, LSTM) use a single recurrent layer with hidden dimension d'; the Transformer variant applies sinusoidal positional encoding and a standard encoder with a padding mask. In all variants, K logits are converted to $f_\theta(\varphi_t) \in \Delta^K$ via softmax.

B Gridworld: Full Results

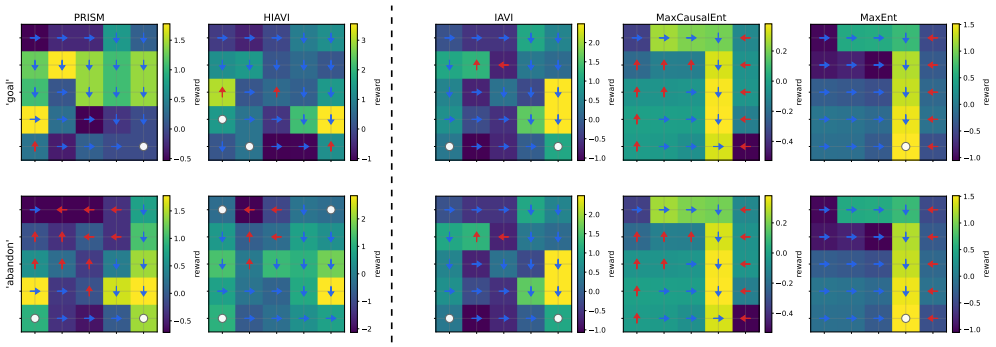

Figure B.2: Recovered per-state reward maps and greedy actions on the frustration gridworld.

Table C.1: Default hyperparameters for the labyrinth and Bridge V2 experiments.

Category	Hyperparameter	Symbol	Labyrinth	Bridge V2
MDP	Num. states	$\lvert\mathcal{S}\rvert$	127	2048
	Num. actions	$\lvert\mathcal{A}\rvert$	4	32
	Discount factor	γ	0.97	0.97
EM	Max EM iterations		180	150
	Random seed		42	42
Intention model	Num. latent intentions	K	3	4
	Architecture		IntentionRNN	IntentionRNN
Intention net	Embedding dim.	d	128	128
	RNN / LSTM hidden dim.	d'	128	128
	Num. layers		1	1
	Attn. heads (Transformer)		4	4
Optimiser	Learning rate		10^{-3}	10^{-3}
	RNN / LSTM epochs per M-step		1	1
	Transformer epochs per M-step		8	8
Regularisation	Regularisation type		KL + L1	
	L1 smoothness weight	λ_{ℓ_1}	2.22	0.0
	KL smoothness weight	λ_{kl}	1.48	0.0

K	Labyrinth		Bridge V2	
	Train LL	Test LL	Train LL	Test LL
1	-0.86801	-0.87071	-2.45171	-2.52187
2	-0.75432	-0.76024	-1.74644	-1.81282
3	-0.63754	-0.64624	-1.36307	-1.43000
4	-0.58141	-0.58714	-1.09903	-1.16584
5	-0.53726	-0.54573	-0.90255	-0.96912
6	-0.51220	-0.52129	-0.75780	-0.82373

Table D.1: Log-likelihood vs number of latent intentions K (IntentionRNN, 5-fold CV, 3 random seeds).

C Default Hyperparameters

D Ablation Experiments

Model	Labyrinth		Bridge V2	
	Train LL	Test LL	Train LL	Test LL
IntentionRNN	-0.63754	-0.64624	-1.09903	-1.16584
IntentionLSTM	-0.60993	-0.62705	-1.10082	-1.16839
IntentionTransformer	-0.86797	-0.87067	-1.12025	-1.18638

Table D.2: Log-likelihood by intention network architecture (d=128, single layer, 5-fold CV).

Figure E.1: More trajectories of intention assignments.

E BridgeData V2: Additional Intention Assignments