
Estimating cognitive biases with attention-aware inverse planning

Sounak Banerjee^{1,*} Daphne Cornelisse¹ Deepak Gopinath² Emily Sumner²
Jonathan DeCastro² Guy Rosman² Eugene Vinitzky¹ Mark K. Ho¹

¹New York University ²Toyota Research Institute

{sounak.banerjee, dc4971, vinitzky.eugene, mark.ho}@nyu.edu
{deepak.gopinath, emily.sumner, jonathan.decastro, guy.rosman}@tri.global

Abstract

People’s goal-directed behaviors are influenced by their cognitive biases, and autonomous systems that interact with people should be aware of this. For example, people’s attention to objects in their environment will be biased in a way that systematically affects how they perform everyday tasks such as driving to work. Here, building on recent work in computational cognitive science, we formally articulate the *attention-aware inverse planning problem*, in which the goal is to estimate a person’s attentional biases from their actions. We demonstrate how attention-aware inverse planning systematically differs from standard inverse reinforcement learning and how cognitive biases can be inferred from behavior. Finally, we present an approach to attention-aware inverse planning that combines deep reinforcement learning with computational cognitive modeling. We use this approach to infer the attentional strategies of RL agents in real-life driving scenarios selected from the Waymo Open Dataset, demonstrating the scalability of estimating cognitive biases with attention-aware inverse planning.

1 Introduction

Conventional wisdom suggests that people are not always rational, and this truism is now backed by over half a century of research in cognitive science [Simon, 1955, Tversky and Kahneman, 1974, Gigerenzer et al., 2000]. It is remarkable, then, that standard approaches to modeling goal-directed human behavior in machine learning and artificial intelligence (e.g., inverse reinforcement learning) assume that people act optimally—or at least approximately optimally—with respect to their goals [Ng and Russell, 2000, Arora and Doshi, 2021]. What explains this disconnect between psychological reality and basic modeling assumptions? How can we bridge this gap?

One reason why machine learning models of human behavior assume (approximate-)optimality is that although this assumption is false in fundamental ways, it underpins *expected utility theory* [von Neumann and Morgenstern, 1947, McFadden, 1974], which itself provides the basis for developing *scalable* algorithms that can make *generalizable* predictions about human behavior [Ho and Griffiths, 2022]. In contrast, general and scalable frameworks that account for why, when, and how people deviate from idealized rationality have not been previously available. This has changed in recent years: Building off of classic ideas in bounded rationality [Simon, 1990], cognitive scientists have begun to go beyond just *describing* deviations from ideal rationality to systematically *explaining*, *predicting*, and *influencing* those deviations by developing general theories of resource-limited cognition [Lewis et al., 2014, Griffiths et al., 2015, Gershman et al., 2015, Shenhav et al., 2017, Bhui et al., 2021]. The present work aims to apply these recent insights to the development of generalizable and scalable algorithms for modeling human decision-making.

Here, we focus specifically on how limited attention affects goal-directed behavior in complex tasks. As an example, consider the everyday activity of driving. When driving, people must systematically prioritize which objects in the environment to attend to (e.g., the car directly ahead, the child entering a blind spot) and which objects to ignore (e.g., the parked car on the opposite side of the street, the pile of leaves in a driveway). Whether an object is prioritized for attention depends on numerous factors, such as its immediate relevance for planning or a person's previous driving experience. Recent work by Ho et al. [2022] has proposed the framework of *value-guided construal*, which formalizes the process of prioritizing attention for planning as a meta-cognitive decision. The value-guided construal framework can be viewed as a principled *forward model* of how unobservable cognitive processes (e.g., attention, planning, and biases) integrate to produce observable behavior.

In this paper, we extend the value-guided construal framework to incorporate attentional biases and then invert it to specify the *attention-aware inverse planning* problem, in which the goal is to infer the biases of an attention-limited decision-maker from their behavior. We then study this approach in tabular domains and compare it to standard modeling approaches that assume decision-makers have *unlimited* attention (e.g., inverse reinforcement learning [Abbeel and Ng, 2004]). Finally, we develop an approach for inferring biases from synthetically generated behavior in real-world driving scenarios selected from the Waymo Open Motion Dataset [Ettinger et al., 2021], demonstrating the feasibility of scaling up attention-aware inverse planning to complex, naturalistic domains.

2 Background

A number of algorithmic approaches have been developed for modeling human behavior. The simplest approaches are based on *imitation learning* [Bain and Sammut, 1995], in which behavior is modeled by directly fitting a policy that predicts actions given observations. Imitation learning is effective when data is abundant and generalization is within-distribution, and has been successfully applied to model human behavior in real-world tasks such as highway driving [Lefèvre et al., 2014, Rudenko et al., 2020]. However, when training data is limited or generalization is out-of-distribution, it is necessary to turn to methods that make stronger assumptions about policy structure. In particular, methods based on *inverse reinforcement learning* (IRL) assume that observed behavior is generated by a policy that is optimal (plus decision-noise) with respect to the true task dynamics and a latent reward function, and attempts to infer the reward function [Abbeel and Ng, 2004, Ziebart et al., 2008].

In human-robot interaction and AI safety, generalizable models of human cognition and action are key for safe and effective interaction [Collins et al., 2024]. Although multi-agent interactions can be modeled without considering human biases via self-play or population-play algorithms [Silver et al., 2017, Jaderberg et al., 2019], agents trained this way perform poorly with people, especially in cooperative tasks [Carroll et al., 2019]. Additionally, accurate representations of how people make decisions are needed for more sophisticated forms of interaction that involve reasoning about the physical capabilities and intentions of human users [Gopinath et al., 2021a], modeling how humans reason about intentions [Dragan et al., 2013], or more complex forms of cooperation [Hadfield-Menell et al., 2016, Kleiman-Weiner et al., 2016]. Separate from this work on behavior modeling, work on attention modeling in the machine learning community has mainly been pursued in the context of specific tasks, such as driving [Kotseruba and Tsotsos, 2022, Gopinath et al., 2021b, Cao et al., 2022]. Although early work used cognitive models that relate attention to behavior [Salvucci, 2006], more recent work is based on data-driven approaches using gaze data and reaction times [Ahlström et al., 2022, Sundareswara et al., 2013, Li et al., 2024, Yang et al., 2020, Xue et al., 2025].

IRL has also inspired computational approaches to studying human social cognition. For example, Baker et al. [2009] proposed that human theory of mind could be conceptualized as *inverse planning*, Bayesian inference over a generative model of rational action. Additionally, as noted in the introduction, a large literature documents human deviations from expected utility theory [Kahneman and Tversky, 1979]. More recent theoretical work (e.g., *resource rational analysis* [Griffiths et al., 2015]) attempts to unify these disparate findings by explicitly modeling people's decision-making as a form of *resource constrained optimization* [Griffiths et al., 2015, Gershman et al., 2015]. Along these lines, Ho et al. [2022] proposed *value-guided construal*, a computational account of how people form simplified but useful world models when planning to make the best use of their limited attentional resources. Follow up work shows how learned biases [Ho et al., 2023] and properties of visuospatial attention [da Silva Castanheira et al., 2025] interact with and bias value-guided construal. The present paper aims to build on these findings and apply them to behavior and attention modeling.

3 Attention-aware inverse planning

This section first reviews preliminaries for sequential decision-making and attention-limited decision-making in the value-guided construal framework. We then describe an extension that incorporates attentional biases and invert this model to define the problem of *attention-aware inverse planning*.

3.1 Preliminaries

We model sequential decision-making problems as Markov Decision Processes (MDPs), which consist of a tuple $\langle S, A, T, R, \gamma \rangle$, where S is a state space, A is an action space, $T : S \times A \times S \rightarrow [0, 1]$ is a transition function, $R : S \times A \rightarrow \mathbb{R}$ is a reward function, and $\gamma \in [0, 1)$ is a discount rate [Sutton and Barto, 2018]. Policies are mappings from states to distributions over actions, $\pi : S \times A \rightarrow [0, 1]$. The expected discounted future reward of following a policy π from a state s defines its *value function*:

$$V(s, \pi) = \sum_a \pi(a | s) \left[R(s, a) + \gamma \sum_{s'} T(s' | s, a) V(s', \pi) \right] \quad (1)$$

An *optimal policy* is one that maximizes value at all states: $\pi^* = \arg \max_{\pi} V(s, \pi)$, for all $s \in S$.

Additionally, we assume an *object-oriented representation* of states [Diuk et al., 2008], in which each state is a subset of objects $\mathcal{O}$ with features $\mathcal{F}$ that can take on values $\mathcal{V}$ that includes scalars and real vectors. That is, define the “sub-state space” corresponding to a subset of objects $O \subseteq \mathcal{O}$ as the set of all maps from features of $o \in O$ to values, $S_O = (O \times \mathcal{F}) \rightarrow \mathcal{V}$. Then the full state space is their union: $S = \bigcup_{O \subseteq \mathcal{O}} S_O$. For example, in a driving setting, objects would correspond to vehicles (including the ego vehicle) or obstacles on the road. A state could then be represented as a map, $s = \{(\text{car_1, location}) \mapsto (0.0, 1.0), (\text{car_1, speed}) \mapsto 2.0, (\text{car_2, location}) \mapsto (5.5, 2.4), \dots\}$. We assume that if the transition function is defined for a state s with objects O , then it is also defined for the *abstracted* state that only includes a subset of objects $O' \subseteq O$, their features, and their values.

3.2 Value-guided construals

The current work aims to characterize attention-aware inverse planning, and our first step towards this goal is to specify a generative model of human planning in which attention plays a central role. To specify such a model, we build on existing work in computational cognitive science on resource-limited planning in humans, specifically the *value-guided construal* approach, which provides a domain-general computational framework for how people form simplified mental representations when planning [Ho et al., 2022, 2023].

In the value-guided construal approach, *top-down* attentional processes are modeled as selecting a subset of relevant details (e.g., objects and/or their affordances) that allow a person to plan an effective solution to a task. As a simple but illustrative example, imagine trying to pass another car on the highway when there are two additional cars, one in the passing lane and one parked on the side of the road. Intuitively, the car being passed and the car in the passing lane are relevant to planning, while the parked car is irrelevant. Value-guided construal formalizes this intuition as a cost-benefit calculation: a *construal* of the situation that includes the cars on the road has a higher *behavioral utility* than one that only includes the car being passed because the latter leads to a plan that could lead to worse outcomes (e.g., potentially crashing into the car in the passing lane). Conversely, the construal with the car being passed and the car in the passing lane is also *less representationally complex* than one that includes all three cars, as it represents fewer entities, but it still has the same behavioral utility since the presence of the parked car does not affect optimal behavior. Thus, value-guided construal predicts that a rational, resource-limited decision-maker trading off behavioral utility and representational complexity will form a construal that only includes the two cars on the road—that is, they will only attend to those two cars.

In the current work, we formalize the process of value-guided construal by controlling which objects are included in a state representation. That is, given a true state s composed of objects O , a *task construal* is a subset of objects $C \subseteq O$, and the corresponding *construed state* is $s[C] = \{(o, f) \mapsto v \in s \mid o \in C\}$, the subset of the object-feature to value pairs in s that only includes the objects in C . Then, we define the *construed policy* as the optimal policy where choices are made with respect

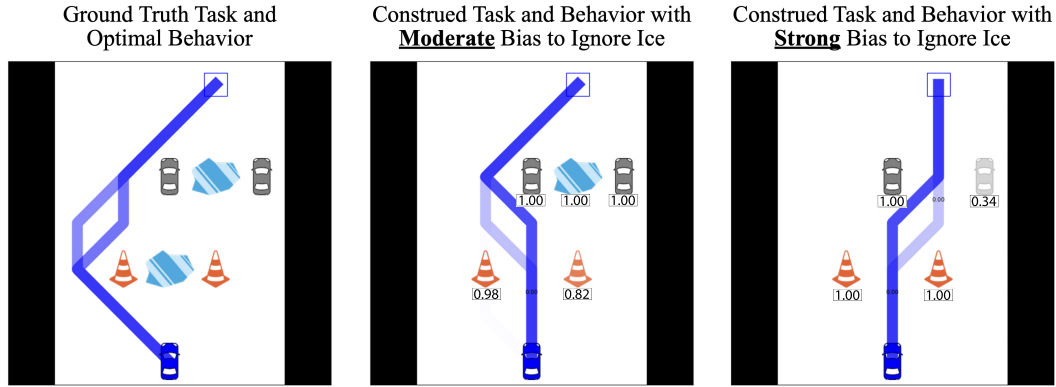


Figure 1: Attentional biases affect planned behaviors. (Left) A simple DrivingWorld scenario in which an agent (blue car) receives +100 reward upon reaching a goal location (blue square). Hitting a traffic cone results in -10 reward, hitting a parked car results in -100 reward and termination, and moving on ice leads to slipping to the left or right with probability 0.4 (see main text for full domain specification). Samples from the optimal policy ($n = 100$, blue lines) avoid both ice patches. (Center) Attention-aware inverse planning assumes that a decision-maker's behavior reflects the formation of simplified task construals that only include task-relevant details [Ho et al., 2022]. In the current scenario, a decision-maker who also has a moderate bias to ignore ice ($\lambda_{\text{Ice}} = -10$) will drive over the first ice patch but not the second. Shown is the average construal and trajectories that the agent expects to occur (that is, it does not consider potentially slipping on the first ice patch). Numbers indicate the marginal probability of attending to an object. (Right) A decision-maker with a strong bias to ignore ice ($\lambda_{\text{Ice}} = -100$) will drive over both ice patches and risk hitting the parked cars. For both, $\lambda_{\text{Cone}} = 10$ and $\lambda_{\text{Parked}} = 0$.

to the construed state: $\pi_C(a | s) = \pi^*(a | s[C])$.¹ The *value of representation* associated with a construal is then a sum of two terms:

$$\text{VOR}(s, C) = V(s, \pi_C) + \text{Cost}(C) \quad (2)$$

The first term is the *behavioral utility* of a construal, the actual value of acting optimally from a state *as if* the construed state were the case. The second term is the *cognitive cost* of the construal, which we assume is simply the negative of the cardinality of the construal (i.e., the number of objects represented) as in previous work [Ho et al., 2022].

Note that while construals are parameterized as sets of objects, the resulting construed policy π_C incorporates all interactions between construed objects. This is because the policy is associated with a value function that is derived from construed reward and transition functions for the task:

$$Q_C(s, a) = R_C(s, a) + \gamma \sum_{s'} T_C(s' | s, a) V_C(s') \quad (3)$$

The reward function $R_C(s, a) = R(s[C], a)$ and transition function $T_C(s' | s, a) = T(s'[C] | s[C], a)$ incorporate the interactions between objects in C (and nothing else). This means the value functions and policy (i.e., V_C , Q_C , and π_C) also incorporate information about interactions.

Previous work on value-guided construal focused on simple discrete domains to ensure tight experimental control for human studies. Here, we extend this work in three distinct ways: (1) we incorporate heuristic biases in the construal selection policy, in Section 3.3; (2) we articulate the *attention-aware inverse planning* problem, in which the goal is to infer attentional biases from behavior, in Section 3.4; and (3) we validate the approach in a tabular domain as well as in a driving simulator with continuous states, in Section 4.

¹This formulation of a construed policy in terms of masking states passed to a general optimal policy is roughly equivalent to previous parameterizations (e.g., Ho et al. [2022]), but allows us to use a single pre-trained policy network instead of explicitly computing separate construed policies, as described in Section 4.3.

3.3 Modeling biased construal formation

The behavioral utility of a construal formalizes top-down, goal-directed influences on attention, but other factors can influence what construals a person will form. For example, some objects are highly salient due to low-level perceptual features, such as an orange traffic cone or red sports car. Other objects are more or less salient because of heuristics that have been learned from previous experiences in similar tasks. For instance, someone who has extensive experience driving in North Dakota might learn that, by default, one should attend to ice on the road since that has been a frequent type of entity with important consequences for driving. In contrast, someone who primarily drives in southern California may have a weaker bias for attending to ice or even a bias for ignoring ice.

We can extend the original value-guided construal framework to incorporate heuristic strategies by adding a *parameterized bias function*, $H_\lambda(s, C)$ that weights a construal C in a state s and is parameterized by λ . A person's *construal selection policy* at a state s is then a softmax distribution over the value of representation and the bias [Ho et al., 2023]:

$$\pi_\lambda(C | s) \propto e^{\text{VOR}(s, C) + H_\lambda(s, C)} \quad (4)$$

At the beginning of an episode, an attention-limited decision-maker samples a construal according to $\pi_\lambda(C | s)$ and then executes its corresponding policy, π_C , in the environment. Note that here, we do not model dynamic construals or switching construals within an episode, but we return to these factors as important directions for future work in the discussion.

3.4 Inferring heuristic biases from behavior

Having specified attention-limited decision making as a form of biased value-guided construal, we can now turn to our main focus: Can we estimate attentional biases from behavior? Of course, in principle, given enough data and an accurate model of a data-generating process, one can estimate the parameters of any stochastic process up to a noise factor. However, it is unclear what this looks like in practice in the context of attention-aware inverse planning. Under what conditions are heuristics going to be identifiable? How much behavioral data is sufficient to obtain accurate estimates? Our goal is to provide preliminary answers to these questions.

Formally, estimating heuristics from behavior can be framed as a probabilistic inference problem. Given a state-action trajectory $\zeta = \langle (s_1, a_1), \dots, (s_T, a_T) \rangle$, *maximum-likelihood attention-aware inverse planning* involves finding the set of heuristic bias parameters λ^* that maximize the probability of the observed behavior:

$$\lambda^* = \arg \max_{\lambda} P(\zeta | \lambda) \quad (5)$$

$$= \arg \max_{\lambda} \sum_C P(\zeta | C) \pi_\lambda(C | s_1) \quad (6)$$

$$= \arg \max_{\lambda} \sum_C \left[\prod_t \pi_C(a_t | s_t) \right] \pi_\lambda(C | s_1) \quad (7)$$

The maximand in Equation 5 can be computed if the space of construals is enumerable and we can compute π_C . For our experiments in a tabular domain, we calculate these quantities exactly using dynamic programming in JAX [Bradbury et al., 2018] and then solve for λ^* using off-the-shelf optimizers [Virtanen et al., 2020]. For experiments involving real-world driving scenarios, we approximate them by using a pre-trained policy network to amortize the computational overhead of planning and masking states to only include subsets of objects.

4 Experiments and Results

We conduct three experiments to validate the attention-aware inverse planning approach to behavior modeling. In Section 4.1 we show how it can be used to infer heuristics and attentional biases from behavior in a tabular setting. In Section 4.2, we provide evidence that, unlike our proposed approach, standard approaches like inverse reinforcement learning [Abbeel and Ng, 2004] are not guaranteed to recover attention-limited decision-making. Finally, in Section 4.3 we describe

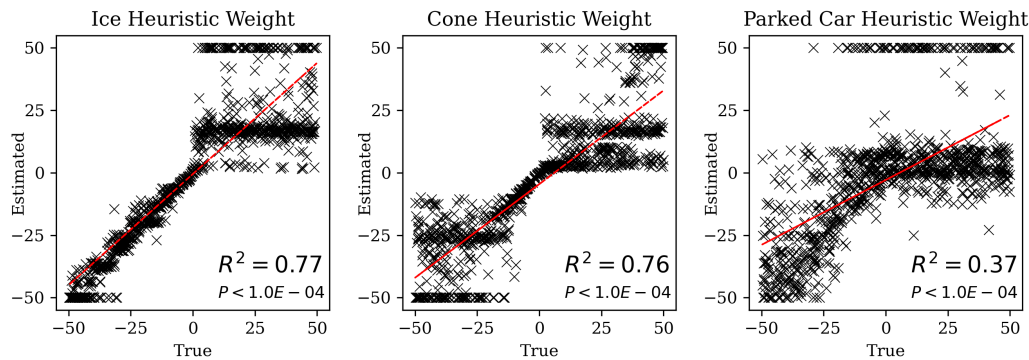


Figure 2: Results of joint maximum-likelihood attention-aware inverse planning: Actual weights of heuristic biases plotted against model estimates, in DrivingWorld (tabular MDP). We simulated the behavior of 1000 agents (with different combinations of weights for three heuristics) by sampling 125 trajectories from 25 different DrivingWorld scenarios (5 from each scenario), for each agent. Then, we attempted to learn the weights jointly from behavior via maximum-likelihood estimation. Each sample represents the true weight (x-axis) of a heuristic for an agent and its estimated value (y-axis) derived from the agent’s behavior. Correspondence between true and estimated weights demonstrates that heuristics that are less consequential (e.g., a bias to ignore cones) are more readily identifiable from behavior (indicated by higher R^2 values). In contrast, biases that could be more consequential (e.g., a bias to ignore parked cars) are less identifiable because they are outweighed by the effect of value-guided construal. These results illustrate the viability and challenges of attention-aware inverse planning in a tabular setting.

an approach to attention-aware inverse planning in a high-fidelity driving simulator using real-world scenarios. Code for replicating the experiments discussed in this section is available at: <https://github.com/sounakban/gpudrive-CoDec/tree/NeurIPS-2025>.

4.1 DrivingWorld

To study how attentional biases affect planned behaviors in a simplified setting, we implemented the tabular DrivingWorld domain shown in Figure 1. The agent controls the blue car and can take the following actions on each timestep: move up 1 cell (−1 reward), move up 2 cells (−1 reward), move diagonally left (−2 reward), and move diagonally right (−2 reward). A scenario consists of parked cars, ice patches, traffic cones, and a goal state. The episode terminates if the agent reaches the goal (+100 reward), hits a parked car (−100 reward), hits a wall (−100 reward) or leaves the grid (0 reward). Driving into a traffic cone is penalized (−10 reward). Finally, there is no additional reward for driving on ice, but with probability 0.4 the car slips either to the left or right and may inadvertently hit another object.

To apply our model of attention-limited decision-making to a DrivingWorld scenario, we define the bias function H_λ using a fixed set of construal feature functions, $\{\phi_i\}$, that map construals and states to real values and parameterize it with linear weights $\{\lambda_i\}$, such that $H_\lambda(s, C) = \sum_i \lambda_i \phi_i(s, C)$. For example, a bias to attend to or ignore ice patches is captured with a weight $\lambda_{\text{Ice}} \in \mathbb{R}$ and feature function $\phi_{\text{Ice}}(s, C)$ that returns the number of ice patches in C that are visible in state s . We assume corresponding weights and feature functions for traffic cones ($\lambda_{\text{Cone}}, \phi_{\text{Cone}}$) and parked cars ($\lambda_{\text{Parked}}, \phi_{\text{Parked}}$). To illustrate the effect of different biases, Figure 1 compares the optimal, attention-unlimited behavior in the task to the behavior of attention-limited decision-makers with a moderate bias to ignore ice ($\lambda_{\text{Ice}} = -10$) and a strong bias to ignore ice ($\lambda_{\text{Ice}} = -100$).

Inferring heuristic biases. To validate our approach for attention-aware inverse planning in DrivingWorld, we apply the procedure to a synthetic dataset generated by first sampling 1000 values of λ_{Ice} , λ_{Cone} , and λ_{Parked} uniformly sampled between −50 and 50, and then sampling 125 construals and trajectories across the 25 scenarios (see Appendix A). We then computed the joint maximum likelihood heuristic weights, $(\lambda_{\text{Ice}}^*, \lambda_{\text{Cone}}^*, \lambda_{\text{Parked}}^*)$. A comparison of the true and estimated values (Figure 2) shows that some parameters can be robustly recovered across a range of values (e.g., λ_{Ice} and λ_{Cone}) whereas others are more difficult to recover accurately (e.g., λ_{Parked}).

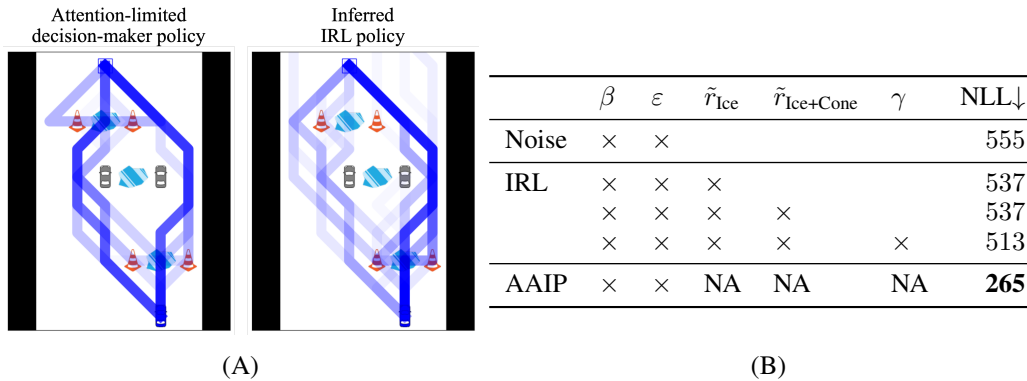


Figure 3: Inverse reinforcement learning (IRL) is not guaranteed to recover attention-limited decision-making. (A) We simulated a ground-truth attention-limited decision-maker with a bias to ignore ice ($\lambda_{\text{Ice}} = -10$) and generated 100 trajectories in the scenario above. This data was passed to an IRL procedure in which we fit the action inverse temperature (β), random action selection (ε), auxiliary reward for driving on ice ($\tilde{r}_{\text{Ice}}$), auxiliary reward for driving on ice between cones ($\tilde{r}_{\text{Ice+Cone}}$), and discount rate (γ). We simulated 100 new trajectories in the same scenario with a policy using the fit parameters. Note that compared to the original policy, the resulting IRL policy is noisier and visits different ice patches. (B) As a quantitative evaluation, we compared fitted negative log-likelihoods (NLL) for a policy given optimal action values with fitted decision-noise parameters β and ε (Noise), three increasingly expressive versions of our IRL procedure, and attention-aware inverse planning (AAIP). Noise and AAIP are intended to serve as a ceiling and floor on negative log-likelihoods, respectively. Even when β , ε , $\tilde{r}_{\text{Ice}}$, $\tilde{r}_{\text{Ice+Cone}}$, and γ are fit, IRL has limited ability to capture the data. In the table, $\times$ indicates that a parameter was fit, otherwise it was set to its true value.

4.2 Comparison with inverse reinforcement learning

Attention-aware inverse planning generalizes models of inverse decision-making, but a natural question to ask is whether existing approaches can capture regularities in behavioral data even if they mis-specify the data-generating process (e.g., assume people are not attention-limited when in fact, they are). Inverse reinforcement learning (IRL) typically assumes that an expert acts based on an accurate and complete representation of the task dynamics but optimizes an unknown reward function [Abbeel and Ng, 2004]. IRL aims to infer this unknown reward function. Crucially, even though IRL applied to human-like behavior (incorrectly) assumes a decision-maker has unlimited attention, a sensible reward function may always exist that sufficiently captures human behavior.

To demonstrate that this is not always the case, we identified a simple DrivingWorld scenario (Figure 3A) in which inferring rewards is not sufficient to capture the behavior of an attention-limited policy. We first generated 100 trajectories from an attention-limited policy with $\lambda_{\text{Ice}} = -10$, $\lambda_{\text{Cone}} = 10$, and $\lambda_{\text{Parked}} = 0$. To simplify the IRL problem, we assumed that the true reward function is known (hitting cones and parked cars is penalized, reaching the goal is incentivized) but that there is an unknown auxiliary reward added to the true reward. This auxiliary reward added a bonus $\tilde{r}_{\text{Ice}} \in \mathbb{R}$ for visiting any ice patch and $\tilde{r}_{\text{Ice+Cone}} \in \mathbb{R}$ for visiting an ice patch in between cones. We then calculated a combination of the auxiliary rewards parameters, softmax inverse temperature, ε random choice, and discount rate that maximizes the log-likelihood of the generated data.

As shown in Figure 3A, the IRL policy that best captures the data misses out on several key properties of the original policy. First, it is much noisier and sometimes even fails to reach the goal altogether. Second, whereas the original policy visits the two ice patches between the cones in roughly equal proportion, the IRL policy only visits the first patch and largely avoids the second. We further confirmed that the IRL policy substantially deviates from the original policy with a quantitative comparison of the negative log-likelihoods (Figure 3B).

4.3 GPUDrive: A high fidelity driving simulator with real-world data

Next, we turn to methods for scaling up to more complex, continuous domains. Specifically, we test an approach to computing the quantities required for attention-aware inverse planning that involves

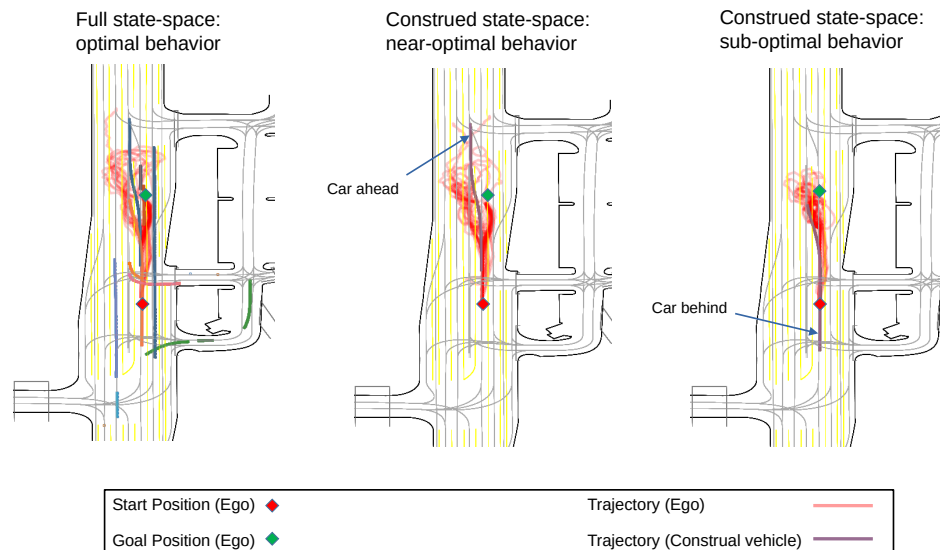


Figure 4: This demonstrates the driving behavior of a PPO agent controlling the ego vehicle in a Wymo highway scenario under three conditions: (i) when it can see every vehicle in the scene (left; optimal); (ii) when it can see a single vehicle ahead, that it needs to pass before it reaches the goal (middle; near-optimal); and (iii) when it can see a single vehicle at its rear that has no impact on the agent’s plan (right; sub-optimal). The trajectories, in translucent red, represent the behavior of the ego vehicle controlled by the PPO agent across 40 trials in each condition. Solid lines of all other colors represent the trajectories of other vehicles in the scene. The purple solid lines in the middle and right plots show the trajectories of the single vehicles being observed in each of the two conditions. The behavior of our PPO agent (distribution of ego trajectories) in the optimal (left) and near-optimal (middle) conditions are similar, where the agent goes around the vehicle ahead. In the sub-optimal condition (right), the agent attempts to drive straight towards the goal position, resulting in a crash.

passing masked states, $s[C]$, to a pre-trained *generalist policy* representing π^* in order to obtain and evaluate construed policies π_C . This approach effectively amortizes the cost of computing construed policies, enabling us to scale up to more complex and naturalistic tasks, such as driving.

For our experiments, we use the GPUDrive driving simulator [Kazemkhani et al., 2025] to generate synthetic trajectory data in scenes obtained from the Waymo Open Motion dataset [Ettinger et al., 2021]. Vehicle trajectories were first generated using a biased generative model parameterized by heuristic bias weights $\{\lambda_H\}$. We then showed that it is possible to retrieve the true λ_H , by performing inference over the synthetic trajectory data, as described in equation 5. We describe the synthetic data generation and inference processes in greater detail in Sections 4.3.1 and 4.3.2, respectively.

4.3.1 Synthetic data generation

Ten scenarios were manually selected (illustrations in Appendix B) from the Waymo Open Motion dataset, where the movement trajectory of the agent-controlled (ego) vehicle presented high degrees of interaction with other vehicles in the scene. As a result of our selection process, a large portion of our simulation dataset contains scenarios where the ego vehicle yields to other vehicles, merges into highways, or changes lanes on multi-lane roads. These are all cases where failing to attend to the relevant objects can lead to catastrophic outcomes (crashes).

Behavioral Utility. For each scenario, we computed the behavioral utility, $\hat{V}(s, \pi_C)$, of all possible single-vehicle construed (any one vehicle is observed in each construed) using Monte-Carlo estimation. The behavioral utility of a construed was computed by taking the average reward earned by a construed policy—the generalist policy that received a masked (i.e., construed) state that only included a subset of the vehicles from the true state—across N rollouts in the true state (algorithm pseudocode available in Appendix D). For example, if there were 15 vehicles in a scene, then 15 different construed models

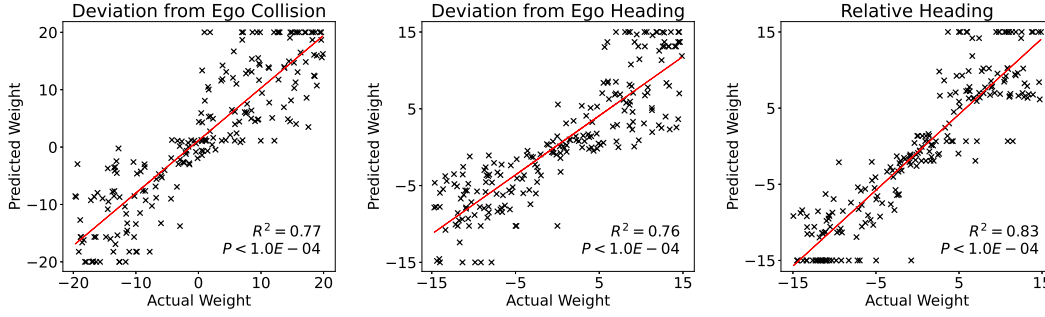


Figure 5: Attention-aware inverse planning results with Waymo Open dataset scenarios in GPU-Drive [Ettinger et al., 2021, Kazemkhani et al., 2025]. We simulated 215 agents with different heuristic weights by sampling 80 trajectories from 10 scenarios per agent. Maximum-likelihood weights were calculated using Bayesian Optimization [Nogueira, 2014]. Each points represents the true weight of a heuristic for an agent and its estimated value derived from the agent’s behavior. Correspondence between true/estimated weights demonstrates that this approach can recover underlying agent biases in complex domains such as real-world driving scenarios.

were each rolled out N times, and the behavioral utility of each construal was computed as the average reward of the construed model across these rollouts:

$$\hat{V}(s, \pi) = \frac{1}{N} \sum_{i=1}^N \sum_{t=0}^{T(i)} R(s_t^{(i)}, a_t^{(i)}) \quad (8)$$

The value of representation was then obtained for each construal based on Equation 2.

Heuristic Biases We computed a construal selection policy, $\pi_\lambda(C \mid s)$, that incorporated three heuristics encoding information about the relative motion of objects, which are features that have been shown to be associated with hazard perception in driving [Lacherez et al., 2014]. The biases were: (i) Deviation from Ego Heading (DfEH), (ii) Relative Heading (RH), and (iii) Deviation from Ego Collision (DfEC) (also see Appendix C). The heuristics were computed at the beginning of each episode, similar to the procedure described in Section 3.3. To perform inference over the heuristic weights of agents for the 10 scenes, we needed ground-truth data containing state information and corresponding agent actions across entire trajectories. However, the Waymo dataset only provides state representations. So, we used a deep-RL PPO model [Cornelisse et al., 2025] pre-trained on GPU-Drive as a generalist model to generate sample state-action trajectories $\{\zeta\}$ in the simulator. All other vehicles in the scene followed the logged (human) trajectories from the dataset.

Generative Model To generate a trajectory dataset that represents the behavior of an agent with a specific set of parameter values for the three biases $\{\lambda_{\text{DfEH}}, \lambda_{\text{RH}}, \lambda_{\text{DfEC}}\}$, a biased construal selection policy for the specific set of λ -values was first used to sample eight construals (with replacement) from each of the 10 scenes. Finally, state-action trajectories were generated by rolling out our generalist policy on construed (masked) state-spaces, based on the 8 sampled construals for each of the 10 scenes. This 80-trajectory dataset then served as the synthetic ‘ground-truth’ behavioral data. 80 trajectories equate to 12 minutes of driving data, which was found to be the minimum amount of data necessary for our algorithm to converge (more details in Appendix G). All simulations were run on an Ubuntu High Performance Computing cluster. A single RTX8000 GPU and 40GB of memory were allocated for the process.

4.3.2 Results: inferring heuristic parameters

To validate our approach, we performed a procedure similar to that described in Section 4.1: We uniformly sampled 215 sets of λ -values for the three heuristics, within a reasonable range of values for each heuristic. For each set, we then sampled 80 trajectories using the generative model described above. Finally, we performed maximum likelihood estimation using Bayesian optimization [Nogueira,

2014] to recover the true λ s (algorithm available in Appendix E). The results of our inference process are presented in Figure 5, which demonstrate that our algorithm for attention-aware inverse planning can reliably recover underlying heuristic biases of simulated agents in real-world scenarios. Details about program execution latency for the current experiment can be found in Appendix F.

5 Limitations and future work

We study attention-aware inverse planning and show how it can scale to continuous, naturalistic domains such as driving, but there are limitations to the current work. First, we constrain the decision-maker to a single fixed construal, chosen at the beginning of an episode, rather than modeling dynamic changes in construal. This can be addressed in future studies by incorporating dynamic updating of task construals in response to task-related events, such as the sudden appearance of new objects. Second, we focus exclusively on how construal leads to systematic deviations from idealized rationality, but people's decisions could be shaped by a variety of other cognitive biases [Lieder and Griffiths, 2020, Schulze et al., 2025] or perceptual mechanisms [Yang et al., 2020]. Future work will need to incorporate these factors. Third, the current work assumes a pre-specified set of possible attentional heuristics, which can be useful for interpretability and data efficiency. However, in situations where interpretability is not a priority or domain knowledge is limited, but behavioral data is abundant, it will be important to extend our approach to learning parameterized heuristics.

Additionally, our work could be extended to incorporate joint inference about rewards and attentional biases since here we assume a known reward model. Previous work in cognitive science has explored how people make these joint inferences [Rane et al., 2023], and theoretical work on “robust IRL” has shown that mis-specification of an expert's dynamics model can lead to systematic error between the value between policies estimated by IRL and the expert policy [Xu and Liu, 2024]. Indeed, the results of section 4.2 suggest that it is not difficult to find scenarios where this error is meaningful and that the recovery gap may be significant in practice. Finally, we apply attention-aware inverse planning in real-world scenarios taken from the Waymo Open Dataset but estimate heuristics from synthetic data. While this approach provides insight into the feasibility of estimating heuristics in complex tasks, validation of our algorithm on human behavior in more tasks is an important next step.

6 Conclusion

This paper aims to bridge recent insights on resource-limited decision-making from cognitive science [Griffiths et al., 2015, Gershman et al., 2015, Lieder and Griffiths, 2020] with scalable human behavior modeling [Lefèvre et al., 2014, Rudenko et al., 2020, Abbeel and Ng, 2004, Ziebart et al., 2008]. Specifically, we extend the framework of *value-guided construal* [Ho et al., 2022, 2023], a formal account of how people allocate limited attention during planning, and articulate the *attention-aware inverse planning* problem, in which the goal is to estimate, from a person's actions, what attentional biases influence their behavior. We conducted two sets of experiments to validate this approach to behavior modeling. Our first set of experiments demonstrated the feasibility of estimating cognitive biases from data in a tabular setting and show that standard inverse reinforcement learning can sometimes fail to capture the behavior of attention-limited decision-makers. Our second set of experiments further validate this approach in a high-fidelity driving simulator [Kazemkhani et al., 2025] using real-world scenarios taken from the Waymo Open Dataset [Ettinger et al., 2021].

Aside from demonstrating the feasibility of scaling up attention-aware inverse planning to complex domains such as driving, the current work shows how principled modeling approaches from computational cognitive science [Griffiths et al., 2024] can integrate effectively with modern machine learning methods (e.g., deep RL). In doing so, it contributes both to the development of safe, interpretable, and trustworthy systems that interact with people [Ho and Griffiths, 2022] as well as to the creation of generalizable cognitive models and theories that can capture the full range of natural human behavior [Wise et al., 2024, Carvalho and Lampinen, 2025].

Acknowledgements

This work was supported by NSF Award #2348442 and the TRI University Research Program.

References

- Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the 21st International Conference on Machine Learning*, ICML '04, page 1, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138385. doi: 10.1145/1015330.1015430.
- Christer Ahlström, George Georgoulas, and Katja Kircher. Towards a context-dependent multi-buffer driver distraction detection algorithm. *IEEE Transactions on Intelligent Transportation Systems*, 23(5):4778–4790, 2022. doi: 10.1109/TITS.2021.3060168.
- Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, August 2021. ISSN 0004-3702. doi: 10.1016/j.artint.2021.103500.
- Michael Bain and Claude Sammut. A framework for behavioural cloning. In *Machine intelligence 15*, pages 103–129, 1995.
- Chris L. Baker, Rebecca Saxe, and Joshua B. Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3):329–349, December 2009. ISSN 0010-0277. doi: 10.1016/j.cognition.2009.07.005.
- Rahul Bhui, Lucy Lai, and Samuel J Gershman. Resource-rational decision making. *Current Opinion in Behavioral Sciences*, 41:15–21, 2021. ISSN 2352-1546. doi: 10.1016/j.cobeha.2021.02.015.
- James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL <http://github.com/jax-ml/jax>.
- Zhangjie Cao, Erdem Biyik, Guy Rosman, and Dorsa Sadigh. Leveraging smooth attention prior for multi-agent trajectory prediction. In *International Conference on Robotics and Automation*, pages 10723–10730. IEEE, 2022.
- Micah Carroll, Rohin Shah, Mark K. Ho, Thomas L. Griffiths, Sanjit A. Seshia, Pieter Abbeel, and Anca Dragan. *On the utility of learning about humans for human-AI coordination*. Curran Associates Inc., Red Hook, NY, USA, 2019.
- Wilka Carvalho and Andrew Lampinen. Naturalistic computational cognitive science: Towards generalizable models and theories that capture the full range of natural behavior, 2025. URL <https://arxiv.org/abs/2502.20349>.
- Katherine M. Collins, Ilia Sucholutsky, Umang Bhatt, Kartik Chandra, Lionel Wong, Mina Lee, Cedegao E. Zhang, Tan Zhi-Xuan, Mark Ho, Vikash Mansinghka, Adrian Weller, Joshua B. Tenenbaum, and Thomas L. Griffiths. Building machines that learn and think with people. *Nature Human Behaviour*, 8:1851–1863, October 2024. ISSN 2397-3374. doi: 10.1038/s41562-024-01991-9.
- Daphne Cornelisse, Aarav Pandya, Kevin Joseph, Joseph Suárez, and Eugene Vinitsky. Building reliable sim driving agents by scaling self-play, 2025. URL <https://arxiv.org/abs/2502.14706>.
- Jason da Silva Castanheira, Nicholas Shea, and Stephen M. Fleming. How attention simplifies mental representations for planning, 2025. URL <https://arxiv.org/abs/2506.09520>.
- Carlos Diuk, Andre Cohen, and Michael L. Littman. An object-oriented representation for efficient reinforcement learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 240–247, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390187.
- Anca D Dragan, Kenton CT Lee, and Siddhartha S Srinivasa. Legibility and predictability of robot motion. In *International Conference on Human-Robot Interaction*, pages 301–308. IEEE, 2013.

- Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R. Qi, Yin Zhou, Zoey Yang, Aur'elien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, October 2021.
- Samuel J. Gershman, Eric J. Horvitz, and Joshua B. Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245):273–278, 2015. doi: 10.1126/science.aac6076.
- Gerd Gigerenzer, Peter M Todd, the ABC Research Group, et al. *Simple heuristics that make us smart*. Oxford University Press, 2000.
- Deepak Gopinath, Mahdieh Nejati Javaremi, and Brenna Argall. Customized handling of unintended interface operation in assistive robots. In *IEEE International Conference on Robotics and Automation*, pages 10406–10412. IEEE, 2021a.
- Deepak Gopinath, Guy Rosman, Simon Stent, Katsuya Terahata, Luke Fletcher, Brenna Argall, and John Leonard. MAAD: A model and dataset for "attended awareness" in driving. In *IEEE/CVF International Conference on Computer Vision*, pages 3426–3436, 2021b.
- Thomas L. Griffiths, Falk Lieder, and Noah D. Goodman. Rational use of cognitive resources: Levels of analysis between the computational and the algorithmic. *Topics in Cognitive Science*, 7(2): 217–229, April 2015. ISSN 1756-8757. doi: 10.1111/tops.12142.
- Thomas L Griffiths, Nick Chater, and Joshua B Tenenbaum. *Bayesian models of cognition: Reverse engineering the mind*. MIT Press, 2024.
- Dylan Hadfield-Menell, Stuart J Russell, Pieter Abbeel, and Anca Dragan. Cooperative inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 29, 2016.
- Mark K. Ho and Thomas L. Griffiths. Cognitive science as a source of forward and inverse models of human decisions for robotics and control. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(1):33–53, May 2022. ISSN 2573-5144. doi: 10.1146/annurev-control-042920-015547.
- Mark K. Ho, David Abel, Carlos G. Correa, Michael L. Littman, Jonathan D. Cohen, and Thomas L. Griffiths. People construct simplified mental representations to plan. *Nature*, 606(7912):129–136, May 2022. ISSN 1476-4687. doi: 10.1038/s41586-022-04743-9.
- Mark K. Ho, Jonathan D. Cohen, and Thomas L. Griffiths. Rational simplification and rigidity in human planning. *Psychological Science*, 34(11):1281–1292, October 2023. ISSN 1467-9280. doi: 10.1177/09567976231200547.
- Max Jaderberg, Wojciech M. Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio Garcia Castañeda, Charles Beattie, Neil C. Rabinowitz, Ari S. Morcos, Avraham Ruderman, Nicolas Sonnerat, Tim Green, Louise Deason, Joel Z. Leibo, David Silver, Demis Hassabis, Koray Kavukcuoglu, and Thore Graepel. Human-level performance in 3d multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859–865, 2019. doi: 10.1126/science.aau6249.
- Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263, March 1979. ISSN 0012-9682. doi: 10.2307/1914185.
- Saman Kazemkhani, Aarav Pandya, Daphne Cornelisse, Brennan Shacklett, and Eugene Vinitsky. Gpudrive: Data-driven, multi-agent driving simulation at 1 million fps. In *International Conference on Learning Representations (ICLR)*, 2025. URL <https://arxiv.org/abs/2408.01584>.
- Max Kleiman-Weiner, Mark K Ho, Joseph L Austerweil, Michael L Littman, and Joshua B Tenenbaum. Coordinate to cooperate or compete: abstract goals and joint intentions in social interaction. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 38, 2016.
- Iuliia Kotseruba and John K Tsotsos. Attention for vision-based assistive and automated driving: A review of algorithms and datasets. *IEEE Transactions on Intelligent Transportation Systems*, 23(11):19907–19928, 2022.

- Philippe Lacherez, Sandra Au, and Joanne M. Wood. Visual motion perception predicts driving hazard perception ability. *Acta Ophthalmologica*, 92(1):88–93, 2014. doi: <https://doi.org/10.1111/j.1755-3768.2012.02575.x>.
- St  phanie Lef  vre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH Journal*, 1(1), July 2014. ISSN 2197-4225. doi: 10.1186/s40648-014-0001-z.
- Richard L. Lewis, Andrew Howes, and Satinder Singh. Computational rationality: linking mechanism and behavior through bounded utility maximization. *Topics in Cognitive Science*, 6(2):279–311, April 2014. ISSN 1756-8765. doi: 10.1111/tops.12086.
- Chenran Li, Aolin Xu, Enna Sachdeva, Teruhisa Misu, and Behzad Dariush. Optimal driver warning generation in dynamic driving environment. In *IEEE International Conference on Robotics and Automation*, pages 14184–14190. IEEE, 2024.
- Falk Lieder and Thomas L. Griffiths. Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, 43:e1., February 2020. ISSN 1469-1825. doi: 10.1017/S0140525X1900061X.
- Daniel McFadden. Conditional logit analysis of qualitative choice behavior. In Paul Zarembka, editor, *Frontiers in Econometrics*, pages 105–142. Academic press, New York, 1974.
- Andrew Y. Ng and Stuart J. Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the 17th International Conference on Machine Learning, ICML ’00*, page 663–670, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1558607072.
- Fernando Nogueira. Bayesian Optimization: Open source constrained global optimization tool for Python, 2014. URL <https://github.com/bayesian-optimization/BayesianOptimization>.
- Sunayana Rane, Mark Ho, Ilia Sucholutsky, and Thomas L. Griffiths. Concept alignment as a prerequisite for value alignment, 2023. URL <https://arxiv.org/abs/2310.20059>.
- Andrey Rudenko, Luigi Palmieri, Michael Herman, Kris M Kitani, Dariu M Gavrilu, and Kai O Arras. Human motion trajectory prediction: A survey. *The International Journal of Robotics Research*, 39(8):895–935, 2020. doi: 10.1177/0278364920917446.
- Dario D. Salvucci. Modeling driver behavior in a cognitive architecture. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(2):362–380, June 2006. ISSN 1547-8181. doi: 10.1518/001872006777724417.
- Christin Schulze, Ada Aka, Daniel M. Bartels, Stefan F. Bucher, Jake R. Embrey, Todd M. Gureckis, Gerald H  ubl, Mark K. Ho, Ian Krajbich, Alexander K. Moore, Gabriele Oettingen, Joan D.K. Ongchoco, Ryan Oprea, Nicholas Reinholtz, and Ben R. Newell. A timeline of cognitive costs in decision-making. *Trends in Cognitive Sciences*, 29(9):827–839, 2025. ISSN 1364-6613. doi: <https://doi.org/10.1016/j.tics.2025.04.004>.
- Amitai Shenhav, Sebastian Musslick, Falk Lieder, Wouter Kool, Thomas L. Griffiths, Jonathan D. Cohen, and Matthew M. Botvinick. Toward a rational and mechanistic account of mental effort. *Annual Review of Neuroscience*, 40(1):99–124, July 2017. ISSN 1545-4126. doi: 10.1146/annurev-neuro-072116-031526.
- David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dhharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm, 2017. URL <https://arxiv.org/abs/1712.01815>.
- Herbert A. Simon. A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1):99, February 1955. ISSN 0033-5533. doi: 10.2307/1884852.
- Herbert A. Simon. *Bounded Rationality*, page 15–18. Palgrave Macmillan UK, 1990. ISBN 9781349205684. doi: 10.1007/978-1-349-20568-4_5.

- Rashmi Sundareswara, Mike Daily, Mike Howard, Howard Neely, Rajan Bhattacharyya, and Craig Lee. Using a distracted driver's behavior to inform the timing of alerts in a semi-autonomous car. In *2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, pages 199–202. IEEE, 2013.
- Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT Press, 2018.
- Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974. doi: 10.1126/science.185.4157.1124.
- Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.
- John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*, 2nd rev. Princeton University Press, Princeton, NJ, USA, 1947. ISBN 978-1-40082946-0.
- Toby Wise, Kara Emery, and Angela Radulescu. Naturalistic reinforcement learning. *Trends in Cognitive Sciences*, 28(2):144–158, February 2024. ISSN 1364-6613. doi: 10.1016/j.tics.2023.08.016.
- Sheng Xu and Guiliang Liu. Robust inverse constrained reinforcement learning under model misspecification. In *Proceedings of the 41st International Conference on Machine Learning*, ICML. JMLR.org, 2024.
- Ruoyu Xue, Jingyi Xu, Sounak Mondal, Hieu Le, Greg Zelinsky, Minh Hoai, and Dimitris Samaras. Few-shot personalized scanpath prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13497–13507, 2025.
- Zhibo Yang, Lihan Huang, Yupei Chen, Zijun Wei, Seoyoung Ahn, Dimitris Samaras, Gregory Zelinsky, , and Minh Hoai. Predicting goal-directed human attention using inverse reinforcement learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*, AAAI'08, page 1433–1438. AAAI Press, 2008. ISBN 9781577353683.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The limitations are discussed in the 'Limitations and future work' Section of the paper.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Theories and necessary proofs are available in Section 3 of the the manuscript.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: The authors performed experiments in two different environments to validate the proposed theory: a simple gridworld environment, and a high-fidelity driving simulator. The necessary information for reproducing the gridworld experiments are provided in Sections 4.1 and 4.2, and Appendix A, while additional information for the simulator-based experiments are provided in Section 4.3 and Appendix B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Details for the gridworlds experiment only require information about gridworld configurations, which are provided in Appendix A. Information about hyperparameters for the simulator-based experiment is provided in Section 4.3.2, and scenes that were used for the experiment are illustrated in Appendix B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The gridworlds paradigm is a standard paradigm that can be run on almost any modern Personal Computer with relative ease. The resources used to run the simulator have been mentioned at the end of Section 4.3.1 as well as Appendix F. More information about the requirements for the simulator is available in the cited material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The positive impacts of our proposed theory on machine teaching are discussed in the introduction.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: No such risks

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The GPUDrive simulator, a PPO model trained on the simulator, and the Waymo Open Motion dataset were used for the second set of experiments. All three are credited in Section 4.3.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Human data was not collected

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Human data was not collected

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs were not used

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Estimating heuristics in tabular setting

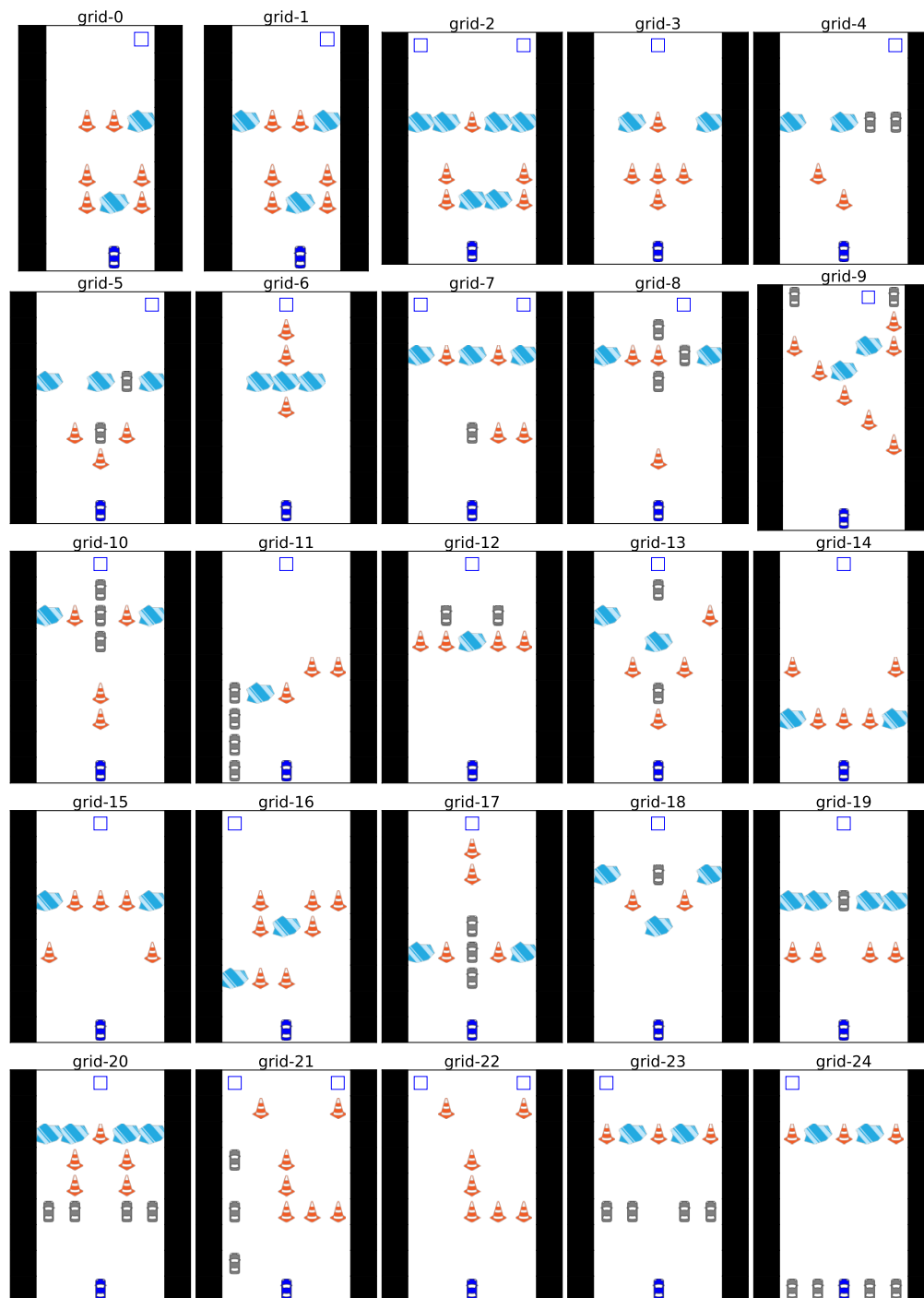


Figure 6: DrivingWorld scenarios used for estimating heuristics in a tabular setting. The true reward function and dynamics are the same as those described in Figure 1.

B Waymo Open Dataset scenes selected for our experiment

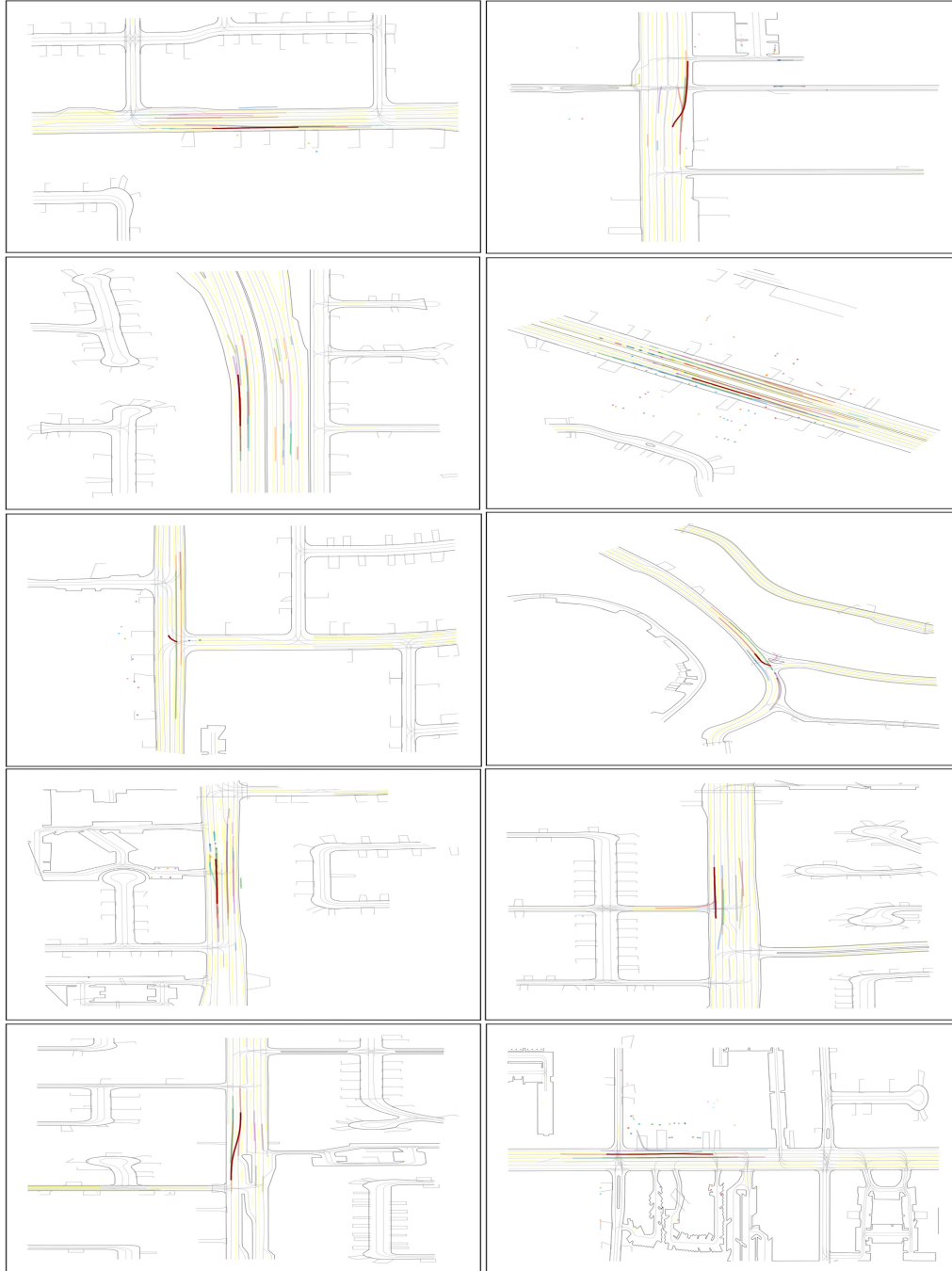


Figure 7: Trajectory illustrations for the 10 scenes chosen from the Waymo dataset. Trajectories marked in dark red represent the movement of the ego vehicle as logged in the original dataset. Other colors represent the movements of other vehicles in the scene.

C GPUDrive Heuristic Biases

C.1 Deviation from Ego Heading bias

The DfEH bias $\{\phi_{DfEH}\}$ was computed by taking the average angle (in radians) between the direction of heading of the ego-vehicle and the positional direction of other vehicles (relative to the ego) in the construal.

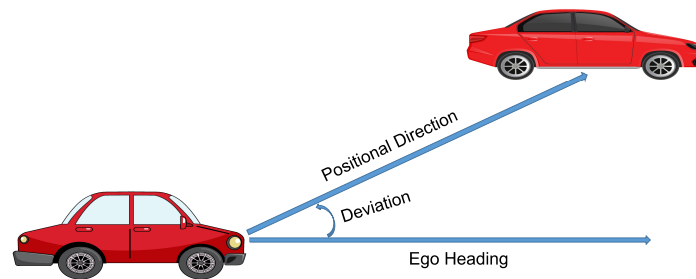


Figure 8: Visual representation for the calculation of Deviation from Ego Heading bias $\{\phi_{DfEH}\}$. NOTE: Car cliparts obtained from freepik

C.2 Relative Heading bias

The RH bias $\{\phi_{RH}\}$ was computed by taking the average angle (in radians) between the direction of heading of the ego-vehicle and all other vehicles in the construal.

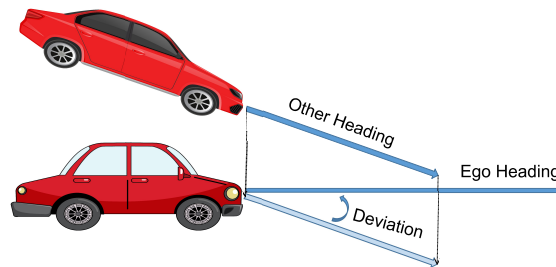


Figure 9: Visual representation for the calculation of Relative Heading bias $\{\phi_{RH}\}$. NOTE: Car cliparts obtained from freepik

C.3 Deviation from Ego Collision bias

The DfEC bias $\{\phi_{DfEC}\}$ was computed by taking the average value of the dot product of the relative velocity and the relative displacement of each vehicle in the construal, relative to the ego:

$$\phi_{DfEC} = \frac{1}{N} \sum_N^{i=1} (V_i - V_e) \cdot (Pos_i - Pos_e) \quad (9)$$

Where, N is the number of vehicles in a construal; V_i and V_e represent the velocities of the current vehicle and the ego vehicle, respectively; and Pos_i and Pos_e represent the positions of the current and ego vehicles, respectively. Values for this bias were normalized to $[-1, 1]$. Where 1 indicates that vehicles in the construal are heading towards the ego vehicle, while a value of -1 indicates that vehicles are moving further away from the ego.

D Algorithm: behavioral utility

The algorithm block summarizes our approach for computing the behavioral utility (VOR) of each construal in each scene. The construed model policy π_C is obtained by masking the state-space of the optimal baseline model.

Algorithm 1 Behavioral Utility Calculation

```

for scene in all_scenes do
   $s \leftarrow S_0(scene)$  ▷ Start state for the scene
  for  $C$  in construals( $s$ ) do ▷ Loop through list of construals given  $s$ 
     $N \leftarrow 40$ 
     $construal\_rewards \leftarrow empty\_list()$ 
     $\pi_C \leftarrow \pi(C)$  ▷ Get construed model policy
    while  $N \neq 0$  do
       $reward \leftarrow simulate(s, \pi_C)$ 
       $construal\_rewards.insert(reward)$ 
       $N \leftarrow N - 1$ 
    end while
     $behavioral\_utilities(scene, C) \leftarrow Average(construal\_rewards)$ 
  end for
end for

```

E Algorithm: λ_H inference

The algorithm block describes our approach for inference of heuristic biases (λ_H). The estimation is performed by maximizing the probability of observing a set of sampled trajectories over values of λ_H . The maximization in our case was performed using a Bayesian optimization technique for GPU Drive, and a gradient-based optimizer for the DrivingWorld paradigm.

Algorithm 2 Inference Logic

```

procedure  $Get\_P_{traj|\lambda}(sampled\_trajs, \lambda_{arg})$ 
   $\lambda_{arg\_list} \leftarrow empty\_list()$ 
  for  $\zeta$  in  $sampled\_trajs$  do
     $\zeta\_list \leftarrow empty\_list()$ 
     $s \leftarrow S_0(\zeta)$  ▷ Start state for trajectory  $\zeta$ 
    for  $C$  in  $construals(s)$  do
       $\pi_C \leftarrow \pi(C)$  ▷ Get construed model policy
       $P_{\zeta|C} \leftarrow \prod_{(s_t, a_t) \in \zeta} \pi_C(a_t | s_t)$ 
       $P_{C|\lambda_{arg}} \leftarrow e^{VOR(s, C) + H_{\lambda_{arg}}(s, C)}$ 
       $\zeta\_list.insert(P_{\zeta|C} * P_{C|\lambda_{arg}})$ 
    end for
     $P_{\zeta|\lambda_{arg}} \leftarrow \sum \zeta\_list$ 
     $\lambda_{arg\_list.insert}(P_{\zeta|\lambda_{arg}})$ 
  end for
   $P_{sample|\lambda_{arg}} \leftarrow \prod \lambda_{arg\_list}$ 
  return  $P_{sample|\lambda_{arg}}$ 
end procedure

 $sampled\_trajs \leftarrow simulate(\pi_{\lambda_{True}})$  ▷ Sample trajectories for agent with  $\lambda_{True}$  heuristics
 $\lambda_{init} \leftarrow sample(\{\lambda_{DFEH}, \lambda_{RH}, \lambda_{DFEC}\})$  ▷ Randomly initialize  $\lambda$ 
 $\lambda_{est} \leftarrow maximize_{\lambda} Get\_P_{traj|\lambda}(sampled\_trajs, \lambda_{init})$  ▷ Estimate  $\lambda$ 

```

F Time Complexity

In this section, we report our algorithm execution time for our experiments with GPU Drive. This is because the data generated by the simulator is a good approximation of real-world high-dimensional complex data.

We had to overcome several computational bottlenecks to perform inference on real-world Waymo data using GPU Drive. The complete pipeline involves: (1) computing an optimal policy, (2) computing behavioral utilities of construals by simulating the optimal policy is construed observation spaces, and (3) performing maximum likelihood estimation over continuous multi-dimensional lambda values.

One of the algorithmic contributions of our paper is a more efficient way to compute the optimal policy, which essentially involves amortizing the cost by pre-training a generalist policy using standard RL methods. This can be done on a standard GPU within a day. GPU Drive includes a pre-trained generalist policy that we used for our experiments.

Computation of behavioral utilities requires simulating a masked policy on the true dynamics. It took an hour to evaluate 166 construals across 10 Waymo scenarios by sampling 40 trajectories per construal, on a Ubuntu High Performance Computing cluster with a single RTX8000 GPU and 40GB of RAM allocated to the process. We were only able to partially parallelize the evaluation process due to limitations in the current implementation of the Python interface, but we note that evaluation can be significantly sped up by optimally parallelizing the process.

Finally, we performed inference over heuristic parameters on a laptop computer (with a Intel Core Ultra 7 165H CPU and 32GB of RAM). It took about 3 min and 30 sec to perform inference over eighty 9-second trajectories (roughly 12 minutes of driving data) for a single set of parameters. Our

current implementation of the inference algorithm is a sequential CPU-based process, which can be parallelized and executed on GPUs.

G Sample Efficiency

To test the sample efficiency of our approach in GPU Drive, we evaluated the performance of our algorithm on a range of dataset sizes (expressed as durations of observed real-world driving data), from ten 9-second trajectories (the equivalent of 1.5 minutes) to five hundred 9-second trajectories (75 minutes). Evaluation of performance for different durations was measured over 30 randomly sampled lambda values. Our findings from this experiment are reported in Figure 10.

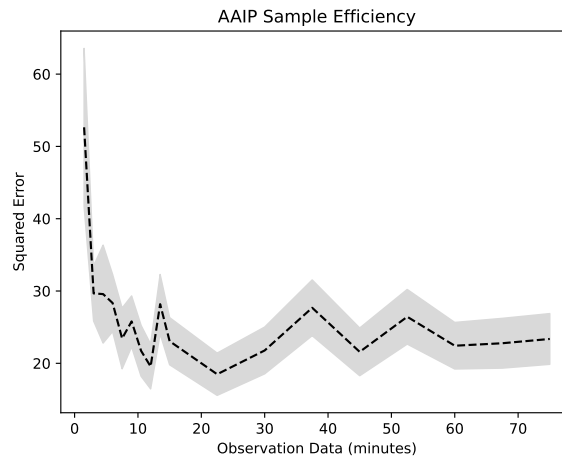


Figure 10: Plot for the efficiency of the AAIP algorithm in estimating true values of lambda with different sample sizes of observation data. The x-axis represents the amount of observation data (in minutes) used for the inference process, while the y-axis represents the mean (and standard error) squared error between true and estimated lambdas. The trend suggests that the algorithm converges at around 12 minutes of observed driving (eighty trajectories).