Article:
Wei, R., Garcia, A., McDonald, A. et al. (3 more authors) (2025) Learning an Active Inference Model of Driver Perception and Control: Application to Vehicle Car-Following. IEEE Transactions on Intelligent Transportation Systems, 26 (7). 9475 -9490. ISSN 1524-9050

https://doi.org/10.1109/tits.2025.3574552

Learning An Active Inference Model of Driver Perception and Control: Application to Vehicle Car-Following

Ran Wei, Alfredo Garcia, Anthony McDonald, Gustav Markkula, Johan Engstrom, Matthew O'Kelly

***Abstract*— In this paper we introduce a general estimation methodology for learning a model of human perception and control in a sensorimotor control task based upon a finite set of demonstrations. The model's structure consists of *(i)* the agent's internal representation of how the environment and associated observations evolve as a result of control actions and *(ii)* the agent's preferences over observable outcomes. We consider a model's structure specification consistent with *active inference*, a theory of human perception and behavior from cognitive science. According to active inference, the agent acts upon the world so as to minimize *surprise* defined as a measure of the extent to which an agent's current sensory observations differ from its preferred sensory observations. We propose a bi-level optimization approach to estimation which relies on a structural assumption on prior distributions that parameterize the statistical accuracy of the human agent's model of the environment. To illustrate the proposed methodology, we present the estimation of a model for car-following behavior based upon a naturalistic dataset. Overall, the results indicate that learning active inference models of human perception and control from data is a promising alternative to black-box models of driving.**

***Index Terms*— Human perception and action, Partially Observable Markov Decision Process, Active inference, inverse reinforcement learning.**

I. Introduction

In many control tasks requiring mind and motor resources by a human agent, the observation space can be high-dimensional and complex. Empirical evidence indicates that humans agents use a simpler, lower dimensional representation of the environment in sensorimotor control tasks [1], [2]. The Bayesian brain hypothesis [3], [4] posits that the human brain uses the information provided by sensory data to update a representation of the world in the form of a conditional probability distribution. To account for the structure of human perception and control in a task involving motor and mind resources, a model must separately describe *(i)* the agent's internal representation of the world as the environment evolves as a result of control actions and *(ii)* the agent's preferences over observable outcomes. Equipped with data in the form of demonstrations (i.e. sequences of recorded observation-action pairs), the learning task is to estimate the agent's preferences as well as its internal representations leading to a behavior policy that best fits data.

In machine learning, this estimation problem is known as *inverse* reinforcement learning (IRL) in an *off-line* setting [5]. In contrast to reinforcement learning (wherein the goal is to identify a control policy based upon reward and state observations), the goal of IRL is to estimate the reward function and transition probabilities from observed trajectories of state-action pairs [6]–[8]. The estimated reward provides an interpretation of the agent's behavior and the reward function can be used to design policies in domains where manual reward specification is difficult, e.g., in autonomous driving [9]. There is a significant literature on identification and estimation of models of human control when the state is observable [10]–[12]. In contrast, model identification and estimation when the state is only partially observable (as in models accounting for human perception) has received less attention. Notable exceptions include [13]–[16]. However, the environments considered in these papers are either low dimensional as in [13], [14], restricted to a linear-quadratic control [15] or customized for a specific control task [16].

In this paper, we introduce a Bayesian estimation methodology for learning a structural model of perception and control in general control tasks in higher dimensional settings. We use the formalism of a Partially Observable Markov Decision Process (POMDP) in which the agent's preferences are modeled by a reward function and the agent's internal representation of the environment consists of observation and transition probabilities. However, a POMDP model of a human agent's perception and control policy based solely on demonstrations is in general *non-identifiable*, i.e. there may be several different combinations of reward and internal model of the environment that rationalize the same demonstrations dataset. This is because in planning control tasks, different combinations of reward and internal dynamics model could result in the same inter-temporal reward trade-offs. To address this issue, we make a structural assumption on prior distributions that parameterize the statistical accuracy of the human agent's model of the environment. Specifically, we assume (*i*) the agent's preferences and model of the environment are independent and (*ii*) the distribution of the model of the environment parameters concentrates on values with higher fit to the data (i.e. higher log-likelihood). In words, this assumption restricts our estimation to agents with reasonably accurate models of the environment whose preferences over states of the world are not determined by their perception of the environment. This allows us to formulate the *Maximum A Posteriori* (MAP) estimator as the solution to a bi-level optimization problem. The upper-level problem is the maximization of the posterior distribution and the lower-level problem is the computation of optimal policy for the given reward *and* model of the

This work was supported in part by Army Research Office ARO under grant W911NF-22-1-0213.

environment. We approximate the solution to this bi-level optimization problem by a stochastic gradient algorithm with a nested policy optimization step.

To illustrate the proposed methodology, we consider an application to highway driving using a naturalistic dataset. We specify the structure of the model in accordance to "active inference" [17]–[19], a novel framework for modeling human perception and behavior in sensorimotor control tasks [20]. Active inference is related to the Bayesian Brain hypothesis that posits prediction as the fundamental task of cognition [21], [22], i.e. the brain minimizes prediction error by updating beliefs about the states of the world consistent with data. Active inference takes a conceptual leap from this view in that minimizing prediction error can also be attained by both updating beliefs and acting upon the world to approximately induce a preferred distribution of the states of the world. The active inference framework is summarized by a principle of *free energy minimization*: forward (action) and backward (belief) updating processes work in tandem to minimize "surprise" with respect to a preferred belief distribution about the states of the world.

Our ultimate goal is to provide a model that is *interpretable* –in terms of a cognitive model of perception and action– and that exhibits a statistical performance that is similar or arguably superior to other black-box models. To this end, we compare the learned structural model of perception and action (with an active inference reward specification) with two baseline models based on Behavior Cloning (BC), a common machine learning approach to driving behavior modeling [23]–[26], and the Intelligent Driver Model (IDM), a class of rule-based models widely used by traffic simulation software [27]–[29]. It is important to emphasize that the naturalistic dataset does not include collisions and only includes relatively few extreme observations (e.g. extremely low relative distance or high relative velocity). Thus, our model only provides an account of driver perception and control behavior in *average* conditions and our testing of model performance focuses on *aggregate* measures. The results indicate that the active inference-based model outperforms those obtained by imitation learning-based models from the machine learning literature. However, the learned model is inaccurate in extreme scenarios that are poorly covered in the dataset and does exhibit higher collision rates than IQM when tested online. This is due to the limitations of the dataset which poorly covers extreme scenarios so that the learned model does poorly in extreme scenarios due to distribution shift [30], [31]. Overall, the results indicate that learning active inference models from data is a promising alternative to black-box models of driving as it provides a way to trace driving behaviors back to a human drivers' perception and preferences.

The structure of this paper is as follows. In section II, we start by describing a Partially Observable Markov Decision Process (POMDP) of perception and control. In section III, we describe the inverse estimation problem, i.e. based upon sequences of observations and implemented actions to estimate the primitives of the POMDP model (reward, partially observable state transition and observation probabilities). In section IV, we describe the specification of reward function based upon active inference, a novel framework for cognition and behavior. In section V, we describe the application of the proposed estimation algorithm to obtain an active inference model for car-following behavior by human drivers. We compare the active inference model with Behavior Cloning (BC) and the Intelligent Driver Model (IDM). Finally, in section VI, we close with concluding remarks about the promise and challenges of learning perception & control models based upon naturalistic datasets.

II. A POMDP Model of Perception and Control

We start by providing a description of a partially Observable Markov Decision Process (POMDP) of perception and control. This encompasses neuroscience modeling frameworks for human perception and action in sensorimotor tasks such as *active inference* [17] and the *expected value of control* (EVC) [32].

In the proposed POMDP framework (see Figure 1), the human agent maintains an internal model of the world (or representation) so that high-dimensional observations $o_t \in O \subset \mathbb{R}^n$ (sensory stimuli) are represented with a lower dimensional *hidden* state $s_t \in S \subset \mathbb{R}^m$ where S is the state space and $m << n$. If the hidden state is s_t and $a_t \in A$ is implemented, the agent accrues a reward $r(s_t, a_t)$. The agent's internal representation includes state dynamics, i.e. a transition to a new state s_{t+1} takes place with probability $\mathbb{T}(s_{t+1}|s_t, a_t)$ and a new observation o_{t+1} is obtained with probability $\mathbb{O}(o_{t+1}|s_{t+1})$.

After $t > 0$ time periods, the observable history of observations and actions is denoted by

$$h_t := \{o_t, ..., o_0, a_{t-1}, ..., a_0\} \in H_t \subset O^{t+1} \times A^t.$$

We consider randomized or stochastic policies π that are adapted to the history of the process, i.e. given history h_t action $a \in A$ is implemented with probability $\pi(a|h_t) \in [0, 1], a \in A$ and $\sum_{a \in A} \pi(a|h_t) = 1$ for all $h_t \in H_t$.

In the proposed POMDP model of perception and action, the human agent aims to maximize the expected value of discounted reward net of information processing costs:

$$U_\tau(h_\tau) \triangleq \sup_{\pi \in \Pi} \mathbb{E}\Big[\sum_{t \geq \tau} \gamma^{t-\tau}[r(s_t, a_t) - c(\pi(\cdot|h_t))]\Big] \quad (1)$$

where $\gamma \in (0, 1)$ is the discount factor, Π is the set of randomized policies that are adapted to the history process and $c(\pi(\cdot|h_t))$ is a per-period *information processing cost*. As human agents may differ in their ability to process task-relevant information or to attend to the task at hand [33]–[37], the cost $c(\pi(\cdot|h_t))$ models the fact that *low* entropy behavioral policies are consistent with *high* information processing effort or attention.

The combination of additive reward structure and Markovian dynamics allows for a recursive characterization of the optimal policy as follows:

$$U_t(h_t) = \max_{\pi(\cdot|h_t)} \Bigg\{ \sum_{a_t} \sum_{s_t} r(s_t, a_t) b_t(s_t) \pi(a_t|h_t) - c(\pi(\cdot|h_t)) + \gamma \sum_{o_{t+1}} \sum_{a_t} \mathbb{P}(o_{t+1}|h_t, a_t) \pi(a_t|h_t) U_{t+1}(h_{t+1}) \Bigg\}. \quad (2)$$

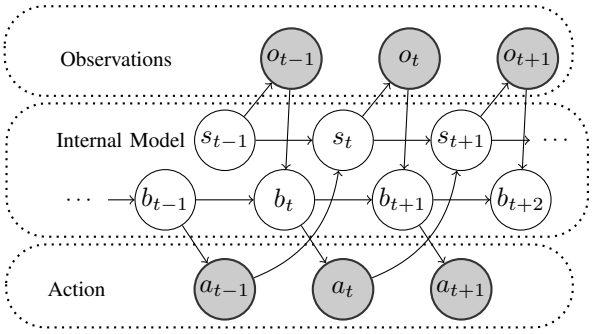


Fig. 1: Graphical Model of Perception and Control.

Let $b_t \in \Delta^S$ denote the Bayes updated belief distribution on the state, i.e. $b_t(s) := \mathbb{P}(s_t = s|h_t)$, where Δ^S is the set of probability distributions on state space S.

Let us denote by $\sigma(o_{t+1}|b_t, a_t)$ the probability of recording observation o_{t+1} when action a_t is implemented and the current belief distribution is b_t, i.e.

$$\sigma(o_{t+1}|b_t, a_t) := \sum_{s_{t+1}} \sum_{s_t} \mathbb{O}(o_{t+1}|s_{t+1}) \mathbb{T}(s_{t+1}|s_t, a_t) b_t(s_t)$$

Using standard POMDP arguments in Proposition 1 below we show that with no loss of optimality, the search for optimal policy can be restricted to *Markovian* policies, say $\Pi^M \subset \Pi$ that only depend b_t, the current Bayes updated belief as opposed to the whole history $h_t \in H_t$.

Proposition 2.1: Let $V_t(b)$ be recursively defined as follows:

$$V_t(b) = \max_{\pi(\cdot|b)} \Big\{ \sum_s \sum_a r(s,a)\pi(a|b)b(s) - c(\pi(\cdot|b)) + \gamma \sum_a \sum_{o'} \sigma(o'|b,a)\pi(a|b) V_{t+1}(b') \Big\}$$

where $b'(s) = \mathbb{P}(s_{t+1} = s|h_t \cup (a, o'))$, i.e. the resulting Bayes update after action a is implemented and observation o' are recorded. Then, the Bayes updated belief $b_t = \mathbb{P}(\cdot|h_t)$ is a sufficient statistic for solving (2), i.e. $U_t(h_t) = V_t(b_t)$ for all h_t.

Proof: See Appendix. ∎

We now state and prove the *soft* Bellman equation for the value function $V_t(b)$ when the information processing cost is proportional to the Kullback-Leibler divergence between the control policy and a default policy π^0, i.e. $c(\pi(\cdot|b_t)) = \alpha \mathcal{D}_{KL}(\pi(\cdot|b_t)||\pi^0(\cdot|b_t))$ where:

$$\mathcal{D}_{KL}(\pi(\cdot|b)||\pi^0(\cdot|b)) := \sum_{a \in A} \pi(a|b) \log \frac{\pi(a|b)}{\pi^0(a|b)}. \quad (3)$$

and $\alpha > 0$.

Let $\mathcal{Q}$ be the Banach space of bounded, measurable functions $Q : \Delta^S \to \mathbb{R}$ under the supremum norm $||.||$. Define the *soft* Bellman operator $\mathcal{B} : \mathcal{Q} \to \mathcal{Q}$ by

$$[\mathcal{B}Q](b,a) := \sum_s r(s,a)b(s) + \gamma \sum_{o'} \sigma(o'|b,a)\alpha \log \Big(\sum_{a'} \pi^0(a'|b') \exp\big(\frac{1}{\alpha} Q(b',a')\big) \Big), \quad (4)$$

where b' is the resulting Bayes update after action a and observation o' are recorded.

Theorem 2.2: (a) $\mathcal{B} : \mathcal{Q} \to \mathcal{Q}$ is a contraction mapping with modulus $\gamma \in (0,1)$ with unique fixed point Q^*, i.e.

$$Q^*(b,a) = \sum_s r(s,a)b(s) + \gamma \sum_{o'} \sigma(o'|b,a)\alpha \log \Big(\sum_{a'} \pi^0(a'|b') \exp\big(\frac{1}{\alpha} Q^*(b',a')\big) \Big),$$

(b)

$$V^*(b) = \max_{\hat{\pi}(\cdot|b)} \Big[\sum_a \hat{\pi}(a|b) Q^*(b,a) - \alpha \mathcal{D}_{KL}\big(\hat{\pi}(\cdot|b)||\pi^0(\cdot|b)\big) \Big] = \alpha \log \sum_a \pi^0(a|b) \exp\big(\frac{1}{\alpha} Q^*(b,a)\big)$$

(c) the optimal policy is of the form:

$$\pi^*(a|b) = \frac{\pi^0(a|b) \exp\big(\frac{1}{\alpha} Q^*(b,a)\big)}{\sum_{a' \in A} \pi^0(a'|b) \exp\big(\frac{1}{\alpha} Q^*(b,a')\big)}. \quad (5)$$

Proof: See Appendix. ∎

Remark 1: Note that as $\alpha \to +\infty$, information processing effort is arbitrarily costly and in the limit, the agent implements the default policy $\pi^* \to \pi^0$. Conversely, as $\alpha \to 0^+$, we recover the optimal solution without information processing cost since $V^*(b) \to \max_{a \in A} Q^*(b,a)$.

Remark 2: In the remainder of the paper we shall use $\alpha = 1$ and the default policy is the uniformly random policy $\pi^0(a|b) = \frac{1}{|A|}$. With these choices the optimal policy takes the form:

$$\pi^*(a|b) = \frac{\exp Q^*(b,a)}{\sum_{a' \in A} \exp Q^*(b,a')}. \quad (6)$$

Remark 3 (Finite Horizon): It can be easily verified that Proposition 1 and Theorem 1 continue to hold for the case in which the controller is solving a finite horizon problem. Evidently, the results in this case require that the state-action function Q_t and the conditional choice probabilities π_t are time-dependent t. Formally, for a planning horizon of length $H > 0$, the optimal policy at time $t \in \{0, 1, \ldots, H\}$ is of the form:

$$\pi^*_{t,H}(a|b) = \frac{\exp Q^*_{t,H}(b,a)}{\sum_{a' \in A} \exp Q^*_{t,H}(b,a')} \quad (7)$$

and

$$Q^*_{t,H}(b,a) = \sum_s r(s,a)b(s) + \sum_{o'} \sigma(o'|b,a) V^*_{t+1,H}(b') \quad (8)$$

where b' is the resulting Bayes update after action a and observation o' are recorded and

$$V^*_{t+1,H}(b') = \log \Big(\sum_{a'} \exp Q^*_{t+1,H}(b',a') \Big) \qquad t \leq H-1 \quad (9)$$

and $V^*_{H+1,H} = 0$.

III. Estimation Methodology

Equipped with a model of perception and action as described in the previous section, we consider the estimation of the primitives based upon demonstrations, that is, sequences of observations and implemented actions of the form $\tau = \{(o_0, a_0), (o_2, a_1), \ldots, (o_T, a_T)\}$. We shall denote by $\mathcal{D}$ the finite dataset of distinct sequences of observation-action pairs.

The primitives of the perception & control model are parametrized as follows:

- *Perception*: We assume the agent's internal representation of hidden state dynamics and observation probabilities is parametrized with $\theta_1 \in \mathbb{R}_1^p$ so that the likelihood of observation o_{t+1} given beliefs b_t and action a_t is $\sigma_{\theta_1}(o_{t+1}|b_t, a_t)$.
- *Preferences*: A reward function $r_{\theta_2}(b, a)$ which is parametrized by $\theta_2 \in \mathbb{R}_2^p$.

Assuming the data is generated by an agent who uses a *receding horizon* plan with horizon H according to (7), the log-likelihood of a sequence $\tau \in \mathcal{D}$ can be written as

$$\mathbb{P}(\tau|\theta) = \prod_{t=0}^{T} \Big(\pi_\theta^*(a_t|b_{\theta_1,t}) \mathbb{P}\big(o_{t+1}|h_t \cup \{a_t\}\big) \Big)$$

where to alleviate notation we write $\pi_\theta^*(\cdot|b)$ to refer to the first-period optimal policy with a planning horizon $H > 0$. $\mathbb{P}(o_{t+1}|h_t \cup \{a_t\})$ is the external observation-generating distribution that is *independent* of the agent's internal representation. Hence, the log-likelihood of dataset $\mathcal{D}$ can be written as:

$$\begin{aligned}
\log \mathbb{P}(\mathcal{D}|\theta) &= \log \prod_{\tau \in \mathcal{D}} \mathbb{P}(\tau|\theta) \\
&= \mathbb{E}_{\tau \sim \mathcal{D}} \Big[\sum_{t=0}^{T} \log \Big(\pi_\theta^*(a_t|b_{\theta_1,t}) \mathbb{P}\big(o_{t+1}|h_t \cup \{a_t\}\big) \Big) \Big] |\mathcal{D}| \qquad (10) \\
&= \mathbb{E}_{\tau \sim \mathcal{D}} \Big[\sum_{t=0}^{T} \log \pi_\theta^*(a_t|b_{\theta_1,t}) \Big] |\mathcal{D}| + \text{constant} \qquad (11)
\end{aligned}$$

where the expectation is taken with respect to the empirical measure $\bar{\mathbb{P}}(\tau) = \frac{1}{|\mathcal{D}|}$ and

$$\pi_\theta^*(a|b) = \frac{\exp Q_\theta^*(b, a)}{\sum_{a' \in A} \exp Q_\theta^*(b, a')} \qquad (12)$$

Condition (12) imposes model in the form of the first period policy of a receding horizon plan.

We take a Bayesian approach to finding an estimator and make an additional assumption on the structure of the prior distribution of parameters denoted by $P(\theta)$:

Assumption 1: (a) $P(\theta) = P(\theta_1)P(\theta_2)$. (b) The distribution of θ_1 is of the form:

$$P(\theta_1) \propto \exp \Big(\lambda \mathbb{E}_{\tau \sim \mathcal{D}} \big[\sum_{t=0}^{T} \log \sigma_{\theta_1}(o_{t+1}|b_{\theta_1,t}, a_t) \big] |\mathcal{D}| \Big) \qquad (13)$$

for some $\lambda > 0$.

Assumption 1(a) restricts our estimation to agents whose preferences (parameterized by θ_2) over states of the world are not determined by their perception of the environment (parameterized by θ_1). Under assumption 1(b) on the prior distribution, parameter values θ_1 with higher fit to the sequences of observations in the data are more likely. Increasing values of λ imply the agent has (a priori) an increasingly accurate model of the environment.

Assuming a uniform prior $P(\theta_2)$ on a compact subset $\Theta_2 \subset \mathbb{R}_2^p$, the log of the posterior distribution can be written as:

$$\begin{aligned}
\log P(\theta|\mathcal{D}) &= \log P(\mathcal{D}|\theta) + \log P(\theta_1) + \text{constant} \\
&= \mathbb{E}_{\mathcal{D}} \Big[\log \sum_{t=0}^{T} \pi_\theta^*(a_t|b_{\theta_1,t}) \Big] |\mathcal{D}| + \\
&\quad \lambda \mathbb{E}_{\mathcal{D}} \Big[\sum_{t=0}^{T} \log \sigma_{\theta_1}(o_{t+1}|b_{\theta_1,t}, a_t) \Big] |\mathcal{D}| + \text{constant}
\end{aligned} \qquad (14)$$

We are ready to formulate the estimation problem as the following bi-level optimization problem:

$$\max_{(\theta_1, \theta_2)} \; \mathbb{E}_{\mathcal{D}} \Big[\log \sum_{t=0}^{T} \pi_\theta^*(a_t|b_{\theta_1,t}) + \lambda \sum_{t=0}^{T} \log \sigma_{\theta_1}(o_{t+1}|b_{\theta_1,t}, a_t) \Big] \qquad (15)$$

$$\text{s.t.} \quad \pi_\theta^* = \arg \max_{\pi \in \Pi^H} \mathbb{E} \Big[\sum_{h \le H} [r_\theta(b_h, a_h) - \log \pi(\cdot|b_h)] \Big]$$

where here again we write $\pi_\theta^*(\cdot|b)$ to refer to the first-period optimal policy with a planning horizon $H > 0$ with initial belief b. The algorithm for approximating a solution, say $\widehat{\theta}$, to (15) is described in Algorithm 1 below. The estimated model structure is summarized as follows:

Structural Model of Perception and Control	
Perception	
Observations	$\mathbb{O}_{\widehat{\theta}_1}(o_t|s_t)$
Transitions	$\mathbb{T}_{\widehat{\theta}_1}(s_{t+1}|s_t, a_t)$
Generative Model	$\sigma_{\widehat{\theta}_1}(o_{t+1}|b_t, a_t)$
Control	
Preferences (reward)	$r_{\widehat{\theta}_2}(s_t, a_t)$
Control Policy	$\pi_{\widehat{\theta}}(a_t|b_t) = \frac{\exp Q_{\widehat{\theta}}^*(b_t, a_t)}{\sum_{a' \in A} \exp Q_{\widehat{\theta}}^*(b_t, a')}$

Algorithm 1 Bayesian MAP Estimation of Perception & Control Model

Require: Dataset $\mathcal{D} = \{\tau\}$, perception model $\sigma_{\theta_1}(o'|b, a)$, preference model $r_{\theta_2}(b, a)$, initial value $\theta_0 = (\theta_{1,0}, \theta_{2,0})$, hyperparameter $\lambda > 0$ and learning rate $\rho > 0$.

1: **for** $k = 0 : K$ **do**
2: Compute the optimal policy $\pi_{\theta_k}^*$ using value-iteration
3: Evaluate the log posterior $\log P(\theta_k|\mathcal{D})$ according to (14).
4: Compute the gradient of $\nabla_\theta \log P(\theta_k|\mathcal{D})$
5: Perform parameter update

$$\theta_{k+1} = \theta_k + \rho \nabla_\theta \log P(\theta_k|\mathcal{D})$$

6: **end for**

IV. An Active Inference Specification

In this section we describe a specification of the reward function consistent with active inference [17]. Active inference is a novel framework for cognition and behavior according to which the agent jointly *perceives* and *acts* upon the world so as to maximize the match between *perceived* vs *preferred* states of the world.

The process of matching the *perceived* vs *preferred* distribution of the states of the world follows a principle of *free energy minimization*: forward (action) and backward (belief) updating processes work in tandem to minimize a measure of "surprise" or free energy. For backward (belief) updating, free energy is minimized when the agent's belief distribution b_t corresponds to the Bayes updated belief distribution on the state s_t. For forward (action) selection processes, surprise is measured with respect to a *preferred* distribution $\tilde{P}(s_{t+1})$ over states of the environment. The immediate "surprise" associated with action a_t when current beliefs are b_t is quantified by the *expected free energy* defined as:

$$EFE(b_t, a_t) = \mathbb{E}\big[D_{KL}\big(b_{t+1}||\tilde{P}\big)\big] + \mathbb{E}\big[\mathcal{H}(\mathbb{O}(\cdot|s_{t+1}))\big] \tag{16}$$

where the expectation is taken with respect to $o_{t+1} \sim \mathbb{O}(\cdot|s_{t+1}), s_{t+1} \sim \sum_s \mathbb{T}(\cdot|s, a_t)b_t(s)$ with

$$b_{t+1}(s) = \mathbb{P}(s_{t+1} = s|h_t \cup \{a_t, o_{t+1}\})$$

and $\mathcal{H}(\mathbb{O}(\cdot|s_{t+1}))$ is the entropy of the resulting generative model of observations, i.e.:

$$\mathcal{H}(\mathbb{O}(\cdot|s_{t+1})) := -\sum_{o'} \mathbb{O}(o'|s_{t+1}) \log\Big(\mathbb{O}(o'|s_{t+1})\Big).$$

The first term in (16) quantifies the extent to which the belief distribution on the states of the world b_{t+1} (resulting from implementing action a_t and recording observation o_{t+1}) differs from the preferred distribution of the states of the world $\tilde{P}(\cdot)$. This term is usually referred to as "risk" because of its relationship to the deviation from an agent's goal [38]. Selecting policies that generate preferred observations minimizes risk. The second term in (16) is a measure of the observation uncertainty induced by action a_t. This term is referred to as "ambiguity" and represents the value of obtaining reliable information that may help to resolve uncertainty about future states [18], [38]. Defining ambiguity hinges on having a model of the world.

In [39] an interpretation of active inference (when the state is observable) is given in terms of Markov decision processes. In a similar manner, by setting the reward function as $r(b_t, a_t) := -EFE(b_t, a_t)$, the active inference model can be seen as a particular instance of the class of POMDP models described in section II [40]. However, the ability to consider trade-offs between the described measures of risk vs. ambiguity presents an advantage of the active inference formulation compared to traditional RL/IRL formulations.

V. Application: Learning a Model of Perception and Control in Car Following Behavior

In this section, we describe the application of Algorithm 1 to estimate an active inference model for car-following behavior by human drivers. [1] Computational models of human performance in such task have been amply studied by traffic engineers and psychologists, see e.g., [41]–[44]. However, our goal here is to *learn* a model that is motivated by cognitive science (active inference) based upon a naturalistic dataset of task demonstrations. In this sense, the closest paper to our work is [16] which assumes the agent's decisions are based upon a *state* estimate (speed, relative speed and distance) and a predictive model of the lead vehicle. In contrast, in the proposed POMDP model, the variables speed, relative speed and distance are *observations* which are used by the agent to form *current* and *future* beliefs about the states of the environment which are discrete.[2]. In addition, the policy in [16] is deterministic and the model is not based on agent's preferences.

We compare the active inference model, referred to as Active Inference Driving Agent (AIDA), with two baseline models: Behavior Cloning (BC), a common machine learning approach to driving behavior modeling [23]–[26], and the Intelligent Driver Model (IDM), a rule-based model used by most traffic simulation software [27]–[29]. We begin by describing the baseline models and the dataset used to estimate the parameters of the models. We then describe the protocols for evaluating the models' ability to replicate human driving behavior in the dataset. Lastly, we present the model evaluation results and demonstrate AIDA's interpretability advantages. Implementation details are provided in Appendix VII-C.

A. Models and parameterization

Behavior Cloning: BC trains neural networks to map observations or a history of observations to control actions in the dataset. The policy parameters, denoted with θ, are estimated using maximum likelihood estimation of the dataset actions:

$$\max_{\theta} \mathcal{L}(\theta) = \mathbb{E}_{\mathcal{D}}\left[\sum_{t=0}^{T} \log \pi_\theta(a_t|h_t)\right] \tag{17}$$

We implement two BC approaches in this work, a standard multi-layer neural network approach (BC-MLP) and a recurrent neural network approach (BC-RNN). These approaches are strong baselines for simulated driving agents. [25], [26], [46], [47].

Intelligent Driver Model: The IDM [27] accepts observations of vehicle speed v, relative speed to the lead vehicle Δv, and distance headway to the lead vehicle d as inputs and outputs acceleration a (the control action) using the following rule:

$$a_t = a_{max}\left[1 - \left(\frac{v_t}{\tilde{v}}\right)^4 - \left(\frac{\tilde{d}}{d_t}\right)^2\right] \tag{18}$$

[1]The source code is available at `https://github.com/ran-weii/interactive_inference`.

[2]In this sense, the generative model in the proposed POMDP model can be seen as categorical. There is evidence to support the categorical nature of human perception in psycho-physical experiments [45] that are simpler than the one considered in this paper

where $\tilde{v}$ is a desired speed and $\tilde{d}$ is the desired distance headway defined as:

$$\tilde{d} = d_0 + v_t\tau - \frac{v_t \Delta v_t}{2\sqrt{a_{max} b}} \tag{19}$$

The IDM has the following parameters: $\tilde{v}$ the desired speed, a_{max} the maximum acceleration rate which can be implemented by the driver, d_0 the minimum allowable distance headway, τ the desired headway time, and b the maximum deceleration rate. To capture heterogeneity in data, we specify an "ensemble" of IDM models, i.e. a distribution over actions given observations. The ensemble model parameters, i.e. mean and variance of action given observations, are obtained by maximizing the likelihood of the ensemble model predictions to the dataset of observed actions subject to the mean action satisfying (18).

Active Inference Driving Agent: We parameterized the state transition probability distributions $\mathbb{T}_{\hat{\theta}_1}(s'|s,a)$ as categorical distributions and the observation probability distributions $\mathbb{O}_{\hat{\theta}_1}(o|s)$ using normalizing flows [48], specifically, a shared inverse autoregressive flow with a set of Gaussian base distributions [49]. Thus, observations that have high density under the conditional distribution of each state represent the "prototypical" observation for that state. The active inference preference distribution $\tilde{P}_{\hat{\theta}_2}(s)$ is parameterized using a categorical distribution. We then obtained the finite horizon policy in (12) by computing the value function in (8) using a finite number of value iterations steps using the QMDP method [50]. By optimizing the policy log likelihood (15), both the preferences and prototypical observations are fitted to explain actions in the dataset.

B. Dataset

We trained and evaluated AIDA, BC, and IDM using the INTERACTION dataset [51], a publicly available driving dataset recorded using drones on fixed road segments in the USA, Germany, and China. The dataset provides a lanelet2 format map [52] and a set of time-indexed trajectories of the positions, velocities, and headings of each vehicle in the scene in the map's coordinate system at a sampling frequency of 10 Hz, and the vehicle's length and width for each road segment. The dataset contains a variety of traffic behaviors, including car following, free-flow traffic, and merges.

Due to our emphasis on modeling longitudinal control behavior in car following, we selected a subset of the data to include car following data from a two-way, seven-lane highway segment in China with a total distance of 175 m. We focused on vehicles in the four middle lanes shown in Figure 2, where the blue west-bound lanes have denser traffic and more stop-and-go behavior and the orange east-bound lanes have sparser traffic at higher speed. We further filtered the remaining vehicles according to two criteria: 1) there was a lead vehicle with a maximum distance headway of 60 m, and 2) the ego vehicle was not performing a merge or lane change. This focus facilitates algorithm comparisons by removing environmental artifacts. We identified merging and lane change behavior using an automated logistic regression-based approach and validated the classifications with a manual review of a subset of trajectories. We also removed all trajectories with length shorter than 10 seconds for the dense lanes and 5 seconds for the sparse lanes, leaving a total of 1,254 trajectories in the dense lanes and 290 trajectories in the sparse lanes with an average length of 14 seconds. We only used the dense lane data for training models.

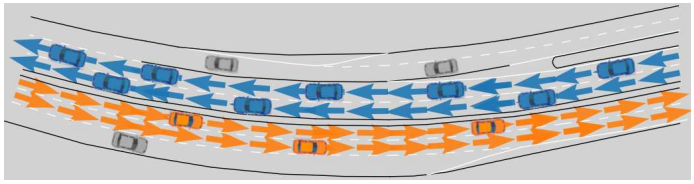


Fig. 2: Top-down view of the roadway explored in the analysis. The west-bound lanes (blue) have denser traffic and more stop-and-go behavior whereas the east-bound lanes (orange) have sparser traffic and higher speed. We trained the models to emulate the behavior of the blue cars and evaluated the models' ability to predict the behavior of the blue and orange cars. Grey cars in the merging lanes were excluded.

1) Feature Computation: The input features to the IDM are defined in (18) and (19). For BC and the AIDA, we used d and Δv but excluded v to prevent the models from achieving spuriously high training accuracy by computing acceleration predictions from past ego velocities, a well-known phenomenon reported in prior studies [25], [26], [53]. Furthermore, we included an additional feature τ^{-1} in BC and AIDA which is a visual estimate of inverse time-to-collision defined as the rate of change of the visual angle of the lead vehicle from the ego driver's seat position divided by the angle itself [54]. This feature was chosen to account for speed control and puts the information contained in the inputs to BC and the AIDA on a similar level to the IDM as the IDM implicitly accounts for time-to-collision in its desired distance headway computation in (19). It also makes our model consistent with recent family of driver models [55]–[57].

We computed all features in the Frenet frame (i.e., lane-centric coordinates [58]), by first transforming vehicle positions, velocities, and headings using the current lane center line as the reference path and then computing the features from the transformed positions and velocities. We obtained the drivers' instantaneous longitudinal control inputs (i.e., accelerations) from the dataset by differentiating the Frenet frame longitudinal velocities. For BC and the AIDA, we discretized the continuous control inputs into discrete actions using a Gaussian mixture model of 15 Gaussian components with mean and variance parameters chosen with the Bayesian Information Criteria [59].

C. Model Evaluation and Comparison

We evaluated and compared our models' ability to generate behavior similar to the human drivers in the dataset using both open-loop offline predictions and closed-loop online simulations. In both cases, we evaluated the models (15 seeds for each model class) on two different held-out testing datasets. The first dataset includes vehicles from the same dense lanes as the training dataset. This dataset tests whether the models can generalize to unseen vehicles in the same traffic condition.

We obtained this dataset by dividing trajectories in the dense lanes using a 7-3 train-test ratio. The second dataset includes vehicles from the sparse lanes. This dataset tests whether the models can generalize to unseen vehicles in novel traffic conditions, since the traffic in the east-bound lanes have on average higher speed and less density.

1) Offline Evaluation: The goal of the offline evaluation was to assess each model's ability to predict a driver's next action based on the observation-action history recorded in the held-out testing dataset. This task evaluates the models' ability to be used as a short-horizon predictor of other vehicles' behavior in an on-board trajectory planner [60]. We measured a model's predictive accuracy using Mean Absolute Error (MAE) of the predicted control inputs (unit=m/s^2) on the entire held-out testing datasets. For the IDM, the predicted control inputs were given by the IDM rule, i.e., we discarded the variance used for model fitting. For BC and the AIDA, predicted action is produced by first sampling a discrete action from the action distribution predicted by the models and then sampling the mean of the selected Gaussian component from the Gaussian mixture model used to perform action discretization. MAE of each dataset action was calculated as the average of 30 samples.

2) Online Evaluation: Rather than predicting instantaneous actions, the goal of the online evaluation was to assess the models' ability to generate trajectories similar to human drivers such that they can be used as simulated agents in automated vehicle training and testing environments [24]. This is fundamentally different from offline predictions because the models need to choose actions based on observation-action history generated by its own actions rather than those stored in the fixed, offline dataset. This can introduce significant distribution shift [61] sometimes resulting in situations outside the model's training data, which can lead to poor action selection.

We built a single-agent simulator where the ego vehicle's longitudinal acceleration is controlled by the trained models and its lateral acceleration is controlled by a feedback controller for lane-centering. The lead vehicle simply plays back the trajectory recorded in the dataset. Other vehicles do not have any effect on the ego vehicle, given our observation space does not contain other vehicle related features. We tested the models on 100 randomly chosen trajectories in each of the dense-lane and sparse-lane settings.

Following [23], we measured the similarity between the generated trajectories and the true trajectories using the following metrics:

1) Average deviation error (ADE; unit=m): deviation of the Frenet Frame position from the dataset averaged over all time steps in the trajectory.
2) Lead vehicle collision rate (LVCR; unit=$\%$): percentage of testing trajectories containing collision events with the lead vehicle. A collision is defined as an overlap between the ego and lead vehicles' bounding boxes.

3) Statistical Evaluation: Following the recommendations in [62], [63] for evaluating learned control policies in stochastic environments with a finite number of testing runs, we represented the central tendency of a model's offline prediction and online control performance using the interquartile mean (IQM) of the offline MAEs and online ADEs. The IQMs are computed by 1) ranking all tested trajectories by their respective performance metrics and 2) computing the mean of the performance metrics ranked in the middle 50%. Collision rates are computed as the percentage of testing runs that resulted in a collision. It should be noted that IQM makes the difference between each model's performance central tendency more salient at the expense of removing the tails of the performance distribution. Thus, we also provide the average performance results in appendix (VII-E). To compare the central performance difference between the AIDA and baseline models, we performed two-sided Welch's t-tests with 5 percent rejection level on the MAE-IQM and ADE-IQM values computed from different random seeds with the assumption that the performance distributions between two models may have different variances [62], [63].

D. Results and Discussion

1) Offline Performance Comparison: Figure 3 shows the offline evaluation results for each model with the model type on the x-axis and the IQMs of acceleration prediction MAEs averaged across the testing dataset on the y-axis. The color of the points in the figure represents the testing condition and each point corresponds to the result of a model initialized from a different random seed. The points are randomly distributed around each x-axis label for clarity. Dispersion on the y-axis indicates sensitivity in the model to initial training conditions. The plot illustrates that the AIDA had the lowest MAE-IQM in the sparse-lane tests, followed by BC-RNN, IDM, and BC-MLP. The corresponding pairwise Welch's t-test results in Table V (Appendix VII-F) show that the differences between AIDA and baseline models are significant. The difference between IDM and BC-RNN was surprisingly small and BC-MLP had substantially larger MAE. This was likely because the IDM rule was well-suited to capture behavior in this traffic condition, whereas the accuracy of BC-MLP was restricted by the features it had access to and action discretization. In the sparse-lane tests, AIDA performed similarly to BC models with a few seeds substantially better than BC models. IDM performed substantially worse and also with much higher variance across different seeds. Given IDM trained from different initializations converged to similar final parameters, this result was most likely due to the distribution shift between training and testing sets and IDM rule's lack of adaptability to different traffic conditions. However, the poor performance of IDM may be specific to the dataset considered in this paper (see Appendix, Section VII-E).

To understand each model's actual behavior, Figure 4 compares the predicted actions of each model's best performing seed versus the ground truth on a randomly selected trajectory in the dense-lane (left) and sparse-lane (right) settings, respectively, where shading corresponds to 1 standard deviation of the predictive distribution represented by 30 samples as described in section V-C.1. In the dense-lane setting, all models captured the variation of actions in the dataset, i.e., acceleration first decreased and then increased. However, the acceleration magnitudes predicted by IDM were substantially

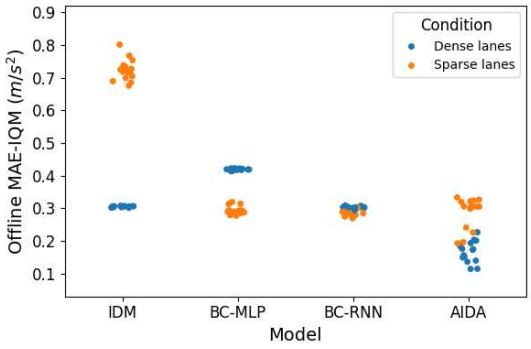


Fig. 3: Offline evaluation MAE-IQM. Each point corresponds to a random seed used to initialize model training and its color corresponds to the testing condition of either dense-lane or sparse-lane.

smaller than the ground truth. In the sparse-lane setting, the prediction interval of all BC models and AIDA were able to cover ground truth actions. However, IDM predictions were substantially lower than the ground truth. These patterns are consistent with the aggregate measures in Figure 3.

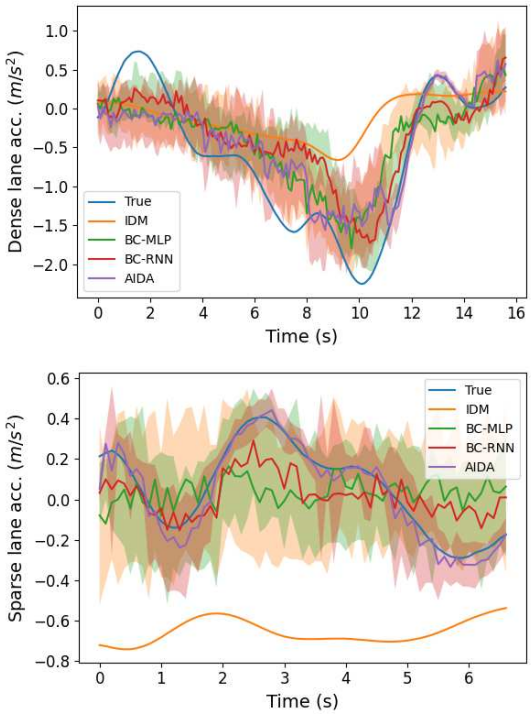


Fig. 4: Example offline predictions in the dense-lane (top) and sparse-lane (bottom) settings. Each line except for the blue line represents the mean prediction of the corresponding model. Shading represents 1 standard deviation of prediction interval. The prediction intervals for BC and AIDA are computed by drawing 30 samples from the models' predictive distributions. IDM has no prediction interval because it's deterministic.

2) Online Performance Comparison: Figure 5 shows the IQM of each model's ADEs from data set trajectories in the online evaluations using the same format as the offline evaluation results. In the dense-lane testing condition, all models had ADE-IQM values between 1.8 m and 2.8 m, which is less than the length of a standard sedan ($\approx$ 4.8 m; [64]). Among all models, BC-MLP achieved the lowest ADE values for both the dense-lane and sparse-lane conditions, followed by the AIDA, IDM, and BC-RNN. Furthermore, both the AIDA and BC models achieved lower ADE-IQM in the sparse lane settings compared to the dense-lane setting, however the IDM achieved higher ADE-IQM in the sparse-lane setting. The Welch's t-test results in Table VI show that AIDA's online test performances are significantly different from all baseline models in both the dense-lane and sparse-lane settings ($P \leq 0.01$). These findings confirm that the AIDA and BC models generalized better to the sparse-lane setting than the IDM and suggest that the AIDA's average online trajectory-matching ability is *on average* better than IDM and BC-RNN, although BC-MLP is better than the AIDA. However, it should be noted that the tail-end behavior of AIDA and BC-RNN can be worse when evaluated under average ADE (i.e., without IQM; see Figure 12 in Appendix VII-E) where the worst AIDA seed performed approximately equal to the worst BC-RNN seed, both of which would increase online ADE by 1 m.

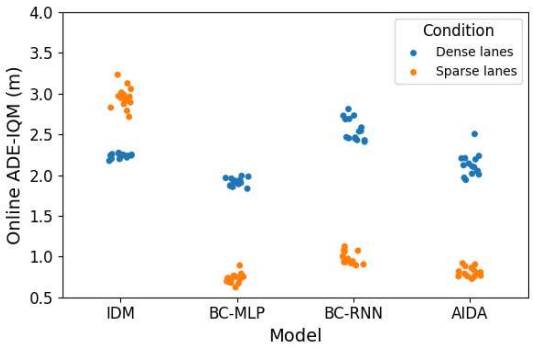


Fig. 5: Online evaluation ADE-IQM. Each point corresponds to a random seed used to initialize model training and its color corresponds to the testing condition of either dense-lane or sparse-lane.

To understand how trajectory deviations were generated, Figure 6 shows the ADE of the best seed of each model averaged over all testing episodes for each time step in the dense-lane (left) and sparse-lane (right) scenarios. We truncated the plots at 10 s and 4 s because there are very few trajectories longer than those horizons making the curves highly oscillatory. The amount of deviations generated by different models are consistent with the prior study [46]. The ranking of model performance is also consistent with the aggregated measures in Figure 5. In the dense-lane settings (Figure 6 left), model performance started to differentiate around 4 s but the differences were not substantial (i.e., up to 2 m). In contrast, in the sparse-lane setting, IDM generated substantially larger deviations from the beginning and BC-MLP and AIDA had nearly matching ADE at all time steps.

Figure 7 shows the lead vehicle collision rates for each random seed and model using the same format as Figure 5. The figure illustrates that in the dense-lane condition, the random seeds for BC-MLP, BC-RNN, and the AIDA had more

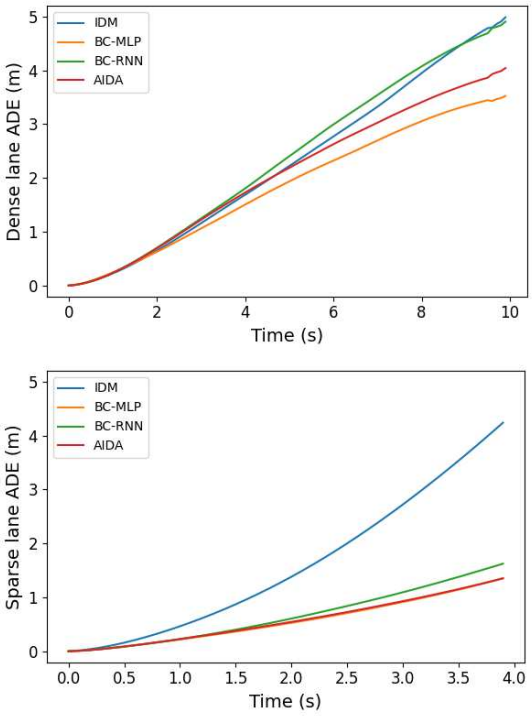


Fig. 6: Online evaluation ADE for each time step averaged over all online testing episodes for the dense-lane (top) and sparse-lane (bottom) settings by the best seed of each model.

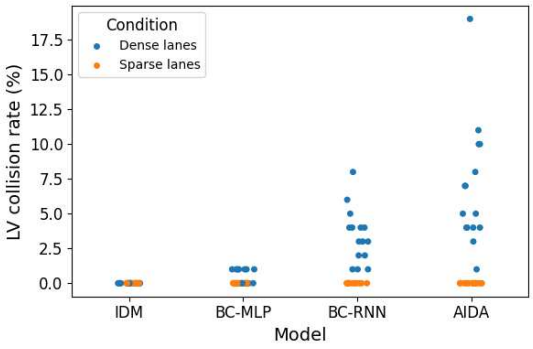


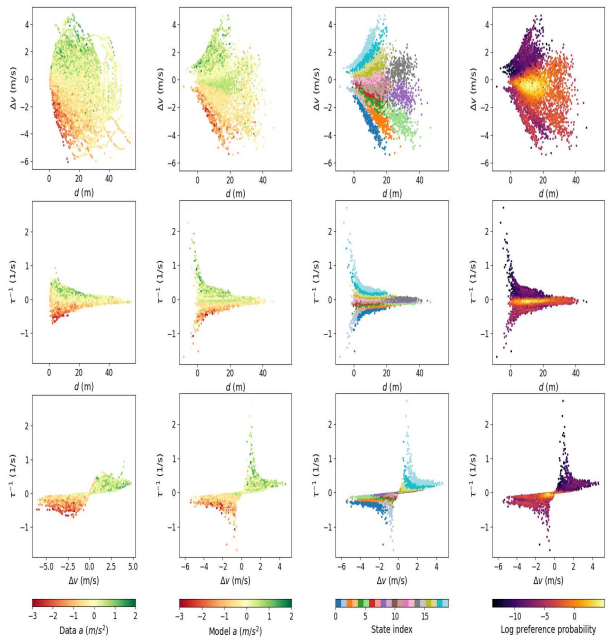
Fig. 7: Lead vehicle collision rate in online evaluation. Each point corresponds to a random seed used to initialize model training and its color corresponds to the testing condition of either dense-lane or sparse-lane.

collisions than the IDM (0% collision rate across all seeds). In particular, BC-RNN and the AIDA had substantial differences across random seeds compared to the other models. However, the minimum collision rates for BC-MLP, BC-RNN, and the AIDA were consistent (less than or equal to 1%). In the sparse-lane condition, the collision rate was 0% for all four models. The higher collision rates in the dense-lane data are likely due to the traffic density and complexity, which were higher in the dense-lane condition compared to the sparse-lane condition. This is also due to the way we defined a collision in section V-C.2 as any overlapping of vehicle bounding boxes. As we show later in section V-D.5, many collision events were due to insufficient braking magnitude despite correct braking intent, part of which can be attributed to discrete belief and action spaces. This puts ego vehicle's stopping position slightly ahead of the no-collision position without generating a large position deviation as commonly seen in machine-learned driving agents [46].

3) AIDA Interpretability Analysis: The previous sections suggest that the AIDA can capture driver car following behaviorcomparably if not better than baseline models. However, the findings have yet addressed the interpretability of the AIDA. Interpretability represents the ability to understand the relationship between model input and output and is a crucial element of model deployment success [65]. While there is no established metric for model interpretability, Räukur et. al. [66] recommend assessments based on the ease of comprehending the connection between model input and output and tracing model predictive errors to internal model dynamics. Given that the AIDA's decisions are emitted from a two-step process, i.e., (1) forming beliefs about the environment and (2) selecting control actions that realized preferred states (i.e., minimize free energy), the model's interpretability depends on the two sub-processes both independently and jointly. Thus, we examined the learned input-output mechanism by visualizing the components (i.e., the observation, transition, and preference distributions) of the best performing AIDA seed and verified them against expectations guided by driving theory [67]–[69]. We then examined the joint belief-action process by replaying the AIDA beliefs and diagnosing its predictions of recorded human drivers in the offline setting and its own decisions in the online setting.

4) AIDA Component Interpretability: Initial insights into the model input and output connections can be gained by visualizing the AIDA components, specifically its policy (Figure 8b), observation distribution (Figure 8c), and preference distribution (Figure 8d). These figures show 200 random "prototypical" samples from the observation distribution $\mathbb{O}(o|s)$ of each state, plotted on each pair of observation modalities. The top row shows the samples using distance headway (d; x-axis) by relative velocity to the lead vehicle (Δv; y-axis), the middle row shows distance headway by τ^{-1}, and the bottom row shows relative velocity by τ^{-1}. Color is used to highlight relevant quantities of interest. We further used samples drawn from the INTERACTION dataset, plotted in Figure 8a and colored by the recorded accelerations, to facilitate interpreting the the AIDA samples. The shape of the sampled points matches the contour of the empirical dataset (Figure 8a), particularly in the middle and bottom visualizations, which suggests that the model's learned observation model aligns with the recorded observations in the dataset. However, the learned distributions also showed longer tails at the edge of the data distribution. This was expected because the dataset does not contain samples that correspond to these extreme conditions. Thus the model could not learn accurate kinematics in these regions. Nevertheless, this does not affect the interpretability analysis.

Figure 8b illustrates the observation samples by the model's chosen control actions. Darker green and red colors correspond



(a) (b) (c) (d)

Fig. 8: Visualizations of the best performing AIDA seed. In panel (a), we plotted observations sampled from the dataset. In panels (b), (c), and (d) we illustrate AIDA's learned policy, observation model, and preference model via 200 "prototypical" samples from each state's conditional observation distribution $\mathbb{O}(o|s)$ and plotted the samples for each pair of observation feature combinations. The points in each panel are colored by: (a) accelerations from the dataset, (b) predicted accelerations upon observing the sampled signals from a uniform prior belief, (c) state indices (d) log probabilities of the preference distribution.

to larger acceleration and deceleration magnitudes, respectively, and light yellow color corresponds to near zero control inputs. The color gradient at different regions in Figure 8b is consistent with that of the empirical dataset shown in Figure 8a. This shows that the model learned a similar control rule (i.e., observation to action mapping) as the empirical dataset. The control rule can be interpreted as the tendency to choose negative accelerations when the relative speed and τ^{-1} are negative and the distance headway is small, and positive accelerations in the opposite case. Furthermore, the sensitivity of the red and green color gradients with respect to distance headway shows that the model tends to accelerate whenever there is positive relative velocity, regardless of the distance headway. However, it tends to input smaller deceleration at large distance headway for the same level of relative speed.

Figure 8c shows the observation samples colored by their associated discrete states. The juxtaposition of color clusters in the top panel shows that the AIDA learned to categorize observations by relative speed and distance headway and its categorization for relative speed is more fine-grained at small distance headways and spans a larger range of values. The middle and bottom panels show that its categorization of relative speed is highly correlated with τ^{-1} as the ordering of colors along the y-axis is approximately the same as in the top panel. The middle and bottom panels show that the AIDA's categorization of high τ_1 magnitude states (blue and cyan clusters) have a larger span than that of low τ^{-1} magnitude states. These patterns further establish that the AIDA has learned a representation of the environment consistent with the dataset. At the same time, it can be interpreted as a form of satisficing in that the model represents low urgency large distance headway states with less granularity [70].

Figure 8d shows the observation samples by the log of its preference probability, $\tilde{P}(o) = \sum_s \tilde{P}(s)\mathbb{O}(o|s)$, where higher preference probability (i.e., desirability) corresponds to brighter colors (e.g., yellow) and lower desirability corresponds to darker colors (e.g., purple). The figure shows that the highest preference probability corresponds to observations of zero τ^{-1}, zero relative velocity, and a distance headway of 18 m (see the center region of the middle chart, and the yellow circle at the left-center of the top chart). This aligns with the task-difficulty homeostasis hypothesis that drivers prefer states in which the crash risk is manageable [67] and not increasing. It is also consistent with the observed driver behavior in Figure 8a where drivers tend to maintain low accelerations (light yellow points) within the same regions.

Overall, these results show a clear mapping between the AIDA's perceptual (Figure 8c) and control (Figure 8d and 8b) behavior that is both consistent with the observed data and straightforwardly illustrated using samples from the fitted model distributions. This mapping facilitates predictions of the AIDA's reaction to observations without querying the model, which is an important dimension of interpretability in real world model validation [66].

5) AIDA Decision Diagnostics: While the previous analysis illustrates the interpretability of individual model components, the overall model interpretability is also contingent upon understanding the interaction between components. To address this, we analyzed two dense-lane scenarios where the AIDA made sub-optimal decisions in the model testing phase — one from the offline evaluations where the AIDA's predictions had the largest MAE and one from the online evaluations where the AIDA generated a rear-end collision with the lead vehicle. We first visualized the AIDA's beliefs and policies as the model generated actions and then used those visualizations to demonstrate how the transparent input-output mechanism in the AIDA can be used to mitigate the sub-optimal decisions.

The chosen offline evaluation trajectory is visualized in Figure 9. The left column charts show the data of the three observation features over time. The right column charts show the time-varying ground truth action probabilities over time (top), action probabilities predicted by the AIDA over time (middle), and environment state probabilities $P(s|h)$ inferred by the AIDA over time (bottom). In the right-middle and right-bottom charts, the action and belief state indices are sorted by the mean acceleration and τ^{-1} value of each state to facilitate alignment with the left and top-right charts. We labeled the actions by the corresponding means but not the belief states because they represent multi-dimensional observation categorizations (see Figure 8c). The bottom-right chart shows that the inferred belief patterns closely followed the observed relative speed and τ^{-1} in the left-middle and left-

bottom charts with high precision, i.e., close to probability of 1. The predicted action probabilities in the right-middle chart followed the trend of the ground truth actions, however, they exhibited substantially higher uncertainty at most time steps and multi-modality at $t = 1$ s and $t = 12$ s, where one of the predicted modes coincided with the true actions. Given the inferred beliefs were precise, uncertain and multi-model actions were likely caused by inter-driver variability in the dataset, where drivers experienced similar belief states but selected different actions. Alternatively, this uncertainty may be caused by actual drivers having highly different beliefs after experiencing similar observations, In either case, the error in AIDA predictions can be attributed to inconsistency between the belief trajectories and action predictions.

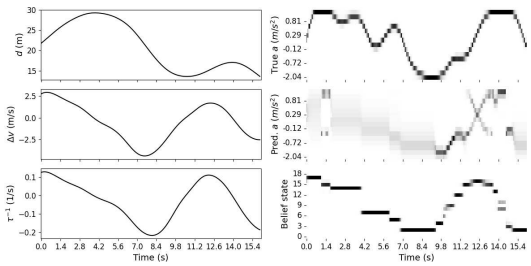


Fig. 9: Visualizations of a dense-lane offline evaluation trajectory where the AIDA had the highest prediction MAE. The charts in the left column show distance headway, relative speed, and τ^{-1} signals observed by the model over time. The binary heat maps in the right column show the ground truth action probabilities (top), action probabilities predicted by the AIDA (middle), and the corresponding belief states (bottom) over time (x-axis), where darker colors correspond to higher probabilities. The belief state and action indices are sorted by the mean τ^{-1} and acceleration value of each state, respectively.

The chosen online evaluation trajectory which resulted in a rear-end collision with the lead vehicle is shown in Figure 10 plotted using the same format as Figure 9. The duration of the crash event is highlighted by the red square in the bottom-left chart, where the sign of τ^{-1} values instantly inverted when overlapping bounding boxes between the ego and lead vehicle first occurred and eventually ended. The AIDA initially made the correct and precise decision of braking, however, its predictions for high magnitude actions became substantially less precise prior to the collision ($t > 1$ s; see right middle chart). This led to the model failing to stop fully before colliding with the lead vehicle. The belief pattern shows that the AIDA tracked the initial decreasing values of relative speed and τ^{-1} but did not further respond to increasing magnitude of τ^{-1} 3 seconds prior to the crash (starting at $t = 1.6$ s). These findings show that the model exhibited the correct behavior of being "shocked" by out-of-sample near-crash observations, however, the learned categorical belief representation was not able to extrapolate beyond the data from the crash-free INTERACTION dataset.

The analysis of the near-crash AIDA beliefs suggests that editing the AIDA's learned environment dynamics model (i.e., the transition and observation distributions) to properly recognize near-crash observation signals can likely avoid the current crash.

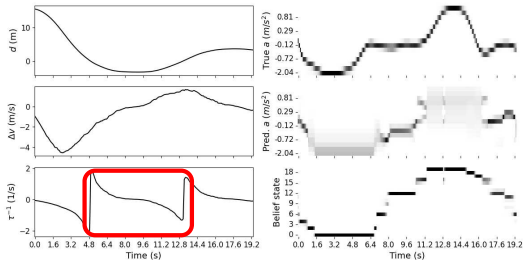


Fig. 10: Visualizations of a dense-lane online evaluation trajectory where the AIDA generated a rear-end collision with the lead vehicle. This figure shares the same format as Figure 9. The red square in the bottom-left chart represents the duration of the rear-end crash event where the vehicle controlled by the AIDA had overlapping bounding box with the lead vehicle.

The analyses in this section show that the decision making structure in the AIDA enables modelers to reason about the training dataset's effect on the learned model behavior. To the best of our knowledge, this analysis is not possible with neural network BC models using existing interpretability tools. Thus AIDA represents a significant step forward for interpretable perception and control models of human control behavior.

VI. Conclusions

We consider the problem of learning a model of human perception and control based on data in the form of observations and implemented actions. We posit a POMDP model and formulated a bi-level optimization formulation of Maximum A Posteriori (MAP) estimate for the primitives of the model. To illustrate the estimation methodology we develop a model of driver behavior (AIDA) with the reward specification motivated by the active inference framework from cognitive science. Using car following data, we showed that the AIDA performed comparably and in certain cases better than the rule-based IDM and data-driven neural network benchmarks. Using an interpretability analysis, we showed that the structure of the AIDA provides superior transparency of its input-output mechanics than the neural network models. Future work should focus on training with data from more diverse driving environments and examining model extensions that can capture heterogeneity across human agents.

Acknowledgements

Support for this research was provided in part by the U.S. Department of Transportation (DOT), University Transportation Centers Program to the Safety through Disruption University Transportation Center (451453-19C36) and the UK Engineering and Physical Sciences Research Council (EPSRC; EP/S005056/1). Thanks to advisers, J. Engstrom and M. O'Kelly, from Waymo, who helped set the technical direction, identified relevant published research, and advised on the scope and structuring of this publication, independent of the support this research received from USDOT.

References

[1] D. Badre, A. Bhandari, H. Keglovits, and A. Kikumoto, "The dimensionality of neural representations for control," *Current Opinion in Behavioral Sciences*, vol. 38, pp. 20–28, 2021.

[2] H. Op de Beeck, J. Wagemans, and R. Vogels, "Inferotemporal neurons represent low-dimensional configurations of parameterized shapes," *Nature neuroscience*, vol. 4, pp. 1244–52, 01 2002.

[3] D. C. Knill and A. Pouget, "The Bayesian brain: the role of uncertainty in neural coding and computation," *Trends in Neurosciences*, vol. 27, no. 12, pp. 712–719, 2004.

[4] K. Friston, "The history of the future of the Bayesian brain," *NeuroImage*, vol. 62, no. 2, pp. 1230–1233, 2012.

[5] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, *An Algorithmic Perspective on Imitation Learning*, vol. 7 of *Foundations and Trends in Robotics*. 2018.

[6] A. Y. Ng, S. J. Russell, *et al.*, "Algorithms for inverse reinforcement learning.," in *Icml*, vol. 1, p. 2, 2000.

[7] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, J. Peters, *et al.*, "An algorithmic perspective on imitation learning," *Foundations and Trends® in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.

[8] N. Ab Azar, A. Shahmansoorian, and M. Davoudi, "From inverse optimal control to inverse reinforcement learning: A historical review," *Annual Reviews in Control*, vol. 50, pp. 119–138, 2020.

[9] T. Phan-Minh, F. Howington, T.-S. Chu, M. S. Tomov, R. E. Beaudoin, S. U. Lee, N. Li, C. Dicle, S. Findler, F. Suarez-Ruiz, B. Yang, S. Omari, and E. M. Wolff, "Driveirl: Drive in real life with inverse reinforcement learning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1544–1550, 2023.

[10] E. Boer and R. Kenyon, "Estimation of time-varying delay time in nonstationary linear systems: an approach to monitor human operator adaptation in manual tracking tasks," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 28, no. 1, pp. 89–99, 1998.

[11] K. van der El, D. M. Pool, H. J. Damveld, M. R. M. van Paassen, and M. Mulder, "An empirical human controller model for preview tracking tasks," *IEEE Transactions on Cybernetics*, vol. 46, no. 11, pp. 2609–2621, 2016.

[12] F. M. Drop, D. M. Pool, M. R. M. van Paassen, M. Mulder, and H. H. Bülthoff, "Objective model selection for identifying the human feedforward response in manual control," *IEEE Transactions on Cybernetics*, vol. 48, no. 1, pp. 2–15, 2018.

[13] C. Baker, J. Jara-Ettinger, R. Saxe, and J. Tenenbaum, "Rational quantitative attribution of beliefs, desires and percepts in human mentalizing," *Nature: Human Behavior*, no. 4, pp. 1–10, 2017.

[14] Y. Chang, A. Garcia, Z. Wang, and L. Sun, "Structural Estimation of Partially Observable Markov Decision Processes," *IEEE Transactions on Automatic Control*, vol. 68, no. 8, pp. 5135–5141, 2023.

[15] D. Straub and C. A. Rothkopf, "Putting perception into action with inverse optimal control for continuous psychophysics," *eLife*, vol. 11, p. e76635, sep 2022.

[16] J. Pekkanen, O. Lappi, P. Rinkkala, S. Tuhkanen, R. Frantsi, and H. Summala, "A computational model for driver's cognitive state, visual perception and intermittent attention in a distracted car following task," *Royal Society Open Science*, vol. 5, no. 9, p. 180194, 2018.

[17] K. Friston, "The free-energy principle: a unified brain theory?," *Nature reviews neuroscience*, vol. 11, no. 2, pp. 127–138, 2010.

[18] T. Parr, G. Pezzulo, and K. Friston, *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior*. MIT Press, 2022.

[19] D. Maisto, F. Donnarumma, and G. Pezzulo, "Interactive Inference: A Multi-Agent Model of Cooperative Joint Actions," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–12, 2023.

[20] J. Engström, R. Wei, A. D. McDonald, A. Garcia, M. O'Kelly, and L. Johnson, "Resolving uncertainty on the fly: modeling adaptive driving behavior as active inference," *Frontiers in Neurorobotics*, vol. 18, 2024.

[21] A. Clark, *Surfing Uncertainty: Prediction, Action, and the Embodied Mind*. Oxford University Press, 2015.

[22] J. Hohwy, *The Predictive Mind*. Oxford University Press, 2015.

[23] S. Suo, S. Regalado, S. Casas, and R. Urtasun, "Trafficsim: Learning to simulate realistic multi-agent behaviors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10400–10409, 2021.

[24] M. Igl, D. Kim, A. Kuefler, P. Mougin, P. Shah, K. Shiarlis, D. Anguelov, M. Palatucci, B. White, and S. Whiteson, "Symphony: Learning realistic and diverse agents for autonomous driving simulation," *arXiv preprint arXiv:2205.03195*, 2022.

[25] F. Codevilla, E. Santana, A. M. López, and A. Gaidon, "Exploring the limitations of behavior cloning for autonomous driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9329–9338, 2019.

[26] M. Zhou, X. Qu, and X. Li, "A recurrent neural network based microscopic car following model to predict traffic oscillation," *Transportation research part C: emerging technologies*, vol. 84, pp. 245–264, 2017.

[27] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical review E*, vol. 62, no. 2, p. 1805, 2000.

[28] M. Treiber and A. Kesting, "Microscopic calibration and validation of car-following models–a systematic approach," *Procedia-Social and Behavioral Sciences*, vol. 80, pp. 922–939, 2013.

[29] A. Kesting, M. Treiber, and D. Helbing, "Agents for traffic simulation," *Multi-agent systems: Simulation and applications*, vol. 5, 2009.

[30] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*. The MIT Press, 2009.

[31] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," *arXiv preprint arXiv:2005.01643*, 2020.

[32] A. Shenhav, M. M. Botvinick, and J. D. Cohen, "The expected value of control: An integrative theory of anterior cingulate cortex function," *Neuron*, vol. 79, no. 2, pp. 217–240, 2013.

[33] N. Tishby and D. Polani, "Information theory of decisions and actions," in *Perception-action cycle*, pp. 601–636, Springer, 2011.

[34] P. A. Ortega and D. A. Braun, "Thermodynamics as a theory of decision-making with information-processing costs," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 469, no. 2153, p. 20120683, 2013.

[35] F. Matějka and A. McKay, "Rational inattention to discrete choices: A new foundation for the multinomial logit model," *American Economic Review*, vol. 105, pp. 272–98, January 2015.

[36] D. Fudenberg, R. Iijima, and T. Strzalecki, "Stochastic choice and revealed perturbed utility," *Econometrica*, vol. 83, no. 6, pp. 2371–2409, 2015.

[37] L. P. Hansen and J. Miao, "Aversion to ambiguity and model misspecification in dynamic stochastic environments," *Proceedings of the National Academy of Sciences*, vol. 115, no. 37, pp. 9163–9168, 2018.

[38] A. Tschantz, A. K. Seth, and C. L. Buckley, "Learning action-oriented models through active inference," *PLoS computational biology*, vol. 16, no. 4, p. e1007805, 2020.

[39] J. Y. Shin, C. Kim, and H. J. Hwang, "Prior preference learning from experts: Designing a reward with active inference," *Neurocomputing*, vol. 492, pp. 508–515, 2022.

[40] R. Wei, "Value of information and reward specification in active inference and pomdps," *arXiv preprint arXiv:2408.06542*, 2024.

[41] M. Taieb-Maimon and D. Shinar, "Minimum and comfortable driving headways: Reality versus perception," *Human Factors*, vol. 43, no. 1, pp. 159–172, 2001. PMID: 11474761.

[42] S. H. Hamdar, M. Treiber, H. S. Mahmassani, and A. Kesting, "Modeling driver behavior as sequential risk-taking task," *Transportation Research Record*, vol. 2088, no. 1, pp. 208–217, 2008.

[43] S. H. Hamdar, H. S. Mahmassani, and M. Treiber, "From behavioral psychology to acceleration modeling: Calibration, validation, and exploration of drivers' cognitive and safety parameters in a risk-taking environment," *Transportation Research Part B: Methodological*, vol. 78, pp. 32–53, 2015.

[44] F. W. Siebert, M. Oehl, F. Bersch, and H.-R. Pfister, "The exact determination of subjective risk and comfort thresholds in car following," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 46, pp. 1–13, 2017.

[45] S. K. Reed, "Pattern recognition and categorization," *Cognitive Psychology*, vol. 3, no. 3, pp. 382–407, 1972.

[46] R. Bhattacharyya, B. Wulfe, D. Phillips, A. Kuefler, J. Morton, R. Senanayake, and M. Kochenderfer, "Modeling human driving behavior through generative adversarial imitation learning," *arXiv preprint arXiv:2006.06412*, 2020.

[47] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 204–211, IEEE, 2017.

[48] G. Papamakarios, E. T. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference.," *J. Mach. Learn. Res.*, vol. 22, no. 57, pp. 1–64, 2021.

[49] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," *Advances in neural information processing systems*, vol. 29, 2016.

[50] M. L. Littman, A. R. Cassandra, and L. P. Kaelbling, "Learning policies for partially observable environments: Scaling up," in *Machine Learning Proceedings 1995*, pp. 362–370, Elsevier, 1995.

[51] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clausse, M. Naumann, J. Kummerle, H. Konigshof, C. Stiller, A. de La Fortelle, *et al.*, "Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps," *arXiv preprint arXiv:1910.03088*, 2019.

[52] F. Poggenhans, J.-H. Pauls, J. Janosovits, S. Orf, M. Naumann, F. Kuhnt, and M. Mayr, "Lanelet2: A high-definition map framework for the future of automated driving," in *2018 21st international conference on intelligent transportation systems (ITSC)*, pp. 1672–1679, IEEE, 2018.

[53] P. De Haan, D. Jayaraman, and S. Levine, "Causal confusion in imitation learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[54] D. N. Lee, "A theory of visual control of braking based on information about time-to-collision," *Perception*, vol. 5, no. 4, pp. 437–459, 1976.

[55] M. Svärd, G. Markkula, J. Bärgman, and T. Victor, "Computational modeling of driver pre-crash brake response, with and without off-road glances: Parameterization using real-world crashes and near-crashes," *Accident Analysis & Prevention*, vol. 163, p. 106433, 2021.

[56] J. Engström, G. Markkula, Q. Xue, and N. Merat, "Simulating the effect of cognitive load on braking responses in lead vehicle braking scenarios," *IET Intelligent Transport Systems*, vol. 12, no. 6, pp. 427–433, 2018.

[57] A. D. McDonald, H. Alambeigi, J. Engström, G. Markkula, T. Vogelpohl, J. Dunne, and N. Yuma, "Toward computational simulations of behavior during automated driving takeovers: a review of the empirical and modeling literatures," *Human factors*, vol. 61, no. 4, pp. 642–688, 2019.

[58] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *2010 IEEE International Conference on Robotics and Automation*, pp. 987–993, IEEE, 2010.

[59] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[60] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions.," in *Robotics: Science and systems*, vol. 2, pp. 1–9, Ann Arbor, MI, USA, 2016.

[61] J. Spencer, S. Choudhury, A. Venkatraman, B. Ziebart, and J. A. Bagnell, "Feedback in imitation learning: The three regimes of covariate shift," *arXiv preprint arXiv:2102.02872*, 2021.

[62] C. Colas, O. Sigaud, and P.-Y. Oudeyer, "A hitchhiker's guide to statistical comparisons of reinforcement learning algorithms," *arXiv preprint arXiv:1904.06979*, 2019.

[63] R. Agarwal, M. Schwarzer, P. S. Castro, A. C. Courville, and M. Bellemare, "Deep reinforcement learning at the edge of the statistical precipice," *Advances in neural information processing systems*, vol. 34, pp. 29304–29320, 2021.

[64] "Sedan vehicle dimensions." `https://www.mathworks.com/help/driving/ref/sedan.html`, 2022. Accessed: 2022-12-15.

[65] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[66] T. Räukur, A. Ho, S. Casper, and D. Hadfield-Menell, "Toward transparent ai: A survey on interpreting the inner structures of deep neural networks," *arXiv preprint arXiv:2207.13243*, 2022.

[67] R. Fuller, "Towards a general theory of driver behaviour," *Accident analysis & prevention*, vol. 37, no. 3, pp. 461–472, 2005.

[68] J. Engström, J. Bärgman, D. Nilsson, B. Seppelt, G. Markkula, G. B. Piccinini, and T. Victor, "Great expectations: a predictive processing account of automobile driving," *Theoretical issues in ergonomics science*, vol. 19, no. 2, pp. 156–194, 2018.

[69] H. Summala, "Hierarchical model of behavioural adaptation and traffic accidents," *Traffic and transport psychology. Theory and application*, 1997.

[70] P. Hancock, "Is car following the real question–are equations the answer?," *Transportation research part F: traffic psychology and behaviour*, vol. 2, no. 4, pp. 197–199, 1999.

[71] A. Tamar, Y. Wu, G. Thomas, S. Levine, and P. Abbeel, "Value iteration networks," *Advances in neural information processing systems*, vol. 29, 2016.

[72] P. Karkus, D. Hsu, and W. S. Lee, "Qmdp-net: Deep learning for planning under partial observability," *Advances in neural information processing systems*, vol. 30, 2017.

[73] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[74] S. Hoogendoorn and R. Hoogendoorn, "Calibration of microscopic traffic-flow models using multiple data sources," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 368, no. 1928, pp. 4497–4517, 2010.

VII. Appendix

A. Proofs

1) Proof of Proposition 1: *Proof:* The proof is by induction. Assume $U_{t+1}(h_{t+1}) = V_{t+1}(b_{t+1})$, then

$$\begin{aligned}
U_t(h_t) &= \max_{\pi(\cdot|h_t)} \Bigg\{ \sum_a \sum_s r(s,a)\mathbb{P}(s_t = s|h_t)\pi(a|h_t) - c((\pi(\cdot|h_t)) \\
&\quad + \gamma \sum_a \sum_{o_{t+1}} \mathbb{P}(o_{t+1}|h_t, a)\pi(a|h_t)V_{t+1}(b_{t+1}) \Bigg\} \\
&= \max_{\pi(\cdot|b_t)} \Bigg\{ \sum_a r(s,a)b_t(s)\pi(a|b_t) - c((\pi(\cdot|b_t)) \\
&\quad + \gamma \sum_a \sum_{o_{t+1}} \sigma(o_{t+1}|s_t, a)\pi(a|b_t)V_{t+1}(b_{t+1}) \Bigg\} \\
&= V_t(b_t)
\end{aligned}$$

where $b_{t+1}(s) = P(s_{t+1} = s|h_t \cup \{a, o_{t+1}\})$ and the second equality follows from

$$\begin{aligned}
\mathbb{P}(o_{t+1}|h_t, a) &= \sum_{s_t} \sum_{s_{t+1}} \mathbb{O}(o_{t+1}|s_{t+1})\mathbb{T}(s_{t+1}|s_t, a)b_t(s_t) \\
&= \sigma(o_{t+1}|b_t, a)
\end{aligned}$$

B. Proof of Theorem 1

To prove *(a)*, let $Q_1, Q_2 \in \mathcal{Q}$ and $\epsilon = \|Q_1 - Q_2\|$. Then

$$\begin{aligned}
&\log\left(\sum_a \pi^0(a|b)\exp\left(\frac{1}{\alpha}Q_1(b,a)\right)\right) \\
&\leq \log\left(\sum_a \pi^0(a|b)\exp\left(\frac{1}{\alpha}Q_2(b,a) + \epsilon\right)\right) \\
&= \log\left(\exp(\epsilon)\sum_a \pi^0(a|b)\exp\left(\frac{1}{\alpha}Q_2(b,a)\right)\right) \\
&= \epsilon + \log\left(\sum_a \pi^0(a|b)\exp\left(\frac{1}{\alpha}Q_2(b,a)\right)\right)
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
&\log\left(\sum_a \pi^0(a|b)\exp\left(\frac{1}{\alpha}Q_1(b,a)\right)\right) \\
&\geq -\epsilon + \log\left(\sum_a \pi^0(a|b)\exp\left(\frac{1}{\alpha}Q_2(b,a)\right)\right)
\end{aligned}$$

Hence, we obtain that

$$\|\mathcal{B}Q_1 - \mathcal{B}Q_2\| \leq \gamma\|Q_1 - Q_2\| = \gamma\epsilon.$$

To prove *(b)*, consider the policy of the form:

$$\pi^*(a|b) = \frac{\pi^0(a|b)\exp\left(\frac{1}{\alpha}Q^*(b,a)\right)}{\sum_{a' \in A}\pi^0(a'|b)\exp\left(\frac{1}{\alpha}Q^*(b,a')\right)}.$$

where Q^* is the unique fixed point of $\mathcal{B}$. We first note that

$$\log \pi^*(a|b) = \log \pi^0(a|b) + \frac{1}{\alpha}Q^*(b,a) - \log \sum_{a'} \pi^0(a'|b) \exp\big(\frac{1}{\alpha}Q^*(b,a')\big)$$

Thus,

$$\frac{1}{\alpha}\Big[\sum_a Q^*(b,a)\pi^*(a|b) - \alpha \mathcal{D}_{KL}(\pi^*(\cdot|b)||\pi^0(\cdot|b))\Big]$$
$$= \sum_a \pi^*(a|b)\Big[\log \sum_{a'} \pi^0(a'|b)\exp\big(\frac{1}{\alpha}Q^*(b,a')\big) + \log\frac{\pi^*(a|b)}{\pi^0(a|b)}\Big] - \mathcal{D}_{KL}(\pi^*(\cdot|b)||\pi^0(\cdot|b))$$
$$= \log \sum_{a'} \pi^0(a'|b)\exp\big(\frac{1}{\alpha}Q^*(b,a')\big) \tag{20}$$

Moreover, for any policy $\pi \neq \pi^*$, it holds that

$$\frac{1}{\alpha}\Big[\sum_a Q^*(b,a)\pi(a|b) - \alpha \mathcal{D}_{KL}(\pi(\cdot|b)||\pi^0(\cdot|b))\Big]$$
$$= \sum_a \pi(a|b)[\frac{1}{\alpha}Q^*(b,a) - \log\frac{\pi(a|b)}{\pi^0(a|b)}]$$
$$= \sum_a \pi(a|b)[\log\frac{\pi^*(a|b)}{\pi^0(a|b)} + \log\sum_{a'}\pi^0(a'|b)\exp\big(\frac{1}{\alpha}Q^*(b,a')\big) - \log\frac{\pi(a|b)}{\pi^0(a|b)}]$$
$$= -D_{KL}(\pi(\cdot|b)||\pi^*(\cdot|b)) + \log\sum_{a'}\pi^0(a'|b)\exp\big(\frac{1}{\alpha}Q^*(b,a')\big) \tag{21}$$

Since $D_{KL}(\pi(\cdot|b)||\pi^*(\cdot|b)) \geq 0$ we conclude from (20) and (21) that

$$\alpha \log \sum_{a'} \pi^0(a'|b)\exp\big(\frac{1}{\alpha}Q^*(b,a')\big) = \sum_a Q^*(b,a)\pi^*(a|b) - \alpha\mathcal{D}_{KL}(\pi^*(\cdot|b)||\pi^0(\cdot|b))$$
$$= \max_{\pi(\cdot|b)}[\sum_a Q^*(b,a)\pi(a|b) - \alpha\mathcal{D}_{KL}(\pi(\cdot|b)||\pi^0(\cdot|b))] = V^*(b)$$

To prove *(c)*, we apply (20) and (21) to π to conclude that

$$V^*(b) = \max_{\pi(\cdot|b)}[\sum_a Q^*(b,a)\pi(a|b) - \alpha\mathcal{D}_{KL}(\pi(\cdot|b)||\pi^0(\cdot|b))]$$
$$= \max_{\pi(\cdot|b)}[\sum_a\sum_s r(s,a)b(s)\pi(a|b) - \alpha\mathcal{D}_{KL}(\pi(\cdot|b)||\pi^0(\cdot|b))) + \gamma\sum_a\sum_{o'}\sigma(o'|b,a)\pi(a|b)V^*(b')]$$

where b' is the updated Bayes belief distribution after action a is implemented and observation o' is recorded and the optimality of π^* follows from Proposition 1.

C. Implementation details

The source code is available at `https://github.com/ran-weii/interactive_inference`.

1) BC Implementation: For BC-MLP, we used a two-layer MLP network with ReLU activation and 40 hidden units in each layer. For BC-RNN, we used a two-layer MLP network on top of a single-layer GRU network with ReLU activation and 30 hidden units in each layer. The GRU layer only takes in past observations but not past actions. We found that larger number of hidden units in the BC-RNN model led to significant overfitting. Both BC-MLP and BC-RNN receive 3 input observations and output probability distributions over 15 discrete actions.

2) AIDA Implementation: The AIDA implementation follows the value-iteration network and QMDP network [71], [72] to enable end-to-end training in Pytorch [73]. We used a state dimension of 20, action dimension of 15, and a planning horizon of 30 steps (3 seconds). Discrete state transition probabilities are parameterized using categorical distributions. The continuous observation distributions are parameterized using a set of Gaussian distributions, one for each discrete state, and a shared noramlizing flow network to transform the base Gaussian distributions into more flexible density estimators. Specifically, we use inverse autoregressive flow [49] parameterized by a two-layer MLP network with ReLU activation and 30 hidden units in each layer.

For each mini-batch of observation-action sequences, we first computed the likelihood of the observations at all time steps and compute the belief at each time step as:

$$b(s_t) = \frac{P(o_t|s_t)\sum_{s_{t-1}}P(s_t|s_{t-1},a_{t-1})b(s_{t-1})}{\sum_{s_t}P(o_t|s_t)\sum_{s_{t-1}}P(s_t|s_{t-1},a_{t-1})b(s_{t-1})}. \tag{22}$$

We then computed the value function (8) for the EFE reward and the resulting optimal policy in (7) for each inferred belief using the QMDP approximation method [50]. The QMDP method assumes the belief-action value can be approximated as a weighted-average of the state-action value:

$$Q^*(b_t,a_t) = \sum_{s_t} b_t(s_t)\mathcal{Q}^*(s_t,a_t), \tag{23}$$

where

$$Q^*(s_t,a_t) = r(s_t,a_t) + \log\pi(a_t|s_t) + \sum_{s_{t+1}}P(s_{t+1}|s_t,a_t)V^*(s_{t+1}), \tag{24}$$

$$r(s_t,a_t) = EFE(s_t,a_t) = D_{KL}(P(s_{t+1}|s_t,a_t)||\tilde{P}(s_{t+1})) + \mathbb{E}_{P(s_{t+1}|s_t,a_t)}[\mathcal{H}(P(o_{t+1}|s_{t+1}))]. \tag{25}$$

and $\forall s \in \mathcal{S}, Q^*(s_{t+H+1}) = 0$. We further approximate the observation entropy using the entropy of the Gaussian base distributions of the normalizing flows which can be computed in closed form.

The combination of QMDP approximation and computing the observation entropy in (25) using the Gaussian base distributions reduced the model's ability to evaluate state uncertainty. However, given the low state uncertainty shown in Figure 9 and Figure 10 (i.e., the nearly deterministic belief states in the lower right charts), these approximations do not significantly impact the current results while providing the benefit of computational tractability.

3) Training and hyperparameters: We trained all models using the Adam optimizer for a fixed number of epochs which are selected upon visual inspection of convergence, i.e., the loss function no longer changes significantly. For all models, we use a batch size of 100. For AIDA, we use a smaller learning rate of 0.001 for the normalizing flow network than the rest of the model because of its sensitivity to large learning rates. Additional hyperparameters are reported in table I.

TABLE I: Training hyperparameters

Hyperparameter	IDM	BC-MLP	BC-RNN	AIDA
Learning rate	0.005	0.001	0.001	0.01
Training epochs	300	500	500	500

TABLE II: Model input features

Feature	IDM	BC-MLP	BC-RNN	AIDA
Distance headway (d)	Yes	Yes	Yes	Yes
Relative speed (Δv)	Yes	Yes	Yes	Yes
Speed (v)	Yes	No	No	No
τ^{-1}	No	Yes	Yes	Yes

TABLE III: Model parameter counts

	IDM	BC-MLP	BC-RNN	AIDA
Count	6	4125	6465	7670

TABLE IV: Fitted IDM parameters: mean and standard deviations across 15 seeds.

$\tilde{v}$	τ	d_0
12.2 $\pm$ 0.2	0.83 $\pm$ 0.03	1.07 $\pm$ 0.07
a_{max}	b	σ
0.21 $\pm$ 0.006	2.68 $\pm$ 0.19	0.46 $\pm$ 0.004

TABLE V: Two-sided Welch's t-test results of offline MAE-IQM against baseline models. Asterisks indicate statistical significance with $\alpha = 0.05$.

Baseline	Comparison	t(df=14)	p-value
IDM	dense-lane	t=16.38	p<0.001*
BC-MLP	dense-lane	t=29.74	p<0.001*
BC-RNN	dense-lane	t=16.03	p<0.001*
IDM	sparse-lane	t=29.11	p<0.001*
BC-MLP	sparse-lane	t=0.44	p=0.66
BC-RNN	sparse-lane	t=-0.04	p=0.97

TABLE VI: Two-sided Welch's t-test results of online ADE-IQM against baseline models. Asterisks indicate statistical significance with $\alpha = 0.05$.

Baseline	Comparison	t(df=14)	p-value
IDM	dense-lane	t=3.05	p<0.01*
BC-MLP	dense-lane	t=-5.46	p<0.001*
BC-RNN	dense-lane	t=8.73	p<0.001*
IDM	sparse-lane	t=58.18	p<0.001*
BC-MLP	sparse-lane	t=-3.77	p<0.001*
BC-RNN	sparse-lane	t = 6.87	p<0.001*

D. Simulation platform

We built a single-agent simulator for online evaluation of the trained models. The simulator plays back lead vehicle trajectories from the dataset which is converted into the Frenet frame to compute LV related observations for the ego vehicle. We model the effect of ego control actions on its own position and velocity in the Frenet frame using linear dynamics:

$$\begin{bmatrix} x' \\ y' \\ v_x' \\ v_y' \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} + \begin{bmatrix} 0.5\Delta t^2 & 0 \\ 0 & 0.5\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a_x \\ a_y \end{bmatrix} \tag{26}$$

Since the models only control longitudinal action a_x, we used a simple feedback controller for lateral actions with position and velocity gain $[-0.01, -0.2]$.

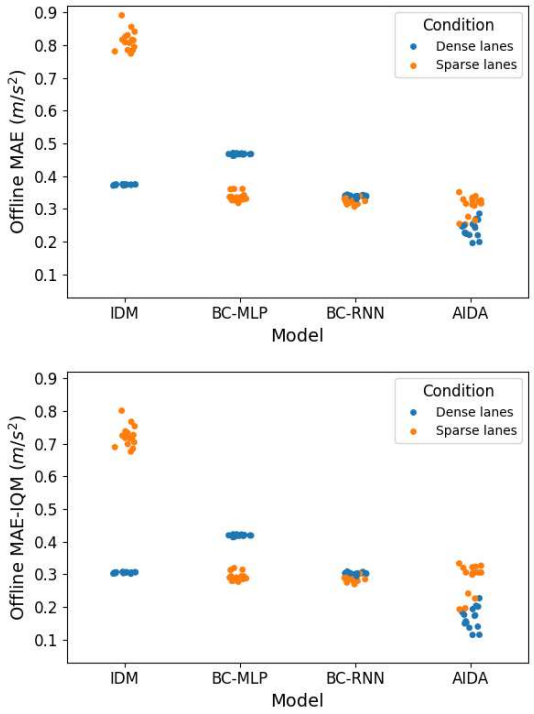


Fig. 11: Comparison of offline MAE by each model with IQM (top) and without IQM (bottom).

E. Comparison of average metrics and IQM

This section compares model offline (Figure 11) and online (Figure 12) evaluation performance with and without IQM. In the offline evaluation setting, IQM did not substantially affect model performance, e.g., the IDM MAE in the sparse-lane setting was only reduced by 0.1 m/s^2. In the online evaluation setting, IQM reduced the average ADE of BC-RNN and AIDA by 1 m and substantially reduced the upper tail of AIDA seeds. However, IDM also increased the difference between IDM's ADE in the dense-lane and sparse-lane settings, which is consistent with IDM having worse offline prediction accuracy in the sparse-lane setting as shown in Figure 3. Overall, IQM did not change the ranking of models. It should be noted that

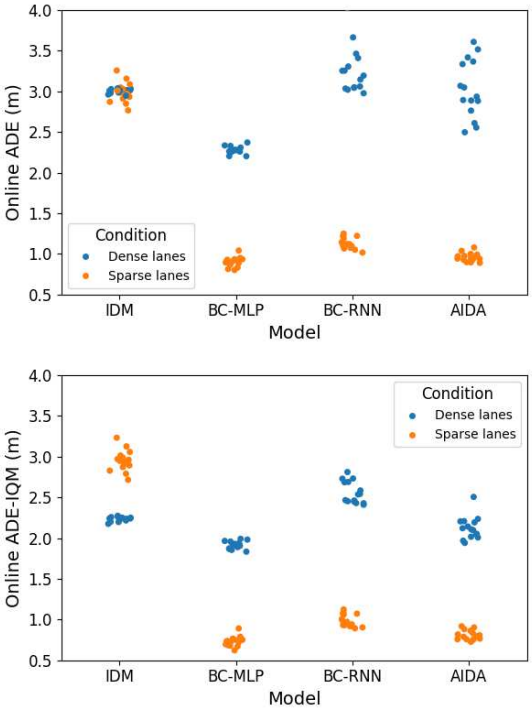


Fig. 12: Comparison of online ADE of each model with IQM (top) and without IQM (bottom).

in the estimated IDM model, the parameter estimate a_{max} is significantly lower than that reported in [74].

F. Statistical testing results

Statistical tests using the two-sided Welch's t-test with 5 percent rejection level are shown in Table V and VI for offline and online evaluations, respectively.