Learning Human Driving Behaviors with Sequential Causal Imitation Learning

Kangrui Ruan, Xuan Di

Columbia University, New York, NY, USA
kr2910@columbia.edu, sharon.di@columbia.edu

Abstract

Learning human driving behaviors is an efficient approach for self-driving vehicles. Traditional Imitation Learning (IL) methods assume that the expert demonstrations follow Markov Decision Processes (MDPs). However, in reality, this assumption does not always hold true. Spurious correlation may exist through the paths of historical variables because of the existence of unobserved confounders. Accounting for the latent causal relationships from unobserved variables to outcomes, this paper proposes Sequential Causal Imitation Learning (SeqCIL) for imitating driver behaviors. We develop a sequential causal template that generalizes the default MDP settings to one with Unobserved Confounders (MDPUC-HD). Then we investigate conditions when ignoring causality leads to poor performances in MDPUC-HD with a sufficient graphical criterion. Through the framework of Adversarial Imitation Learning, we develop a procedure to imitate the expert policy by blocking π-backdoor paths at each time step. Our methods are evaluated on a synthetic dataset and a real-world highway driving dataset, both demonstrating that the proposed procedure significantly outperforms non-causal imitation learning methods.

1 Introduction

Imitation Learning (IL) presents a promising paradigm for autonomous driving (Pomerleau 1989; Bojarski et al. 2016, 2017; Bansal, Krizhevsky, and Ogale 2018; Codevilla et al. 2019). There are two major types in IL: behavioral cloning (BC) and inverse reinforcement learning (IRL). BC methods directly learn an approximate conditional distribution from any given state to the expert's action. Instead, IRL learns the implicit reward function, which is optimal to the expert behaviors; then using this learned reward function, RL methods are employed to obtain a policy. Inspired by Generative Adversarial Networks (GANs) and IRL (Goodfellow et al. 2014; Gulrajani et al. 2017), the framework of Adversarial Imitation Learning (AIL) is proposed (Ho and Ermon 2016; Li, Song, and Ermon 2017; Fu, Luo, and Levine 2017). In AIL, the policy is trained by producing expert-like state-action pairs to fool the discriminator. While training, the discriminator can provide a reward signal to help the policy proceed to expert-like zones.

However, IL methods rely on the assumption that all expert features are fully observed and recorded. When there exist unobserved variables inside the expert's demonstrations, critical problems arise. Causal Inference (CI) addresses those issues by investigating the causal relationships among observed and unobserved variables (Spirtes et al. 2000; Pearl 2009; Peters, Janzing, and Schölkopf 2017). There has been some research focused on the combination of IL and Causality. Causal Imitation Learning (CIL) was recently proposed by (Zhang, Kumor, and Bareinboim 2020), which focuses on learning a policy within the limit of non-sequential one-stage settings. The authors of (Etesami and Geiger 2020) study the causal transfer problem by assuming that the relationships among variables are linear. The paper (de Haan, Jayaraman, and Levine 2019) ignores unobserved confounders and assumes the reward and the expert can be easily accessible.

So far we found one paper (Kumor, Zhang, and Bareinboim 2021) related to SeqCIL. Their diagrams are within 2-stage settings, and their experiments are mainly based on synthetic datasets with simplified reward functions. However, we work in the γ-discounted infinite horizon setting, and our experiments include a realistic dataset whose actual reward function cannot even be accessed. We also show the internal connection between GAIL and SeqCIL.

Little research has been conducted for sequential imitation learning when unobserved variables exist. To demonstrate the issue of ignoring unobserved variables resulting in an unacceptable policy, we begin with an introductory example.

Example 1. Suppose a scenario when an imitator needs to learn how to correctly accelerate A_t. S_t denotes the velocity and locations of the ego vehicle and the surrounding vehicles. In reality, there exist some unobserved variables inside the expert demonstrations. Human drivers utilize the vehicle light information $U_t^{(L)}$, i.e., the tail light from the front vehicle 1 and the turn signal from the left vehicle 2. However, $U_t^{(L)}$ may not be recorded, which is common in some real-world datasets, such as NGSIM (Alexiadis et al. 2004) or highD (Krajewski et al. 2018). When the demonstrator drove, the road can be slippery $U_t^{(S,R)}$, which is unknown to the imitator. The level of the expert driving skills U_π is ignored. H_{t-1} encodes the history information, which the

imitator does not take into account. The imitator only takes S_t as input, while the expert makes a decision based on $U_\pi, H_{t-1}, S_t, U_t^{(L)}$. The latent reward is evaluated based on $S_t, A_t, U_t^{(S,R)}$. Fig. 1 depicts this situation and its graphical representation.

Consider a numerical instance where variables $A_t, S_t, R_t, H_{t-1}, U_t^{(L)}, U_t^{(S,R)}, U_\pi \in \{0, 1\}$; their values are generated based on the functions: $U_\pi \leftarrow 0$, $S_t \leftarrow H_{t-1} \oplus U_t^{(S,R)}$, $U_t^{(L)} \leftarrow S_t$, the expert policy $\pi_E : A_t \leftarrow U_\pi \oplus H_{t-1} \oplus S_t \oplus U_t^{(L)}$, the implicit reward $R_t \leftarrow \neg(A_t \oplus U_t^{(S,R)} \oplus S_t)$; variables $H_{t-1}, U_t^{(S,R)}$ are uniformly sampled over $\{0, 1\}$; the operator $\oplus$ means *exclusive-or*. The performance of the demonstrator is $\mathbb{E}[R_t] = 1$, which is optimal. However, when ignoring the unobserved variables, the imitator with $\pi(a_t|s_t)$ is only able to achieve $\mathbb{E}[R_t|do(\pi)] = 0.5$. This result shows that even though the imitator $\pi(a_t|s_t)$ can copy the demonstrator's actions given some states, the obtained policy is still not optimal.

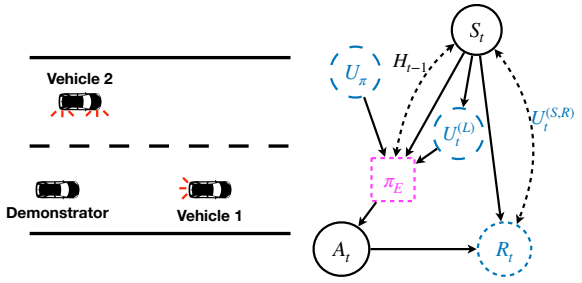

Figure 1: Left: The imitator learns to accelerate like the demonstrator under the same S_t, while the light information $U_t^{(L)}$, the demonstrator level U_π, and the slippery indicator $U_t^{(S,R)}$ are not recorded. Right: Graphical representation for Example 1.

Motivated by this example, we try to explain this phenomenon, and understand how to learn sequential behaviors when unobserved confounders cannot be ignored. We address these issues by Sequential Causal Imitation Learning (SeqCIL) and the framework of Adversarial Imitation Learning (AIL). Our major contributions are summarized as follows:

- We formalize the problem of SeqCIL, and introduce a causal template, MDPUC-HD, which generalizes MDPs with fewer constraints.
- By leveraging causal relationships among observed and unobserved variables, we provide a sufficient graphical criterion to explain why a non-causal imitator fails, and develop a procedure to learn a sequential causal imitator by blocking π-backdoor paths at each step.
- With causal Markov property, we instantiate our SeqCIL framework by GAIL, and show the internal connection between these two techniques.
- We demonstrate the superiority of our proposed method over non-causal baselines by conducting experiments on a synthetic dataset and a real-world highway dataset.

The rest of this paper is organized as follows: Section 2 exhibits the preliminaries and formally defines the problem of SeqCIL. Section 3 presents the proposed causal template for human driving behaviors, i.e., MDPUC-HD. Based on this template, Section 4 explicates why unobserved variables cannot be ignored, and provides the sequential causal imitator to address such issues. We propose to solve SeqCIL through the framework of AIL in Section 5, and experiments are conducted in Section 6.

2 Sequential Causal Imitation Learning

Conventions: In this paper, capitalized letters represent random variables, e.g. R and lowercase letters are their specific values, e.g. r; the probability distribution of a random variable is capital $P(R)$, and $p(r)$ is for the mass at value $R = r$. $\boldsymbol{S}$ denotes a set. D_{KL} represents the Kullback–Leibler (KL) divergence between two distributions. Without explicit remark, all proofs are shown in the appendix of this paper.

2.1 Preliminaries on Imitation Learning

Imitation Learning: $\mathcal{S}$ represents the state space, and $\mathcal{A}$ represents the action space. The expert state space $\mathcal{S}_{\pi_E}$ and the chosen imitator state space $\mathcal{S}_\pi$ can be different[1]. The policy space Π is the set of all stationary stochastic policies that take actions in $\mathcal{A}$ given states in $\mathcal{S}_\pi$, denoted by $\{\pi : \mathcal{S}_\pi \mapsto \mathcal{A}\}$.

Occupancy measure: Under the policy π, the state-action occupancy measure $\rho_\pi(s, a) : \mathcal{S}_\pi \times \mathcal{A} \to \mathbb{R}$ is defined as $\rho_\pi(s, a) = \rho(s, a|do(\pi)) = \rho(s, a; \pi) = \rho_\pi(s)\pi(a|s)$, where $\rho_\pi(s)$ denotes the unnormalized discounted future state distribution when following the policy π: $\rho_\pi(s) = \sum_{t=0}^{\infty} \gamma^t p_\pi(s_t = s)$.

Expert Regime: π_E always refers to the expert policy. $P_{\pi_E}(\cdot)$ denotes the distribution for any feasible random variable under the expert π_E regime[2]. $P(\cdot)$ denotes the observational distribution ("demonstrations"). In this paper, demonstrations are generated by the expert, so that $P_{\pi_E}(\cdot) = P(\cdot)$.

Imitator Regime: $P_\pi(\cdot)$ denotes the distribution under the imitator π regime. Generally, $P_\pi(\cdot) \neq P(\cdot)$, except the situations when some graphical conditions are satisfied. (More details can be found in Appendix Sec. A)

Adversarial Imitation Learning (AIL): AIL learns a policy by matching occupancy measures between the expert and the imitator, and its training procedure includes updating the discriminator and the policy simultaneously. Generative Adversarial Imitation Learning (GAIL) (Ho and Ermon 2016) minimizes the Jensen-Shannon divergence between ρ_π and ρ_{π_E}. However, when using it, non-smooth distance and mode collapse issues commonly occur (Gulrajani et al. 2017; Li, Song, and Ermon 2017). To address such

[1] $\mathcal{S}_\pi$ is manually chosen. People usually choose $\mathcal{S}_\pi$ the same as $\mathcal{S}_{\pi_E}$, but they can be different.

[2] To illustrate the idea of "regime", we take $\rho_{\pi_1}(s)$ and $\rho_{\pi_2}(s)$ as an example. $\rho_{\pi_1}(s)$ is under policy π_1 regime. $\rho_{\pi_2}(s)$ is under policy π_2 regime. We distinguish different regimes, because different policies generally lead to different state distributions, i.e., $\rho_{\pi_1}(s) \neq \rho_{\pi_2}(s)$. (More details are in Appendix Sec. A.)

issues, InfoGAIL (Li, Song, and Ermon 2017) proposes utilizing the Wasserstein distance (Arjovsky, Chintala, and Bottou 2017), i.e., $\min_\pi \max_D \mathbb{E}_{\pi_E}[D(s,a)] - \mathbb{E}_\pi[D(s,a)] - \lambda H(\pi)$, where π denotes the policy, D is the discriminator, and $H(\pi)$ is the γ-discounted causal entropy of the policy π.

2.2 Preliminaries on Causality

Structural Causal Models (SCMs) (Pearl 2009; Spirtes et al. 2000) A Structural Causal Model $\mathcal{M}$ is denoted as a tuple $(\boldsymbol{U}, \boldsymbol{V}, \mathcal{F}, P(\boldsymbol{U}))$, where $\boldsymbol{U}$ denotes a set of exogenous variables; $\boldsymbol{V}$ denotes a set of endogenous variables; $\mathcal{F}$ is a set of structural functions, and each f_i is mapping from U_i and its parents to V_i, $V_i \leftarrow f_{V_i}(Parent_{V_i}, U_{V_i})$; $P(\boldsymbol{U})$ denotes the joint distribution which generates the values of U.

Intervention and $\mathrm{do}(\cdot)$ operator: Intervention answers the question "what if I do X?" (Bareinboim et al. 2020). $\mathrm{do}(\cdot)$ operator defines this action in a probabilistic way w.r.t. any random variable in $\boldsymbol{V}$. Specifically, $\mathrm{do}(\boldsymbol{X} = \boldsymbol{x})$ intervenes all values of $\boldsymbol{X}$ to $\boldsymbol{x}$, by replacing the original functions $f_{\boldsymbol{X}}$.

π-backdoor paths at step t: We generalize the single-stage π-backdoor criterion (Zhang, Kumor, and Bareinboim 2020) to a sequential setting. Given a causal diagram $\mathcal{G}$ and a policy space Π, and the paths $A_t \rightarrow R_t, S_t \rightarrow R_t$ exist inside $\mathcal{G}$, a set $\boldsymbol{S}_{t,causal}$ is said to satisfy the π-backdoor criterion with respect to $\langle \mathcal{G}, \Pi \rangle$ **if and only if** $\pi(\cdot|\boldsymbol{s}_{t,causal}) \in \Pi$ and $(R_t \perp\!\!\!\perp A_t | \boldsymbol{S}_{t,causal})_{\mathcal{G}_{\underline{A_t}}}$, which is called the π-backdoor admissible set at step t with respect to $\langle \mathcal{G}, \Pi \rangle$.

2.3 Problem Formulation

Human drivers make sequential decisions with varying temporal duration. It is impossible to imitate a good policy restricted to single-stage decision-making. There should usually be a reward after each time step, e.g., whether collided or driving off-road or not. First, we formally define the problem of Sequential Causal Imitation Learning (SeqCIL) in general: *given a causal diagram $\mathcal{G}$ and a policy space Π, by choosing a set $\boldsymbol{S}_{\mathrm{causal}}$ w.r.t. $\langle \mathcal{G}, \Pi \rangle$, the sequential causal imitator is able to learn a stationary policy $\pi(a \mid \boldsymbol{s}_{\mathrm{causal}})$, whose performance matches the performance of the expert, i.e.,*

$$\underbrace{\mathbb{E}\left[\sum_t \gamma^t R_t \mid \mathrm{do}(\pi)\right]}_{\text{expected discounted return of } \pi} = \underbrace{\mathbb{E}\left[\sum_t \gamma^t R_t\right]}_{\text{expected discounted return of } \pi_E} \tag{1}$$

where the l.h.s. characterizes the performance measure under the imitator π regime, and the r.h.s. denotes the observational distribution ("demonstrations") generated by π_E.

3 Causal Template for Human Driving Behaviors

In this paper, we propose a causal template: Markov Decision Processes with Unobserved Confounders for Human Driving (MDPUC-HD), depicted in Fig. 2.

When using real-world driving datasets, it is inappropriate to assume that expert covariates are fully observed. Similar to Example 1, U_t^L accounts for the missing light information in real-world driving datasets. U_π represents the internal latent code, e.g., levels of driving skills and inconsistent policies. Additionally, human drivers usually take actions based on previous actions, suggesting a cause-effect relationship between A_{t-1} and A_t (de Haan, Jayaraman, and Levine 2019; Codevilla et al. 2019). U_t^A denotes those unknown actions which affect the environment dynamics, e.g., the actions from other vehicles. $U_t^{(S,R)}$ are the unobserved variables that affect both the reward and the state. The expert policy $\pi_E(a_t|a_{t-1}, u_\pi, s_t, u_t^L)$ is a conditional distribution which is unknown to the imitator. The implicit reward R_t is decided by $S_t, A_t, U_t^{(S,R)}$. The observability of each mentioned variable is summarized in Table 1.

Variables	Expert	Imitator
U_π	known	unknown
U_t^L	known	unknown
$U_t^{(S,R)}$	known	unknown
U_t^A	unknown	unknown
R_t	known	unknown

Table 1: The observability of variables for the expert and the imitator

Compared to an MDP, an MDPUC-HD is more general and realistic to practical applications: (1) An MDPUC-HD **does not** assume that the expert covariates are fully observed, because of U_t^L and U_π; (2) In an MDPUC-HD, R_t can depend on variables not inside $\mathcal{S}_{\pi_E}$ or $\mathcal{A}$, i.e., $U_t^{(S,R)}$; (3) The transition dynamics may not be successfully recovered, because of the unobserved variables U_t^A and $U_t^{(S,R)}$; (4) Following a MDPUC-HD, the dataset does not need to satisfy the constraint $A_t \perp\!\!\!\perp (S_{t-1}, A_{t-1})|S_t$. While following MDPs, the dataset should meet this constraint. In summary, the proposed causal template is a generalized version of MDPs, which imposes fewer constraints.

MDPUC-HDs are orthogonal and complementary to POMDPs with different focuses (Zhang and Bareinboim 2016). MDPUC-HDs concentrate on the causal relationships. It is possible that unobserved confounders still exist in POMDPs. In MDPUC-HDs, not all expert inputs are observed, e.g., U_t^L. In POMDPs, the actual states are hidden, and all decisions are made according to the entire history, i.e., POMDPs are non-Markovian. However, the crucial Causal Markov property, as stated below, still holds in MDPUC-HDs.

3.1 Causal Markov Property

Proposition 1 (Causal Markov Property[3])**.** Following the causal diagram $\mathcal{G}$ in Fig. 2, the following statements hold:

1. Given the current state S_t and action A_t, the transition to

[3] All strict proofs are provided in Appendix.

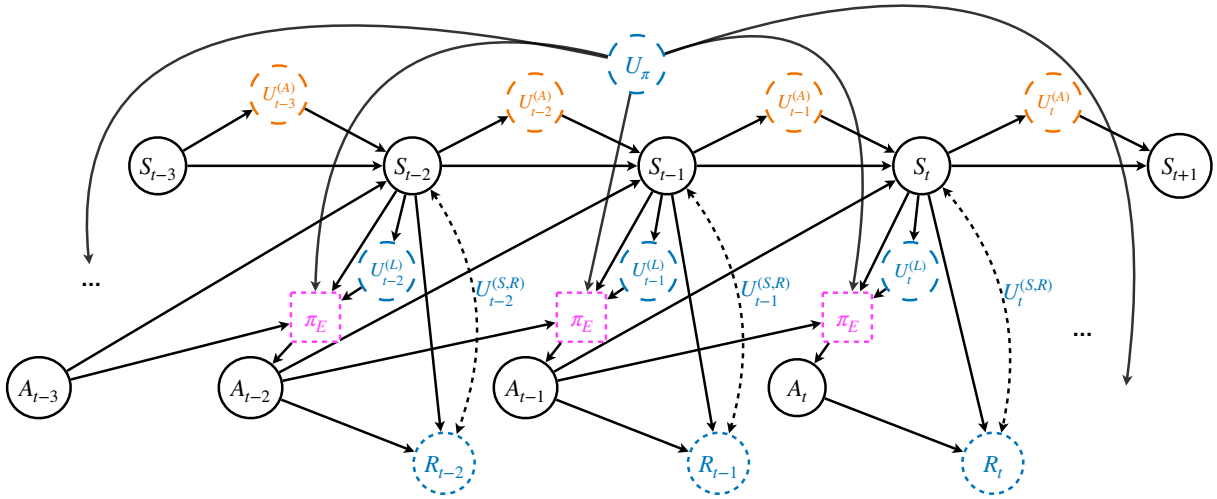

Figure 2: Proposed causal template for SeqCIL. $U^{(A)}$ is unknown to both the expert and the imitator (Orange Border). Blue shapes are variables known to the expert but not to the imitator. Squared node π_E denotes an implicit conditional distribution (Magenta). Directed arrows are solid, and bi-directed arrows are dashed.

the next state S_{t+1} is independent of the past history:

$$\begin{aligned} &P(S_{t+1}|S_t, A_t, S_{t-1}, A_{t-1}, \ldots, S_0, A_0) \\ =&P(S_{t+1}|S_t, A_t) \end{aligned} \tag{2}$$

2. Given the 2-step information tuple $(S_t, A_t, S_{t-1}, A_{t-1})$, the reward R_t is independent of the past history:

$$\begin{aligned} &P(R_t|S_t, A_t, S_{t-1}, A_{t-1}, \ldots, S_0, A_0) \\ =&P(R_t|S_t, A_t, S_{t-1}, A_{t-1}) \end{aligned} \tag{3}$$

In general, Prop. 1 is valid for a series of graphs:

Proposition 2. Given the causal diagram $\mathcal{G}$ following Fig. 2, removing any bi-directed arrow or unobserved variable, Causal Markov Property (Prop. 1) is still tenable.

Prop. 2 is intuitive: after removing any bi-directed arrow or unobserved variable, the new graph still satisfies the original (conditional) independent constraints.

4 Sequential Causal Imitator

With Prop. 1 and 2 defined above, in this section, we first explain why the non-causal imitator fails, and then propose a solution using the sequential causal imitator.

4.1 Why Non-causal Imitator Fails

To analyze the failure reason for the non-causal imitator, we begin with Example 2, which shares the similar driving setting as Example 1.

Example 2. Consider an instance where demonstrations follow Fig. 2. When the history information $\{S, A, U^A, U^{(L)}, U^{(S,R)}\}_{0:t-1}$ is summarized as $\boldsymbol{H}_{t-1}$, the right part in Fig. 1 encodes the similar causal relationships among $S_t, A_t, U^{(L)}_t, R_t$ as in Fig. 2.

Variables $A_t, S_t, R_t, U^{(S,R)}_t, U^{(A)}_t, U^{(L)}_t \in \{0, 1\}$; their values are generated by processes: $U_\pi \leftarrow 0$, $U^{(S,R)}_t \sim Bern(0.5)$, $U^{(L)}_t \leftarrow S_t$, $U^{(A)}_t \leftarrow S_t$; at $t = 0$, $S_0 \leftarrow U^{(S,R)}_0$. Later, when $t \geq 1$, S_t is determined by the function: $S_t \leftarrow U^{(A)}_{t-1} \oplus S_{t-1} \oplus A_{t-1} \oplus U^{(S,R)}_t$. The reward R_t is defined as $R_t \leftarrow \neg(S_t \oplus A_t \oplus U^{(S,R)}_t)$. The implicit expert policy π_E is defined as:

$$A_t \leftarrow \begin{cases} S_0 \oplus U^{(L)}_0 \oplus U_\pi, & \text{if } t = 0 \\ A_{t-1} \oplus S_t \oplus U^{(L)}_t \oplus U_\pi, & \text{otherwise} \end{cases} \tag{4}$$

; the operator $\oplus$ means *exclusive-or*. Non-causal imitator $\pi(a_t|s_t)$, is only capable of achieving sub-optimal performance.

When unobserved confounders exist, as Example 2 demonstrates, the non-causal imitator $\pi(a_t|s_t)$ fails to recover a good policy. The underlying reason is related to the π-backdoor paths between R_t and A_t: (1) $A_t \leftarrow S_t \rightarrow R_t$. (2) $A_t \leftarrow A_{t-1} \rightarrow S_t \leftrightarrow R_t$, where S_t is a collider. When S_t is conditioned, path (1) is blocked but the colliding path (2) is open, so that some spurious correlation is learned. Formally, we prove a generalized theorem to analyze the criterion when the non-causal imitator fails:

Theorem 1 (Not Imitable in the Same C-Component)**.** *Given a causal diagram $\mathcal{G}$ and a policy space Π, at time step t, if π_E and R_t are in the same C-Component of R_t's ancestral graph, then there exist two models M_1, M_2, which have the same distribution of observed variables, but there exists no policy $\pi \in \Pi$ such that $P(R_t; M_1) = P(R_t|do(\pi); M_1)$ and $P(R_t; M_2) = P(R_t|do(\pi); M_2)$, i.e., $P(R_t)$ is not imitable.*

C-Component (Tian 2002) is a set of nodes that are connected by bi-directed arrows. Here we use a fake node X as an example. A directed path from a node X to R_t is a path composed of directed edges ($X \rightarrow \cdots \rightarrow R_t$). If there is a directed path from a node X to R_t or $X = R_t$, then this node

X is an ancestor of R_t. R_t's ancestral graph is composed of nodes that are ancestors of R_t. To illustrate the idea of the same C-Components of R_t's ancestral graph, take S_t as example. S_t is connected with R_t by bi-directed arrows, and there is a directed path from S_t to R_t.

In Example 1, π_E and S_t are R_t's ancestors because of paths $\pi_E \to A_t \to R_t$ and $S_t \to R_t$, and the path $\pi_E \leftrightarrow S_t \leftrightarrow R_t$ makes π_E and R_t in the same C-Components. Therefore, π_E and R_t are in the same C-Component of R_t's ancestral graph, and the non-causal imitator $\pi(a_t|s_t)$ fails. Intuitively, Thm. 1 also suggests that if more variables are observed, less confounding effect exists. This result can causally justify why more sensors are required to obtain a better policy, when recording human driving demonstrations (Jeyachandran 2020).

4.2 Sequential Causal Imitator

In this section, we circumvent the above issue by exploiting topological graphical conditions.

Definition 1 (Sequential Causal Imitator)**.** Given a causal diagram $\mathcal{G}$ following Fig. 2 and a policy space Π, $\pi(a_t|\boldsymbol{s}_{t,causal})$ is called "sequential causal imitator" with respect to $\langle \mathcal{G}, \Pi \rangle$, if the set $\boldsymbol{S}_{t,causal}$ blocks all π-backdoor paths between R_t and A_t, i.e., $(R_t \perp\!\!\!\perp A_t|\boldsymbol{S}_{t,causal})_{\mathcal{G}_{\underline{A_t}}}$.

Although there exist diverse sets that can block all π-backdoor paths between R_t and A_t, we only consider the minimal set[4]. Later, we will show that a larger set does not necessarily lead to a better performance.

Below, two propositions show the warrant for the sequential causal imitator with a satisfactory performance in multiple graphs.

Proposition 3. Given the causal diagram $\mathcal{G}$ following Fig. 2 and a policy space Π, the sequential causal imitator $\pi(a_t|\boldsymbol{s}_{t,causal})$ is guaranteed to match the performance of the expert, $p_\pi(r_t) = p(r_t)$, at time step t. (The strict proof is in Appendix.)

Prop. 3 shows: without knowing the expert policy π_E, the sequential causal imitator $\pi(a_t|\boldsymbol{s}_{t,causal})$ can still achieve $p_\pi(r_t) = p(r_t)$. Prop. 4 generalizes it to graphs with fewer bi-directed arrows or unobserved variables.

Proposition 4. Given the causal diagram $\mathcal{G}$ following Fig. 2, removing any bi-directed arrow or unobserved variable, Prop. 3 is still tenable.

Consider Example 2 again, denote the set $\{S_t, A_{t-1}, S_{t-1}\}$ by the set $\boldsymbol{S}_{t,causal}$, which blocks all π-backdoor paths between R_t and A_t. The causal relationships among S_t, A_{t-1}, S_{t-1} are depicted in Fig. 3. With Prop. 3, the sequential causal imitator $\pi(a_t|\boldsymbol{s}_{t,causal})$ is able to obtain the optimal expert performance $\mathbb{E}[R_t|do(\pi)] = 1$, for $t \geq 1$.

[4]A set is considered to be minimal when there does not exist a subset which can block all π-backdoor paths between R_t and A_t w.r.t. $\langle \mathcal{G}, \Pi \rangle$.

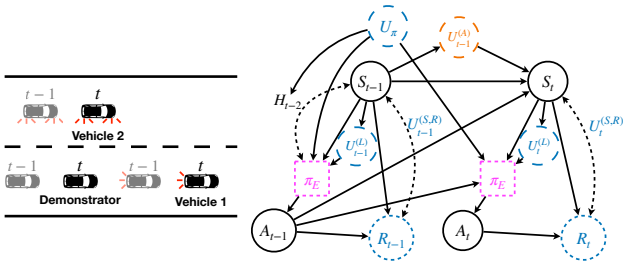

Figure 3: Left: The grey vehicles represent S_{t-1}. The imitator now has access to $\{S_t, A_{t-1}, S_{t-1}\}$. Right: Graphical representation for Example 2, where $\boldsymbol{H}_{t-2} = \{S, A, U^A, U^{(L)}, U^{(S,R)}\}_{0:t-2}$.

5 Adversarial Imitation Learning

To obtain the sequential causal imitator, we propose to use the framework of AIL. Allowing the agent to explore the environment, AIL addresses the issue of "covariate shift" (Ross, Gordon, and Bagnell 2011). Apart from that, AIL facilitates the developed SeqCIL to explicitly match the performances of the expert and the imitator. To demonstrate how to leverage AIL for SeqCIL, we will implement GAIL below.

f_r represents the implicit reward function, which takes $s_t, a_t, u_t^{(S,R)}$ as input. Although $U_t^{(S,R)}$ affects the true reward R_t and the reward function f_r is unknown, a surrogate reward signal can still be studied from demonstrations. With $\boldsymbol{S}_{t,causal}$ equal to $\{S_t, A_{t-1}, S_{t-1}\}$, Prop. 1 implies $R_t \perp\!\!\!\perp S_{t-2}, A_{t-2}, \ldots, S_0, A_0|(\boldsymbol{S}_{t,causal}, A_t)$. That is, given $(\boldsymbol{S}_{t,causal}, A_t)$, the history $(S_{t-2}, A_{t-2}, \ldots, S_0, A_0)$ does not tell any newer information related to the reward R_t. This suggests that $(\boldsymbol{S}_{t,causal}, A_t)$ can be treated as a proxy for the whole trajectory as the input to the discriminator (D).

We utilize the Wasserstein distance (Arjovsky, Chintala, and Bottou 2017) for occupancy measure matching between different regimes. When the discriminator cannot distinguish the causal state-action pairs coming from the imitator π or the expert π_E, the occupancy measure is matched, as summarized in Prop. 5. The objective of the generator and the discriminator is given by:

$$\min_\pi \max_D \mathbb{E}_{\pi_E}\left[D(\boldsymbol{s}_{causal}, a)\right] - \mathbb{E}_\pi\left[D(\boldsymbol{s}_{causal}, a)\right] - \lambda H(\pi) \tag{5}$$

where π denotes the policy of the sequential causal imitator, D is the discriminator taking causal state-action pair $(\boldsymbol{s}_{causal}, a)$ as input. $H(\pi)$ is the γ-discounted causal entropy of the policy π, defined as $H(\pi) \triangleq \mathbb{E}_\pi[-\log \pi(a \mid \boldsymbol{s}_{causal})]$ (Bloem and Bambos 2014). When updating the parameters of the policy, $r'_t = D(s_{t,causal}, a_t)$ plays like a proxy reward signal.

Proposition 5. When the discriminator D cannot distinguish $(\boldsymbol{s}_{causal}, a)$ generated from the sequential causal imitator π or the expert π_E, $\rho_\pi(\boldsymbol{s}_{causal}, a)$ matches $\rho_{\pi_E}(\boldsymbol{s}_{causal}, a)$, and $\pi(a|\boldsymbol{s}_{causal})$ matches $p_{\pi_E}(a|\boldsymbol{s}_{causal})$.

The convergence of the discriminator indicates the performance match, described in Eq. (1).

Under the imitator π regime, the relationship between l.h.s. of Eq. (1) and the occupancy measure $\rho_\pi\left(\boldsymbol{s}_{causal}, a\right)$ is given by:

$$\begin{aligned}
&\mathbb{E}\left[\sum_t \gamma^t R_t \mid \mathrm{do}(\pi)\right] \\
&= \sum_{\boldsymbol{s}_{\mathrm{causal}}, a, u^{(S,R)}} p_\pi(\boldsymbol{s}_{causal}, a, u^{(S,R)}) f_r(s, a, u^{(S,R)}) \\
&= \sum_{\boldsymbol{s}_{\mathrm{causal}}, a, u^{(S,R)}} \underbrace{\rho_\pi\left(\boldsymbol{s}_{causal}, a\right)}_{\text{occupancy measure of } \pi} p_\pi(u^{(S,R)} | \boldsymbol{s}_{causal}, a) \\
&\quad \cdot f_r(s, a, u^{(S,R)})
\end{aligned} \tag{6}$$

With Eq. (6), we will prove Prop. 5 in Appendix.

Proximal Policy Optimization (PPO). PPO is a model-free on-policy algorithm (Schulman et al. 2017). There are two principal variants: PPO-penalty and PPO-clip. PPO-penalty turns the trust-region constraint into a penalty $D_{\mathrm{KL,forward}}(\pi_{\mathrm{old}} \| \pi_\theta)$ to approximately restrict the size of the update. Instead, PPO-clip clips the probability ratio directly via:

$$\begin{aligned}
\mathcal{L}_{\mathrm{CLIP}}(\theta) &= \mathop{\mathbb{E}}_{a,s\sim\pi_{\mathrm{old}}}\left[\min\left(\mathrm{surrogate}_1, \mathrm{surrogate}_2\right)\right] \\
\mathrm{surrogate}_1 &= \frac{\pi_\theta(a \mid s)}{\pi_{\mathrm{old}}(a \mid s)} A^{\pi_{\mathrm{old}}}(s, a) \\
\mathrm{surrogate}_2 &= \mathrm{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\mathrm{old}}(a \mid s)}, 1-\epsilon, 1+\epsilon\right) A^{\pi_{\mathrm{old}}}(s, a)
\end{aligned} \tag{7}$$

where ϵ is a hyper-parameter to control the clip value, π_{old} denotes the policy prior to the update, $A^{\pi_{\mathrm{old}}}$ denotes the advantage function for the policy π_{old}.

Suggested by (Hsu, Mendler-Dünner, and Hardt 2020), to better keep the new policy near the old policy and avoid common failure modes, our PPO objective is formulated as:

$$\begin{aligned}
\mathcal{L}_{\mathrm{PPO}}(\theta) = &\mathcal{L}_{\mathrm{CLIP}}(\theta) \\
&- \beta_1 D_{\mathrm{KL,forward}}(\pi_{\mathrm{old}} \| \pi_\theta) \\
&- \beta_2 D_{\mathrm{KL,reverse}}(\pi_\theta \| \pi_{\mathrm{old}})
\end{aligned} \tag{8}$$

where $\mathcal{L}_{\mathrm{CLIP}}(\theta)$ is defined in Eq. (7) and D_{KL} represents the KL divergence between two distributions, which is asymmetric. We use two soft constraints because generally $D_{\mathrm{KL,forward}}(\pi_{\mathrm{old}} \| \pi_\theta) \neq D_{\mathrm{KL,reverse}}(\pi_\theta \| \pi_{\mathrm{old}})$.

To summarize, Algorithm 1 reveals the training procedure of how to obtain a sequential causal imitator through GAIL. The convergence of the discriminator can be interpreted as the explicit signal that the performance is finally matched. Similar to the previous research (Ho and Ermon 2016; Fu, Luo, and Levine 2017), we implement the policy and the discriminator with Neural Networks and use Monte-Carlo methods to approximate expectations.

Algorithm 1: Finding $\pi(a|s_{causal})$ by GAIL

Input: $\mathcal{G}$, Π, Expert demonstrations τ_E, initial parameters θ_0 for policy and ψ_0 for discriminator

1: **for** iteration $i = 0, 1, 2, \ldots$ **do**
2: Collect trajectories τ_i from current policy π_{θ_i}
3: Update the discriminator D_{ψ_i} based on Eq. (5) with training data sampled from τ_E and τ_i
4: Train the policy parameters from θ_i to θ_{i+1} by maximizing Eq. (8) with the causal entropy of the policy using PPO
5: **end for**

6 Experiments

We conduct experiments mainly on two classic driving datasets: a synthetic dataset simulating drivers' car-following behaviors and a real-world highway driving dataset, Next Generation Simulation (NGSIM) (Alexiadis et al. 2004). Our experiments seek to answer the following questions: **(1)** Is the sequential causal imitator better than the non-causal imitator? **(2)** Is the proposed method robust to the real-world dataset, even without the ground truth reward function? **(3)** Will more temporal information, e.g., 3-step GAIL or recurrent policy, show a better result?

6.1 Synthetic Car-Following Dataset

Car-following is one of the most common and frequent scenarios in reality. The primary mission of car-following is to maintain safety, efficiency, and comfort by controlling the longitudinal dynamics, i.e., accelerating or braking.

State. In this experiment, the properties of the leading vehicle and ego vehicle are initialized randomly. At each time step t, the state S_t is composed of the velocity of the ego vehicle, the velocity of the leading vehicle, and the gap between them, which is the same as the observation of the Intelligent Driver Model (IDM) (Treiber, Hennecke, and Helbing 2000).

Action. The action A_t is the acceleration[5] of the ego vehicle, which is continuous.

Reward. To simulate the complex real-world driving target, we need to design the reward by meeting the demands for safety, comfort, and efficiency. Safety is the major challenge. Therefore, we make use of time-to-collision (TTC), which measures the risk of collision if the speed difference of two vehicles is maintained (Hayward 1972). To quantify the level of human comfort when driving, we use jerk, the first derivative of acceleration (Bellem et al. 2018). For efficiency, to avoid driving too slow, the vehicles need to maintain a proper time headway. Finally, the reward R_t is also dependent on $U_t^{(S,R)}$.

Unobserved Variables. $U_t^{(A)}$ denotes the acceleration of the leading vehicle, which affects the state transition but is unknown to both the expert and the imitator. $U_t^{(L)}$ represents the tail light indicator, when the leading vehicle brakes $U_t^{(L)} := 1$, otherwise 0. When the road is slippery,

[5]When the action is negative, it is the deceleration.

$U_t^{(S,R)} = 1$, only the TTC (safety) part of the reward is concerned, and the state s_t is also affected.

Expert demonstrations are generated by running PPO-clip on the ground truth reward, which is similar to the settings of GAIL (Ho and Ermon 2016). During imitation learning, the environment reward is not provided anymore.

Performance Metrics: Similar to the settings in IL, we compare the average cumulative reward directly.

Results: In Fig. 4, causal 2-step GAIL is much better than the well-established baselines.

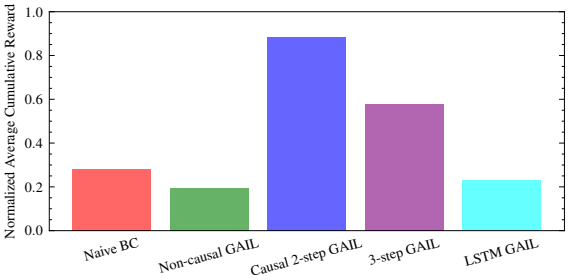

Figure 4: Average cumulative reward for the car-following dataset. The result values are normalized so that 1.0 represents the performance of demonstrations.

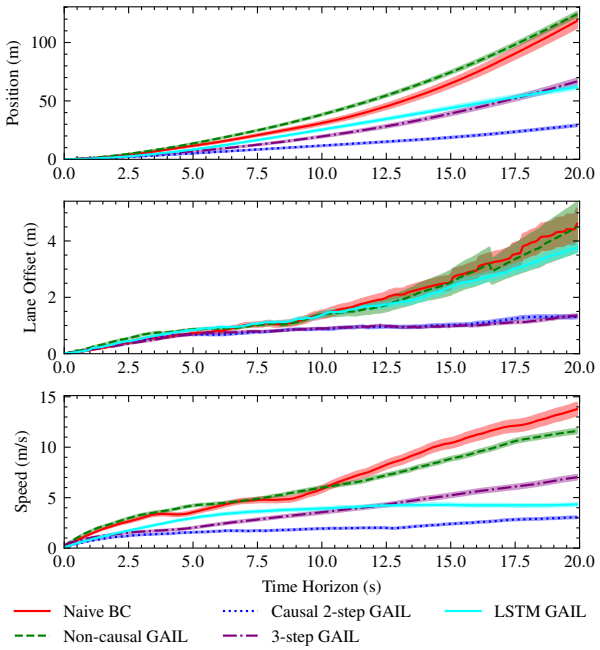

Figure 5: The root mean squared error calculated for each variable v.s. time horizon. The sequential causal imitator better captures how the expert drives.

6.2 Real-world NGSIM Dataset

The public NGSIM data records the real-world human driving trajectories for US Highway 101, and Interstate 80 Freeway (Alexiadis et al. 2004). Each dataset contains a total of 45 minutes of trajectories recorded per 0.1s, including precise information for location, speed, acceleration, surrounding vehicles. We focus on US Highway 101 here.

To offer a realistic driving environment, we adapt a Julia-based NGSIM simulator (Kuefler et al. 2017; Bhattacharyya et al. 2018). In the beginning, the environment is reset arbitrarily to choose a frame from NGSIM data. Inside this chosen frame, the ego-vehicle is randomly selected among all vehicles. Other traffic participants follow the replayed trajectory safely with proper extra braking mechanisms to avoid a potential collision with the ego vehicle.

State. At each time step t, the state S_t consists of the following types of features extracted from the NGSIM data: (1) The features for the ego-vehicle, such as the width and the length, local lane curvature, the distance to the left and the right lane markers. (2) Temporal features, e.g., the time gap and time-to-collision (TTC). (3) Information of surrounding vehicles. Specifically speaking, multiple LiDAR beams are sent out by the ego vehicle and detect first struck objects with the information of the relative position and the range rate. (4) The information for the vehicle ahead of the fore vehicle. The simulator is done whenever the ego vehicle collides, drives off-road, or drives backward.

Action. In this experiment, longitudinal and lateral motions are based on the dynamics of a bicycle model. The action A_t includes the acceleration and turn rate.

Performance Metrics: It is infeasible to directly compare the cumulative reward because we do not know the ground truth reward function when people drive. Additionally, there are infinitely many reward functions to make the expert demonstrations optimal, because the IRL problem is ill-posed (Nguyen, Low, and Jaillet 2015). Therefore, trajectories are used as a proxy to check. To characterize the difference, we make use of Root Mean Squared Error (RMSE), KL-divergence, and undesirable behaviors (Kuefler et al. 2017; Bhattacharyya et al. 2018).

RMSE. Suppose the imitator generates m trajectories. Each predicted trajectory corresponds to one ground truth trajectory. v is the variable of interest we want to compare. $v_{\pi,t}^{(i)}$ denotes the simulated value executing the policy π in the ith trajectory at time step t. $v_{\pi_E,t}^{(i)}$ denotes the true value generated by the expert. $\mathrm{RMSE}(v)_t$ is defined below:

$$\mathrm{RMSE}(v)_t = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(v_{\pi_E,t}^{(i)} - v_{\pi,t}^{(i)}\right)^2} \tag{9}$$

We extract RMSE for the global position, lane offset, and speed over an entire time horizon up to 20 seconds.

KL divergence. KL divergence is to compare the dissimilarity for the predicted and actual actions distribution. A satisfied policy should share a similar distribution with expert demonstrations over the acceleration and turn rate.

Metrics on undesirable behaviors. Undesirable behaviors include off-road duration, collision rate, and hard braking rate, capturing the feasibility and reliability of the model. The collision rate is the ratio of trajectories when the ego vehicle collides with other vehicles. The hard brake rate calculates the situations when the deceleration is higher than a threshold. Off-road duration exhibits the average number of

time steps when the moving ego vehicle is outside the road.

Models	Acceleration	Turn rate
Naive BC	0.677	0.174
Non-causal GAIL	0.412	0.872
Causal 2-step GAIL	**0.061**	**0.083**
3-step GAIL	0.291	**0.084**
LSTM GAIL	0.088	0.313

Table 2: KL-divergence for actions between each model and the expert.

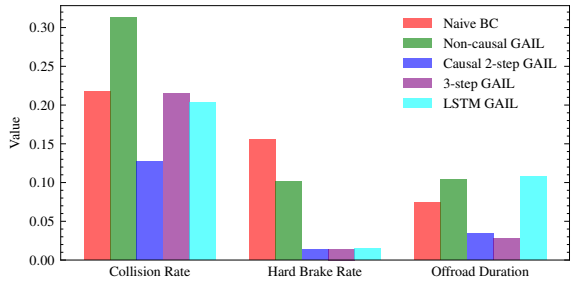

Figure 6: Metrics on undesired behaviors for each model. All expert values are not drawn because their values are zero.

Results: Fig. 5 shows the results of RMSE. For each curve, the line corresponds to the mean value, and the shaded area represents the variance. Causal 2-step GAIL performs best among all models, while 3-step GAIL performs similarly in terms of the RMSE of lane offset. Table 2 shows the results of KL-divergence. Causal 2-step GAIL performs best for both motions, although 3-step GAIL also performs well for the turn rate. The results for undesired behaviors are shown in Fig. 6, demonstrating that causal 2-step GAIL is less likely to have undesired behaviors, and acts more like human beings.

To summarize, the sequential causal imitator learned through AIL performs the best with or without knowing the ground truth reward function, and more temporal information is not guaranteed to give us better results.

7 Conclusions

This paper exploited the qualitative knowledge for human drivers and proposed a causal template, MDPUC-HD. We explained why unobserved variables cannot be ignored for sequential demonstrations, and proposed a principled and general approach to solving SeqCIL with GAIL, with the potential to apply in other domains. The take-away message is: sequential causal imitators can better capture the expert behaviors than non-causal imitators.

Acknowledgment

This work is partially sponsored by the He Research Fund for Artificial Intelligence, Robotics and/or Autonomous Vehicles under Columbia's SEAS Interdisciplinary Research Seed (SIRS).

References

Alexiadis, V.; Colyar, J.; Halkias, J.; Hranac, R.; and McHale, G. 2004. The next generation simulation program. *Institute of Transportation Engineers. ITE Journal*, 74(8): 22.

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein GAN. arXiv:1701.07875.

Bansal, M.; Krizhevsky, A.; and Ogale, A. 2018. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*.

Bareinboim, E.; Correa, J. D.; Ibeling, D.; and Icard, T. 2020. On Pearl's hierarchy and the foundations of causal inference. *ACM Special Volume in Honor of Judea Pearl (provisional title)*, 2(3): 4.

Bellem, H.; Thiel, B.; Schrauf, M.; and Krems, J. F. 2018. Comfort in automated driving: An analysis of preferences for different automated driving styles and their dependence on personality traits. *Transportation research part F: traffic psychology and behaviour*, 55: 90–100.

Bhattacharyya, R. P.; Phillips, D. J.; Wulfe, B.; Morton, J.; Kuefler, A.; and Kochenderfer, M. J. 2018. Multi-agent imitation learning for driving simulation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1534–1539. IEEE.

Bloem, M.; and Bambos, N. 2014. Infinite time horizon maximum causal entropy inverse reinforcement learning. In *53rd IEEE conference on decision and control*, 4911–4916. IEEE.

Bojarski, M.; Del Testa, D.; Dworakowski, D.; Firner, B.; Flepp, B.; Goyal, P.; Jackel, L. D.; Monfort, M.; Muller, U.; Zhang, J.; et al. 2016. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*.

Bojarski, M.; Yeres, P.; Choromanska, A.; Choromanski, K.; Firner, B.; Jackel, L.; and Muller, U. 2017. Explaining how a deep neural network trained with end-to-end learning steers a car. *arXiv preprint arXiv:1704.07911*.

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

de Haan, P.; Jayaraman, D.; and Levine, S. 2019. Causal confusion in imitation learning. *Advances in Neural Information Processing Systems*, 32: 11698–11709.

Etesami, J.; and Geiger, P. 2020. Causal transfer for imitation learning and decision making under sensor-shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 10118–10125.

Fu, J.; Luo, K.; and Levine, S. 2017. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. 2017. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*.

Hayward, J. C. 1972. NEAR-MISS DETERMINATION THROUGH USE OF A SCALE OF DANGER. *Highway Research Record*.

Ho, J.; and Ermon, S. 2016. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29: 4565–4573.

Hsu, C. C.-Y.; Mendler-Dünner, C.; and Hardt, M. 2020. Revisiting design choices in proximal policy optimization. *arXiv preprint arXiv:2009.10897*.

Jeyachandran, S. 2020. Introducing the 5th-generation Waymo Driver: Informed by experience, designed for scale, engineered to tackle more environments. *Waymo LLC, březen*.

Krajewski, R.; Bock, J.; Kloeker, L.; and Eckstein, L. 2018. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2118–2125. IEEE.

Kuefler, A.; Morton, J.; Wheeler, T.; and Kochenderfer, M. 2017. Imitating driver behavior with generative adversarial networks. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, 204–211. IEEE.

Kumor, D.; Zhang, J.; and Bareinboim, E. 2021. Sequential Causal Imitation Learning with Unobserved Confounders. Technical report, Technical Report R-76, Causal AI Lab.

Li, Y.; Song, J.; and Ermon, S. 2017. Infogail: Interpretable imitation learning from visual demonstrations. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 3815–3825.

Nguyen, Q. P.; Low, B. K. H.; and Jaillet, P. 2015. Inverse Reinforcement Learning with Locally Consistent Reward Functions. *Advances in Neural Information Processing Systems*, 28: 1747–1755.

Pearl, J. 2009. *Causality*. Cambridge university press.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Pomerleau, D. A. 1989. Alvinn: An autonomous land vehicle in a neural network. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA ARTIFICIAL INTELLIGENCE AND PSYCHOLOGY.

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

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

Spirtes, P.; Glymour, C. N.; Scheines, R.; and Heckerman, D. 2000. *Causation, prediction, and search*. MIT press.

Tian, J. 2002. *Studies in causal reasoning and learning*. University of California, Los Angeles.

Treiber, M.; Hennecke, A.; and Helbing, D. 2000. Congested traffic states in empirical observations and microscopic simulations. *Physical review E*, 62(2): 1805.

Zhang, J.; and Bareinboim, E. 2016. Markov decision processes with unobserved confounders: A causal approach. Technical report, Technical Report R-23, Purdue AI Lab.

Zhang, J.; Kumor, D.; and Bareinboim, E. 2020. Causal imitation learning with unobserved confounders. *Advances in neural information processing systems*, 33.