Intent-Aware Planning in Heterogeneous Traffic via Distributed Multi-Agent Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract: Navigating safely and efficiently in dense and heterogeneous traffic scenarios is challenging for autonomous vehicles (AVs) due to their inability to infer the behaviors or intentions of nearby drivers. In this work, we propose a distributed multi-agent reinforcement learning (MARL) algorithm with trajectory and intent prediction in dense and heterogeneous traffic scenarios. Our approach for intent-aware planning, iPLAN, allows agents to infer nearby drivers' intents solely from their local observations. We model two distinct *incentives* for agents' strategies: *Behavioral Incentive* for agents' long-term planning based on their driving behavior or personality; *Instant Incentive* for agents' short-term planning for collision avoidance based on the current traffic state. We design a two-stream inference module that allows agents to infer their opponents' incentives and incorporate their inferred information into decision-making. We perform experiments on two simulation environments, Non-Cooperative Navigation and Heterogeneous Highway. In Heterogeneous Highway, results show that, compared with centralized MARL baselines such as QMIX and MAPPO, our method yields a 4.0% and 35.7% higher episodic reward in *mild* and *chaotic* traffic, with 48.1% higher success rate and 80.6% longer survival time in *chaotic* traffic. We also compare with a decentralized baseline IPPO and demonstrate a higher episodic reward of 9.2% and 10.3% in *mild* traffic and *chaotic* traffic, 25.3% higher success rate, and 13.7% longer survival time.

Keywords: Autonomous Driving, Multi-agent Reinforcement Learning, Representation Learning

1 Introduction

Autonomous vehicles (AVs) face numerous challenges when navigating through dense and heterogeneous traffic scenarios. One of the major difficulties in such scenarios is accurately inferring the behaviors and intentions of diverse traffic agents [1]. Agents can differ in their type, such as two-wheelers, three-wheelers, cars, buses, or trucks, or they can vary in their driving behavior, ranging from aggressive to conservative. While trajectory and intent prediction methods [2, 3, 4, 5] have been extensively studied in both industry and academia, most of the existing approaches are evaluated on datasets like Waymo [6] and NuScenes [7], which lack significant variation in driver behavior and primarily consist of homogeneous traffic [4]. As a result, these methods often struggle to reliably predict the intentions of other agents in complex and heterogeneous traffic situations.

Simulators offer promise in addressing this challenge by generating diverse and kinodynamically feasible AVs and human drivers in dense and heterogeneous scenarios. This enables the modeling of multiple AVs simultaneously and facilitates the study of their interactions using multi-agent reinforcement learning (MARL). Existing MARL algorithms can be categorized as centralized or distributed (decentralized). Centralized approaches utilize centralized credit assignment networks [8, 9] or communication of information among agents [10, 11], resulting in better convergence properties. However, centralized MARL algorithms are not practical for autonomous driving scenarios, as implementing a central controller for vehicles distributed across a large-scale traffic scenario is infeasible. Distributed MARL algorithms allow agents to act independently, but training and convergence of such methods are challenging due to the well-documented non-stationary problem [12, 13]. Some approaches to mitigate the non-stationary problem include parameter sharing [14] or passing a global reward to all agents [15], but these mechanisms introduce strong and unrealistic assumptions for applications like autonomous driving.

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.

Intent prediction, on the other hand, focuses on inferring neighboring agents' behavior using local information, effectively addressing the non-stationarity problem. In the context of autonomous driving, some studies have approached intent prediction by classifying driving behaviors into predefined classes for trajectory forecasting, which oversimplifies the complexity of driving behavior and real-world traffic [4]. Other works [16, 17] aim to learn a low-dimensional vector that enables a more diverse policy and behavior representation. However, their training processes require the involvement of the same group of agents between episodes, which is not practical in the real world. In reality, AVs encounter different types of vehicles and drivers without prior knowledge of specific entities. This observation motivates the exploration of new mechanisms to infer opponents' intentions in diverse and heterogeneous environments with variable agents.

Main Contributions: In this paper, we propose a new intent-aware planning algorithm for autonomous driving in dense and heterogeneous traffic environments. We cast the autonomous driving problem as a hidden parameter partially observable stochastic game (HiP-POSG) [18, 19] and solve it using a distributed MARL framework, called iPLAN, built around an intent-aware trajectory prediction encoder-decoder architecture. Given the current traffic state along with past observations, iPLAN computes the optimal multi-agent policy for each agent in the environment, relying solely on local observations without information-sharing or communication. Our main contributions include:

1. To the best of our knowledge, we propose the first MARL algorithm embedded with a classical trajectory forecasting architecture for intent-aware planning in dense and heterogeneous environments. Our algorithm is fully decentralized without parameter sharing, communication, or centralized critics, and can handle variable agents across episodes.
2. We provide an explicit representation of agents' private incentives to their strategies that include (i) *Behavioral Incentive* for long-term planning tied to an agent's driving behavior or personality and (ii) *Instant Incentive* for short-term collision avoidance related to current traffic state. We propose a two-stream incentive inference mechanism that allows agents to infer incentives from their opponents and incorporate their inferred information into decision-making.
3. We perform experiments on two simulation environments, Non-Cooperative Navigation [10] and Heterogeneous Highway [20]. The results show that, compared to centralized MARL baselines like QMIX and MAPPO, our method yields a 4.0% and 35.7% higher episodic reward in *mild* and *chaotic* traffic and is 48.1% more successful with an 80.6% longer survival time in *chaotic* traffic in Heterogeneous Highway. Compared to the decentralized baseline IPPO, we demonstrate a higher episodic reward of 9.2% and 10.3% in *mild* traffic and *chaotic* traffic, a 25.3% higher success rate, and 13.7% longer survival time on Heterogeneous Highway.

2 Related Work

Trajectory and Intent Prediction for Autonomous Driving. Trajectory prediction is a fundamental task in autonomous driving [21, 22, 23]. TraPHic and RobustTP [24, 3] use an LSTM-CNN framework to predict trajectories in dense and complex traffic. TNT [25] uses target prediction, motion estimation, and ranking-based trajectory selection to predict future trajectories. DESIRE [26] uses sample generation and trajectory ranking for trajectory prediction. PRECOG [27] combines conditioned trajectory forecasting with planning objectives for AVs. Additionally, many methods focus on intent prediction to gain a better understanding of interactions between vehicles when predicting trajectories. Intent prediction can be done by physical-based methods like Kalman filter [28] or Monte Carlo [29], classical machine learning like Gaussian processes (GP) [30], Hidden Markov Model (HMM) [31], and Monte Carlo Tree Search (MCTS) [32], or deep learning-based methods such as Trajectron++ and CS-LSTM [2, 33]. [34] uses a Seq2Seq framework to encode agents' observations over neighboring vehicles as their social context for trajectory forecasting and decision-making. [35] uses temporal smoothness in attention modeling for interactions and a sequential model for trajectory prediction. However, most methods omit the difference in driving behaviors, which deteriorates their reliability in heterogeneous traffics.

Intent-aware Multi-agent Reinforcement Learning. As a large-scale and non-cooperative [36] scenario, the awareness of opponents' incentives is quite important when implementing MARL in autonomous driving. Intent-aware multi-agent reinforcement learning [5] estimates an intrinsic value that represents opponents' intentions for communication [37] or decision-making. Many intent inference modules are based on Theory of Mind (ToM) [38] reasoning or social attention-related mechanisms [39, 40]. [41] uses ToM reasoning over opponents' reward functions from their historical behaviors in performing multi-agent inverse reinforcement learning (MAIRL). [42] uses game theory ideas to reason about other agents' incentives and help decentralized planning among strategic agents. However, many prior works oversimplify the intent inference and make some prior assumptions about the content of intent. In the real world, agents' incentives are more complex

and intractable during interactions among large groups of agents, so a more general and high-level incentive representation is needed in intent-aware MARL.

Opponent Modeling. Opponent modeling [43] in multi-agent reinforcement learning usually uses some inference mechanisms to model opponents' policies. Opponent modeling could be done by either estimating others' actions and safety via Gaussian Process [44] or by generating an embedding of opponents' observations and actions [45]. Inferring opponents' policies helps to interpret peer agents' actions [46] and make agents more adaptive when encountering new partners [47]. Notably, many works [48, 49] reveal the phenomenon whereby ego agents' policies also influence opponents' policies. To track the dynamic variation of opponents' strategies made by an ego agent's influence, [16, 17] propose the latent representation to model opponents' strategies and influence based on their findings on the underlying structure in agents' strategy space. [50] provides a causal influence mechanism over opponents' actions and defines an influential reward over actions with high influence over others' policies. [51] proposes an optimization objective that accounts for the long-term impact of ego agents' behavior in opponent modeling. However, most existing methods assume agents are interacting with the same group of opponents across episodes. This is unrealistic in autonomous driving scenarios as human drivers always encounter different drivers and vehicles, and they should infer opponents' incentives without much prior knowledge.

3 Problem Formulation

Problem Setting and Assumptions: We consider a multi-agent scenario with $N \geq 2$ non-cooperative agents [52], *i.e.*, agents are controlled by distinct policies that maximize their own reward without parameter sharing or communication. In each episode, agents interact with one another and gain general experience without any prior knowledge about a specific agent from previous episodes. Agents' strategies remain the same within one episode, though strategies may evolve between episodes. We assume that all agents have some motivations behind their actions that come from their instant reactions to environments or long-lasting preferences. We denote them as *incentives* for agents' strategies. Such incentives are private to others but detectable from the observation of their strategies. In this work, we explicitly model these private incentives with hidden parameters representing latent states. Therefore, we formulate this problem as a multi-agent hidden parameter partially observable stochastic game [53], or HiP-POSG[1].

Task and objective: We consider the tuple

$$\left\langle N, \mathcal{S}, \{\mathcal{A}_i\}_{i=1}^N, \{\mathcal{O}_i\}_{i=1}^N, \{\Omega_i\}_{i=1}^N, \{\mathcal{Z}_i\}_{i=1}^N, \{\mathcal{F}_i\}_{i=1}^N, \mathcal{T}, \{\mathcal{R}_i\}_{i=1}^N, \gamma \right\rangle, \tag{1}$$

where N is the number of agents. $\mathcal{S}$ is the set of states. $\mathcal{A}_i$ is the set of actions for agent i. $\mathcal{O}_i$ is the observation set of agent i of the global state $S \in \mathcal{S}$, generated by agent i's observation function $\Omega_i : \mathcal{S} \to \mathcal{O}_i$. In our problem, agent i's observation $\mathbf{o}_i^t$ at time t could be further specified as $\mathbf{o}_i^t = \{o_{i,j}^t\}_{j \in \mathcal{N}_i}$, where $\mathcal{N}_i$ refers to the set of agents j in the neighborhood of i. The bold $\mathbf{o}_i^t$ denotes the set of agent i's observation of its neighbors at time t. We denote the sequence of agent i's historical observations $o_{i,j}$ of opponent j up to time t as $h_{i,j}^t = \{o_{i,j}^k\}_{k=1}^t$. The bold $\mathbf{h}_i^t = \{\mathbf{o}_i^k\}_{k=1}^t$ denotes agent i's observation history of its neighbors. Here, we indicate that agent i's observation history of agent j only consists of its observation of agent j's states, while agent j's actions and rewards are unobservable information by others. $\mathcal{Z}_i$ denotes the latent state space that represents the *incentive* of agent i's strategy. $f_i : \mathcal{O}_i^1 \times \mathcal{O}_i^2 \times \ldots \times \mathcal{O}_i^t \times \mathcal{Z}_j \to \mathcal{Z}_j$ is agent i's incentive inference function that makes an estimation $\hat{z}_{i,j}$ of its opponent j's actual incentive z_j from its observation history of opponent $h_{i,j}^t$ up to time t and its past estimation of z_j. Here, we assume agent i's estimations of agent j's incentive $\hat{z}_{i,j}$ belongs to the same latent state space $\mathcal{Z}_j$ as agent j's actual incentive z_j. $\mathcal{T} : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times \ldots \times \mathcal{A}_N \to \Delta(\mathcal{S})$ is the (stochastic) transition matrix between global states. $\mathcal{R}_i : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times \ldots \times \mathcal{A}_N \to \mathbb{R}$ is the reward function for agent i. γ is the reward discount factor. Agent i decides its action $a_i \in \mathcal{A}_i$ with policy $\pi_i : \mathcal{O}_1^t \times \mathcal{O}_2^t \times \ldots \times \mathcal{O}_N^t \times \mathcal{Z}_1 \times \mathcal{Z}_2 \times \ldots \times \mathcal{Z}_N \to \Delta(\mathcal{A}_i)$ with its observations $\mathbf{o}_i^t$, own incentive z_i, and estimated opponents' incentives $\hat{z}_{i,j}^t$ at time t.

The objective of agent i is to find the optimal policy π_i^* that maximizes their individual γ-discounted cumulative rewards over an episode with length T. The objective equation is given by

$$\pi_i^* = \arg\max_{\pi_i} \mathbb{E}_{\pi_i} \left[\sum_{t=1}^{T} \gamma^t r_i \left(s^t, \{a_i^t\}_{i=1}^N \right) \right] \tag{2}$$

[1] an extension of the HiP-POMDP [18, 19]

where r_i is the reward function of agent i.

Two-incentive Latent Strategy Representation. In this work, we assume that agents' actions are motivated by (i) long-term planning tied to an agent's driving behavior or personality and (ii) short-term collision avoidance related to current traffic state. To this end, we decouple agent i's incentive z_i into a two-element vector $z_i = \{\beta_i, \zeta_i\}$. The **Behavioral Incentive** β_i reflects agents' *personalities* or some long-lasting preferences in their decision-making. As stated in [54], vehicles' velocities and accelerations are indicators of driver aggressiveness, so an agent's personality could be reflected by its historical states. When querying an agent's historical states, some identical patterns could be found in their strategies, *e.g.*, aggressive drivers tend to have higher speeds and maintain smaller inter-vehicle spaces. The **Instant Incentive,** ζ_i**,** models the agents' instant reactions to their current surroundings, like their opponents' current position distribution or velocities. With this information, agents could make decisions by anticipating opponents' future states. Unlike instant incentives, behavioral incentives are more stable and invariant to agents' surroundings. Agents use their own incentives and their estimations over opponents' incentives in their decision-making. Since all agents' incentives for their strategies are private, agents preserve an inference module to estimate their opponents' incentives with their local observations.

Incentive Inference. To address two different incentives, we decouple agent i's incentive inference function f_i into two distinct functions, $f_{i,\beta}$ and $f_{i,\zeta}$: $\hat{\beta}^t_{i,j} \sim f_{i,\beta}(\cdot|h^t_{i,j}, \hat{\beta}^{t-1}_{i,j})$ uses agent i's historical observation $h^t_{i,j}$ of opponent j up to time t and its previous estimation of opponent j's behavioral incentive $\hat{\beta}^{t-1}_{i,j}$ to estimate opponent j's new behavioral incentive $\hat{\beta}^t_{i,j}$ at time t. $\hat{\zeta}^t_{i,j} \sim f_{i,\zeta}(\cdot|o^t_{i,j}, \hat{\beta}^t_{i,j}, \hat{\zeta}^{t-1}_{i,j})$ uses agent i's observation $o^t_{i,j}$ of opponent j at time t, its current estimation over opponent j's behavioral incentive $\hat{\beta}^t_{i,j}$ and its previous estimation of opponent j's instant incentive $\hat{\zeta}^{t-1}_{i,j}$ to estimate opponent j's new instant incentive $\hat{\zeta}^t_{i,j}$ at time t. With the estimation over opponents' incentives, agent i's policy $a^t_i \sim \pi(\cdot|\mathbf{o}^t_i, \hat{\boldsymbol{\beta}}^t_i, \hat{\boldsymbol{\zeta}}^t_i)$ decides its action a^t_i with its local observation, ego incentive, and estimations over opponents' incentives. Here, $\hat{\boldsymbol{\beta}}^t_i$ denotes the combination of agent i's behavioral incentive β_i and its estimations over all its opponent agents' behavioral incentives$\{\hat{\beta}^t_{i,j}\}^N_{j=1,j\neq i}$ at time t. $\hat{\boldsymbol{\zeta}}^t_i$ denotes the combination of agent i's instant incentive ζ_i and its estimations over all its opponent agents' instant incentives $\{\hat{\zeta}^t_{i,j}\}^N_{j=1,j\neq i}$ at time t.

4 iPLAN: Methodology

We demonstrate the overall architecture of our proposed framework in Figure 1. Agents interact with the environment with continuous state space $\mathcal{S}$. Here, we denote that an agent's state includes its ID, current position, and current velocity. An agent's observation includes the states of its neighbors within its observation scope. An agent i records its historical observations of its opponents' states for incentive inference. With historical observations $h^t_{i,j}$ and intermediate observations $\mathbf{o}^t_i$, agent i estimates opponent j's behavioral incentive β_j and instant incentive ζ_j. The controller of agent i decides action a^t_i based on its local observation $\mathbf{o}^t_i$, ego and opponents' estimated behavioral incentives $\hat{\boldsymbol{\beta}}^t_i$, and instant incentives $\hat{\boldsymbol{\zeta}}^t_i$. The action space $\mathcal{A}$ of the environment is discrete and consists of the following high-level actions: {*lane left*, *idle*, *lane right*, *faster*, *slower*} in our Heterogeneous Highway environment, or {*idle*, *up*, *down*, *left*, *right*} in our Non-cooperative Navigation environment (details in Section 5 and Appendix A), while a low-level motion controller (*e.g.*, IDM model [55]) converts the high-level actions into a sequence of x, y coordinates.

4.1 Behavioral Incentive Inference

The behavioral incentive inference module intends to estimate opponents' behavioral incentives by generating latent representations from their historical states. At time step t, agent i queries a sequence of historical observations h^t_{ij} for opponent j from its observation history profile as the input of the behavioral incentive inference module. For ease of computing, we truncate the full historical interaction sequence into a fixed-length sequence that includes the observation history from the previous t_h steps. We introduce an encoder $\mathcal{E}_i$ to update opponents' behavioral incentive estimation and a decoder $\mathcal{D}_i$ to reconstruct opponents' state sequences in the next t_h steps with current historical observations and behavioral incentive estimation. In practice, we parameterize encoder $\mathcal{E}_i$ with $\theta_{\mathcal{E}_i}$, and decoder $\mathcal{D}_i$ with $\theta_{\mathcal{D}_i}$. Hence, the encoder $\mathcal{E}_i$ approximates the behavioral incentive inference function $\hat{\beta}^t_{i,j} \sim f_\beta(\cdot|h^t_{i,j}, \hat{\beta}^{t-1}_{i,j})$.

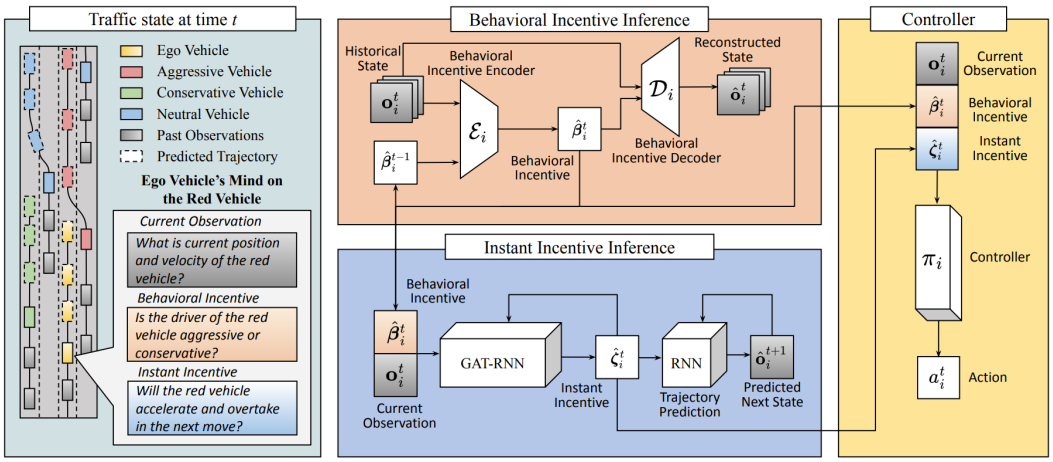


Figure 1: **Intent-aware planning in heterogeneous traffic:** At time t, we show current vehicle states in solid colors: ego vehicles i (solid yellow vehicle), aggressive vehicles (solid red), conservative vehicles (solid green), and neutral vehicles (solid blue). The future states of each vehicle are shown with dotted colors. At time t, the ego-agent observes nearby vehicles and infers their behavioral and instant incentives. The behavioral incentive inference (red block) uses agent i's historical observations $\mathbf{h}_i^t$ of other vehicle states (stacked gray boxes of current observations, $\mathbf{o}_i^t$) to infer their behavioral incentives and reconstruct state sequences with behavioral incentive inferences. The instant incentive inference (blue block) uses agent i's current observations $\mathbf{o}_i^t$ (single gray box) and its inference of others' behavioral incentives $\hat{\boldsymbol{\beta}}_i^t$ (single red box) to infer other vehicles' instant incentives $\hat{\boldsymbol{\zeta}}_i^t$ for trajectory prediction. Agent i's controller (yellow block) selects its action a_i^t with its current observations $\mathbf{o}_i^t$ (gray) and its inference of others' behavioral incentives $\hat{\boldsymbol{\beta}}_i^t$ (red) and instant incentives $\hat{\boldsymbol{\zeta}}_i^t$ (blue).

To address the temporal relation within opponents' state observation sequences, the encoder $\mathcal{E}_i$ uses a recurrent network that takes h_{ij}^t as a temporal sequence. Its output is the new estimation of the behavioral incentive of opponent j. As cognitive science suggests, human social attention does not change rapidly [56], so we take the behavioral incentive inference over opponents as a converging procedure towards actual opponents' behavioral incentives without sudden changes between updates. Starting from blank prior estimation over opponents' behavioral latent states, agents make new candidates over opponents' behavioral incentive estimations at each time step but use a soft updating policy with an extra coefficient η to update the behavioral incentives estimation. With this mechanism, agents could make more authentic estimations over opponents' behavioral incentives when controlling variance between estimation updates, which stabilizes agents' policies

$$\hat{\beta}_{i,j}^t = \eta \mathcal{E}_i(h_{i,j}^t, \hat{\beta}_{i,j}^{t-1}) + (1-\eta)\hat{\beta}_{i,j}^{t-1}. \tag{3}$$

The decoder $\mathcal{D}_i$ uses another recurrent network that concatenates agent i's historical observations h_{ij}^t of opponent j with its current behavioral incentive estimation $\hat{\beta}_{ij}^t$. The output is the reconstructed state sequence $\hat{h}_{i,j}^{t+t_h}$ of opponent j from t to $t+t_h$. We train our encoder and decoder with behavioral incentive inference loss $\mathcal{J}_{\beta_i}$, given by an average L1-norm error between the reconstructed state sequence $\hat{h}_{i,j}^{t+t_h} = \mathcal{D}_i(h_{i,j}^t, \hat{\beta}_{i,j}^t)$ and the ground truth $h_{i,j}^{t+t_h}$.

$$\mathcal{J}_{\beta_i} = \min_{\mathcal{E}_i, \mathcal{D}_i} \frac{1}{N t_h} \sum_{j=1}^{N} \left\| \mathcal{D}_i(h_{i,j}^t, \hat{\beta}_{i,j}^t) - h_{i,j}^{t+t_h} \right\|_1 . \tag{4}$$

4.2 Instant Incentive Inference for Trajectory Prediction

The instant incentive inference module intends to estimate opponents' instant incentives from current observations of surrounding agents and their behaviors, which is used for trajectory prediction. Similar to the behavioral incentive inference, we introduce another encoder-decoder structure with encoder ϕ_i parameterized by θ_{ϕ_i} and decoder ψ_i parameterized by θ_{ψ_i}. The encoder ϕ_i approximates the instant incentive inference function $\hat{\zeta}_{i,j}^t \sim f_{i,\zeta}(\cdot|o_{i,j}^t, \hat{\beta}_{i,j}^t, \hat{\zeta}_{i,j}^{t-1})$ from agent i's current observations $\mathbf{o}_i^t$ of agent i, current behavioral incentive estimations $\hat{\boldsymbol{\beta}}_i^t$, and previous instant incentive estimations $\hat{\boldsymbol{\zeta}}_i^{t-1}$. The instant latent state encoder ϕ_i uses a sequential structure with two networks.

The first network is a Graph Attention Network (GAT) [57]. For agent i, GAT reads its observation $\mathbf{o}_i^t$ at time t and the current behavioral incentive estimation $\hat{\boldsymbol{\beta}}_i^t$. The output of GAT is fed to an undirected graph $\mathcal{G}_i^t$ that represents instantaneous interactions among agents at time t. Every node in $\mathcal{G}_i^t$ represents an agent in the environment, while the attention weight over the edge between node i and node j encodes the interaction between agent i and j with its relative importance. The second part of the encoder ϕ_i is a recurrent neural network (RNN) to extract the temporal information from interaction history. The RNN uses the graphical representation $\mathcal{G}_i^t$ of interactions as the input and previous instant incentive estimation $\hat{\boldsymbol{\zeta}}_i^{t-1}$ as the hidden state. The output hidden state of this RNN $\hat{\boldsymbol{\zeta}}_i^t$ is the updated instant incentive estimation over all opponents of agent i.

The decoder ψ_i predicts all opponents' trajectories over a pre-defined length t_p from instant incentive estimations $\hat{\boldsymbol{\zeta}}_i^t$. We use another RNN that takes agent i's current observation $\mathbf{o}_i^t$ as the input and its current instant incentive estimation $\hat{\boldsymbol{\zeta}}_i^t$ as the hidden state. The first output of this RNN is the prediction of opponents' states $\hat{\mathbf{o}}_i^{t+1}$ at the next time step $t+1$. Then we use $\hat{\mathbf{o}}_i^{t+1}$ as the new input of RNN and iteratively predict opponents' states. The sequence of opponents' state predictions $\{\hat{\mathbf{o}}_i^{t+k}\}_{k=1}^{t_p} \sim \psi_i(\mathbf{o}_i^t, \hat{\boldsymbol{\zeta}}_i^t)$ is the trajectory prediction from $t+1$ to $t+t_p$ for all opponents of agent i. We train our encoder and decoder with instant incentive inference loss $\mathcal{J}_{\zeta_i}$, given by an average L1-norm error between predicted trajectories $\{\hat{\mathbf{o}}_i^{t+k}\}_{k=1}^{t_p}$ and ground truth trajectories $\{\mathbf{o}_i^{t+k}\}_{k=1}^{t_p}$.

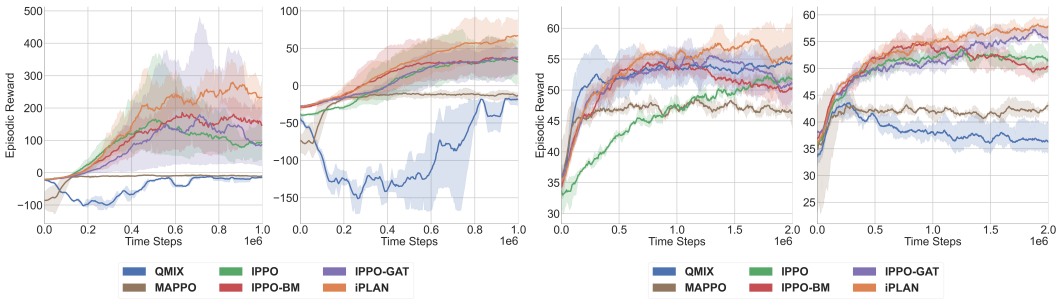
$$\mathcal{J}_{\zeta_i} = \min_{\phi_i, \psi_i} \frac{1}{N t_p} \sum_{j=1}^{N} \sum_{k=0}^{t_p-1} \left\| \psi_i(\mathbf{o}_i^t, \phi_i(\mathbf{o}_i^t, \hat{\boldsymbol{\beta}}_i^t, \hat{\boldsymbol{\zeta}}_i^{t-1})) - \mathbf{o}_i^{t+k+1} \right\|_1 \tag{5}$$

4.3 Implementation

The pseudocode of our algorithm is provided in Algorithm 1. For each environmental step t in the execution (line 4), agent i gathers its current and historical observations $\mathbf{o}_i^t$ and $\boldsymbol{h}_i^t$ (line 6), and uses this information to infer their opponents' behavioral incentives $\boldsymbol{\beta}_i^t$ and instant incentives $\boldsymbol{\zeta}_i^t$ (lines 7 and 8). After that, agent i's policy π_i selects action a_i^t (line 9). The backbone algorithm for each agent's controller is PPO [58], which includes a policy network π_i and a critic network Q_i. For each gradient step in training, agent i updates its policy π_i and critic Q_i (line 15) with sampled trajectories, computes the behavioral incentive inference loss $\mathcal{J}_{\beta_i}$ (line 16) to update its behavioral incentive inference encoder $\theta_{\mathcal{E}_i}$ and decoder $\theta_{\mathcal{D}_i}$ with $\mathcal{J}_{\beta_i}$, and uses instant incentive inference loss $\mathcal{J}_{\zeta_i}$ (line 17) to update its instant incentive inference encoder θ_{ϕ_i} and decoder θ_{ψ_i}.

Algorithm 1 iPLAN: Intent-aware Planning in Heterogeneous Traffic via Distributed MARL

Require: Number of agents N
1: **Initialize:** Agent i's network parameters $\theta_{\pi_i}, \theta_{Q_i}, \theta_{\mathcal{E}_i}, \theta_{\mathcal{D}_i}, \theta_{\phi_i}, \theta_{\psi_i}, i = 1, 2, \ldots, N$
2: **Initialize:** Replay Buffer $\mathcal{B} \leftarrow \emptyset$, Incentive Inferences $\boldsymbol{\beta}_i^0 \leftarrow \overrightarrow{0}, \boldsymbol{\zeta}_i^0 \leftarrow \overrightarrow{0}$ for $i = 1, 2, \ldots, N$
3: **for** each environmental step t **do**
4: **for** $i = 1, 2, \ldots, N$ **do**
5: Gather current and historical observation $\mathbf{o}_i^t$ and $\boldsymbol{h}_i^t$
6: Infer behavioral incentives $\hat{\boldsymbol{\beta}}_i^t$ with $\mathcal{E}_i(\boldsymbol{h}_i^t, \hat{\boldsymbol{\beta}}_i^{t-1})$
7: Infer instant incentives $\hat{\boldsymbol{\zeta}}_i^t$ with $\phi_i(\mathbf{o}_i^t, \hat{\boldsymbol{\beta}}_i^t, \hat{\boldsymbol{\zeta}}_i^{t-1})$
8: Select action a_i^t with $\pi_i(\cdot|\mathbf{o}_i^t, \hat{\boldsymbol{\beta}}_i^t, \hat{\boldsymbol{\zeta}}_i^t)$
9: **end for**
10: **end for**
11: **for** each gradient step **do**
12: Sample trajectories from the replay buffer $\mathcal{B}$
13: **for** $i = 1, 2, \ldots, N$ **do**
14: Update policy θ_{π_i} and critic θ_{Q_i} of the PPO controller
15: Update behavioral incentive encoder $\theta_{\mathcal{E}_i}$ and decoder $\theta_{\mathcal{D}_i}$ with $\mathcal{J}_{\beta_i}$ in (4)
16: Update instant incentive encoder θ_{ϕ_i} and decoder θ_{ψ_i} with $\mathcal{J}_{\zeta_i}$ in (5)
17: **end for**
18: **end for**
19: **Output:** $\mathcal{E}_i^*, \phi_i^*, \pi_i^*$ for each i.



(a) **Non-Cooperative Navigation:** with 3 agents in the (left) *easy* and (right) *hard* scenarios. 50 steps/episode.

(b) **Heterogeneous Highway:** with 5 agents in (left) *mild* and (right) *chaotic* scenarios. 90 steps/episode.

Figure 2: Comparison of average episodic reward in the Non-Cooperative Navigation and Heterogeneous Highway environments. **Conclusion:** iPLAN (orange) outperforms centralized approaches like QMIX (blue) and MAPPO (brown) as well as IPPO (green) in heterogeneous traffic environments.

5 Empirical Results and Discussion

We perform experiments over two non-cooperative environments, Non-Cooperative Navigation [10] and Heterogeneous Highway [20]. Experiments are designed from two perspectives. The first compares our method's performance with other distributed and centralized MARL approaches in non-cooperative environments. In this paper, we compare our method with two centralized MARL baselines, QMIX [8] and MAPPO [59], and one distributed MARL baseline, IPPO [60]. QMIX uses a central network to assign credits among agents with respect to their Q-values and global states. MAPPO uses a central critic that reads the observation of all agents and generates a critic value to update distributed actors. IPPO uses a distinct PPO policy to control each agent without any centralized training, communication, or inference module. The other perspective is to show the necessity of instant and behavioral incentive inference, especially under highly heterogeneous scenarios. We further design two scenarios with different heterogeneity levels in both environments and perform ablation studies over two variants of our method, including IPPO-BM, a vanilla IPPO controller without the instant incentive inference module, and IPPO-GAT, a vanilla IPPO controller without behavioral incentive inference module. Details regarding the experiment environment design are given in Appendix A. Further details regarding implementation, visual results, module design, and hyper-parameter study are given in Appendix B, C, D, and E, respectively.

5.1 Results on Non-Cooperative Navigation

Non-Cooperative Navigation involves n agents maximizing coverage over n landmarks without cooperation or communication. Agents select, reach, and stay at landmarks while avoiding conflicts. The *easy* scenario has three controllable agents with different kinematics, and the *hard* scenario adds an uncontrollable agent taking random actions. Figure 2a compares episodic rewards in *easy* and *hard* scenarios. iPLAN outperforms other methods with low deviation. IPPO-GAT and vanilla IPPO have larger deviations, indicating the benefit of behavioral incentive inference in stabilizing strategies. QMIX and MAPPO perform poorly with negative episodic rewards in both scenarios.

In Non-Cooperative Navigation, agents are attracted to the closest landmark at each time step, allowing multiple agents to target the same landmark simultaneously. As there is no consensus in destination assignment, agents must observe and infer others' strategies to modify their own. This reliance on observations and inference contributes to the superior performance of distributed MARL approaches over centralized MARL approaches.

5.2 Results on Heterogeneous Highway

In Heterogeneous Highway, vehicle agents should stay collision-free and maintain a proper speed when driving through heterogeneous traffic. Agents should infer their opponents' intentions and kinematics from their observations without much prior knowledge. Two scenarios in *mild* and *chaotic* traffic are provided. The majority of vehicles in the *mild* traffic scenario have normal driving behavior, and more aggressive vehicles exist in the *chaotic* traffic scenario. Figure 2b compares episodic rewards in the *mild* and *chaotic* traffic scenarios of the Heterogeneous Highway. We find that iPLAN has the best episodic reward in both the *mild* and *chaotic* traffic. IPPO-GAT, IPPO-BM, and vanilla IPPO have similar performances in *mild* traffic scenarios, but IPPO-GAT is slightly worse than iPLAN in *chaotic* traffic. Notably, two centralized MARL baselines have much lower episodic rewards than distributed MARL approaches in *chaotic* traffic, and QMIX has a significant collapse compared with its performance in *mild* traffic.

In addition to the episodic reward curve comparison, we evaluate our method and baselines over several navigation metrics, including:

Episodic Average Speed. Agents' average speed during their lifetime in an episode. Agents are encouraged to drive faster when driving between 20 and 30 m/s.

Average Survival Time. The average time steps passed over all agents before they collide or reach the end of this episode. Longer survival time reflects agents' better ability to avoid collisions.

Success Rate. The percentage of vehicles that still stay collision-free when an episode ends.

Table 1 shows navigation metrics for *mild* and *chaotic* traffic. High speed (closer to 30) correlates with low survival time and success rate. This is because aggressive reward-exploiting policies increase collision risk, reducing long-term reward. Approaches like iPLAN and IPPO-GAT drive slower (closer to 20) for safety and higher episodic reward. Instant incentive inference improves episodic reward and success rates, especially in *chaotic* traffic. iPLAN maintains similar success rates but higher average speed in *mild* traffic, being more conservative and dependent in heterogeneous traffic. Comparing iPLAN and IPPO-GAT, iPLAN drives faster in both scenarios for higher episodic reward. IPPO-GAT has longer survival time in *mild* traffic, but the opposite in *chaotic* traffic. This indicates that agents are more dependent on their instant incentive inference in *mild* traffic when opponents' trajectories are more predictable, and more dependent on their behavioral incentive inference in *chaotic* traffic due to aggressive vehicles' unpredictable behaviors. QMIX performs well in *mild* traffic but poorly in *chaotic* traffic (success rate $< 20\%$) due to environmental heterogeneity affecting its credit assignment mechanism.

	Approach	Avg. Speed (m/s)	Avg. Survival Time (# Time Steps) ↑	Success Rate (%) ↑
Mild	QMIX [8]	21.24 ± 0.09	$\mathbf{75.98 \pm 3.67}$	67.50 ± 6.34
	MAPPO [59]	$\mathbf{27.85 \pm 0.40}$	48.94 ± 3.11	32.81 ± 5.22
	IPPO [60]	22.63 ± 0.17	66.13 ± 4.13	49.06 ± 7.35
	IPPO-GAT	22.05 ± 0.11	75.54 ± 3.61	$\mathbf{68.44 \pm 6.64}$
	IPPO-BM	22.61 ± 0.16	64.11 ± 4.28	45.63 ± 6.33
	iPLAN	22.91 ± 0.15	70.56 ± 3.81	$\mathbf{68.44 \pm 5.86}$
Chaotic	QMIX [8]	27.06 ± 0.47	39.38 ± 2.64	19.69 ± 3.72
	MAPPO [59]	$\mathbf{29.46 \pm 0.05}$	42.31 ± 2.43	16.25 ± 3.76
	IPPO [60]	22.28 ± 0.13	67.01 ± 3.64	42.50 ± 7.12
	IPPO-GAT	20.91 ± 0.13	71.24 ± 3.83	61.88 ± 6.41
	IPPO-BM	21.65 ± 0.28	63.20 ± 3.51	35.31 ± 5.66
	iPLAN	21.61 ± 0.16	$\mathbf{76.20 \pm 3.33}$	$\mathbf{67.81 \pm 5.91}$

Table 1: **Navigation metrics in Heterogeneous Highway:** Metrics are averaged over 64 episodes with 0.95 confidence. iPLAN outperforms all other approaches in its highest success rate and survival time, though it tends to be conservative in its average speed.

5.3 Discussion

According to the experimental results, we conclude that iPLAN has the best performance in its episode reward and safety metrics. Two centralized MARL approaches perform worse as the environmental heterogeneity increases due to the absence of consensus among agents in heterogeneous environments. On the other hand, the awareness of opponents' strategies becomes more important in agents' decision-making when the environment is heterogeneous, especially the awareness of agents' instant reactions to surroundings. This need for increased awareness makes intent-aware distributed MARL algorithms perform better in these environments.

As for the two incentive inference modules, instant incentive inference helps to achieve a higher reward, especially in highly heterogeneous environments. However, solely depending on instantaneous information also leads to unstable strategies. Behavioral incentive inference helps to stabilize agents' strategies, but behavioral incentive inference alone does not contribute to a better-performing strategy. Agents perform the best with both incentive inference modules since they have a better understanding of their opponents' intentions.

6 Conclusion, Limitations, and Future Work

This paper introduces a new intent-aware distributed multi-agent reinforcement learning algorithm for planning and navigation in heterogeneous traffic. Our work has a few limitations. First, we have evaluated our approach exclusively in simulation and not in real-world autonomous driving scenarios. Further evaluation is required in more complex traffic scenarios with real vehicle trajectory recordings. Second, our contributions are backed by empirical evidence, and not by strong theoretical foundations since theoretical research in MARL is still in its infancy. Strict guarantees on safety are especially needed for autonomous driving applications. Follow up research on this line of work will study theoretical bounds on safety and convergence. Apart from addressing these limitations, we are excited to test the capabilities of our algorithm in even more challenging traffic conditions including diverse weather, night time traffic, and traffic where drivers may not follow traffic rules.

References

[1] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang, and H. Chen. A survey on trajectory-prediction methods for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 7(3):652–674, 2022. doi:10.1109/TIV.2022.3167103.

[2] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, editors, *Computer Vision – ECCV 2020*, pages 683–700, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58523-5.

[3] R. Chandra, U. Bhattacharya, A. Bera, and D. Manocha. Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8483–8492, 2019.

[4] R. Chandra, T. Guan, S. Panuganti, T. Mittal, U. Bhattacharya, A. Bera, and D. Manocha. Forecasting trajectory and behavior of road-agents using spectral clustering in graph-lstms. *IEEE Robotics and Automation Letters*, 2020.

[5] S. Qi and S.-C. Zhu. Intent-aware multi-agent reinforcement learning. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 7533–7540. IEEE, 2018.

[6] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. R. Qi, Y. Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, 2021.

[7] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[8] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *The Journal of Machine Learning Research*, 21(1):7234–7284, 2020.

[9] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[10] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*, 2017.

[11] Y. Liu, W. Wang, Y. Hu, J. Hao, X. Chen, and Y. Gao. Multi-agent game abstraction via graph attention neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7211–7218, 2020.

[12] T. Chu, J. Wang, L. Codecà, and Z. Li. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 21(3): 1086–1095, 2019.

[13] K. Zhang, Z. Yang, and T. Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pages 321–384, 2021.

[14] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning*, pages 5872–5881. PMLR, 2018.

[15] C. Wang, I. Durugkar, E. Liebman, and P. Stone. Dm2: Decentralized multi-agent reinforcement learning via distribution matching. 2023.

[16] A. Xie, D. Losey, R. Tolsma, C. Finn, and D. Sadigh. Learning latent representations to influence multi-agent interaction. In *Conference on robot learning*, pages 575–588. PMLR, 2021.

[17] W. Z. Wang, A. Shih, A. Xie, and D. Sadigh. Influencing towards stable multi-agent interactions. In *Conference on robot learning*, pages 1132–1143. PMLR, 2022.

[18] F. Doshi-Velez and G. Konidaris. Hidden parameter markov decision processes: A semiparametric regression approach for discovering latent task parametrizations. In *IJCAI: proceedings of the conference*, volume 2016, page 1432. NIH Public Access, 2016.

[19] J. Song, H. Ren, D. Sadigh, and S. Ermon. Multi-agent generative adversarial imitation learning. *Advances in neural information processing systems*, 31, 2018.

[20] E. Leurent. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018.

[21] W. Luo, C. Park, A. Cornman, B. Sapp, and D. Anguelov. JFP: Joint future prediction with interactive multi-agent modeling for autonomous driving. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=Y42uoIekm5b.

[22] A. Farid, S. Veer, B. Ivanovic, K. Leung, and M. Pavone. Task-relevant failure detection for trajectory predictors in autonomous vehicles. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=oPRhm0Aben_.

[23] P. Bhattacharyya, C. Huang, and K. Czarnecki. SSL-lanes: Self-supervised learning for motion forecasting in autonomous driving. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=fXMV2CEwNVo.

[24] R. Chandra, U. Bhattacharya, C. Roncal, A. Bera, and D. Manocha. Robusttp: End-to-end trajectory prediction for heterogeneous road-agents in dense traffic with noisy sensor inputs. In *Proceedings of the 3rd ACM Computer Science in Cars Symposium*, pages 1–9, 2019.

[25] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, C. Li, and D. Anguelov. Tnt: Target-driven trajectory prediction. In *Conference on Robot Learning*, 2020.

[26] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2165–2174, 2017.

[27] N. Rhinehart, R. Mcallister, K. Kitani, and S. Levine. Precog: Prediction conditioned on goals in visual multi-agent settings. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2821–2830, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society. doi:10.1109/ICCV.2019.00291. URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00291.

[28] V. Lefkopoulos, M. Menner, A. Domahidi, and M. N. Zeilinger. Interaction-aware motion prediction for autonomous driving: A multiple model kalman filtering scheme. *IEEE Robotics and Automation Letters*, 6(1):80–87, 2020.

[29] K. Okamoto, K. Berntorp, and S. Di Cairano. Driver intention-based vehicle threat assessment using random forests and particle filtering. *IFAC-PapersOnLine*, 50(1):13860–13865, 2017.

[30] J. Joseph, F. Doshi-Velez, A. S. Huang, and N. Roy. A bayesian nonparametric approach to modeling motion patterns. *Autonomous Robots*, 31:383–400, 2011.

[31] J. Li, W. Zhan, Y. Hu, and M. Tomizuka. Generic tracking and probabilistic prediction framework and its application in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 21(9):3634–3649, 2019.

[32] C.-J. Hoel, K. Driggs-Campbell, K. Wolff, L. Laine, and M. J. Kochenderfer. Combining planning and deep reinforcement learning in tactical decision making for autonomous driving. *IEEE transactions on intelligent vehicles*, 5(2):294–305, 2019.

[33] N. Deo and M. M. Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1468–1476, 2018.

[34] S.-W. Yoo, C. Kim, J. Choi, S.-W. Kim, and S.-W. Seo. Gin: Graph-based interaction-aware constraint policy optimization for autonomous driving. *IEEE Robotics and Automation Letters*, 8(2):464–471, 2022.

[35] Z. Cao, E. Biyik, G. Rosman, and D. Sadigh. Leveraging smooth attention prior for multi-agent trajectory prediction. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10723–10730. IEEE, 2022.

[36] A. Liniger and J. Lygeros. A noncooperative game approach to autonomous racing. *IEEE Transactions on Control Systems Technology*, 28(3):884–897, 2019.

[37] Y. Wang, F. Zhong, J. Xu, and Y. Wang. Tom2c: Target-oriented multi-agent communication and cooperation with theory of mind. *arXiv preprint arXiv:2111.09189*, 2021.

[38] N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S. A. Eslami, and M. Botvinick. Machine theory of mind. In *International conference on machine learning*, pages 4218–4227. PMLR, 2018.

[39] A. Vemula, K. Muelling, and J. Oh. Social attention: Modeling attention in human crowds. In *2018 IEEE international Conference on Robotics and Automation (ICRA)*, pages 4601–4607. IEEE, 2018.

[40] Z. Dai, T. Zhou, K. Shao, D. H. Mguni, B. Wang, and H. Jianye. Socially-attentive policy optimization in multi-agent self-driving system. In *Conference on Robot Learning*, pages 946–955. PMLR, 2023.

[41] H. Wu, P. Sequeira, and D. V. Pynadath. Multiagent inverse reinforcement learning via theory of mind reasoning. *arXiv preprint arXiv:2302.10238*, 2023.

[42] R. Chandra, R. Maligi, A. Anantula, and J. Biswas. Socialmapf: Optimal and efficient multi-agent path finding with strategic agents for social navigation. *IEEE Robotics and Automation Letters*, 2023.

[43] H. He, J. Boyd-Graber, K. Kwok, and H. Daumé III. Opponent modeling in deep reinforcement learning. In *International conference on machine learning*, pages 1804–1813. PMLR, 2016.

[44] Z. Zhu, E. Bıyık, and D. Sadigh. Multi-agent safe planning with gaussian processes. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6260–6267. IEEE, 2020.

[45] G. Papoudakis, F. Christianos, and S. Albrecht. Agent modelling under partial observability for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 34: 19210–19222, 2021.

[46] D. P. Losey, M. Li, J. Bohg, and D. Sadigh. Learning from my partner's actions: Roles in decentralized robot teams. In *Conference on robot learning*, pages 752–765. PMLR, 2020.

[47] S. Parekh, S. Habibian, and D. P. Losey. Rili: Robustly influencing latent intent. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 01–08. IEEE, 2022.

[48] F. B. Von Der Osten, M. Kirley, and T. Miller. The minds of many: Opponent modeling in a stochastic game. In *IJCAI*, pages 3845–3851, 2017.

[49] K. K. Ndousse, D. Eck, S. Levine, and N. Jaques. Emergent social learning via multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 7991–8004. PMLR, 2021.

[50] N. Jaques, A. Lazaridou, E. Hughes, C. Gulcehre, P. Ortega, D. Strouse, J. Z. Leibo, and N. De Freitas. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. In *International conference on machine learning*, pages 3040–3049. PMLR, 2019.

[51] D.-K. Kim, M. Riemer, M. Liu, J. Foerster, M. Everett, C. Sun, G. Tesauro, and J. P. How. Influencing long-term behavior in multiagent reinforcement learning. *Advances in Neural Information Processing Systems*, 35:18808–18821, 2022.

[52] J. Nash Jr. Non-cooperative games. In *Essays on Game Theory*, pages 22–33. Edward Elgar Publishing, 1996.

[53] E. A. Hansen, D. S. Bernstein, and S. Zilberstein. Dynamic programming for partially observable stochastic games. In *AAAI*, volume 4, pages 709–715, 2004.

[54] E. Cheung, A. Bera, E. Kubin, K. Gray, and D. Manocha. Identifying driver behaviors using trajectory features for vehicle navigation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3445–3452. IEEE, 2018.

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

[56] J. Swettenham, S. Baron-Cohen, T. Charman, A. D. Cox, G. Baird, A. Drew, L. M. Rees, and S. J. Wheelwright. The frequency and distribution of spontaneous attention shifts between social and nonsocial stimuli in autistic, typically developing, and nonautistic developmentally delayed infants. *Journal of child psychology and psychiatry, and allied disciplines*, 39 5:747–53, 1998.

[57] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

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

[59] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in Neural Information Processing Systems*, 35:24611–24624, 2022.

[60] C. S. de Witt, T. Gupta, D. Makoviichuk, V. Makoviychuk, P. H. S. Torr, M. Sun, and S. Whiteson. Is independent learning all you need in the starcraft multi-agent challenge?, 2020.

[61] A. Mavrogiannis, R. Chandra, and D. Manocha. B-gap: Behavior-rich simulation and navigation for autonomous driving. *IEEE Robotics and Automation Letters*, 7(2):4718–4725, 2022.

[62] P. Polack, F. Altché, B. d'Andréa Novel, and A. de La Fortelle. The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles? In *2017 IEEE intelligent vehicles symposium (IV)*, pages 812–818. IEEE, 2017.