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

Xiyang Wu[1], Rohan Chandra[2], Tianrui Guan[3], Amrit Singh Bedi[3] and Dinesh Manocha[1]

Full version at https://arxiv.org/pdf/2306.06236.pdf

***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 introduce a distributed multi-agent reinforcement learning (MARL) algorithm that can predict trajectories and intents 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 high-level decision-making based on their driving behavior or personality and *Instant Incentive* for motion planning for collision avoidance based on the current traffic state. Our approach enables agents to infer their opponents' behavior incentives and integrate this inferred information into their decision-making and motion-planning processes. We perform experiments on two simulation environments, Non-Cooperative Navigation and Heterogeneous Highway. Results show that iPLAN has a better performance than baselines in both environments in terms of episode rewards and navigation metrics. We open source our code at https://github.com/wuxiyang1996/iPLAN.**

I. INTRODUCTION

In this work, we consider the task of trajectory planning for autonomous vehicles in dense and heterogeneous traffic. This complexity arises from both the vehicle density and driving style diversity, vehicle dynamics, and types, ranging from motorcycles to trucks [3]. The key challenge to efficient trajectory planning in such environments is to infer the behavior of these heterogeneous agents [5]. Therefore, many solutions perform trajectory planning by jointly predicting the agents' future *trajectories* along with their *intent* [7].

Trajectory prediction predicts future states like spatial coordinates and heading angles of an agent, possibly incorporating velocity [18]. Intent prediction, in autonomous driving, focuses on inferring neighbors' behavior using local information [23] or categorizes inferred behaviors into types like aggressive and conservative [6], [5]. While various methods for joint trajectory and intent prediction exist [27], [4], [7], [23], most are tested on datasets [14], [2] that lack driver behavior variation [7]. Consequently, they falter in predicting diverse agent intentions in cluttered traffic [8].

Simulators like CARLA produce diverse agent behaviors [13], addressing dataset shortcomings. Though many prediction methods work with such simulators [25], [18], they often store data offline, undermining the simulator's purpose [26]. Contrarily, simulators can model agent interactions through multi-agent reinforcement learning (MARL), where the learning algorithm can engage with the simulation environment. MARL has demonstrated remarkable success in many different multi-agent domains such as Go [29], Dota2 [1], and StarCraft [34]. However, their applicability to autonomous driving has been relatively sparse [17].

Recent advances in MARL for autonomous driving emerged with the Highway-Env [19] environment proposed in the author's doctoral thesis [20]. Since then, some deep MARL techniques have been developed [37], [9] for trajectory planning, but they do not accommodate diverse traffic and assume agents can share information with each other. Currently, no decentralized MARL approach exists for predicting both intent and trajectory in mixed traffic.

Main Contributions: In this paper, we propose a new intent-aware trajectory 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) [12], [30] and solve it using a DTDE MARL framework, called iPLAN, built around a joint intent and trajectory prediction encoder-decoder architecture. Given the current traffic conditions and historical observations, iPLAN computes the optimal multi-agent policy for each agent in the environment, relying solely on local observations without weight-sharing or communication. We perform experiments on two simulation environments, Non-Cooperative Navigation [21] and Heterogeneous Highway [19]. Results show that iPLAN outperforms both centralized training decentralized execution (CTDE) MARL baselines like QMIX and MAPPO and DTDE baseline IPPO in terms of episodic reward and navigation metrics.

II. PROBLEM FORMULATION

Problem Setting and Assumptions: We consider a multi-agent scenario with $N \geq 2$ non-cooperative agents [22], *i.e.*, agents are controlled by individual policies that maximize their own reward without weight sharing or communication. In each episode, agents interact and gain experience, not relying on prior knowledge about specific agents from past episodes. Agents' strategies remain the same within one episode, though strategies may evolve between episodes. We assume that all agents are driven by motivations behind their actions. These motivations can arise from instantaneous reactions to environmental changes or more enduring preferences. These motivations, termed *incentives*, are not explicitly known to other agents but can be inferred by observing 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 [16], or HiP-POSG[1].

Task and objective: We consider the tuple

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

[1] Author are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD, USA {wuxiyang, dmanocha}@umd.edu

[2] Author is with the Department of Computer Science, University of Texas, Austin, TX, USA rchandra@utexas.edu

[3] Author are with the Department of Computer Science, University of Maryland, College Park, MD, USA {rayguan, amritbd}@umd.edu

[1] an extension of the HiP-POMDP [12], [30]

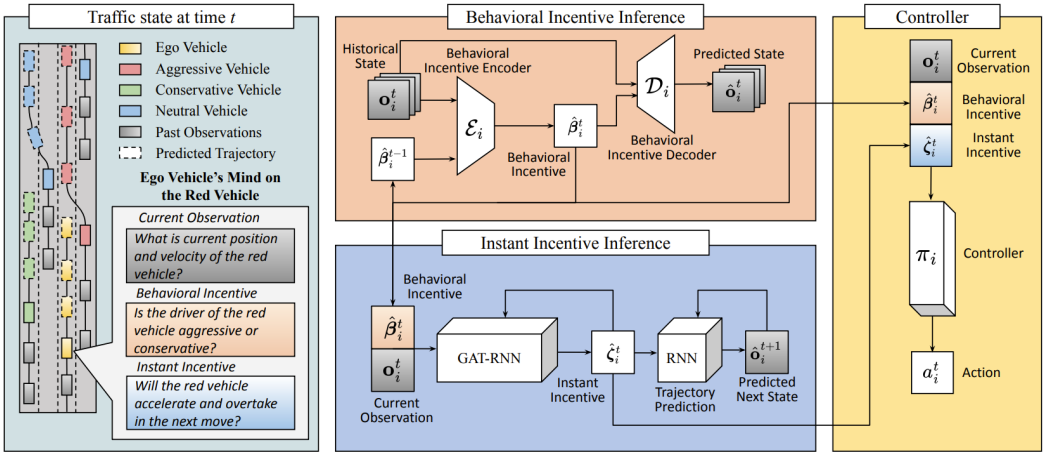


Fig. 1: **Intent-aware planning in heterogeneous traffic:** At time t, solid colors depict current vehicle states: ego vehicles i (yellow), aggressive (red), conservative (green), and neutral (blue). Dotted colors indicate future states. At time step t, the ego-agent observes nearby vehicles and infers their behavioral and instant incentives. The behavioral incentive inference (red block) uses agent i's historical observations $\mathbf{h}_i^t$ (stacked gray boxes of current observations, $\mathbf{o}_i^t$) to infer their behavioral incentives and predict future 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 inferred 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).

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. O_i is the observation set of agent i of the global state $S \in \mathcal{S}$, generated by agent i's observation function $\Omega_i : \mathcal{S} \to O_i$. 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. $f_i : O_i^1 \times O_i^2 \times \ldots \times O_i^t \times \mathcal{Z}_j \to \mathcal{Z}_j$ 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. $r_i : \mathcal{S} \times \mathcal{A}_1 \times \mathcal{A}_2 \times \ldots \times \mathcal{A}_N \to \mathbb{R}$ 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 $\pi_i : O_1^t \times O_2^t \times \ldots \times O_N^t \times \mathcal{Z}_1 \times \mathcal{Z}_2 \times \ldots \times \mathcal{Z}_N \to \Delta(\mathcal{A}_i)$ 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^*, maximizing its γ-discounted cumulative rewards over an episode of length T. The objective equation is given by

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

where r_i is the reward function of agent i.

Incentive Latent 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 the current traffic state. To this end, we decouple agent i's incentive z_i into a vector $z_i = \{\beta_i, \zeta_i\}$. Our formulation is related to the task and motion planning literature [15] where the behavior incentive follows a high-level decision-making strategy that sets planning sub-goals whereas the instant incentive refers to the low-level motion planning that executes the sub-goals. The behavior incentive biases the motion forecasting in a behavior-aware manner to be better suited for heterogeneous traffic.

Behavioral Incentive β_i models drivers' driving styles which are deeply rooted in their *personalities* [10]. Given the observations for the previous few seconds, behavior incentive performs high-level decision-making and plans actions, or sub-goals, and asks, "*What's the most likely action of this driver to take next?*". The answer is encoded via $\hat{\beta}_i^t$. This tells an agent whether to speed up in empty traffic or slow down in dense traffic and also allows an agent to reason between aggressive and conservative drivers.

Instant Incentive ζ_i signifies drivers' instantaneous responses to proximate traffic, taking into account the positions and speeds of neighboring vehicles. Instant incentive then asks, "*How should I execute this sub-goal/high-level action/plan using my controller so that I'm safe and still on track towards my goal?*". Instant incentive measures classical efficiency metrics defined in robotics literature such as collision avoidance (safety), distance from goal, and smoothness.

III. 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 $h_{i,j}^t$, and intermediate observations $\mathbf{o}_i^t$, agent i estimates opponent j's behavioral incentive β_j and instant incentive ζ_j. The controller of agent i decides action a_i^t based on its local observation $\mathbf{o}_i^t$, ego, and opponents' estimated behavioral incentives $\hat{\boldsymbol{\beta}}_i^t$, and instant incentives $\hat{\boldsymbol{\zeta}}_i^t$.

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 IV), while a low-level motion controller (*e.g.*, IDM model [32]) converts the high-level actions into a sequence of x, y coordinates.

A. 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 fixed-length sequence of historical observations h_{ij}^t from the previous t_h steps for opponent j from its observation history profile as the input of the behavioral incentive inference module. We introduce an encoder $\mathcal{E}_i$ to update opponents' behavioral incentive estimation and a decoder $\mathcal{D}_i$ to predict 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 $\hat{\beta}_{i,j}^t \sim f_\beta(\cdot|h_{i,j}^t, \hat{\beta}_{i,j}^{t-1})$.

To capture the sequential nature within opponents' state observation sequences, the encoder $\mathcal{E}_i$ employs a recurrent network that processes h_{ij}^t as a time series. This produces a new estimate of the behavioral incentive of opponent j. According to [31], we interpret the behavioral incentive inference for opponents as a smooth converging process toward the ground-truth Starting with an initial neutral estimation of opponents' behavioral latent states, agents propose new estimates for opponents' behavioral incentives at each time step. However, they employ a gentle update strategy, using an additional coefficient η, to refine the behavioral incentive estimates.

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

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 predicted 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 predicted 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 $h_{i,j}^{t+t_h}$.

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

B. 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, aiding trajectory prediction. Like the behavioral incentive inference, another encoder-decoder structure is employed 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 prior instant incentive estimations $\hat{\boldsymbol{\zeta}}_i^{t-1}$. The instant latent state encoder ϕ_i uses a sequential structure with two networks: a Graph Attention Network (GAT) [33] to extract the spatial relation from instantaneous interactions among agents and a recurrent neural network (RNN) to extract the temporal information from interaction history. 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 fixed 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. This RNN repeatedly predicts opponent states, forming a sequence $\{\hat{\mathbf{o}}_i^{t+k}\}_{k=1}^{t_p} \sim \psi_i(\mathbf{o}_i^t, \hat{\boldsymbol{\zeta}}_i^t)$ that denotes the trajectory for agent i's opponents from $t+1$ to $t+t_p$. Encoder and decoder training is guided by 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 $\{\mathbf{o}_i^{t+k}\}_{k=1}^{t_p}$.

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

C. Implementation

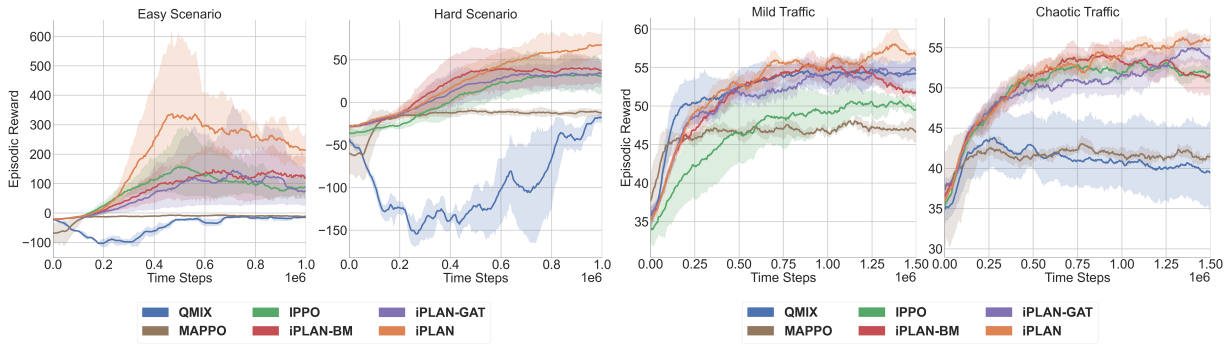
For each environmental step t in execution, agent i gathers its current and historical observations $\mathbf{o}_i^t$ and $\boldsymbol{h}_i^t$, and uses this information to infer their opponents' behavioral incentives $\boldsymbol{\beta}_i^t$ and instant incentives $\boldsymbol{\zeta}_i^t$. After that, agent i's policy π_i selects action $a_i^t \sim \pi(\cdot|\mathbf{o}_i^t, \hat{\boldsymbol{\beta}}_i^t, \hat{\boldsymbol{\zeta}}_i^t)$. The backbone algorithm for each agent's controller is PPO [28], including 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 with sampled trajectories, computes the behavioral incentive inference loss $\mathcal{J}_{\beta_i}$ to update its behavioral incentive inference encoder $\theta_{\mathcal{E}_i}$ and decoder $\theta_{\mathcal{D}_i}$, and uses instant incentive inference loss $\mathcal{J}_{\zeta_i}$ to update its instant incentive inference encoder θ_{ϕ_i} and decoder θ_{ψ_i}.

IV. Empirical Results and Discussion

We perform experiments over two non-cooperative environments, Non-Cooperative Navigation [21] and Heterogeneous Highway [19]. Experiments are designed from two perspectives. The first is to compare our approach's performance with other CTDE and DTDE MARL approaches in non-cooperative environments. In this paper, we compare our method with two CTDE MARL baselines, QMIX [24] and MAPPO [36], and one DTDE MARL baseline, IPPO [11]. 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 IPLAN-BM a vanilla IPPO controller without the instant incentive inference module, and IPLAN-GAT, a vanilla IPPO controller without behavioral incentive inference module. More details about environments and experiment results could be found in [35].

A. Experiment Results

Figure 2a compares episodic rewards in *easy* and *hard* scenarios. IPLAN outperforms other methods with low deviation. Two CTDE baselines, 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. The lack of consensus



(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.

Fig. 2: Average episodic reward in the Non-Cooperative Navigation and Heterogeneous Highway environments. **Conclusion:** iPLAN (orange) outperforms CTDE approaches like QMIX (blue) and MAPPO (brown), as well as DTDE approaches, like IPPO (green) in both environments.

	Approach	Avg. Speed (m/s)	Avg. Survival Time (# Time Steps) ↑	Success Rate (%) ↑
Mild	QMIX [24]	21.24 ± 0.09	$\mathbf{75.98 \pm 3.67}$	67.50 ± 6.34
	MAPPO [36]	$\mathbf{27.85 \pm 0.40}$	48.94 ± 3.11	32.81 ± 5.22
	IPPO [11]	22.63 ± 0.17	66.13 ± 4.13	49.06 ± 7.35
	iPLAN-GAT	22.05 ± 0.11	75.54 ± 3.61	$\mathbf{68.44 \pm 6.64}$
	iPLAN-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 [24]	27.06 ± 0.47	39.38 ± 2.64	19.69 ± 3.72
	MAPPO [36]	$\mathbf{29.46 \pm 0.05}$	42.31 ± 2.43	16.25 ± 3.76
	IPPO [11]	22.28 ± 0.13	67.01 ± 3.64	42.50 ± 7.12
	iPLAN-GAT	20.91 ± 0.13	71.24 ± 3.83	61.88 ± 6.41
	iPLAN-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 I: **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.

in destination assignment favors DTDE MARL approaches and inference modules, which contributes to their better performance over CTDE MARL approaches.

Figure 2b compares episodic rewards in the *mild* and *chaotic* traffic scenarios of the Heterogeneous Highway. iPLAN leads in episodic rewards for both traffic scenarios. In *mild* traffic, iPLAN-GAT, iPLAN-BM, and IPPO are comparable, but iPLAN-GAT lags in *chaotic* scenarios. CTDE MARL baselines fare worse than DTDE MARL in *chaotic* traffic, with QMIX notably declining from its *mild* performance. Beyond episodic reward curves, we also assess methods on navigation metrics: **Episodic Average Speed**: encouraging speeds between 20 and 30 m/s, **Average Survival Time**: indicating agents' ability of collision avoidance. **Success Rate**: measuring the ratio of collision-free vehicles at the episode's end.

Table I shows navigation metrics for *mild* and *chaotic* traffic. High speed (closer to 30) correlates with low survival time and success rate due to collision risks from aggressive policies. iPLAN and iPLAN-GAT opt for slower speeds (closer to 20), prioritizing safety and reward. Instant incentive inference improves episodic reward and success rates, especially in *chaotic* traffic. iPLAN is conservative, showing consistent success rates but faster speeds in *mild* traffic. Comparatively, iPLAN generally drives faster iPLAN-GAT. iPLAN-GAT has a longer survival time in *mild* traffic, but the opposite in *chaotic* traffic. QMIX performs well in *mild* traffic but poorly in *chaotic* traffic (success rate $< 20\%$) due to environmental heterogeneity effect on its credit assignment.

B. Discussion

Centralized versus Decentralized Training Regime. We used the decentralized training regime with the assumption that agents should learn navigation policies in a DTDE manner without centralization in training. Empirically, we find that CTDE MARL approaches perform worse as the environmental heterogeneity increases due to lack of consensus among agents. 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.

Decoupled Incentive Inference. Individually, the incentives yield some benefit over a baseline controller. For example, we find that both the behavior and instant incentive inference modules individually help to achieve a higher reward, especially in more heterogeneous environments (See Figure 2). However, our system works best when both incentives are jointly activated, for example in Table I, we find that the success rate drops significantly for iPLAN-GAT, compared to iPLAN (61.88% versus 67.81%). This clearly indicates autonomous vehicles need the behavior incentive module to survive in the more heterogeneous chaotic traffic scenario.

V. CONCLUSIONS

This paper presents a novel intent-aware distributed multi-agent reinforcement learning algorithm for heterogeneous traffic navigation. We model two agent incentives: behavioral and instant. Our method enables agents to infer opponents' behavioral incentives, integrating this knowledge into decisions and motion planning. Our approach outperforms baselines in the Non-Cooperative Navigation and Heterogeneous Highway in episodic rewards and navigation metrics. Our future work includes exploring using global information in decision-making through communication or weight sharing, and refining our approach for real traffic complexities and unpredictable road conditions with edge cases from unfamiliar roads with sudden-changing behaviors. We aim to enhance the real-world applicability and generalizablity of our approach through further research.

ACKNOWLEDGMENT

We would like to thank Caroline Wang for helpful discussions and feedback during the course of this paper. This research was supported by Army Cooperative Agreement W911NF2120076.

References

[1] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemys law Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

[2] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar 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.

[3] Rohan Chandra. *Towards Autonomous Driving in Dense, Heterogeneous, and Unstructured Traffic*. PhD thesis, University of Maryland, College Park, 2022.

[4] Rohan Chandra, Uttaran Bhattacharya, Aniket Bera, and Dinesh 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.

[5] Rohan Chandra, Uttaran Bhattacharya, Trisha Mittal, Aniket Bera, and Dinesh Manocha. Cmetric: A driving behavior measure using centrality functions. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2035–2042. IEEE, 2020.

[6] Rohan Chandra, Uttaran Bhattacharya, Trisha Mittal, Xiaoyu Li, Aniket Bera, and Dinesh Manocha. Graphrqi: Classifying driver behaviors using graph spectrums. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4350–4357. IEEE, 2020.

[7] Rohan Chandra, Tianrui Guan, Srujan Panuganti, Trisha Mittal, Uttaran Bhattacharya, Aniket Bera, and Dinesh Manocha. Forecasting trajectory and behavior of road-agents using spectral clustering in graph-lstms. *IEEE Robotics and Automation Letters*, 2020.

[8] Rohan Chandra, Xijun Wang, Mridul Mahajan, Rahul Kala, Rishitha Palugulla, Chandrababu Naidu, Alok Jain, and Dinesh Manocha. Meteor: A dense, heterogeneous, and unstructured traffic dataset with rare behaviors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9169–9175. IEEE, 2023.

[9] Baiming Chen, Mengdi Xu, Zuxin Liu, Liang Li, and Ding Zhao. Delay-aware multi-agent reinforcement learning for cooperative and competitive environments. *arXiv preprint arXiv:2005.05441*, 2020.

[10] Ernest Cheung, Aniket Bera, Emily Kubin, Kurt Gray, and Dinesh 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.

[11] Christian Schroeder de Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip H. S. Torr, Mingfei Sun, and Shimon Whiteson. Is independent learning all you need in the starcraft multi-agent challenge?, 2020.

[12] Finale Doshi-Velez and George 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.

[13] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017.

[14] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin 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.

[15] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4:265–293, 2021.

[16] Eric A Hansen, Daniel S Bernstein, and Shlomo Zilberstein. Dynamic programming for partially observable stochastic games. In *AAAI*, volume 4, pages 709–715, 2004.

[17] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A Al Sallab, Senthil Yogamani, and Patrick Pérez. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):4909–4926, 2021.

[18] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher Bongsoo Choy, Philip H. S. Torr, and Manmohan 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.

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

[20] Edouard Leurent. *Safe and efficient reinforcement learning for behavioural planning in autonomous driving*. PhD thesis, Université de Lille, 2020.

[21] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*, 2017.

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

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

[24] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *The Journal of Machine Learning Research*, 21(1):7234–7284, 2020.

[25] 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.

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

[27] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 683–700, Cham, 2020. Springer International Publishing.

[28] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[29] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[30] Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. Multi-agent generative adversarial imitation learning. *Advances in neural information processing systems*, 31, 2018.

[31] John Swettenham, Simon Baron-Cohen, Tony Charman, Antony D. Cox, Gillian Baird, Auriol Drew, Laura M. Rees, and Sally 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.

[32] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical review E*, 62(2):1805, 2000.

[33] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[34] Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, Alireza Makhzani, Heinrich Küttler, John Agapiou, Julian Schrittwieser, et al. Starcraft ii: A new challenge for reinforcement learning. *arXiv preprint arXiv:1708.04782*, 2017.

[35] Xiyang Wu, Rohan Chandra, Tianrui Guan, Amrit Singh Bedi, and Dinesh Manocha. iplan: Intent-aware planning in heterogeneous traffic via distributed multi-agent reinforcement learning. *arXiv preprint arXiv:2306.06236*, 2023.

[36] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in Neural Information Processing Systems*, 35:24611–24624, 2022.

[37] Haifeng Zhang, Weizhe Chen, Zeren Huang, Minne Li, Yaodong Yang, Weinan Zhang, and Jun Wang. Bi-level actor-critic for multi-agent coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7325–7332, 2020.