

# Communication-Critical Planning via Multi-Agent Trajectory Exchange

Nathaniel Moore Glaser and Zsolt Kira  
Georgia Institute of Technology  
{nglaser, zkira}@gatech.edu

**Abstract**—This paper addresses the task of joint multi-agent perception and planning, especially as it relates to the real-world challenge of collision-free navigation for connected self-driving vehicles. For this task, several communication-enabled vehicles must navigate through a busy intersection while avoiding collisions with each other and with obstacles. To this end, this paper proposes a *learnable* costmap-based planning mechanism, given raw perceptual data, that is (1) *distributed*, (2) *uncertainty-aware*, and (3) *bandwidth-efficient*. Our method produces a costmap and uncertainty-aware entropy map to sort and fuse candidate trajectories as evaluated across multiple-agents. The proposed method demonstrates several favorable performance trends on a suite of open-source overhead datasets as well as within a novel communication-critical simulator. It produces accurate semantic occupancy forecasts as an intermediate perception output, attaining a 72.5% average pixel-wise classification accuracy. By selecting the top trajectory, the multi-agent method scales well with the number of agents, reducing the hard collision rate by up to 57% with eight agents compared to the single-agent version.

## I. INTRODUCTION

For much of its history, robotics has revolved around single-agent systems. Many factors perpetuated this singular focus, such as costly hardware, weak compute power, and spotty inter-agent communication. However, thanks to modern technological advancements, these prohibitive developmental costs are largely reduced. With improved capabilities in hand, researchers have been keen to extend single-agent techniques towards their multi-agent counterparts.

One such extension involves that of deep learning [13, 26]. This class of techniques can infer semantic information about the world by combining neural architectures with data. It has seen remarkable success in areas such as object detection and, more recently, robotic perception and planning [4, 35, 11, 12, 2]. However, again, these algorithms have primarily focused on *single-agent* applications.

To balance out this trend, this paper seeks to apply modern deep learning techniques towards the task of *multi-agent* perception and planning, especially within the setting of collision-free trajectory planning for multiple connected robots. Specifically, we propose a *distributed, communication-enhanced* collision avoidance policy that is learned from observations of multi-agent interactions. We derive a novel trajectory exchange mechanism that leverages several state-of-the-art learning-based techniques, including image-to-costmap generation, deep-learned spatial uncertainty, and costmap-based motion planning.

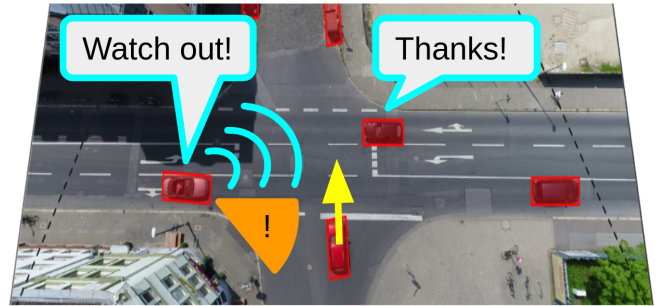


Fig. 1: **Task.** Collision-free planning via communication.

To achieve this end, we distribute copies of an identical neural architecture across multiple connected agents. Using this neural network, each agent then processes its local observations to determine (1) the cost of occupying different parts of its local state space (i.e. each agent generates a local costmap) and (2) the perceptual uncertainty associated with that cost (i.e. a entropy map). We then propose a method whereby a querying agent broadcasts several motion trajectory candidates for distributed evaluation. Then each supporting agent evaluates the incoming motion trajectories against its own egocentric costmap (and corresponding uncertainty map) and returns the sampled costs (and their uncertainties). Importantly, we propose an uncertainty-aware method to fuse the costs and uncertainties from multiple agents. Given the fused cost for each trajectory, the original agent then selects the lowest cost trajectory for execution.

We summarize the contributions of our paper as follows:

- We address a variant on *Multi-Robot Collaborative Perception and Planning* [16, 17] in which a group of decentralized, but communication-enabled, agents must avoid collisions as they move through a shared world.
- To address this challenge, we propose an end-to-end learnable **Multi-Agent Trajectory Exchange** network, **MATE**. We contribute novel techniques that extend the current state of the art, including a *bandwidth-efficient, uncertainty-aware* trajectory exchange mechanism.
- We evaluate our algorithm across a suite of open-source datasets [1, 23], including within a novel, communication-critical simulator, **CoBEV-Sim**.

## II. RELATED WORK

Several prior works are particularly relevant for collaborative perception and planning.

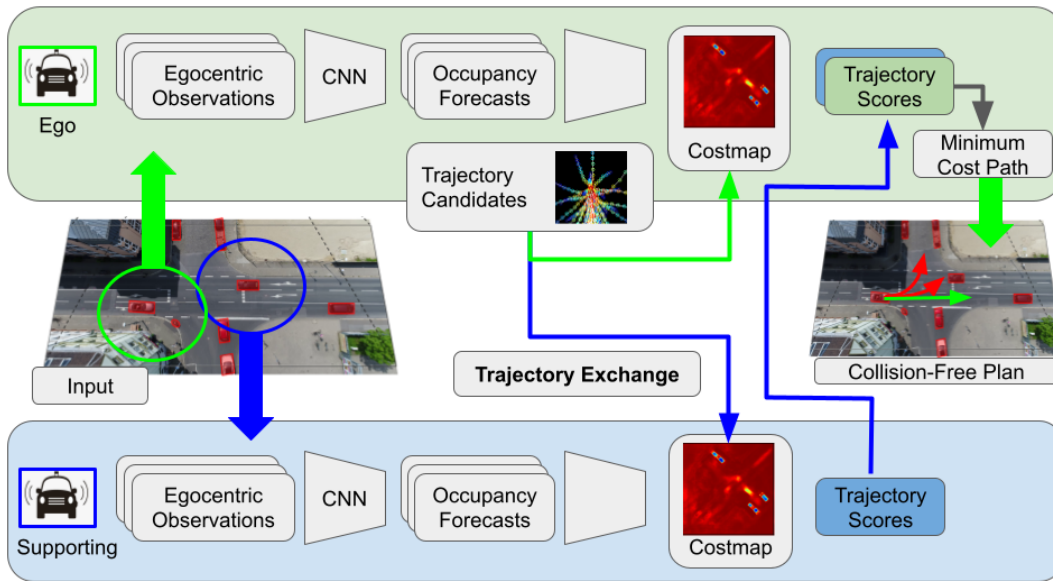


Fig. 2: **Multi-Agent Trajectory Exchange.** Egocentric observations from multiple connected vehicles are locally processed into learnable and interpretable costmaps. Each agent then broadcasts its trajectory candidates for distributed, opportunistic evaluation across any number of supporting agents. The supporting agents then return the costs sampled by the trajectory. The ego agent aggregates these scores using uncertainty-aware fusion and selects the minimum cost trajectory for execution.

One set of prior work [10, 24, 12, 11] addresses how a motion planning policy may be learned from data, either by interacting with an environment (reinforcement learning) or by observing an expert’s interactions with an environment (inverse reinforcement learning). The work of Sadat et al. [24] shows how a neural motion policy may be learned by observing expert driving demonstrations. Their system takes a map and lidar sequence as inputs, and it uses a learned encoder to produce a set of occupancy predictions (e.g. pedestrian, car, bike) for future timesteps. Next, it uses a learned cost function to score a set of sampled trajectories against these occupancy predictions, and the minimum cost trajectory is selected for execution.

Other, more human-centric, works consider the interplay between multiple human agents in a shared scene. For instance, the works of Kollmitz et al. [12, 11] leverage physical interaction and inverse reinforcement learning to adapt the planning behavior of a robot near human actors. However, while the robots in these works explicitly consider the motion of other actors, they do not leverage communication to improve their capabilities.

To this end, another set of prior work has addressed the challenge of *collaborative perception* through learned communication [17, 16, 32]. Learned communication is a relatively recent trend where data from different sensing sources is pegged for exchange, based on a purely learned process. **When2Com** [16] leverages a learned *handshake communication* module to mediate the exchange of visual information between robotic agents, specifically for multi-view image classification and semantic segmentation.

A similar set of prior work has addressed collaborative perception through learned *transformation* and *communication*

of spatial feature maps [14, 7, 4]. For instance, **Disconet** [14] uses a convolutional network to compress top-down 2D lidar rasters into an intermediate feature representation. The spatial cells inside these feature maps are then synchronized across multiple agents using a spatial collaboration graph.

Using similar communication architectures, more recent works have addressed the joint task of *collaborative perception and planning*. **Coopernaut** [5] uses a learned encoder module to compress point cloud information into a relatively compact 2D representation. The 2D representations from multiple allocentric agents are then transmitted, spatially transformed, and aggregated into the perspective of a single egocentric agent. Finally, a learned control module executes a control policy based on the aggregated information. However, these works only implicitly consider the role of uncertainty when aggregating multi-agent information.

An exciting new trend involves leveraging deep-learned uncertainty [9, 22, 30, 28]. Pan et al. [22] argue that the softmax confidences produced by classification networks roughly correlate with the empirical accuracy, as evaluated across a diverse set of common classification tasks. Converse to confidence, several works [30, 22, 20] apply Shannon Entropy [25] as a measure of *uncertainty*, and demonstrate its application to tasks such as domain adaptation, active learning, and crowd counting. Moreover, other works [20, 29, 28] leverage entropy as a spatial map, especially for semantic segmentation tasks. Wang et al. [31] uses an entropy-weighted average for image fusion.

Our approach draws inspiration from each of these bodies of work. Like prior work, our method uses an interpretable costmap-based scoring mechanism for evaluating and selecting trajectories for execution. However, unlike prior

work, our method uses a highly efficient *trajectory* exchange mechanism to improve planning performance, instead of transmitting dense representations (such as spatial feature maps). Moreover, our exchange mechanism explicitly considers perceptual uncertainty as an intuitive way to fuse the trajectory scores from multiple robot sources.

Beyond extending the methodology of prior work, our work also advances the setting of collaborative perception and planning. Several prior works add their own specialized datasets [15, 23] to the corpus of existing open-source datasets [6, 27, 8, 34] and closed-source datasets [32, 3, 19, 33, 18]. **V2X-Sim** [15] primarily focuses on the challenge of collaborative perception, though it only offers relatively simple planning scenarios. For our work, we evaluate our algorithm on a dataset split of **AutoCastSim** [23]. This dataset specifically targets challenging self-driving situations, such as red light infringements and left turns across traffic. However, these scenarios lack diversity and have a limited number of agents available for useful communication. To address these limitations, our work extends the capabilities of these datasets with the novel **CoBEV-Sim**, a simulator which creates a higher volume of diverse, communication-critical scenarios.

### III. METHODOLOGY

Autonomous navigation is an important capability for self-driving vehicles (SDVs). Prior to any widespread adoption, SDVs must first demonstrate a robust ability to move safely in a cluttered, dynamic environment. To this end, we propose a novel approach for performing *safe* autonomous navigation, especially in the context of inter-vehicle communication.

#### A. Task

This work addresses the task of collision-free intersection navigation for connected robotic agents. Namely, given pose in a global coordinate frame, local perception, and inter-agent communication, a group of multiple autonomous robots must plan a sequence of actions that limits dangerous interactions with the environment and with each other, as shown in Fig. 1.

In this setting, each agent has access to three types of data: (1) global pose data, (2) local perception data, and (3) data communicated from nearby agents. For the global pose data, each vehicle has knowledge of its past (and current) positions and orientations with respect to a global coordinate system (as provided by GPS or similar). For the local perception data, each vehicle captures *egocentric* visual information as it travels through the environment. For the communicated data, each vehicle may communicate bandwidth-limited information (as measured in bytes) to other vehicles within a limited range. These limitations are typically dictated by real-world environmental constraints. This constrained communication provides the agents with an indirect ability to *perceive* beyond their local observations and *plan* for future interactions. This setting is further complicated by the presence of both communicating and non-communicating mobile agents.

#### B. Model Architecture

As summarized in Fig. 2 and 3, this work leverages several modules to tackle the task of intersection navigation:

**Motion Forecasting.** First, as with Chen et al. [2], each agent uses a CNN  $\Phi$  to convert a sequence of aligned 2D Bird’s Eye View (BEV) observations  $\mathbf{X}^{T^-:0}$  into a sequence of 2D semantic occupancy forecasts  $\mathbf{P}^{1:T^+}$ :

$$\mathbf{P}_{\{\text{null}, \text{ego}, \text{allo}\}}^{1:T^+} = \Phi(\mathbf{X}^{T^-:0}) \quad (1)$$

The output classes include a *null*, egocentric (*ego*), and allocentric (*allo*) agent class, trained via cross-entropy loss.

**Costmap and Uncertainty-Map Generation.** Next, each agent converts its 2D semantic occupancy forecasts into a costmap and entropy map. The costmap is created by combining its egocentric  $\mathbf{P}_{ego}$  and allocentric  $\mathbf{P}_{allo}$  occupancy forecasts into a binarized occupancy map  $\mathbf{B}$  using threshold  $\theta$  and then computing a signed distance field (SDF) [21]:

$$\mathbf{B}^t = \max(\mathbf{P}_{ego}^t, \mathbf{P}_{allo}^t) > \theta \quad \text{and} \quad \mathbf{D}^t = \text{SDF}(\mathbf{B}^t) \quad (2)$$

To generate the uncertainty map, Shannon entropy is computed across the softmaxed class channels for each spatial cell in the semantic occupancy forecast<sup>1</sup>, similar to the procedures described in [20, 29, 28]:

$$u_{x,y}^t = - \sum_c p_{x,y}^t(c) \log p_{x,y}^t(c) \quad (3)$$

**Trajectory Generation.** Each agent generates a set of  $N$  trajectory candidates,  $\mathbb{S} = \{S^i\}_0^N$ , where each trajectory candidate defines a sequence of agent poses at future timesteps,  $S^i = \{(x_t, y_t)\}_1^{T^+}$ . The trajectories were generated to cover a wide range of feasible vehicle maneuvers, as dictated by linear and angular acceleration limits.

**Bandwidth-Efficient Trajectory Exchange.** Each ego agent ( $e$ ) broadcasts its candidate trajectories for distributed evaluation by the supporting agents ( $a$ ). Each supporting agent transforms these trajectories to its local frame using relative pose matrix  $R_e^a$ . Next, each supporting agent returns values sampled from its costmap and entropy map:

$$S_t^{a,i} = R_e^a S_t^{e,i} = (x(t), y(t)) \quad (4)$$

$$C_t^{a,i} = \mathbf{D}_t^{a,i}[x(t), y(t)] \quad (5)$$

$$U_t^{a,i} = \mathbf{U}_t^{a,i}[x(t), y(t)] \quad (6)$$

**Uncertainty-Aware Multi-Agent Cost Fusion.** After having its suite of candidate trajectories scored by the cooperating agents, the ego agent fuses all of these contributions together via an entropy-weighted average, similar to [31]:

$$F^i = \sum_{a \in \text{allo}} \sum_{t=1}^{T^+} \frac{C_t^{a,i}}{U_t^{a,i}} \quad (7)$$

**Trajectory Prioritization.** Finally, the set of trajectories is sorted based on their fused scores:

$$\mathbb{S}_{\text{prioritized}} = [S_i | \forall i \in \arg \underset{i}{\text{sort}} F^i] \quad (8)$$

<sup>1</sup>In this section, we denote image-structured data with bold uppercase (e.g.  $\mathbf{U}^t$ ) and extracted cells in lowercase (e.g.  $u_{x,y}^t$ ).

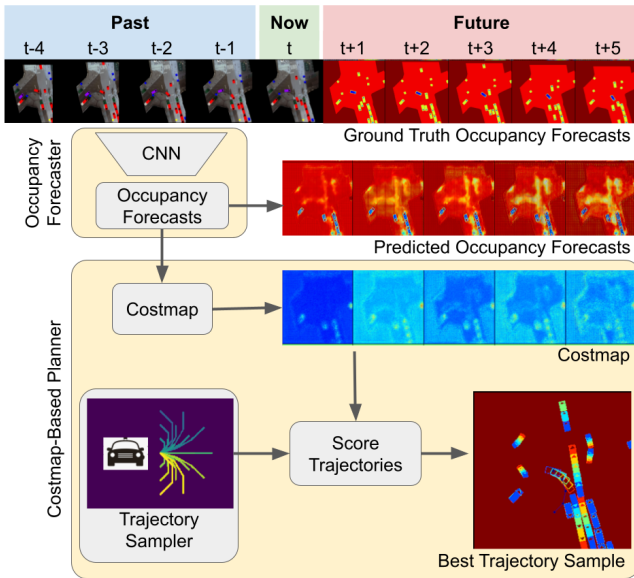


Fig. 3: Overview of qualitative results for a single agent.

### C. Dataset

Our problem domain combines several distinct research directions (motion forecasting, motion planning, communication-criticality, and safety for self-driving vehicles) all within a collaborative multi-robot setting. Datasets exist for each of these domains; however, it is challenging to find a single dataset that fairly unifies all of them. Therefore, we evaluate our approach using both slightly modified open-access datasets as well as a novel communication-critical simulator. We seek to demonstrate that, in certain rare (but important!) scenarios, having access to a team of collaborating agents (and their local perceptions) improves planning performance.

Namely, we include the Intersection Drone Dataset (**InD**), **AutoCastSim**, and our own Collaborative Bird’s Eye View simulator (**CoBEV-Sim**) in our analysis. Each of these datasets is adapted to the same general task: a set of communication-enabled agents must safely maneuver through a shared environment that also includes a set of non-communicative agents.

Specifically, we adapt each dataset to a uniform style of egocentric **BEV** inputs and outputs<sup>2</sup>. For each communication-enabled agent, we capture a limited-range, top-down perspective of the shared environment, centered and oriented with respect to that agent. These top-down perspectives can take the form of a 2d lidar scan, lidar scan raster, and local segmentation mask (i.e. occupancy mask), and we capture them at a constant synchronized frame rate throughout the duration of each planning scenario. Moreover, we record the relative pose between each pair of agents, communicating and non-communicating (the poses of the non-communicating agents are solely used for evaluation). We separate the observed sequence of top-down rasters and

<sup>2</sup>The egocentric BEV perspective is the output of prior research works [15]. For simplicity, we evaluate our algorithm using this data format as a starting point.

relative poses into two categories – past data and future data. The past information is used as input, whereas the future information is used for training and evaluation.

The ultimate task of each dataset is the following: each communicating agent observes a sequence of past, egocentric occupancy masks; then each agent must efficiently leverage the inferences of its peers to propose a set of trajectory candidates that are collision-free for some future time horizon.

1) *Open-Access Datasets*: We port several open-access datasets to our unified collaborative perception and planning setting. Each of these datasets includes motion trajectories for scenarios with many cooperating (i.e. non-colliding) agents. The **InD** dataset consist of extensive real motion tracks of interacting road users from an overhead **BEV** drone perspective. The **AutoCastSim** dataset consists of the same data, though synthesized within the CARLA simulator.

2) *Communication-Critical Simulator*: In addition to open-access datasets, we evaluate the performance of our approach within a novel, communication-critical simulation environment. This simulator addresses several shortcomings observed across many available datasets:

**Existing datasets are cumbersome.** They often require a bulky simulator, complex API calls, or heavy downloads.

**Existing datasets lack diversity.** They support relatively few distinct scenarios, offering insignificant scene diversity.

**Existing datasets cover inconsequential settings.** There is an imbalance between common, benign situations (like lane-following) and rare, dangerous ones (like blind corners).

**Existing datasets do not contain interoperable agents.** Most datasets focus on a single “ego” agent, instead of allowing any agent to become a benefactor of communication.

Our dataset addresses each of these concerns. It is **lightweight**, **randomizeable**, and **customizeable**. It is straightforward to seed situations which are interesting, especially situations which intuitively would benefit from multi-agent communication. Moreover, each agent can be “put into the driver’s” seat, and its planning performance can be fairly and uniformly evaluated.

Specifically, our simulator allows for the dynamic generation of communication-critical multi-agent planning scenarios. First, a user specifies the start and goal locations of any number of agents. These agents can be spawned with a customizeable footprint, either corresponding to mobile road users (such as a car, truck, or pedestrian) or other communication-enabled road infrastructure (such as a smart traffic light or building). Next, the simulator iteratively assigns a randomized straight-line speed profile to each agent such that it avoids collisions with its neighbors. The assignment process is repeated until all agents have a non-colliding, goal-reaching trajectory sequence. Finally, the simulator then renders the local perception data (2D lidar scan, BEV lidar raster, and BEV semantic segmentation map) for each of these agents. These lidar scans take into account realistic limitations, such as sensor range limitations and perceptual occlusions.

Using our simulator, we generate a set of communication-critical scenarios:

	ForecastTime					avg
	t = 1	t = 2	t = 3	t = 4	t = 5	
Mean Accuracy	78.0	78.3	73.0	68.2	65.0	72.5
Mean IoU	49.3	53.4	53.1	51.1	47.5	50.9

TABLE I: Segmentation metrics for motion forecasting.

**CrissCross.** This scenario consists of many straight-line trajectories that chaotically intersect in open space. For each scenario, we randomize the start and goal positions of 20 agents. A random subset of these agents is communication-enabled ( $n_{com} = \{1, 5, 10, 15\}$  for our experiments), while the remaining agents are non-communicative ( $n_{silent} = 20 - n_{com}$ ). This scenario is communication-critical because agents frequently transit through the paths of each other, oftentimes from outside of their egocentric perception range; the communicating agents must rely on collaborators to enhance their effective observation range.

**AutoCastSim.** This scenario includes a mixture of the red light infraction and blind corner cases from the original open-access dataset. Since the original dataset includes only a small number of nearby collaborating agents ( $\sim 3$ ), we randomly insert an additional set of static and trailing agents to the planning scene. We include 8 total agents, with a mix of communicating agents ( $n_{com} = \{1, 2, 4, 8\}$ ) and non-communicating agents ( $n_{silent} = 8 - n_{com}$ ). These supplementary agents undergo the same collision-free randomization described earlier, ensuring novel scenarios after each randomization.

We leverage the randomization component of our simulator during training and evaluation. During each training epoch, we synthesize 10 novel planning scenarios. For evaluation, we use randomization to create a static test suite of 100 planning scenarios.

#### D. Experiments

We evaluate the performance of our proposed model on the **InD**, **AutoCastSim**, and **CoBEV-Sim** datasets. Our model uses an input BEV sequence to infer future segmentation masks and to produce a candidate trajectory output. To train the network, these inferred segmentation masks are compared against the ground truth segmentation masks using a cross entropy loss function. To test the network, the inferred segmentation masks are compared against the ground truth masks to yield accuracy and mean intersection of union (mIoU) metrics. To test planning performance, the trajectories are “rolled-out” in the dataset environment, where potential collisions with the environment and its obstacles are tallied. Both qualitative and quantitative assessments are discussed in the following section.

## IV. RESULTS

The results presented in the beginning of this section highlight the performance of single-agent perception and planning, as demonstrated on the **InD** dataset. Multi-agent perception and planning is discussed towards the end of the section. Those results are evaluated on the **InD**, **AutoCastSim**, and **CoBEV-Sim** datasets.

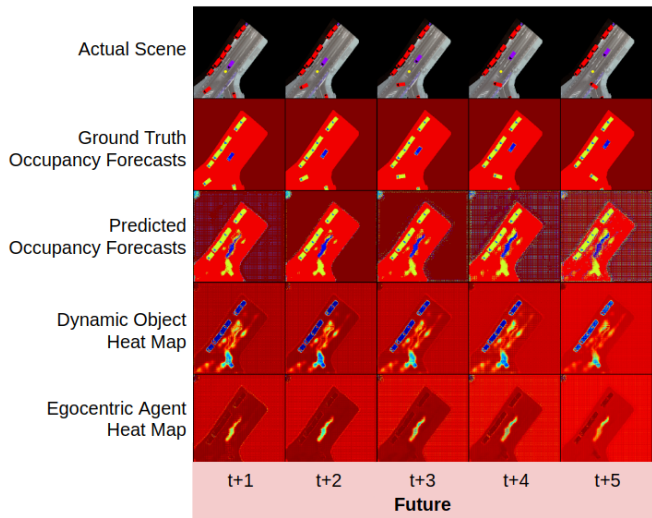


Fig. 4: Sample qualitative results for motion forecasting. Using a past sequence of egocentric data (not pictured), the motion forecasting module predicts future state occupancy.

#### A. Motion Forecasting

We assess perception performance by comparing the inferred occupancy forecasts against ground truth future occupancy. Qualitatively, as shown in Fig 4, the predicted locations of various road users roughly matches what actually unfolded in the scene. In other words, the occupancy heat maps for the egocentric agent and dynamic objects constitute reasonable hypotheses for where different road actors may be found at different points in the future. Moreover, the model seems to have learned that moving objects maintain their speed and heading, within some variability. For instance, the motion forecasting module predicted a diagonal streak of probable locations for the egocentric agent—since the agent was previously travelling in a straight line at constant speed, it will likely continue on that path. The streak is likely stretched in the longitudinal direction to account for uncertainty in velocity. There is less stretching in the transverse direction likely because the road surface constrains that variability. Additionally, the model is able to learn turning behavior and multi-modal behavior, as shown in Fig 3.

Quantitatively, the predicted and ground truth segmentation forecasts are compared using standard accuracy and mean intersection over union (mIoU) metrics. As shown in Table I, the motion forecasting module achieves an average pixel-wise accuracy of 72.5% and mean IoU of 50.9%. As expected, the performance of the motion forecasting module degrades for temporally distant forecasts.

#### B. Single-Agent Trajectory Sorting

The proposed method infers a cost that allows for a set of trajectories to be scored and prioritized. An indirect way to measure the quality of the cost function (and hence the overall learned pipeline) is to see how sorting trajectories by their cost correlates with key metrics, such collision rate. As shown in Fig 5, the inferred trajectory cost correlates with collision rate.

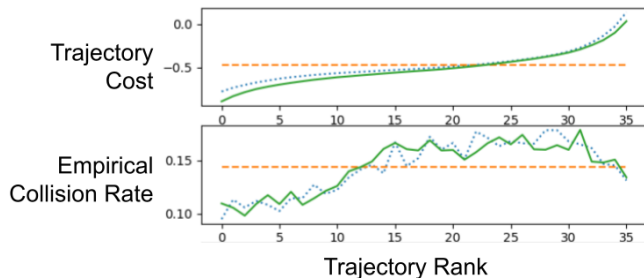


Fig. 5: **Trajectory rank versus cost.** The horizontal orange lines show the average cost of a randomly sampled trajectory. The solid and dotted curves show the cost of a trajectory of a given rank, averaged across the **InD** dataset.

Dataset	Com Agents	Top- $k$ Collision Rate (%)		<i>rand traj</i>
		$k = 1$	$k = 10$	
<b>AutoCastSim</b>	$n = 1$	18.3	19.2	26.0
	$n = 2$	16.2 [-11.2]	15.8 [-17.3]	
	$n = 4$	17.8 [-2.8]	17.5 [-8.7]	
RedLight	$n = 8$	<b>7.9 [-57.0]</b>	10.2 [-46.6]	
	$n = 1$	42.7	41.5	
<b>CoBEV-Sim</b>	$n = 5$	36.7 [-14.1]	36.9 [-11.3]	41.3
	$n = 10$	35.5 [-16.9]	36.8 [-11.5]	
	$n = 15$	<b>30.0 [-29.7]</b>	30.9 [-25.6]	

TABLE II: **Top- $k$  collision rate with  $n$ -agent communication.** This table shows the collision rates of our method compared to a random sampling baseline. The numbers in brackets show the percent change in collision rate as compared to the single-agent ( $n = 1$ ) performance.

### C. Multi-Agent Trajectory Sorting

We evaluate our method on the communication-critical datasets of **AutoCastSim** and **CoBEV-Sim**. These datasets include important scenarios where communication is crucial to improving safety, such as maneuvering past a blind corner at high speed. We present our results in terms of top- $k$  collision rates. The top- $k$  collision rate denotes the frequency of a hard collision as averaged across the  $k$  trajectories that have the lowest cost, as inferred by our multi-agent trajectory exchange method. Oftentimes, several trajectory candidates can safely maneuver through a scenario. We include these top- $k$  metrics to capture the performance of both the best trajectory ( $k = 1$ ) and the best *set* of trajectories ( $k = 10$ ).

As shown in Table II, our method highlights several desirable trends. First, the top-1 trajectory outperforms the top-10 trajectories, and both outperform the random trajectory sampler baseline. This trend indicates that our learned, distributed trajectory prioritization scheme is properly inferring the cost of trajectory samples, proportional to empirical collision rates. Second, the collision rate improves with more inter-agent communication. The top-1 trajectory shows a 57.0% decrease and a 29.7% decrease from the single agent collision rate for **AutoCastSim** and **CoBEV-Sim**, respectively. Third, the top-1 and top-10 collision rates are relatively similar, indicating that our method has prioritized trajectories such that downstream tasks can safely choose between the top- $k$  set of trajectories with minimal penalty, a de-

sirable feature for more downstream planners. Finally, communication is critical in the **CoBEV-Sim** dataset, as shown by the similar collision rates in the non-communicating case ( $n = 1$ ), where single-agent performance matches that of a random trajectory sampler.

For the **InD Dataset**, our multi-agent trajectory exchange mechanism (with support from about 4 agents) yielded a 7.2% decrease in hard collisions over its single-agent counterpart. The solid and dotted lines of Fig 5 show the close relative performance of the multi-agent and single-agent algorithms. This improvement is likely a result of communication being non-essential for this specific dataset: a single independent agent can easily handle most **InD** scenarios.

### D. Bandwidth

Several prior works [14, 5] address multi-agent collaborative perception by exchanging dense representations of perception information. These representations, while compressed, still consume significant amounts of bandwidth, especially since they preserve 2D image structure. Our work takes bandwidth-efficiency to the extreme. For instance, if all agents share a common dictionary of trajectory samples (i.e. 80 trajectory samples with 15 2D pose waypoints), then the egocentric agent only needs to transmit its relative location (i.e. a single  $(x, y, \theta)$  pose); and the allocentric agents can roto-transform their shared dictionaries of trajectories to the egocentric perspective and simply broadcast the values that were sampled from its costmap and uncertainty map. This 1D list of sampled values ( $80 \times 15 \times 2 = 2400$  floating point values) is significantly more compressed than the 2D image-based representations of prior work (i.e. DiscoNet [14] with  $\sim 10^4$  floating point values).

## V. CONCLUSION

This paper presents a novel method for addressing collision-free navigation for connected self-driving vehicles. Namely, it uses a trajectory exchange mechanism to perform distributed validation of trajectory candidates for any number of neighboring vehicles. The method exhibits several desirable traits, including being (1) distributed, (2) uncertainty-aware, and (3) bandwidth-efficient. The method scales with the number of agents, reducing the collision rate up to 57% with eight agents compared to the single-agent version.

**Future Work.** Future work will further investigate the multi-agent setting and architecture. By demonstrating favorable results for an extensible and interpretable neural architecture, this work laid a strong foundation for future multi-agent collaborative planning experiments. Such experiments will explore the impact of localized sensor noise and degradation, reduced individual sensor capability, positioning noise, and variable network connectivity. Each of these parameters adds realism to the multi-agent planning challenge, and future work will address their impact on the overall planning performance of the connected vehicle network.

## VI. ACKNOWLEDGEMENT

This work was supported by ONR grant N00014-18-1-2829.

## REFERENCES

- [1] Julian Bock et al. “The ind dataset: A drone dataset of naturalistic road user trajectories at german intersections”. In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 1929–1934.
- [2] Jianyu Chen, Shengbo Eben Li, and Masayoshi Tomizuka. “Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning”. In: *arXiv preprint arXiv:2001.08726* (2020).
- [3] Qi Chen et al. “Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds”. In: *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 514–524.
- [4] Alexander Cui et al. “LookOut: Diverse Multi-Future Prediction and Planning for Self-Driving”. In: *CoRR* abs/2101.06547 (2021). arXiv: 2101.06547. URL: <https://arxiv.org/abs/2101.06547>.
- [5] Jiaxun Cui et al. “Coopernaut: End-to-End Driving with Cooperative Perception for Networked Vehicles”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. CVPR ’22, 2022.
- [6] Alexey Dosovitskiy et al. “CARLA: An open urban driving simulator”. In: *Conference on robot learning*. PMLR, 2017, pp. 1–16.
- [7] Nathaniel Glaser et al. “Overcoming Obstructions via Bandwidth-Limited Multi-Agent Spatial Handshaking”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2021, pp. 2406–2413. DOI: 10.1109/IROS51168.2021.9636761.
- [8] Matthew Howe, Ian Reid, and Jamie Mackenzie. “Weakly Supervised Training of Monocular 3D Object Detectors Using Wide Baseline Multi-view Traffic Camera Data”. In: *arXiv preprint arXiv:2110.10966* (2021).
- [9] Alex Kendall and Yarin Gal. “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *CoRR* abs/1703.04977 (2017). arXiv: 1703.04977. URL: <http://arxiv.org/abs/1703.04977>.
- [10] B Ravi Kiran et al. “Deep reinforcement learning for autonomous driving: A survey”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.6 (2021), pp. 4909–4926.
- [11] Marina Kollmitz et al. “Learning Human-Aware Robot Navigation from Physical Interaction via Inverse Reinforcement Learning”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2020, pp. 11025–11031. DOI: 10.1109/IROS45743.2020.9340865.
- [12] Marina Kollmitz et al. “Time dependent planning on a layered social cost map for human-aware robot navigation”. In: *2015 European Conference on Mobile Robots (ECMR)*. IEEE, 2015, pp. 1–6.
- [13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.
- [14] Yiming Li et al. “Learning Distilled Collaboration Graph for Multi-Agent Perception”. In: *Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS 2021)*. 2021.
- [15] Yiming Li et al. “V2X-Sim: Multi-Agent Collaborative Perception Dataset and Benchmark for Autonomous Driving”. In: (2022).
- [16] Yen-Cheng Liu et al. “When2com: Multi-agent perception via Communication Graph Grouping”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.
- [17] Yen-Cheng Liu et al. “Who2com: Collaborative Perception Via Learnable Handshake Communication”. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2020.
- [18] Yassine Maalej et al. “Vanets meet autonomous vehicles: A multimodal 3d environment learning approach”. In: *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 2017, pp. 1–6.
- [19] Sivabalan Manivasagam et al. “Lidarsim: Realistic lidar simulation by leveraging the real world”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 11167–11176.
- [20] Yanda Meng et al. *Spatial Uncertainty-Aware Semi-Supervised Crowd Counting*. 2021. DOI: 10.48550/ARXIV.2107.13271. URL: <https://arxiv.org/abs/2107.13271>.
- [21] Helen Oleynikova et al. “Signed distance fields: A natural representation for both mapping and planning”. In: *RSS 2016 Workshop: Geometry and Beyond-Representations, Physics, and Scene Understanding for Robotics*. University of Michigan, 2016.
- [22] Fei Pan et al. “Unsupervised Intra-Domain Adaptation for Semantic Segmentation Through Self-Supervision”. In: June 2020. DOI: 10.1109/CVPR42600.2020.00382.
- [23] Hang Qiu et al. “AutoCast: Scalable Infrastructure-less Cooperative Perception for Distributed Collaborative Driving”. In: *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications, and Services*. MobiSys ’22, 2022.
- [24] Abbas Sadat et al. “Perceive, predict, and plan: Safe motion planning through interpretable semantic representations”. In: *European Conference on Computer Vision*. Springer, 2020, pp. 414–430.
- [25] C. E. Shannon. “A Mathematical Theory of Communication”. In: *SIGMOBILE Mob. Comput. Commun. Rev.* 5.1 (Jan. 2001), pp. 3–55. ISSN: 1559-1662. DOI: 10.1145/584091.584093. URL: <https://doi.org/10.1145/584091.584093>.

- [26] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).
- [27] Simon Suo et al. “TrafficSim: Learning to Simulate Realistic Multi-Agent Behaviors”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2021).
- [28] Junjiao Tian et al. “UNO: Uncertainty-aware noisy-or multimodal fusion for unanticipated input degradation”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5716–5723.
- [29] Johannes H. Uhl et al. “Spatialising uncertainty in image segmentation using weakly supervised convolutional neural networks: a case study from historical map processing”. In: *IET Image Processing* 12.11 (2018), pp. 2084–2091. DOI: <https://doi.org/10.1049/iet-ipr.2018.5484>. eprint: <https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-ipr.2018.5484>. URL: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-ipr.2018.5484>.
- [30] Keze Wang et al. “Cost-Effective Active Learning for Deep Image Classification”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 27.12 (Dec. 2017), pp. 2591–2600. DOI: [10.1109/tcsvt.2016.2589879](https://doi.org/10.1109/tcsvt.2016.2589879). URL: <https://doi.org/10.1109/tcsvt.2016.2589879>.
- [31] Pei Wang, Hui Fu, and Ke Zhang. “A pixel-level entropy-weighted image fusion algorithm based on bidimensional ensemble empirical mode decomposition”. In: *International Journal of Distributed Sensor Networks* 14.12 (2018), p. 1550147718818755. DOI: [10.1177/1550147718818755](https://doi.org/10.1177/1550147718818755). eprint: <https://doi.org/10.1177/1550147718818755>. URL: <https://doi.org/10.1177/1550147718818755>.
- [32] Tsun-Hsuan Wang et al. “V2vnet: Vehicle-to-vehicle communication for joint perception and prediction”. In: *European Conference on Computer Vision*. Springer, 2020, pp. 605–621.
- [33] Zhongyang Xiao et al. “Multimedia fusion at semantic level in vehicle cooperative perception”. In: *2018 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2018, pp. 1–6.
- [34] Haibao Yu et al. “DAIR-V2X: A Large-Scale Dataset for Vehicle-Infrastructure Cooperative 3D Object Detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 21361–21370.
- [35] Wenyuan Zeng et al. “End-to-end interpretable neural motion planner”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 8660–8669.