A. Loss Functions

The loss function for training the factorized joint trajectory decoder (Sec. 3.4) is defined by:

$$\mathcal{L}_2 = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{prop}}, \tag{13}$$

where $\mathcal{L}_{\text{reg}}(\{\hat{Y}_k\}_{k\in[K]}, Y) := \mathcal{L}_{\ell_1}(\{\hat{Y}_k\}_{k\in[K]}, Y)$ is a scene-level smooth ℓ_1 regression loss applied to the best modality of $K = 6$ joint modalities $\{\hat{Y}_k\}_{k\in[K]}$, where the best modality attains the minimum loss:

$$\mathcal{L}_{\ell_1}(\{\hat{Y}_k\}_{k\in[K]}, Y) = \min_{k\in[K]} \frac{1}{A \cdot T_{\text{fut}}} \sum_{a\in[A]} \sum_{t\in[T_{\text{fut}}]} \text{reg}(\hat{Y}^a_{t,k} - Y^a_t), \tag{14}$$

where Y denotes the ground-truth future trajectory coordinates of all A agents in the scene, $\text{reg}(\mathbf{x}) = \sum_i d(x_i)$, x_i is the i'th element of $\mathbf{x}$, and $d(x)$ is the smooth ℓ_1 loss defined by:

$$d(x) = \begin{cases} 0.5x^2, & \text{if } ||x||_1 \leq 1 \\ ||x||_1 - 0.5, & \text{otherwise.} \end{cases} \tag{15}$$

Similarly, the auxiliary decoder loss $\mathcal{L}_{\text{prop}}$ is a scene-level smooth ℓ_1 loss applied to the best of $K = 15$ joint proposals $\{\hat{Y}^{\text{prop}}_k\}_{k\in[K]}$:

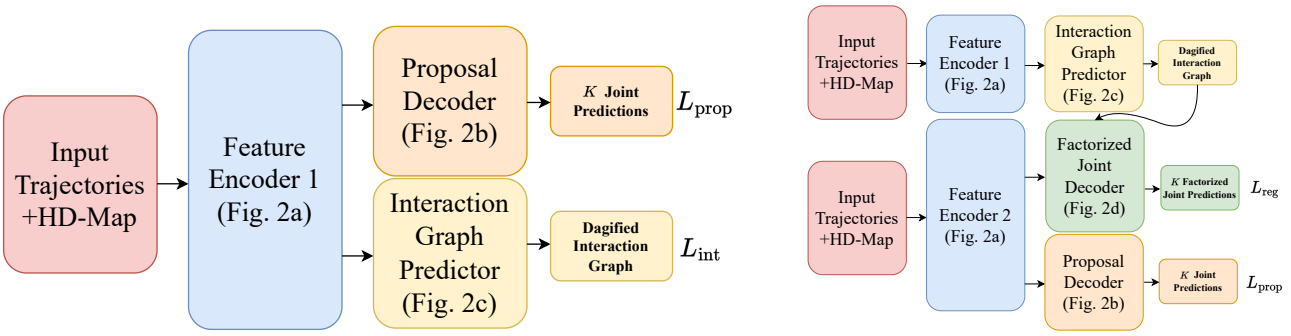
$$\mathcal{L}_{\text{prop}}(\{\hat{Y}^{\text{prop}}_k\}_{k\in[K]}, Y) := \mathcal{L}_{\ell_1}(\{\hat{Y}^{\text{prop}}_k\}_{k\in[K]}, Y). \tag{16}$$

We use the auxiliary proposal loss $\mathcal{L}_{\text{prop}}$ for training both the interaction graph predictor ($\mathcal{L}_1$ in Sec. 3.5) and the factorized joint decoder ($\mathcal{L}_2$ in Sec. 3.5) as both modules require explicit reasoning about interactions in the future trajectories, and thus *future-aware* agent features are beneficial for both modules.

B. FJMP System Diagram

B.1. Training Time

Figure 3 illustrates a high-level schematic of the FJMP architecture training stages at training time. We note that Feature Encoder 1 and Feature Encoder 2 consist of the same architecture as described in Sec. 3.2, but use separate weights.



(a) Interaction Graph Predictor training stage of FJMP.

(b) Factorized Joint Predictor training stage of FJMP.

Figure 3. High-level schematic of the training stages of FJMP.

B.2. Inference Time

Figure 4 illustrates a high-level schematic of the FJMP architecture and data flow at inference time. We note that at inference time the proposal decoders are removed.

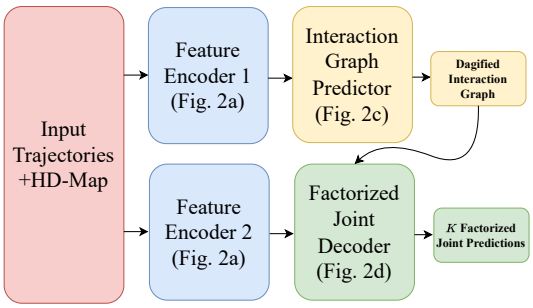


Figure 4. High-level schematic of the FJMP architecture at inference time.

C. Non-Factorized Baseline

We explain the non-factorized baseline described in Sec. 4 in more detail. The non-factorized baseline uses the same feature encoder architecture as FJMP, but the factorized joint decoder is replaced with a DECODE module consisting of a residual block and linear layer for simultaneously decoding K joint future trajectory coordinates, where diverse futures are obtained by appending a one-hot encoding of the modality index to the agent feature representation before feeding it into DECODE, as is done in FJMP. The DECODE module is the same architecture as the DECODE module used in FJMP. The non-factorized baseline is trained with the scene-level winner-takes-all smooth ℓ_1 loss $\mathcal{L}_{\text{reg}}$ that is described in Appendix A. The non-factorized baseline is trained with the same training hyperparameters as FJMP.

D. Non-Factorized Baseline Ablation

Model	Multiple Futures Method	Hyperparameter Configuration	Feature Encoder	minFDE	minADE	SMR	SCR
LaneGCN [30]	Separate Weights	LaneGCN	LaneGCN	0.935	0.300	0.223	0.233
-	One-hot Encoding	LaneGCN	LaneGCN	0.807	0.264	0.142	0.010
-	One-hot Encoding	FJMP	LaneGCN	0.713	0.227	0.113	0.006
Non-Factorized Baseline	One-hot Encoding	FJMP	FJMP	0.643	0.199	0.088	0.004

Table 5. Ablation study of the Non-Factorized Baseline model on the INTERACTION validation set. **Multiple Futures Method** denotes the method used to attain multiple joint futures. **Hyperparameter Configuration** denotes the hyperparameter settings for batch size, learning rate/step, and the number of training epochs. **Feature Encoder** denotes whether we use the LaneGCN feature encoder (LaneGCN) or the simplified LaneGCN feature encoder with fewer components (FJMP).

In Tab. 5, we perform an ablation study on the various components of the non-factorized baseline model on the INTERACTION dataset. First, we ablate using a one-hot encoding for multiple futures (*One-hot Encoding*) compared with using separate decoder weights for each joint future modality (*Separate Weights*), as is done in LaneGCN [30]. The one-hot encoding method significantly improves performance; this is because when using separate weights, the winner-takes-all training process quickly converges to one future joint modality, and thus the other decoders' weights never receive gradients for updating their weights. As a result, the collision rate (SCR) significantly improves when using the one-hot encoding method. Next, we ablate using the default hyperparameter configuration for LaneGCN compared with the FJMP hyperparameter configuration. Namely, LaneGCN trains for 36 epochs with a batch size of 128, with the learning rate decreasing by a factor of 10 at epoch 32. FJMP trains for 50 epochs with a batch size of 64, with the learning rate decreasing by a factor of 5 at epochs 40 and 48. The FJMP hyperparameter configuration significantly improves performance over the LaneGCN hyperparameter configuration. Finally, we ablate using the modified LaneGCN feature encoder (*FJMP*) consisting of a GRU for processing agent trajectories instead of LaneGCN's proposed ActorNet module, 2 MapNet layers instead of 4, and the A2L and L2L blocks removed. These modifications yield further improvements in validation performance.

E. INTERACTION Ablation Study

In Tab. 6, we repeat the FJMP ablation study conducted in Tab. 4 on the INTERACTION dataset. The results are consistent with Argoverse 2, showing that both the proposal decoder and teacher forcing are critical for performance.

Model	Prop?	TF?	minFDE	minADE	iminFDE	iminADE
Non-Factorized	✗	✗	0.643	0.199	0.688	0.210
FJMP	✗	✗	0.647	0.200	0.690	0.212
FJMP	✗	✓	0.644	0.200	0.688	0.212
FJMP	✓	✗	0.636	0.197	0.677	0.208
FJMP	✓	✓	**0.630**	**0.194**	**0.671**	**0.206**

Table 6. Ablation study of FJMP on the INTERACTION validation set. **Prop?** denotes whether we include the proposal decoder during training. **TF?** denotes whether we teacher-force the influencer trajectories during training.

F. Datasets

F.1. INTERACTION

INTERACTION requires predicting 3 seconds into the future given 1 second of past observations sampled at 10 Hz. INTERACTION contains 47,584 training scenes, 11,794 validation scenes, and 2,644 test scenes. A scene consists of a 4 s sequence of observations (1 s past, 3 s future) for each agent. INTERACTION contains pedestrians, bicyclists, and vehicles as context agents but only requires predicting vehicles in their multi-agent challenge. As bounding box length/width information is not provided for the pedestrian/cyclist labels, we set the length and width to a pre-defined value of 0.7m. We note that pedestrians and cyclists are not differentiated in the INTERACTION dataset.

F.2. Argoverse 2

Argoverse 2 requires predicting 6 seconds into the future given 5 seconds of past observations sampled at 10 Hz. Argoverse 2 contains 199,908 training scenes and 24,988 validation scenes. A scene consists of an 11 s sequence of observations (5 s past, 6 s future) for each agent. Argoverse 2 requires predicting 5 agent types: vehicle, pedestrian, bicyclist, motorcyclist, and bus. As bounding box length/width information is not provided in the Argoverse 2 dataset, we use the following predefined length/width in meters for each agent type to construct the interaction labels (length/width): vehicle (4.0/2.0), pedestrian (0.7/0.7), bicyclist (2.0/0.7), motorcyclist (2.0/0.7), bus (12.5/2.5).

G. Training Details

G.1. INTERACTION

The hidden dimension of FJMP is 128 except for the GRU history encoder, which has a hidden dimension of 256. The output of the GRU encoder is mapped to dimension 128 with a linear layer. We set $K = 6$ for the factorized decoder and $K = 15$ for the proposal decoders. For training the interaction graph predictor, we set $\gamma = 5$ and $\alpha = [1, 2, 4]$. We set $\epsilon_I = 2.5$ s. During training, we center and rotate the scene on a random agent, as an input normalization step. During validation and test time, we center and rotate the scene on the agent closest to the centroid of the agents' current positions. We use 2 MapNet layers, 2 L2A layers, and 2 A2A layers, where the L2A and A2A distance thresholds are set to 20 m and 100 m, respectively. We use all agents in the scene for context that contains a ground-truth position at the present timestep. As centerline information is not provided in INTERACTION, for each lanelet we interpolate P evenly-spaced centerline points, where $P = \min\{10, \max\{L, R\}\}$ and L, R are the number of points on the lanelet's left and right boundaries, respectively; that is, we restrict long lanelets to have a maximum of 10 evenly-spaced centerline points. At validation time, we consider for evaluation all vehicles that contain a ground-truth position at both the present and final timesteps. We train our model on the train and validation set with the same training hyperparameters before evaluating FJMP on the INTERACTION test set.

G.2. Argoverse 2

The details in Appendix G.1 apply to Argoverse 2 with the following exceptions. For training the interaction graph predictor, we set $\gamma = 5$ and $\alpha = [1, 4, 4]$. We set $\epsilon_I = 6$ s as interactions are comparatively more sparse in Argoverse 2. At validation time, we center on the ego vehicle. We increase the number of MapNet layers to 4 in Argoverse 2 to handle the larger amount of unique roadway. The L2A threshold is set to 10 m as the centerline points are comparatively more dense in Argoverse 2 than in INTERACTION. We use all scored, unscored, and focal agents in the scene for context that contains a ground-truth position at the present timestep. In the *Scored* validation setting (see Tab. 2), we consider for evaluation all

Actors Evaluated	Model	$\text{SMR}_{\text{Argoverse2}}$
Scored	Non-Factorized	0.264
	FJMP	0.259
	Δ	0.005
All	Non-Factorized	0.259
	FJMP	0.257
	Δ	0.002

Table 7. Non-Factorized Baseline vs. FJMP performance on Argoverse 2 SMR metric on the Argoverse 2 validation set. Δ denotes the difference in performance between FJMP and the Non-Factorized baseline.

scored and focal agents with a ground-truth position at both the present and final timesteps. In the *All* validation setting (see Tab. 2), we consider for evaluation all scored, unscored, and focal agents with a ground-truth position at both the present and final timesteps.

H. Collision Checker

To construct the interaction labels as described in Sec. 3.3, a collision checker is used to identify collisions between all pairs of timesteps in the future trajectories. We use the collision checker provided with the INTERACTION dataset. At each timestep, the collision checker defines each agent by a list of circles, and two agents are defined as colliding if the Euclidean distance between any two circles' origins of the given two agents is lower than the following threshold:

$$\epsilon_C := \frac{w_i + w_j}{\sqrt{3.8}}, \tag{17}$$

where w_i, w_j are the widths of agents i, j.

I. Miss Rate

For both Argoverse 2 and INTERACTION, we use the definition of a miss used in the INTERACTION dataset: a prediction is considered a "miss" if the longitudinal or latitudinal distance between the prediction and ground-truth endpoint is larger than their corresponding thresholds, where the latitudinal threshold is $\epsilon_{\text{lat}} := 1\,\text{m}$ and the longitudinal threshold is:

$$\epsilon_{\text{long}} := \begin{cases} 1, & \text{if } v \leq 1.4\,\text{m/s} \\ 1 + \frac{v-1.4}{11-1.4}, & \text{if } 1.4\,\text{m/s} \leq v \leq 11\,\text{m/s} \\ 2, & \text{otherwise,} \end{cases} \tag{18}$$

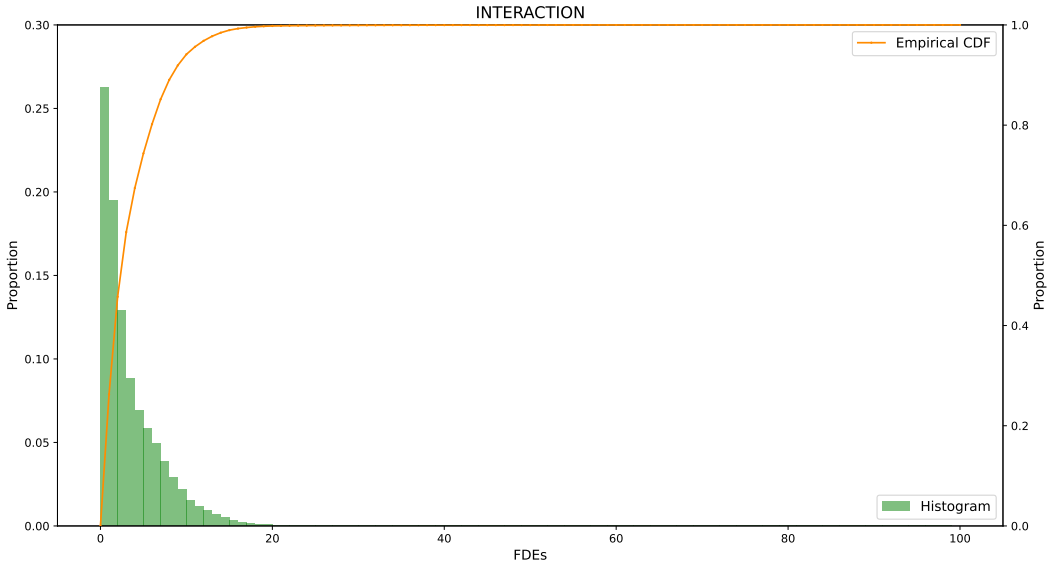
where v is the ground-truth velocity at the final timestep. We note that Argoverse 2 officially defines a miss as a prediction whose endpoint is more than 2 m from the ground-truth endpoint; however, we report all miss rate numbers in Tab. 2 using the miss rate definition in INTERACTION as it is a more robust measure of miss rate that takes into account the agent's velocity. For completeness, we report miss rate numbers for Argoverse 2 using the Argoverse 2 definition of a miss in Tab. 7.

J. Constant Velocity Model

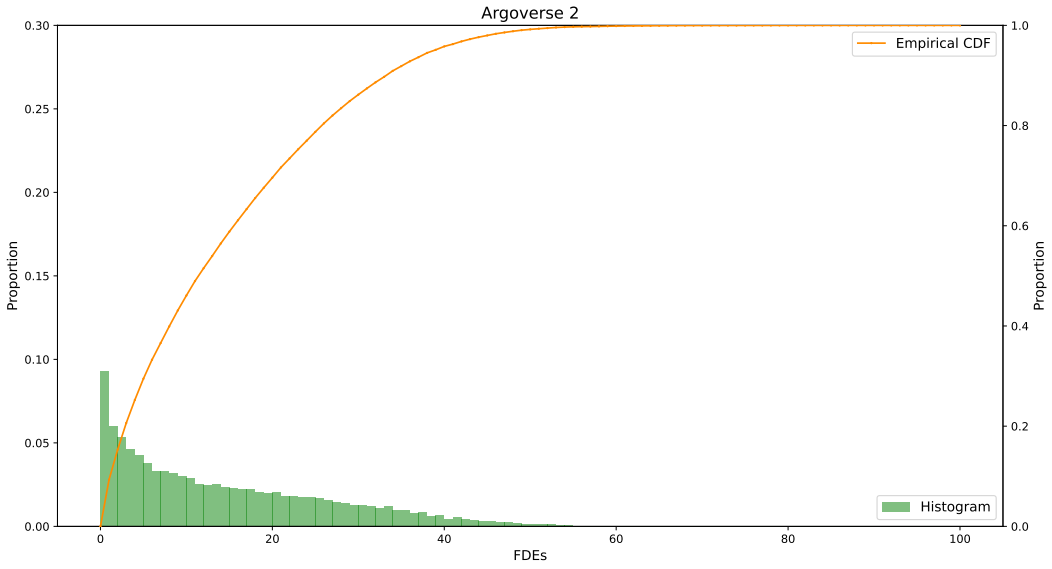
In Sec. 4, we identify the kinematically complex interactive agents in the datasets by filtering for agents that attain at least d m in FDE with a constant velocity model. An interactive agent is defined as an agent with at least one incident edge in the ground-truth interaction graph, where $\epsilon_I = 2.5\,\text{s}$, as is explained in Sec. 4. In this section, we describe the constant velocity model in more detail. The constant velocity model computes the average velocity over the observed timesteps and unrolls a future trajectory using the calculated constant velocity. Namely, the average velocity is calculated as:

$$\mathbf{v}_{\text{avg}} = \frac{1}{T_{\text{obs}}} \sum_{t \in [T_{\text{obs}}]} \mathbf{v}_t, \tag{19}$$

where $\mathbf{v}_t$ is the ground-truth velocity at timestep t. Using the constant velocity model, we calculate the agent-level FDE of all interactive agents in the INTERACTION and Argoverse 2 validation sets, respectively, where the FDE distributions are



(a)



(b)

Figure 5. Histogram of FDEs on interacting agents in (a) the INTERACTION dataset, and (b) the Argoverse 2 dataset. The left y-axis corresponds to the histogram and the right y-axis corresponds to the empirical cumulative distribution function (CDF).

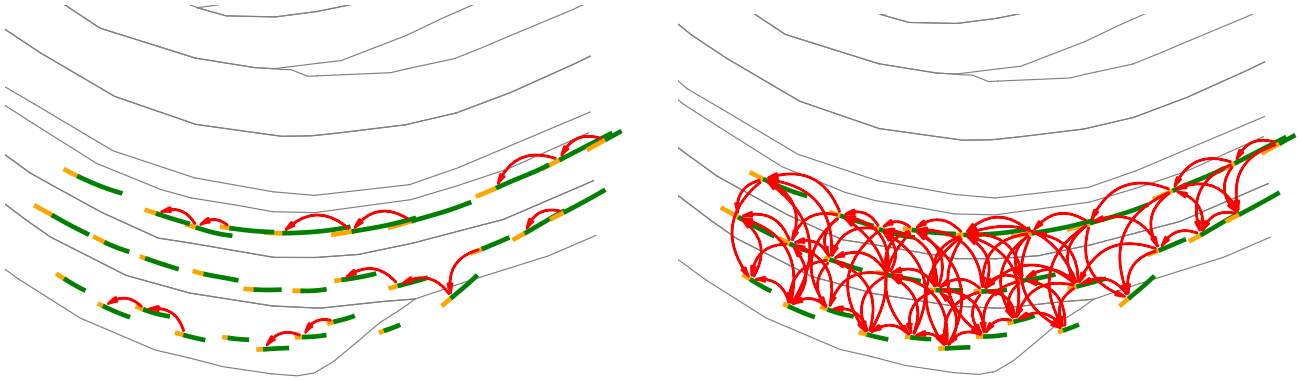
plotted in Fig. 5. We observe that a large proportion of the interactive agents have low FDE with a constant velocity model, especially in the INTERACTION dataset. By filtering out these kinematically simple agents, as is done in Sec. 4, we can assess the model's joint prediction performance on agents that are both interactive and kinematically complex. In Tab. 8, we report the number of interactive agents in the INTERACTION and Argoverse 2 validation sets that attain at least d m in FDE, for $d = 0, 3, 5$. We note that $d = 0$ corresponds to the number of interactive agents in the respective validation sets.

K. FJMP vs. M2I Interaction Graphs

Figure 6 illustrates the ground-truth interaction graph of a congested scene according to the FJMP and M2I heuristics, respectively. We observe that the M2I heuristic adds several superfluous edges, which would lead to unnecessary additional computation for the factorized decoder.

Dataset	d	Count
INTERACTION (112994)	0	50967
	3	21077
	5	13069
Argoverse 2 (248719)	0	37065
	3	29421
	5	26140

Table 8. Number of *interactive* agents in the INTERACTION and Argoverse 2 datasets that attain at least d m in FDE with a constant velocity model. In parentheses, we include the total number of evaluated agents (interactive + non-interactive) in the respective validation sets.



(a) Interaction graph generated with FJMP labeling heuristic.

(b) Interaction graph generated with M2I labeling heuristic.

Figure 6. Comparison of FJMP and M2I labeling heuristics on a congested scene from the INTERACTION dataset. The ground-truth pasts are indicated in yellow and the ground-truth futures are indicated in green. Lane boundaries are depicted as grey lines. Each red arrow points from an influencer agent to its corresponding reactor agent. We note that two agents at the bottom-right of the scene are on the shoulder of the lane.

Dataset	Edge Type	Edge Type Proportion
INTERACTION	`no-interaction`	0.955
	`m-influences-n`	0.037
	`n-influences-m`	0.008
Argoverse 2	`no-interaction`	0.973
	`m-influences-n`	0.015
	`n-influences-m`	0.013

Table 9. Edge type proportions in the INTERACTION and Argoverse 2 training set interaction graphs with the FJMP labeling heuristic.

L. Interaction Graph Predictor Performance

Table 9 reports the proportion of `no-interaction`, `m-influences-n`, and `n-influences-m` edges in the INTERACTION and Argoverse 2 training sets. Due to the severe class imbalance, we employ a focal loss when training the interaction graph predictor, as explained in Sec. 3.3.1. The edge type accuracies of the proposed interaction graph predictor on the INTERACTION and Argoverse 2 validation sets are reported in Tab. 10.

L.1. Ground-truth Interaction Graph Performance

Table 11 compares the performance of FJMP with two modified versions of FJMP: (1) we replace the predicted interaction graphs at inference time with the ground-truth interaction graphs; and (2) we replace the predicted interaction graphs during

Dataset	Edge Type	Edge Type Accuracy
INTERACTION	`no-interaction`	0.992
	`m-influences-n`	0.940
	`n-influences-m`	0.939
Argoverse 2	`no-interaction`	0.990
	`m-influences-n`	0.847
	`n-influences-m`	0.859

Table 10. Accuracy of each edge type on the INTERACTION and Argoverse 2 validation sets with the FJMP interaction graph predictor.

Model	Train IG	Inference IG	minFDE	minADE	iminFDE	iminADE
FJMP	Learned	Learned	1.963	0.812	3.204	1.273
FJMP	Learned	Ground-truth	1.947	0.807	3.165	1.265
FJMP	Ground-truth	Ground-truth	1.888	0.789	2.986	1.220

Table 11. FJMP with ground-truth vs learned interaction graphs at training and inference time on the Argoverse 2 validation set, All setting. For each metric, the best model is **bolded**. **Train IG** indicates the interaction graphs that are used during training, where **Learned** denotes the predicted interaction graphs from the interaction graph predictor and **Ground-truth** denotes the interaction graphs obtained from the labeling heuristic. The **Inference IG** column is interpreted similarly.

training and inference time with the ground-truth interaction graphs. The results in Tab. 11 indicate that the choice of interaction graph has a considerable effect on the performance of the factorized joint predictor, as indicated by an additional 4 cm improvement in iminFDE with the ground-truth interaction graph at inference time over the predicted interaction graph. Moreover, when the model is trained and evaluated with the ground-truth interaction graphs, we see a substantial increase in performance over FJMP with the learned interaction graphs. This indicates that further refinement of the interaction graph predictor may yield additional performance improvements with our FJMP design, which we leave to future work.

M. Qualitative Results

M.1. Argoverse 2

In this section, we show qualitative results on scenes in the Argoverse 2 validation set where we show side-by-side comparisons between FJMP and the Non-Factorized Baseline. In Fig. 7 and Fig. 8, for each row, the left panel shows the non-factorized baseline predictions, the middle panel shows FJMP predictions, and the right panel shows the predicted DAG. We visualize only the best scene-level modality to avoid clutter. In Fig. 7, we show examples where FJMP reasons properly in scenes with interactive pass-yield behaviours. In contrast, the non-factorized baseline incorrectly predicts conservative behaviour where the yielding vehicle avoids the passing vehicle's trajectory. In Fig. 8, we show qualitative examples where FJMP correctly identifies chains of leader-follower interactions, which in turn leads to more accurate leader-follower predictions than the non-factorized baseline. In Fig. 10, we illustrate two failure cases of the FJMP model. In both cases, an erroneous influencer future prediction negatively biases the downstream reactor prediction.

M.2. INTERACTION

Figure 9 shows qualitative results of FJMP on various scenes in the INTERACTION dataset, with all $K = 6$ scene-level modalities visualized. We emphasize FJMP's ability to produce accurate and scene-consistent predictions for scenes with a large number of interacting agents.

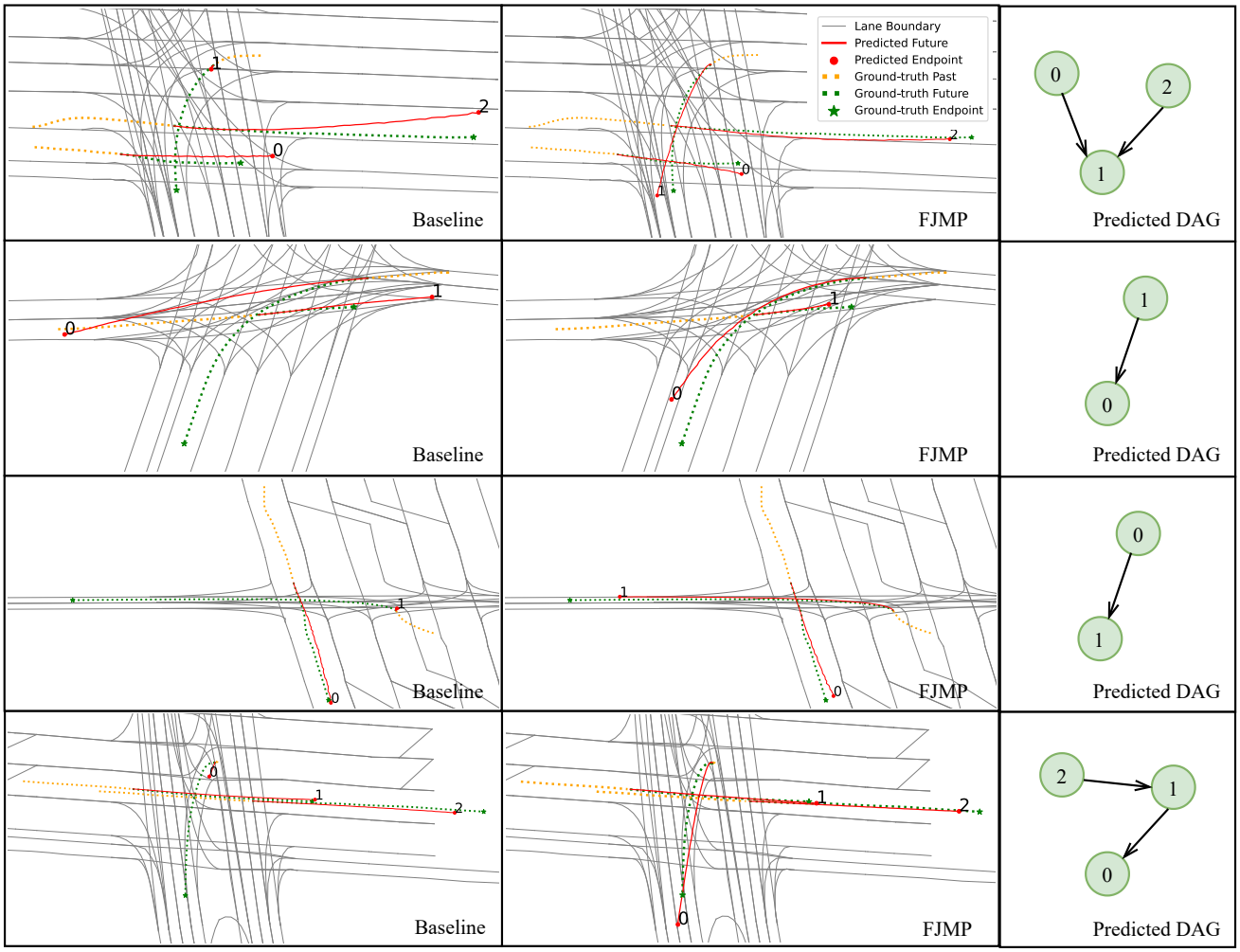


Figure 7. Qualitative examples of left-turn interactive scenes in the Argoverse 2 validation set. All predicted DAGs match the ground-truth DAG. In all scenes, FJMP correctly identifies the passing vehicle as the influencer and the left-turning vehicle as the reactor. The Non-Factorized baseline consistently predicts overly conservative behaviour that avoids the influencer trajectory. In contrast, FJMP consistently captures the proper left-turn behaviour.

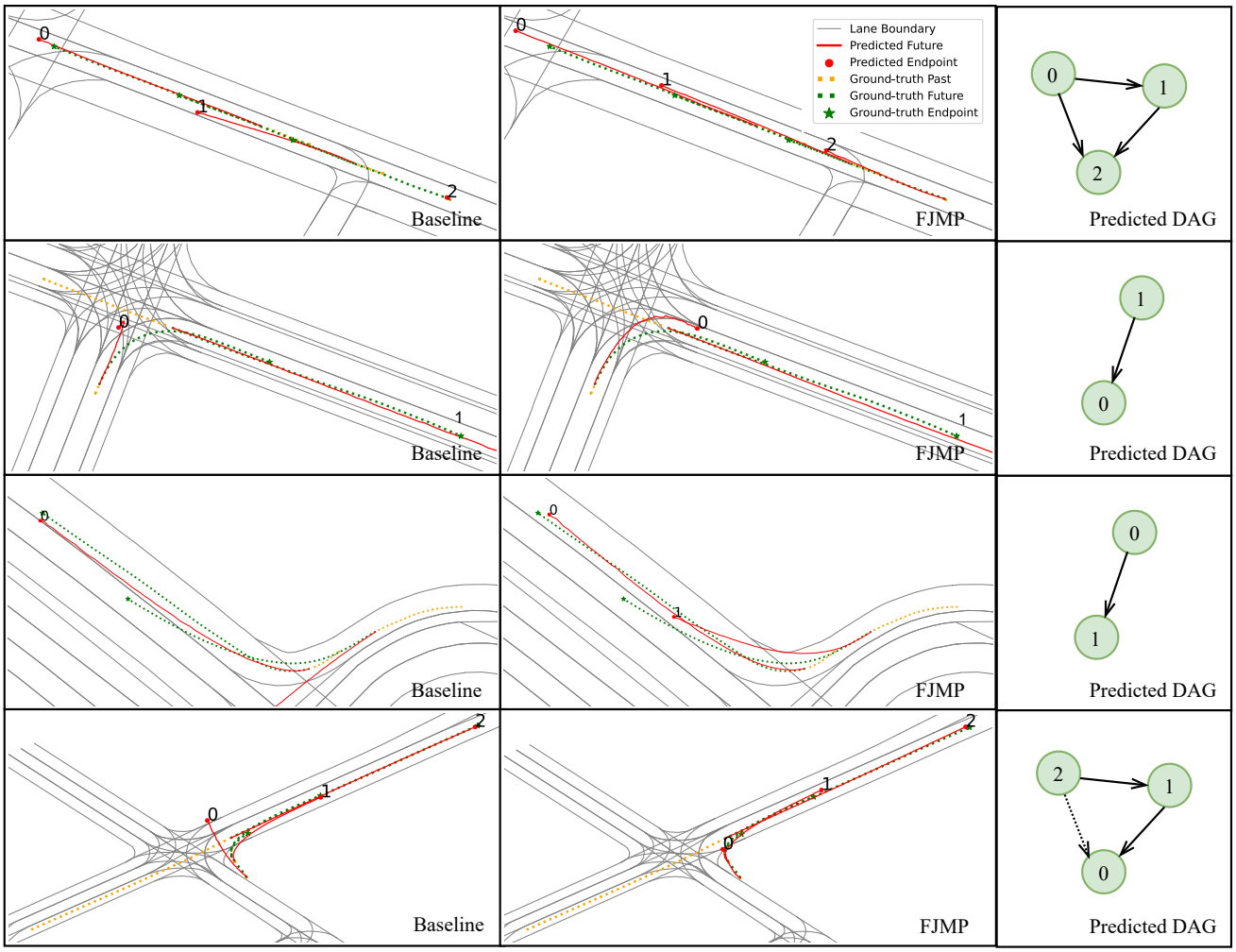


Figure 8. Qualitative examples of leader-follower interactive scenes in the Argoverse 2 validation set. Predicted DAGs are shown on the right, where true positive edges are indicated in solid black and true negative edges are shown in dotted black. In all of the above scenes, FJMP correctly predicts chains of influencer-reactor relationships. In the first row, the non-factorized baseline predicts conservative behaviour for the trailing vehicle. In contrast, FJMP predicts proper leader-follower behaviour for the trailing vehicle (leaf node in the DAG). In the second and third rows, the right-turn mode of the trailing vehicle is missed by the non-factorized baseline, whereas FJMP correctly identifies the right-turn mode due to correctly identifying the leader-follower interaction. In the last row, the non-factorized baseline predicts scene-incompliant behaviour for the trailing vehicle whereas FJMP predicts proper leader-follower dynamics reflecting the predicted DAG.

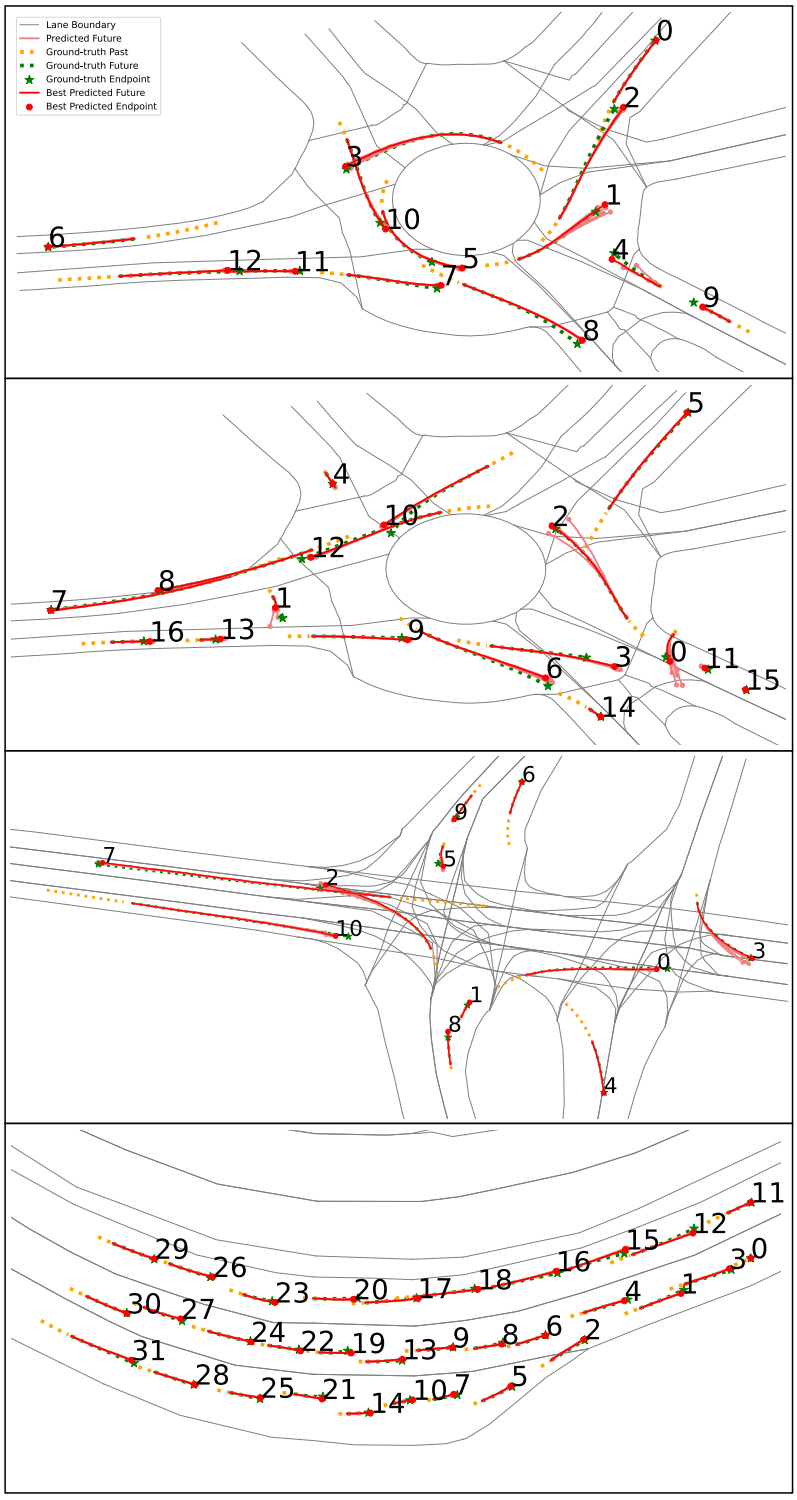


Figure 9. Qualitative examples of FJMP on agent-dense scenes in the INTERACTION dataset.

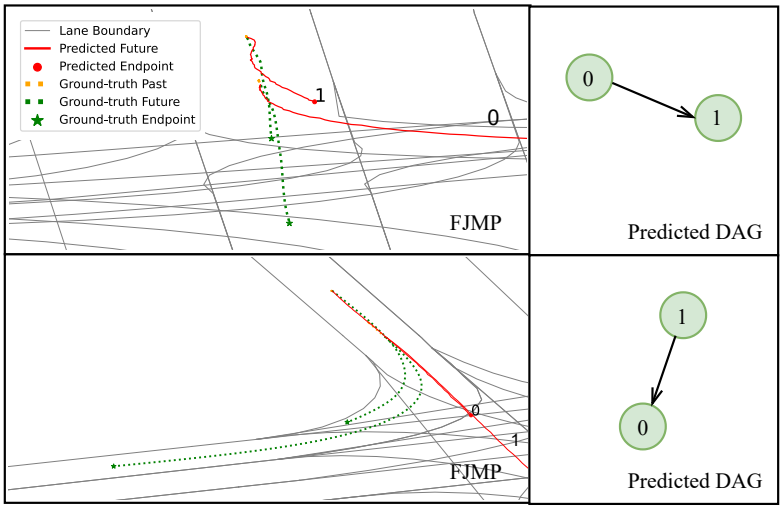


Figure 10. Qualitative examples of failure cases of the FJMP model. All predicted DAGs match the ground truth. In both rows, the interaction graph is correctly predicted; however, the influencer trajectory is erroneously predicted, which negatively biases the reactor's prediction to follow the influencer.