

A MOTIVATING QUESTIONS

Inspired by the appendix of (Karamcheti et al., 2023), in this section, we list some motivating questions that may arise from reading the main paper.

Q1. The features used when computing the preference distance are manually designed, what are the benefits and limitations?

The alignment between generated agents’ behavior and expert demonstrations requires measuring the distance between their occupancy measures in a semantically meaningful feature space. In this work, we use manually designed features that are well-validated and widely used in the autonomous driving industry. This allows us to conduct controlled experiments and evaluate the performance of our approach using proven, effective features. Additionally, in industry settings, it is often necessary to align a pre-trained motion model to specific criteria during post-training, such as progress. Using semantically meaningful features enables efficient and effective alignment for such cases. While encoding the generated traffic simulation into a learnable feature space could potentially provide more information and improve the informativeness of the preference distance, it also risks introducing spurious correlations (Zhang et al., 2020) and requires additional tasks to train the encoder to extract valuable features. Recent work (Tian et al., 2023) proposes to use human input to explicitly calibrate the feature space learning process to reduce the risk of learning spurious correlations, but the proposed method is only validated in simple simulation settings. We are excited to explore this direction in future work to further enhance the performance of our approach.

Q2. The preference distance can better reflect the alignment between a generated motion and the expert demonstrations, can it be used to directly train the motion model end-to-end and provide better results?

In this work, we focus specifically on LLM-type token-prediction models, as they are becoming the backbone of motion models in various embodied tasks. These auto-regressive models typically use a teacher forcing training scheme for efficiency, which is not naturally compatible with the preference distance. While it is possible to use preference distance as a loss signal to train the auto-regressive model, this approach introduces additional complexity in the training pipeline and significantly increases computational cost, as it requires solving the optimal transport problem at each step.

Q3. Why the performance is still worse than SOTA methods even with preference alignment?

In Table 1, while the aligned model underperforms compared to some state-of-the-art (SOTA) models, we believe this is primarily due to the architecture and capacity limitations of the reference model. We anticipate that applying our approach to larger SOTA models would result in significant performance improvements, as our method provides more nuanced alignment and could fully leverage the capabilities of more advanced architectures.

Q4. What makes the paper different from previous works that use optimal transport based reward for robot behavior learning via RL?

The optimal transport method has been used to generate reward signals by measuring the distance between a rollout and expert demonstrations in single-agent RL settings (Xiao et al., 2019; Luo et al., 2023; Tian et al., 2023). In contrast, our work focuses on post-training alignment of multi-agent motion token-prediction models using expert demonstrations. To overcome the overly conservative assumption in previous works, which treat all model-generated samples as unpreferred, we leverage optimal transport to construct preference rankings among sampled rollouts. While it is feasible to convert the preference distance into per-step rewards and align the model using RLHF, this approach would require significantly more computational resources, as optimal transport must be solved in a multi-agent setting at every RL step. We are excited to explore the compute-performance trade-off between using preference distance in direct alignment versus RLHF in future work.

Q5. Why is using expert demonstrations as the preferred samples in preference learning less informative compared to constructing preference data using the generated samples?

The expert demonstrations are first used to fine-tune the model with (1), thus the likelihood of expert demonstrations are already much higher than the sampled generations from the model. If using expert demonstrations as preferred samples and all the sampled generations as unpreferred

samples in preference alignment, the contrastive loss can be improved but this will further suppress the likelihood of the samples overall.

B EXTENDED RELATED WORKS

Motion generation as next-token prediction. Motion generation has traditionally been approached using one-shot prediction techniques, where the entire motion sequence is generated in a single forward pass conditioned on scene information (Nayakanti et al., 2023). However, these approaches often struggle with modeling long-term dependencies and maintaining temporal coherence between actions. Next-token prediction framework — where each subsequent token is predicted based on the previously generated ones — has proven highly successful in language modeling (Achiam et al., 2023). Similar to language, human or robotic motion unfolds as a series of continuous actions over time, with each movement serving as a “token” that depends on prior movements. Next-token prediction provides a natural autoregressive framework for generating these sequential actions, making it a powerful tool for motion generation in domains such as autonomous driving (Seff et al., 2023; Philion et al., 2024), robot manipulation (Brohan et al., 2023), and humanoid locomotion (Radosavovic et al., 2024). In our work, we explore the use of next-token prediction for multi-agent motion generation. However, unlike previous approaches, we focus on aligning a pre-trained motion generation model with human preference.

Alignment of Large Language Models using preference feedback. Next-token prediction optimizes for local coherence between individual tokens but often lacks long-term consistency between the generated sequence and the ground truth. This misalignment between the training objective and the human internal reward function, which governs their behavior, can lead to suboptimal outcomes and even safety-critical motions in embodied settings. To address this misalignment, both online and offline methods have been developed to better align large language models (LLMs) with human values using preference feedback. One prominent online approach is Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), which fine-tunes the model by first learning a reward model and then using reinforcement learning to optimize the generative model’s behavior to maximize the learned reward. However, this two-phase approach introduces significant complexity to the alignment process and incurs substantial computational costs. As an alternative, offline methods—often referred to as direct alignment methods (Rafailov et al., 2024b)—bypass the reward learning step by directly optimizing the model to maximize the margin between the log-likelihood ratio of preferred samples and that of unpreferred ones. While offline methods have demonstrated empirical efficiency and are often more favorable compared to online methods (Tunstall et al., 2023), collecting preference feedback remains time-consuming and difficult to scale. This challenge is especially pronounced in embodied settings, where human annotators must analyze intricate and nuanced multi-agent motions, making the process even more labor-intensive.

C MOTION GENERATION MODEL

We follow the MotionLM architecture to implement our multi-agent motion generation model (Seff et al., 2023).

Our model utilizes an early fusion network as the scene encoder to encode multi-modal scene inputs. The scene encoder integrates multiple input modalities, including the road graph, traffic light states, and the trajectory history of surrounding agents. These inputs are first projected into a common latent space through modality-specific encoders. The resulting latent embeddings for each modality are then augmented with learnable positional encodings to preserve spatial and temporal relationships. The augmented embeddings are concatenated and passed through a self-attention encoder, which generates a scene embedding for each modeled agent. These scene embeddings are subsequently used by the autoregressive model, via cross-attention, to predict the actions of each agent. An agent’s action token is obtained via discretizing acceleration control into a finite number of bins (169) and a joint action token denotes the collection of all agents’ action token in the scene. Please refer to Section App.A of (Seff et al., 2023) about the bins used for tokenization.

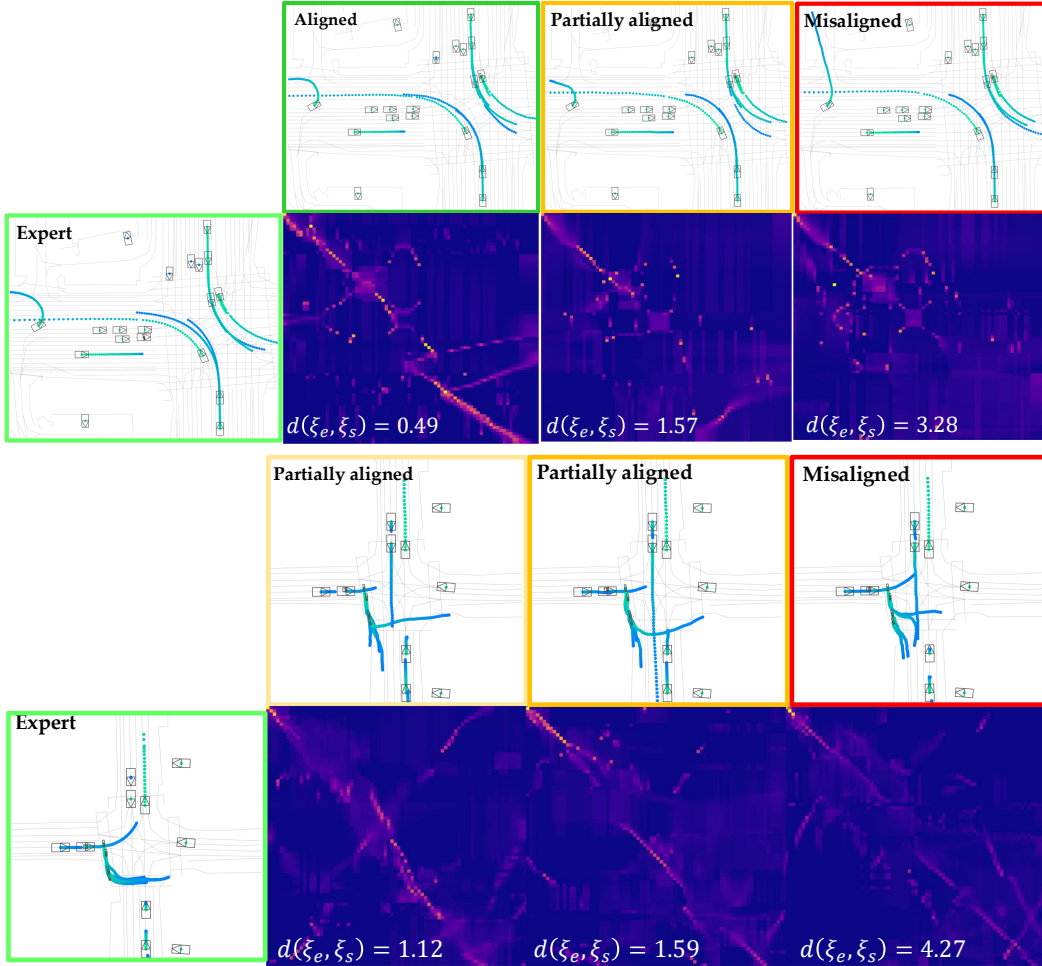


Figure 8: **Alignment visualization.** The heat map visualizes the optimal coupling between a generated traffic simulation and the ground truth scene evolution. More peaks along the diagonal indicate better alignment between the behaviors (i.e., a smaller preference distance).

D QUALITATIVE EXAMPLES OF PREFERENCE DISTANCE

In Figure 8, we show qualitative examples that illustrates how the coupling matrix reflects the behavior alignment between generated traffic simulations and the expert demonstrations. We see that traffic simulations with smaller preference distance demonstrate more aligned behavior compared to the expert demonstration.

E QUALITATIVE EXAMPLES DEMONSTRATING THE EFFECTIVENESS OF OUR APPROACH

In Figure 9, we include more visualizations to demonstrate the performance of our approach. For each model, we sample 64 rollouts from the model and we show the most-likely one.

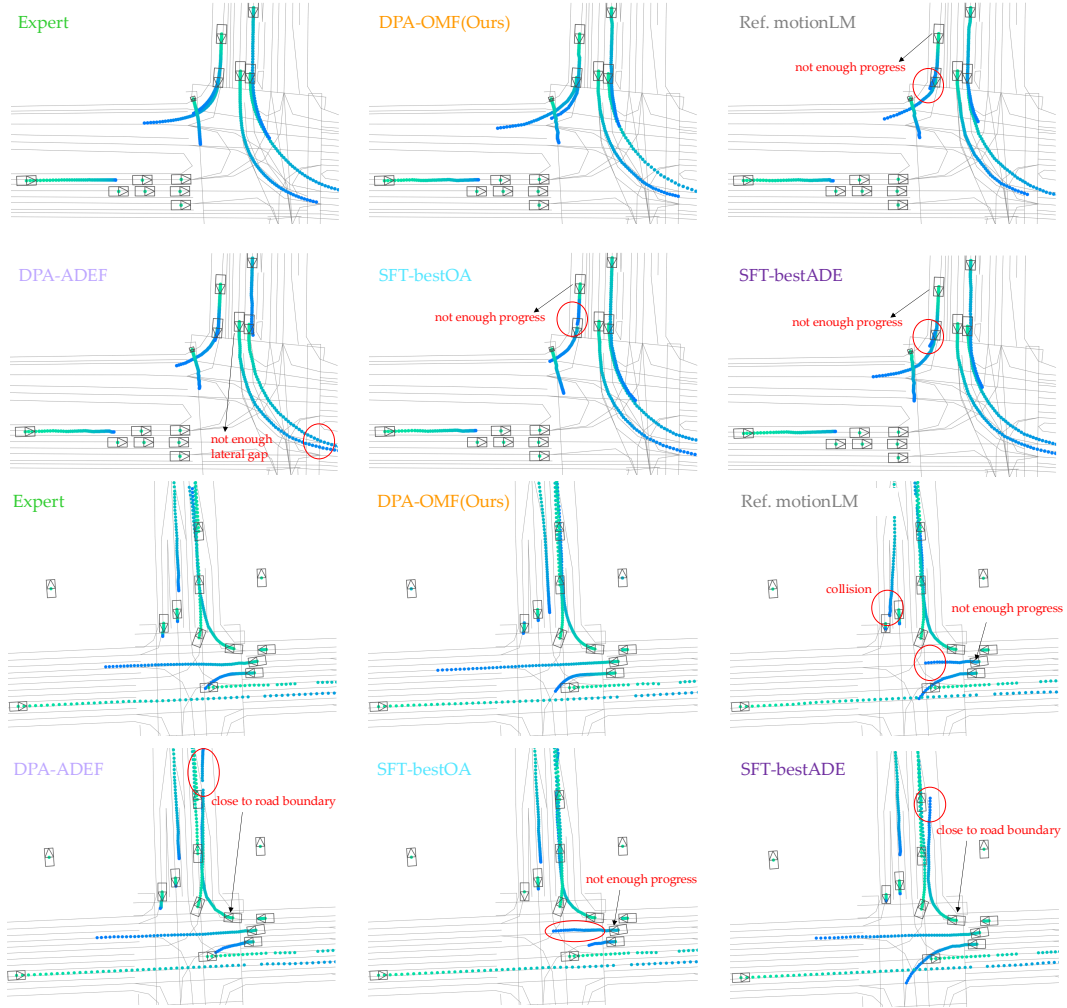


Figure 9: **Traffic simulation generation visualization.** Our approach produces traffic simulations that are more closely aligned with expert demonstrations, while baseline models generate simulations that are only partially aligned or misaligned.

F ADDITIONAL RESULT - BIAS INTRODUCED BY THE HETEROGENEITY OF THE PREFERENCE DATA

In Section 5.3, we show that using expert demonstrations as preferred samples and model generations as unpreferred samples results in increasing the likelihood of expert demonstrations without significantly affecting the likelihood of either preferred or unpreferred generated samples. This suggests that the model struggles to associate the features that make expert demonstrations preferred with the generated preferred samples. To further explore this, we conducted a separate experiment demonstrating how a discriminative objective using expert demonstrations as positive samples and model generations as negative samples can lead to spurious correlations.

In this experiment, we trained a discriminator using a contrastive objective to distinguish between expert demonstrations and model generations. The discriminator achieved a classification accuracy of 0.83 on the evaluation dataset, indicating it can reasonably classify motions as either expert

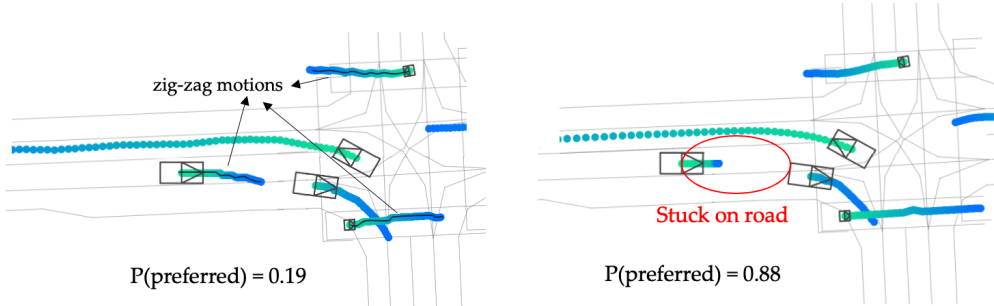


Figure 10: **Spurious correlation introduced by the heterogeneity of the preference data.** The model relies heavily on trajectory smoothness to differentiate between expert demonstrations and model generations, which can lead to incorrect predictions about preferred behaviors. For example, in the traffic simulation on the left, the trajectories exhibit zig-zag patterns but demonstrate more human-like behaviors compared to the simulation on the right. However, the model incorrectly predicts the likelihood of the left human-like simulation being preferred by humans as only 0.19 and predicts the likelihood of the right unhuman-like simulation being preferred by humans as 0.88, highlighting its inability to fully capture nuanced human preferences.

demonstrations or model generations. When the discriminator was used to rank pairs of model-generated motions, we observed a pattern: motions with zig-zag trajectories are often classified as unpreferred, while relatively smooth motions are classified as preferred, even when there is unhuman-like behaviors (e.g., stuck on roads as shown in Figure 10).

This behavior arises because of the heterogeneity of the two data sources: most human demonstrations exhibit smooth motions, while model generations are not constrained by vehicle dynamics. Consequently, the contrastive objective may incentivize the model to pick up this spurious correlation, prioritizing smoothness over other critical attributes such as staying on the road.

G ADDITIONAL RESULT - THE COST OF QUERYING HUMANS FOR PREFERENCES IN MULTI-AGENT TRAFFIC GENERATIONS

To quantify the human cost associated with providing preference rankings for multi-agent traffic simulations, we conducted an Institutional Review Board (IRB)-approved human subject study to measure the effort required. In this study, we presented paired traffic simulations to participants and asked them to rank the pairs based on how realistic the simulations were compared to their personal driving experience. We varied the number of traffic agents in the simulations and recorded the time needed to provide rankings. Five participants ranked 500 pairs of traffic simulations, and Table 4 summarizes the time required to complete this task. The results show a clear trend: as the number of traffic agents increases, the time required for human annotators to rank simulations grows significantly.

Num. of agents in the scene	1	10	20	40	80
Average time used for ranking [s]	0.7	4.9	9.8	29.4	42.1

Table 4: Average time required for a human to rank traffic simulations.

Although this study was conducted under time constraints and is not exhaustive, it provides an useful estimate of the human cost for constructing preference rankings at scale. Specifically, for the preference data used in our experiments, the estimated average time required for one human annotator is approximately **633** days. This result underscores the practical challenges of scaling preference ranking annotations in multi-agent scenarios, motivating our approach to leverage existing demonstrations to construct preference rankings efficiently.