1 Detailed Rebuttal for Reviewer PGtc

We sincerely thank you for your valuable and constructive feedback on our work. We have carefully addressed all of your comments and made corresponding improvements and additions to the manuscript in order to resolve any concerns you may have.

Below, we provide detailed responses to each of your comments. For ease of reference, we denote reviewer Weaknesses as W and Questions as Q in our replies. Relevant figures and tables can be found in the anonymized supplementary material linked accordingly.

• **W1 and Q1: Scalability Comparison.** To ensure fair and comparable evaluation, we adopt similar experimental settings to prior diffusion-based offline MARL works [1-3], where most tasks involve fewer than 10 agents. However, this does not suggest our method is limited to small-scale scenarios.

As detailed in Appendix 7.4.2, we compare the policy sampling time of our method with MADIFF, showing that despite the additional cost from structural diffusion, our approach remains computationally competitive.

To assess scalability and collaborative performance in larger-scale settings, we further conduct experiments on the MPE Spread task with 8, 16, 32, and 64 agents. As presented in Table 1 of the anonymized supplementary material, MCGD consistently outperforms MADIFF and DOM2 across all scales, with performance gains increasing as the number of agents grows. These results demonstrate that our graph diffusion-based design not only scales well but is also more effective in modeling complex multi-agent coordination.

References:

[1] Beyond Conservatism: Diffusion Policies in Offline Multi-agent Reinforcement Learning, Li et al, CoRR 2023.

[2] Madiff: Offline multi-agent learning with diffusion models, Zhu et al, NeurIPS 2024.

[3] Diffusion-based Episodes Augmentation for Offline Multi-Agent Reinforcement Learning, Oh et al, ICML 2024.

2 Rebuttal for Reviewer YBYW

We sincerely thank you for your valuable and constructive feedback on our work. We have carefully addressed all of your comments and made corresponding improvements and additions to the manuscript in order to resolve any concerns you may have.

Below, we provide detailed responses to each of your comments. For ease of reference, we denote reviewer Weaknesses as W and Questions as Q in our replies. Relevant figures and tables can be found in the anonymized supplementary material linked accordingly.

• **W1.1: Action Selection in Continuous Space.** In the policy sampling process, for each agent n_i, we generate N andom action samples from the continuous action space to construct a candidate set. We then evaluate each candidate using the trained Q-function $\mathcal{Q}_{\phi_i}$ and select the action with the highest Q-value (Line 5 in Algorithm 1).

This replaces the Gaussian noise initialization commonly used in prior diffusion-based methods [1, 2], providing a more informed and value-guided sampling strategy. Since the optimal action is selected from a finite set of sampled candidates, this approach is naturally applicable to both discrete and continuous action spaces, and does not require differentiability or closed-form maximization over actions.

The selected action is then used to construct the coordination graph and initialize the graph diffusion process.

• **W1.2 and W3.2: Explanation of Average Observation.** To reduce model complexity and maintain scalability, we process each neighboring observation using a parameter-shared MLP, followed by a mean pooling operation over the resulting features. This replaces concatenation, which can significantly increase parameter count with more neighbors. By limiting the number of neighbors to those with higher similarity, the averaging operation is performed over semantically similar features, thereby mitigating information loss.

We further validate this design through an ablation study on the SMAC benchmark, comparing two variants: MCGD-AO (with observation averaging) and MCGD-FC (with feature concatenation). As shown in Table 2 of the anonymized supplementary material, MCGD-AO achieves both higher average rewards and lower computation cost across all tasks. The inferior performance of MCGD-FC is largely due to increased parameterization and optimization difficulty, supporting the effectiveness and generality of our averaging-based approach.

Regarding testing, since our setup is fully decentralized and other agents' observations are not directly accessible, we follow a standard practice in MARL literature by using the agent's own observation in place of the averaged neighbor input, ensuring consistency between training and testing procedures.

• **W1.3: Gaussian Diffusion over Discrete Actions.** For discrete action spaces, we employ a one-hot encoding for action representation and apply a softmax-based decoding at the output of the diffusion model. This allows us to embed discrete actions into a continuous latent space during the Gaussian diffusion process, while enabling valid discrete action reconstruction at inference.

• **W2: Rationale of Covariance Matrix.** The node attribute A_t represents the individual agent's action within the dynamic coordination graph—effectively forming the joint action when aggregated across all agents. In continuous action spaces, this corresponds to raw action vectors, while in discrete spaces, we use one-hot encoded actions embedded into the continuous diffusion process.

The purpose of anisotropic diffusion is to extend prior diffusion-based offline MARL methods [1-3] by modeling agent-wise action uncertainty while preserving the structural properties of multi-agent coordination. Inspired by [4], we leverage a learnable covariance matrix in the forward noising process to capture directional variances and maintain the consistency of coordination dynamics.

We intentionally avoid modifying the mean of the Gaussian distribution, as prior work has shown that dynamically altering the mean can destabilize training. By customizing only the covariance, we achieve anisotropic diffusion while ensuring convergence and structural integrity throughout the forward process.

• **W3.1: Observation Similarity for Collaboration.** Our method utilizes observation similarity primarily in two components: (1) initializing the dynamic coordination graph, and (2) defining the transition matrix in categorical diffusion.

For graph initialization, we follow prior work on graph-based interaction modeling [5], using observation similarity to identify initial neighbor sets for information sharing. Importantly, this initialization

serves only as a flexible starting point—the subsequent diffusion process adaptively refines the coordination structure, allowing the model to capture meaningful interactions even among agents with differing characteristics. This mitigates over-reliance on raw similarity and promotes more diverse cooperation.

In categorical diffusion, we compute cosine similarity between agent observations to define transition probabilities for edge rewiring. This design aligns with established formulations in categorical diffusion [6], and provides a principled way to control structural evolution based on relational cues.

Moreover, our framework is not limited to cosine similarity-any similarity metric with values scaled to [0,1] can be used, offering flexibility across environments with varying observation semantics. This adaptability enhances the generality and robustness of our approach beyond the tested settings.

• **W4: Learned Coordination Graph.** To illustrate how the learned coordination graph evolves during task execution, we provide a case study in Figure 1 of the anonymized supplementary material, based on the MPE Spread task. The horizontal axis indicates timesteps, and the vertical axis shows different experimental settings.

Initially, despite initializing with a nearest-neighbor graph, large positional differences between agents caused the forward diffusion process to disrupt coordination edges. As a result, the model predicted no edges and agents acted independently. As agents moved closer, the learned graph gradually recovered the underlying coordination structure, enabling effective collaboration such as landmark assignment and collision avoidance.

In a modified setting where Agent 0's speed was reduced, coordination edges emerged primarily between the other agents. Agent 0, being slower, required more timesteps to engage in coordination, delaying the full graph reconstruction.

In a more extreme case, where Agent 0 was inactive, it remained isolated, and coordination was exclusively formed between the other two agents.

These visualizations demonstrate the adaptive nature of the learned graph, which dynamically reflects the agents' interaction context and task demands.

• **W5: Adjusting for Figure 5.** We have revised Figure 5 by reducing the color saturation to improve visual clarity and presentation quality. We hope the updated figure offers a more comfortable and interpretable visualization experience.

References:

[1] Beyond Conservatism: Diffusion Policies in Offline Multi-agent Reinforcement Learning, Li et al, CoRR 2023.

[2] Madiff: Offline multi-agent learning with diffusion models, Zhu et al, NeurIPS 2024.

[3] Diffusion-based Episodes Augmentation for Offline Multi-Agent Reinforcement Learning, Oh et al, ICML 2024.

[4] Directional diffusion models for graph representation learning, Yang et al, NeurIPS 2023.

[5] Graph Convolutional Reinforcement Learning, Jiang et al, ICLR 2020.

[6] Graph-Constrained Diffusion for End-to-end Path Planning, Shi et al, ICLR 2024.

3 Rebuttal for Reviewer PSeB

We sincerely thank you for your valuable and constructive feedback on our work. We have carefully addressed all of your comments and made corresponding improvements and additions to the manuscript in order to resolve any concerns you may have.

Below, we provide detailed responses to each of your comments. For ease of reference, we denote reviewer Weaknesses as W and Questions as Q in our replies. Relevant figures and tables can be found in the anonymized supplementary material linked accordingly.

• **W1: Diversity Metrics in Anisotropic Diffusion.** We appreciate the reviewer's suggestion and agree that better quantification of action diversity strengthens our claim.

While mutual information-based metrics [1] are insightful, they often require estimating conditional entropy involving agent identity variables, which can be challenging and computationally intensive. Instead, we adopt the System Neural Diversity (SND) metric [2], which offers a tractable and reliable measure of behavioral diversity across agents.

In our experiments on SMAC benchmark tasks (3m, 2s3z, 5m6m, and 8m), we compute SND by first sampling N_1 observations, then generating N_2 independent action samples per agent under each MARL method. We construct pairwise distance matrices over these action sets and estimate SND using the Sinkhorn divergence.

As shown in Table 3 of the anonymized supplementary material, MCGD consistently achieves higher SND scores than diffusion-based baselines such as MADIFF and DOM2 across all tasks. Notably, in tasks involving more agents and greater heterogeneity in unit types, the advantage of MCGD becomes more prominent, as it models both coordination structure and per-agent action diversity more effectively.

• **W2 and Q1: Intuition behind Transition Matrix.** The transition matrix Q in Eq. 6 is constructed based on the cosine similarity between agent observations. The core intuition is that agents in similar states are more likely to share functional roles or exhibit similar coordination patterns. Thus, a higher cosine similarity implies a greater likelihood for agent n_j to replace agent n_i in its current coordination structure, reflected by a higher Q_{ij}. This models structural variability in a principled way during categorical noising.

This design is inspired by prior categorical diffusion work [3], which adaptively defines the transition matrix to guide edge perturbations. While we use cosine similarity for its simplicity and bounded range, our framework is compatible with alternative metrics (e.g., RBF kernels or learned embeddings), provided the output can be scaled to valid transition probabilities in [0, 1]. This ensures flexibility across different tasks and observation modalities.

• **W3: Continuous Action Attribute.** In the continuous action space, the node attribute matrix A_t encodes the raw actions of all agents, where each row corresponds to an individual agent and the dimension d matches the dimensionality of the agent's action space. Thus, A_t represents the joint action as a stack of per-agent actions indexed accordingly.

During inference, the predicted action matrix $\hat{A}_t$ maintains the same structure. Under decentralized execution, each agent n_i extracts its own action by retrieving the i-th row of $\hat{A}_t$, ensuring consistency with the fully decentralized setting.

For discrete action spaces, we adapt the Gaussian diffusion process by representing each agent's action using a one-hot vector of dimension d, where d is the number of discrete actions. The denoised continuous output is then passed through a softmax layer, and the final discrete action is selected via an argmax operation over the softmax probabilities.

• **W4, Q2, and Q3: Q-loss in Anisotropic Diffusion.** As highlighted in prior work [4, 5], optimizing only the standard surrogate loss in diffusion models [6] is insufficient for effective offline RL. To address this, we incorporate a conservative Q-loss term into our anisotropic diffusion framework to enhance policy learning under offline constraints.

Specifically, for each agent n_i, we introduce a critic network $\mathcal{Q}_{\phi_i}$, which estimates Q-values based on the agent's own action and the averaged observations of its neighbors. The critic is trained using a temporal-difference loss:

$$\mathcal{L}_{cri}(\phi_i) = \mathbb{E}_\tau \left[\left[r_i^t + \gamma \mathcal{Q}_{\phi_i}(\overline{o}_i^{t+1}, a_i^{t+1}) - \mathcal{Q}_{\phi_i}(\overline{o}_i^t, a_i^t) \right]^2 + \zeta \left[\log \sum_{\tilde{a}_i^t} \exp(\mathcal{Q}_{\phi_i}(\overline{o}_i^t, \tilde{a}_i^t)) - \mathcal{Q}_{\phi_i}(\overline{o}_i^t, a_i^t) \right] \right], \tag{1}$$

where $\overline{o}_i^t$ denotes the mean of the processed neighboring observations, extracted via a shared MLP followed by averaging. This design choice reduces parameter count and encourages generalization. Although averaging may lose some granularity, we mitigate this by limiting neighbor count and ensuring local similarity.

To empirically support this design, we conducted ablations comparing MCGD-AO (with averaged observations) and MCGD-FC (with feature concatenation) on SMAC tasks. As shown in Table 2 of the anonymized supplementary material, MCGD-AO achieves superior performance and lower training cost. This validates both the efficiency and practical advantage of using averaged neighboring observations in Q-value estimation.

While the incorporation of averaged neighbor features into Q-functions is relatively rare, our results suggest it provides a tractable and effective approximation of local agent context in cooperative settings.

• **W5: Generated Coordination Graph.** While the ground truth edge matrix E used for training corresponds to a predefined nearest-neighbor (NN) graph, our denoising network learns to generate coordination structures that go beyond this static prior.

As visualized in Figure 1 of the anonymized supplementary material, the diffusion process does not merely reconstruct the NN graph, but adapts the edge structure based on the evolving agent dynamics. For example, in the early timesteps of the MPE Spread task, the agents are far apart and the denoised graph remains sparsely connected, allowing agents to act independently. As agents converge spatially, the learned graph increasingly resembles the NN structure, enabling effective collaboration.

Importantly, in scenarios with asymmetric dynamics (e.g., Agent 0 moving slowly or being inactive), the predicted graph deviates from the fixed NN structure by prioritizing edges among active agents. Thus, although the NN graph serves as a training prior, the denoising model learns to infer context-aware, task-adaptive coordination structures that outperform static heuristics. This ability to dynamically reshape connectivity is a key advantage of our diffusion-based framework.

• **W6: Capturing Agent Interaction Using Graph Structure.** Thank you for highlighting these important prior works on graph-based representations of agent interactions in multi-agent systems. We appreciate your insights, which help us further contextualize our contributions within the broader literature.

While previous methods [7-10] have explored learning or leveraging interaction graphs in MARL, our work is, to the best of our knowledge, the first to introduce a diffusion-based framework that operates jointly on edge and node attributes to model both structural and action-level diversity in offline MARL settings.

Specifically, we build upon simple yet effective graph construction heuristics based on observation similarity (as in [7]), and extend them via categorical diffusion for edge dynamics and anisotropic Gaussian diffusion for continuous action modeling at the node level. This dual-diffusion approach enables context-aware, adaptive coordination structures that evolve with agent behavior—an aspect not addressed in prior graph-based MARL works.

We also agree that incorporating more sophisticated graph learning techniques from prior works [8–10] into our framework is a promising direction, and we plan to investigate these extensions in future work, including a journal version of this study.

• **W7: Inconsistency between Subscripts and Superscripts.** We have reviewed the manuscript and corrected all inconsistencies in the use of subscripts and superscripts for the timestep variable t to ensure notation remains consistent throughout.

• **W8: Adjusting for Figure 1 and Figure 4.** We have revised Figure 4 to improve temporal interpretability by applying a fading effect to trajectory colors—early timesteps now appear lighter, while later timesteps are rendered in darker tones. This highlights trajectory progression more clearly.

Regarding Figure 1, the right subplot reflects scenarios where a vehicle either slows down or goes offline. In the offline case (Figure 1(b)), the affected vehicle remains stationary throughout the episode. As its position does not change, it visually overlaps with its initial state, which may give the impression of a missing agent. We have clarified this behavior in the caption to avoid confusion.

• **W9: Updating Reference.** We have corrected the references to ensure the years and versions are up-to-date.

References:

[1] Celebrating diversity in shared multi-agent reinforcement learning, Li et al, NeurIPS 2021.

[2] Controlling Behavioral Diversity in Multi-Agent Reinforcement Learning, Bettini at al, ICML 2024.

[3] Graph-Constrained Diffusion for End-to-end Path Planning, Shi et al, ICLR 2024.

[4] Diffusion Policies as an Expressive Policy Class for Offline Reinforcement Learning, Wang et al, ICLR 2023.

[5] Beyond Conservatism: Diffusion Policies in Offline Multi-agent Reinforcement Learning, Li et al, CoRR 2023.

[6] Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps, Lu et al, NeurIPS 2022.

[7] Graph Convolutional Reinforcement Learning, Jiang et al, ICLR 2020.

[8] Discrete GCBF Proximal Policy Optimization for Multi-agent Safe Optimal Control, Zhang et al, ICLR 2025.

[9] Scaling Safe Multi-Agent Control for Signal Temporal Logic Specifications, Eappen et al, CoRL 2024.

[10] Graph Policy Gradients for Large Scale Robot Control, Khan et al, CoRL 2020.

4 Rebuttal for Reviewer DRJr

We sincerely thank you for your valuable and constructive feedback on our work. We have carefully addressed all of your comments and made corresponding improvements and additions to the manuscript in order to resolve any concerns you may have.

Below, we provide detailed responses to each of your comments. For ease of reference, we denote reviewer Weaknesses as W and Questions as Q in our replies. Relevant figures and tables can be found in the anonymized supplementary material linked accordingly.

• **W1: Testing Environments with Dynamic Agent Numbers.** We would like to clarify that our original manuscript already includes evaluation in dynamic settings where agent availability changes. Specifically, during testing on the MPE Spread task, we randomly selected one of the three agents and set its velocity to zero to simulate a sudden offline event. The corresponding results are reported in the "Coordination Structure" column of Table 2. We believe the confusion may stem from Appendix 7.4.2, which focuses on attribute variation (e.g., speed changes) rather than agent removal.

To further address your concern, we have extended our evaluation by increasing the number of agents and landmarks to 8. During testing, we randomly deactivate 1 to 4 agents by setting their velocities to zero. As shown in Table 4 of the anonymized supplementary material, MCGD consistently outperforms baselines such as DOM2 and MADIFF under all conditions. Notably, the performance gap widens as more agents go offline, highlighting MCGD's robustness and adaptability in highly dynamic environments.

These results support our claim that MCGD is well-suited for handling general coordination under dynamic agent configurations.

• **W2: Scalability Comparison.** While our core experiments follow prior diffusion-based offline MARL settings [1-3], which typically involve fewer than 10 agents, this does not indicate a limitation of our framework in large-scale scenarios.

In Appendix 7.4.2, we first evaluate computational efficiency by comparing sampling time under increasing agent numbers (8 to 64). Despite the added complexity of structural diffusion, MCGD remains competitive with existing baselines such as MADIFF.

To further assess planning performance at scale, we additionally conduct experiments on the MPE Spread task with 8, 16, 32, and 64 agents. As reported in Table 1 of the anonymized supplementary material, MCGD consistently outperforms MADIFF and DOM2 across all settings, with the performance gap widening as the number of agents increases. This trend highlights MCGD's ability to effectively model complex collaboration patterns under growing agent populations.

These results confirm that our framework is not only computationally scalable, but also capable of maintaining strong coordination performance in large-scale environments.

• **W3: Real World Applications:** We agree that real-world validation is an important direction to further demonstrate the practical applicability of our framework.

Our team is currently working on deploying the proposed method in real-world multi-robot hunting scenarios. While we do not yet have quantitative results ready for inclusion in this version, we are actively collecting data and refining the deployment process. We plan to report these findings as part of a more extensive evaluation in a future journal extension of this work.

References:

[1] Beyond Conservatism: Diffusion Policies in Offline Multi-agent Reinforcement Learning, Li et al, CoRR 2023.

[2] Madiff: Offline multi-agent learning with diffusion models, Zhu et al, NeurIPS 2024.

[3] Diffusion-based Episodes Augmentation for Offline Multi-Agent Reinforcement Learning, Oh et al, ICML 2024.