INTER-JOINT RELATIONS MATTER: ASYMMETRIC MOTION–DISTANCE DENOISING FOR TEXT-TO-MOTION GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In text-to-motion generation, many described actions hinge on changing relations between body parts, such as the hands meeting at the right moment during clapping. However, most generators leave these relations implicit in joint states or motion codes, using joint-derived distances, if at all, only as losses or sampling guidance, so the denoiser has no explicit relational state to follow. We propose Asymmetric Motion–Distance Denoising (AMDD), which jointly denoises motion and sparse, continuous inter-joint distances. Its two-stream denoiser couples them one way: shielded from noisy motion, the distance stream gives the motion stream a stable relational reference at every denoising step. A geometric consistency objective encourages the predicted motion to realize the predicted distances, and a text-conditioned gate scales the distance stream's updates. AMDD lowers FID by 43–84% on HumanML3D across three diffusion backbones and by 12% on SnapMoGen, and its distance stream reduces inter-joint distance error about twice as much as position error.

1 INTRODUCTION

Text-to-motion generation translates natural-language descriptions into 3D human motion. Many actions, however, are defined by changing relations between body parts. Clapping requires the hands to meet at the right moment; touching one's head requires a hand to approach and remain near it. Generating these actions thus requires not only plausible motion of individual joints, but also coordinated spatial relations over time.

Most text-to-motion methods generate frame-wise kinematic features, such as root motion and joint positions or rotations, or learned codes that decode into those features (Tevet et al., 2023; Chen et al., 2023; Zhang et al., 2023a; Guo et al., 2024). Although the resulting poses determine distances between body parts, these distances are typically consequences of the generated motion rather than generation states in their own right. A loss on joint features, moreover, cannot tell whether an error breaks a relation: two wrist shifts of equal size cost the same, yet only one changes the distance between the hands (Fig. 1a). Body-part decomposition and skeleton-aware networks facilitate coordination by exchanging information across joints or regions (Zhang et al., 2023c; Zou et al., 2024; Hong et al., 2025), but spatial relations remain implicit in the learned motion features.

These relations can be traced by a few inter-joint distances: in Fig. 1b, the kick is a spike in one distance and the repeated punches are peaks in another. A text-to-motion model could therefore generate such distances from the text together with the motion. The methods that make relations explicit do not do this within the body. Distance losses (Liang et al., 2024) act only on predictions during training, leaving the denoiser no relational state during sampling; inference-time constraints (Wang et al., 2024; Liu et al., 2024a) must be specified before sampling, by a user or a language model; generated intermediates (Huang et al., 2024b; Zhang et al., 2025c; Xue et al., 2025) are finished before the motion is produced; and methods that generate relations jointly with motion model them between a person and objects (Diller & Dai, 2024).

To close this gap, we propose Asymmetric Motion–Distance Denoising (AMDD), which generates continuous inter-joint distances from the text and denoises them alongside motion as an explicit state. Joint-pair distances directly capture proximity while remaining invariant to global translation and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

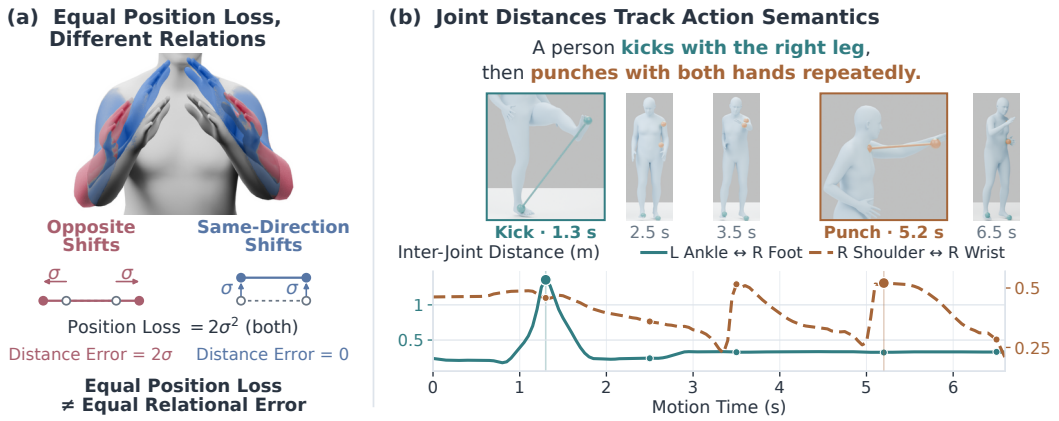

Figure 1: **(a)** Relative to the gray reference pose, equal-magnitude wrist shifts incur the same position loss, but different inter-joint distance errors. **(b)** Inter-joint distances track action semantics over time: the left-ankle–right-foot distance spikes during the kick, and the right-shoulder–right-wrist distance oscillates during the repeated punches.

rotation. We use a sparse set of pairs to focus on changing body-part relations. At every denoising step, the model predicts both the distance and the motion sequence, so the predicted distances serve as relational context for the motion throughout generation.

Motion and distances are different quantities, and denoising both with one set of weights can let them interfere: appending distances as extra channels of one network does not improve the baseline (Sec. 4.3). Following Mixture-of-Transformers (Liang et al., 2025), which decouples parameters by modality, AMDD gives each its own stream and parameters, and the two exchange information through a shared self-attention. We mask this attention to run one way: the motion input is noisy at each step, so a distance stream that read it would be exposed to that noise, whereas one denoised from the text and its own state gives the motion a stable relational reference. Reading lets the motion see the distances but not necessarily follow them, so a geometric consistency objective pulls the distances implied by the predicted motion toward the predicted ones, and not the reverse. Because descriptions involve different joint pairs, such as the hands in clapping or the feet in walking, a text-conditioned gate scales the distance stream's updates. As the relational context comes from the denoiser itself, inference needs no relational targets or test-time optimization.

By integrating these designs, AMDD makes inter-joint relations an explicit state of text-to-motion diffusion and generates more realistic motions. Experiments show that it consistently improves diffusion backbones: on HumanML3D, it reduces the FID of three backbones by 43–84%, and on the strongest one it reaches an FID of 0.046 under the MARDM evaluator (Meng et al., 2025b), the lowest among all methods we compare under this evaluator, including discrete-token ones. On SnapMoGen, it lowers FID by 12% over the same backbone. Ablations and analyses further show that the gains come from generating distances rather than penalizing them, that among the compared couplings only one-way coupling from distances to motion improves the baseline consistently, and that the distance stream reduces inter-joint distance error about twice as much as position error.

2 Related Work

Text-to-motion generation. Existing methods fall mainly into continuous diffusion and discrete token models. MDM (Tevet et al., 2023) and MotionDiffuse (Zhang et al., 2024a) brought diffusion models to text-to-motion generation, and MLD (Chen et al., 2023) then denoised in a latent space for efficiency; T2M-GPT (Zhang et al., 2023a) and MoMask (Guo et al., 2024) instead quantize motion into tokens and predict them autoregressively or by masked modeling. Recent work improves the generative framework: MARDM (Meng et al., 2025b) uses masked autoregressive diffusion, and CMDM (Yu et al., 2026) causal diffusion. Other work improves the representation: ACMDM (Meng et al., 2025a) generates absolute joint coordinates, and MoGenTS (Yuan et al., 2024), ParCo (Zou et al., 2024), SALAD (Hong et al., 2025), and UniMate (Mou et al., 2026) add structural inductive

biases through joint-level tokens, part-wise decomposition, or skeletal topology. These methods change how motion is represented and generated, but the generated quantity remains per-joint states or their codes; we add inter-joint distances to the denoising state explicitly.

Joint relations in motion generation. Beyond the specified constraints and intermediate representations discussed in Sec. 1, joint or interpersonal relations have served as training supervision or discrete descriptions, and have been generated jointly with motion in interaction settings. InterGen (Liang et al., 2024) supervises interpersonal joint distances during training; PoseScript (Delmas et al., 2025) describes poses with distance predicates, and Kinematic Phrases (Liu et al., 2024b) describes motion with discretized distance changes. For interactions, CG-HOI (Diller & Dai, 2024), JointHOI (Song et al., 2026), and SyncDiff (He et al., 2025) jointly generate motion with human–object contacts or relative motion, and Contact Matrix (Chen et al., 2026), InterDist (Zhang et al., 2026), and ARMS (Liu et al., 2026a) model interpersonal contacts, distances, or relative translations. These methods supervise or discretize relations, or model them across entities, typically with inference-time guidance or synchronization; we instead denoise continuous intra-body inter-joint distances jointly with motion in one denoiser, under a training-time geometric consistency objective.

Multi-stream coupling. In other generation tasks, related work either adds auxiliary representations or designs how streams interact. REPA (Yu et al., 2025b) aligns denoiser features with external representations, and ReDi (Kouzelis et al., 2025) jointly generates image latents and such features. Mixture-of-Transformers (MoT) (Liang et al., 2025) uses modality-specific parameters with shared global self-attention; PackDiT (Jiang et al., 2025) couples motion and text through bidirectional cross-attention, whereas UniCon (Yu et al., 2025a) passes information one way, from a frozen diffusion model to a trainable adapter. We adopt the modality-specific parameters of MoT, but our auxiliary stream is derived from the motion itself, and the two streams are trained jointly with information flowing only from distances to motion.

3 METHOD

3.1 PRELIMINARIES

Diffusion backbone. Given text c, we generate motion $x_0^m \in \mathbb{R}^{D_m \times T}$ with a denoising diffusion model, following the x_0-prediction formulation of MDM (Tevet et al., 2023) with classifier-free guidance (Ho & Salimans, 2022).

Motion representation. The standard representation of HumanML3D (Guo et al., 2022) has 263 dimensions, most of them redundant joint rotations, velocities, and foot contacts. Following MARDM (Meng et al., 2025b), which shows that this redundancy hinders diffusion models, we use a compact representation that keeps only the first 67 dimensions, which determine the motion: root angular velocity, horizontal linear velocity, and height (4 dims), plus the positions of 21 non-root joints relative to the root (63 dims). This change alone lowers the FID of both MDM backbones (Appendix D). Likewise, of the 296 dimensions of the standard SnapMoGen representation (Guo et al., 2025a), we keep only the first 148, from which the redundant joint positions, velocities, and foot contacts are derived: the same root information (4 dims), plus global 6D rotations of 24 joints in the root-heading-aligned frame (144 dims). These 148 dimensions are also exactly the input of the official SnapMoGen evaluator.

3.2 RELATIONAL STATE: SPARSE JOINT-PAIR DISTANCES

The motion representation describes each joint on its own, by its position or rotation; how two body parts relate, for example whether the hands meet, follows only indirectly, as a nonlinear function of several coordinates (on SnapMoGen, of all rotations along the kinematic chain between them). Joint-pair distances state such relations directly and are invariant to global translation and rotation, but lack global position, orientation, and chirality, so they cannot replace the motion. The two are complementary, and we denoise distances as a second set of channels alongside the motion.

Definition. We select a set of joint pairs $\mathcal{P} = \{(i_k, j_k)\}_{k=1}^{K}$, $K \ll \binom{J}{2}$ (J is the number of joints). Let Φ recover metric joint positions from the motion representation (reading the position

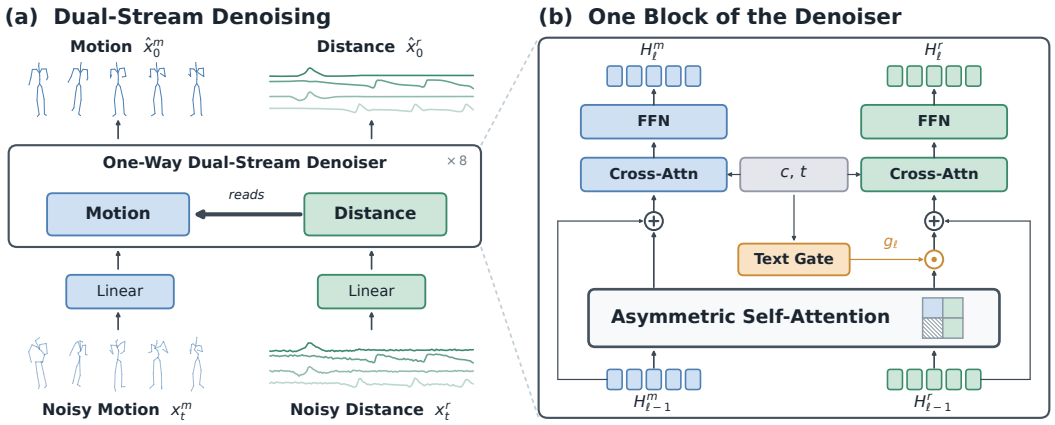

Figure 2: **Overview of AMDD. (a)** Noisy motion and distances are denoised in two streams; the motion stream reads from the distance stream, but not the reverse. **(b)** Within each block, this one-way reading is set by the mask of the asymmetric self-attention. A text gate g_ℓ scales the distance stream's update. The two streams share the condition (c, t) through cross-attention but have separate parameters.

channels for HumanML3D; differentiable forward kinematics for SnapMoGen). Given motion x^m, we compute the per-frame Euclidean distances of the selected pairs, $\mathcal{D}_{\mathcal{P}}(\Phi(x^m)) \in \mathbb{R}^{K\times T}$, and standardize each channel to obtain the relational state x^r.

Combined diffusion state. The relational state and motion are concatenated along the channel axis into a combined state $x_0 = [x_0^m; x_0^r] \in \mathbb{R}^{(D_m+K)\times T}$ and noised under a shared schedule:

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon, \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \qquad x_t = [x_t^m; x_t^r]. \tag{1}$$

Joint-pair selection. Pairs within a near-rigid segment barely change, so we rank pairs by how much their distance varies: the **variance-based family** $\mathcal{P}_K^{\text{var}}$ by temporal variance, and the **rule-based family** $\mathcal{P}_K^{\text{rule}}$ by variance and periodicity, excluding near-rigid and mutually redundant pairs. Both are prefix-nested ($\mathcal{P}_K \subset \mathcal{P}_{K+1}$), so varying K only adds or removes channels, and almost all selected pairs connect different body parts, mostly through distal limb joints (Appendix A).

3.3 AMDD: A Dual-Stream Denoiser with One-Way Coupling from Relational State to Motion

The denoiser G_θ produces two output components in x_0-prediction form (Fig. 2):

$$\hat{x}_0^r = G_\theta^r\big(x_t^r,\, t,\, c\big), \qquad \hat{x}_0^m = G_\theta^m\big(x_t^m,\, x_t^r,\, t,\, c\big). \tag{2}$$

The dependency is one-way: the distance stream is independent of motion, while the motion stream additionally reads from the distance stream.

Dual-stream inputs and modality-specific parameters. G_θ is implemented as two frame-aligned token streams of equal length T. The two streams share a text-and-timestep condition: text encodings and timestep embeddings form a set of memory tokens, and each stream cross-attends to them. All other parameters are modality-specific, following Mixture-of-Transformers (Liang et al., 2025); one block is shown in Fig. 2b.

One-way coupling via masked self-attention. The two streams interact within a single self-attention operation, whose direction is set by a block mask M. Each stream computes queries, keys, and values with its own projections, concatenated along the sequence axis, shown here for the queries (the first T tokens are motion, the last T are distance):

$$Q = \big[H_{\ell-1}^m W_{Q,\ell}^m;\; H_{\ell-1}^r W_{Q,\ell}^r\big], \qquad A = \mathrm{softmax}\Big(\tfrac{QK^\top}{\sqrt{d_h}} + \log M\Big), \quad M = \left[\begin{smallmatrix}1 & 1\\ 0 & 1\end{smallmatrix}\right] \otimes \mathbf{1}_{T\times T}. \tag{3}$$

Text-conditioned gate. Since different descriptions emphasize different joint pairs, we introduce a text-conditioned gate to modulate the distance stream's contribution to the motion stream:

$$g_\ell = \mathbf{1} + \tanh\left(W_\ell^g\, \bar{e}_c\right) \in \mathbb{R}^d, \tag{4}$$

where $\bar{e}_c$ is the pooled text representation and W_ℓ^g is zero-initialized with no bias. The gate has two mutually exclusive mounting positions: **read-side**, scaling only the distance values read by motion queries and leaving the distance stream unchanged; and **write-side**, scaling the distance stream's self-attention output, that is, its own updates (Fig. 2b).

3.4 Training Objective

Joint denoising loss. Both the motion and distance channels serve as denoising targets:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t,\epsilon}\Big[\big\|\hat{x}_0 - x_0\big\|_{\mathbf{v}}^2\Big], \qquad \hat{x}_0 = [\hat{x}_0^m; \hat{x}_0^r]. \tag{5}$$

$\|\cdot\|_{\mathbf{v}}^2$ averages over all valid elements, so $\mathcal{L}_{\text{simple}} = \frac{D_m}{D_m+K}\mathcal{L}_m + \frac{K}{D_m+K}\mathcal{L}_r$, where $\mathcal{L}_m$ and $\mathcal{L}_r$ are the losses on the motion and distance channels: the two streams are weighted by their channel counts, and changing K also changes their relative weight.

Geometric consistency loss. The inter-joint distances recomputed from the predicted motion should agree with the predicted distances. Let $\hat{d}_0^r$ denote the de-standardized (metric) prediction of $\hat{x}_0^r$:

$$\mathcal{L}_{\text{geo}} = \mathbb{E}_{t,\epsilon}\Big[\big\|\, \mathcal{D}_{\mathcal{P}}\big(\Phi(\hat{x}_0^m)\big) - \text{sg}\big[\hat{d}_0^r\big]\,\big\|_{\mathbf{v}}^2\Big], \tag{6}$$

where $\text{sg}[\cdot]$ denotes stop-gradient: this term pulls the motion prediction toward the predicted distances without back-propagating into the distance stream. Motion-side losses still update the distance stream's parameters through the reading path in Eq. (3), so information flows one way while parameters are optimized jointly.

Distance auxiliary loss. $\mathcal{L}_{\text{aux}}$ penalizes the distance errors of a separately fixed pair set $\mathcal{P}'$ directly on the predicted motion, up-weighting small distances, as an additional relational regularizer (definition in Appendix B).

Total objective. The three terms are combined as

$$\mathcal{L} = \mathcal{L}_{\text{simple}} + \lambda_{\text{geo}}\,\mathcal{L}_{\text{geo}} + \lambda_{\text{aux}}\,\mathcal{L}_{\text{aux}}, \tag{7}$$

with $\lambda_{\text{geo}} = \lambda_{\text{aux}} = 1$ unless otherwise noted.

Inference sampling. At inference, both streams are denoised together, and classifier-free guidance may use a separate scale for each stream:

$$\hat{x}_0 = G_\theta(x_t, t, \varnothing) + \mathbf{s} \odot \big(G_\theta(x_t, t, c) - G_\theta(x_t, t, \varnothing)\big), \qquad \mathbf{s} = \big[s_m \mathbf{1}_{D_m};\, s_r \mathbf{1}_K\big], \tag{8}$$

where $\varnothing$ denotes the unconditional branch with the text dropped, and s_m and s_r apply to the motion and distance channels; $s_m = s_r$ unless otherwise noted. The distance channels are then discarded and only the motion $\hat{x}_0^m$ is kept.

4 Experiments

4.1 Experimental Setup

Datasets and metrics. We experiment on HumanML3D (Guo et al., 2022) and SnapMoGen (Guo et al., 2025a), in the representations of Sec. 3.1. On HumanML3D, we use two evaluators: the standard T2M evaluator (Guo et al., 2022), applied to 263-d features reconstructed from joint positions, and the MARDM evaluator (Meng et al., 2025b), applied directly to the 67-d representation. We report FID, R-precision (R@k), MM-Dist (named Matching in the MARDM evaluator), and Diversity. On SnapMoGen, we use the official evaluator and report FID, TOP-k retrieval accuracy, and Matching.

Table 1: Quantitative results on the HumanML3D. Best and second best per group in **bold** and underlined. RAM is listed without REG, as in our setting; the full comparison is in Appendix D. Parentheses: FID reduction over the backbone (T2M / MARDM). $^{\diamond}$ From RAM (263-d). † Our implementation: RAM's frozen encoders and reconstruction branch on MDM's Transformer encoder.

	T2M evaluator					MARDM evaluator		
Method	FID↓	R@1↑	R@3↑	MM-Dist↓	Div→	FID↓	R@1↑	Matching↓
Real	0.002±.000	0.511±.003	0.797±.002	2.974±.008	9.503±.065	0.000±.000	0.503±.002	3.244±.005
Discrete-token methods								
MoMask (Guo et al., 2024)	0.045±.002	0.521±.002	0.807±.002	2.958±.008	–	**0.116**±.006	**0.490**±.004	**3.353**±.010
MMM (Pinyoanuntapong et al., 2024b)	0.089±.005	0.515±.002	0.804±.002	2.926±.007	9.577±.050	0.132±.004	0.487±.003	3.359±.009
BAMM (Pinyoanuntapong et al., 2024a)	0.055±.002	0.525±.002	0.814±.003	2.919±.008	9.717±.089	–	–	–
MoGenTS (Yuan et al., 2024)	0.033±.001	0.529±.003	0.812±.002	2.867±.006	9.570±.077	–	–	–
LaMP (Li et al., 2025)	0.032±.002	0.557±.003	0.843±.001	2.759±.007	9.571±.069	–	–	–
KinMo (Zhang et al., 2025b)	0.039±.003	0.532±.002	0.821±.003	2.901±.010	9.674±.058	–	–	–
MoMask++ (Guo et al., 2025a)	0.072±.003	0.528±.003	0.811±.002	2.912±.008	–	–	–	–
MoScale (Zheng et al., 2026)	0.046±.002	0.540±.002	0.817±.002	2.830±.005	**9.525**±.090	–	–	–
ScaleMoGen (Hwang et al., 2026)	**0.030**±.002	**0.577**±.004	**0.856**±.002	**2.626**±.006	–	–	–	–
Diffusion-based methods								
StableMoFusion (Huang et al., 2024a)	0.098±.003	0.553±.003	0.841±.002	–	9.748±.092	–	–	–
MARDM-SiT (Meng et al., 2025b)	0.061	0.523	0.810	–	–	0.114±.007	0.500±.004	3.270±.009
SALAD (Hong et al., 2025)	0.076±.002	0.581±.003	0.857±.002	2.649±.009	9.696±.096	0.124±.005	0.552±.003	2.990±.010
ACMDM-XL (Meng et al., 2025a)	–	–	–	–	–	0.058±.004	0.522±.002	3.205±.008
CMDM (Yu et al., 2026)	0.068±.003	**0.588**±.004	**0.860**±.003	**2.620**±.010	–	0.078±.003	**0.563**±.004	**2.920**±.007
RAM w/o REG$^{\diamond}$ (Liu et al., 2026b)	0.107±.004	0.563±.002	0.840±.002	2.729±.006	9.711±.066	–	–	–
MDM (CLIP-Enc)	0.236±.007	0.497±.003	0.786±.002	3.069±.007	10.002±.092	0.338±.008	0.476±.004	3.465±.010
+ AMDD(−75% / −84%)	0.059±.004	0.506±.003	0.794±.003	2.982±.010	**9.429**±.053	0.054±.003	0.491±.002	3.301±.007
MDM (BERT-Dec)	0.219±.006	0.547±.003	0.836±.002	2.776±.011	10.042±.087	0.297±.006	0.529±.002	3.134±.008
+ AMDD(−71% / −78%)	0.064±.003	0.547±.003	0.836±.002	2.743±.007	9.580±.060	0.064±.002	0.542±.002	3.007±.008
RAM† w/o REG	0.099±.003	0.552±.002	0.836±.002	2.744±.007	9.790±.068	0.081±.003	0.534±.002	3.071±.009
+ AMDD(−51% / −43%)	**0.048**±.003	0.552±.003	0.838±.002	2.710±.007	9.604±.065	**0.046**±.004	0.541±.002	3.024±.006

Table 2: Quantitative results on the SnapMoGen test set (official evaluator); grouping, marking, inclusion, and citations as in Tab. 1; each method is listed with its main result (other variants in Appendix D). § Retrained by Guo et al. (2025a).

Method	FID↓	TOP-1↑	TOP-3↑	Match.↑
Real	0.001±.000	0.940±.001	0.985±.001	0.837±.000
Discrete-token methods				
MoMask§	17.40±.05	0.777±.002	0.927±.002	0.664±.001
MoMask++	**15.06**±.07	0.802±.001	0.938±.001	0.685±.001
ScaleMoGen	16.35±.09	**0.807**±.004	**0.941**±.003	**0.693**±.001
Diffusion-based methods				
StableMoFusion§	27.80±.06	0.679±.002	0.888±.002	0.605±.001
MARDM§	26.88±.13	0.659±.002	0.860±.002	0.602±.001
CMDM	**14.45**±.07	**0.831**±.004	**0.958**±.002	**0.702**±.001
MDM (BERT-Dec)	18.86±.08	0.785±.003	0.927±.001	0.673±.001
+ AMDD	16.56±.07	0.805±.002	0.941±.001	0.687±.001

Table 3: Component and coupling ablations on HumanML3D (MDM (BERT-Dec)). Best in (a) and (c) in bold.

Configuration	Params	FID (T2M)↓	FID (MARDM)↓
(a) Components (variance family, K=64)			
Base	26.2M	0.214±.012	0.300±.016
+ distance stream	51.5M	0.188±.008	0.217±.007
+ $\mathcal{L}_{\text{aux}}$	51.5M	0.180±.005	0.186±.005
+ $\mathcal{L}_{\text{geo}}$	51.5M	0.152±.008	0.152±.006
+ text gate (AMDD)	61.0M	**0.064**±.003	**0.064**±.002
(b) Parameter-matched Base ($\mathcal{L}_{\text{simple}}$ only)			
2× width	51.6M	0.302±.005	0.485±.014
2× depth	51.5M	0.296±.012	0.440±.020
(c) How distances enter (rule family, K=64)			
Aux. loss only, no stream	26.2M	0.240±.011	0.325±.014
Concat (single stream)	26.3M	0.327±.008	0.419±.007
Bidirectional	51.5M	0.303±.013	0.293±.008
Motion → distance	51.5M	0.228±.014	0.266±.015
Distance → motion (ours)	51.5M	**0.182**±.013	**0.231**±.005

Backbones and implementation. To test whether AMDD depends on a particular backbone, we add it to three diffusion backbones, all trained from scratch on the 67-d representation: two configurations of MDM (Tevet et al., 2023), with a CLIP text encoder and a Transformer encoder (MDM (CLIP-Enc)) or a BERT text encoder and a Transformer decoder (MDM (BERT-Dec)), and RAM (Liu et al., 2026b) without reconstruction-error guidance (REG). In MDM (CLIP-Enc), each stream attends to the text token in a separate attention layer, outside the shared self-attention; our RAM backbone uses the same architecture with RAM's reconstruction branch and the frozen text and motion encoders of a RAM model we trained. Each backbone shares these changes with its AMDD version (Appendix B). On each backbone, AMDD adds the same components: a distance stream over the variance-family pairs (K=64) and the write-side text gate (Sec. 3.3). On SnapMoGen, we use MDM (BERT-Dec) with the same components and the rule-based family (K=64), sampled with s_m=4.5 and s_r=2.5 (Eq. (8); Appendix B). Ablations and analyses use MDM (BERT-Dec) on HumanML3D, where *Base* denotes its single-stream model on the 67-d representation. Implementation, evaluation, and inference-cost details are in Appendix B and Appendix C.

4.2 Comparison with Published Methods

Consistent gains across backbones. Tab. 1 compares each backbone with and without AMDD under the same training protocol. AMDD lowers FID on all three backbones under both evaluators, by more than two thirds on the two MDM configurations and by more than 40% on RAM, the strongest backbone. MM-Dist and Matching improve as well, Diversity moves closer to that of real motions, and R-precision stays on par with the backbones. AMDD thus benefits diffusion backbones of different architectures and strengths.

Comparison with the state of the art. With AMDD, all three backbones reach a lower FID than most published diffusion-based methods under both evaluators (Tab. 1). On HumanML3D, however, the gains in R-precision are limited, and CMDM and SALAD remain ahead; Appendix E analyzes this. On SnapMoGen, AMDD lowers FID by 12% over the same backbone, at every guidance scale we tested (Appendix H), and improves all retrieval metrics (Tab. 2). Among diffusion-based methods, its FID and retrieval accuracy are second only to those of CMDM; among discrete-token methods, MoMask++ and ScaleMoGen reach a lower FID with similar TOP-1.

4.3 Ablation Studies

We ablate the components, capacity, and coupling (Tab. 3) and the relation variables (Tab. 4) on HumanML3D; SnapMoGen ablations are in Appendix H.

Tab. 3a adds the components to Base one at a time. The distance stream alone, trained only with $\mathcal{L}_{\text{simple}}$, already improves Base, whereas penalizing the distance error of the predicted motion with $\mathcal{L}_{\text{aux}}$ does not (first row of Tab. 3c); the same loss helps only on top of the stream. $\mathcal{L}_{\text{geo}}$ adds a further gain by requiring the generated motion to agree with the predicted distances, and the write-side text gate brings the largest single improvement; every step lowers FID under both evaluators. The gain thus comes from generating distances: supervising them complements generation but cannot replace it. The gate's benefit depends on classifier-free guidance. The gate acts only on the text-conditioned prediction and equals 1 when the text is dropped, so guidance amplifies not only the text signal but also the gate's largely text-independent modulation; this contracts the generated motion, bringing its dispersion closer to that of real data, in line with the lower FID. Without guidance, the gated model has no lower FID than the gate-free one in any of three training runs (Appendix G).

Tab. 3b tests whether the gain is merely a matter of capacity. Under the same recipe, widening or deepening Base to the two-stream size makes it worse, not better. Likewise, the text gate's gain does not come from its parameters alone: the far smaller read-side gate also improves on its gate-free counterpart (Appendix F). The improvement therefore reflects what the second stream generates, not the parameters it adds.

Tab. 3c compares ways for distances to enter generation. Appending distances as extra channels of one stream (Concat), letting the two streams read each other (Bidirectional), and reversing the direction (motion $\rightarrow$ distance) all fail to improve on Base under the T2M evaluator; only distance $\rightarrow$ motion improves under both evaluators. Concat loses the separate parameters of the two streams, Bidirectional exposes the distance stream to noisy motion, and the reversed direction leaves the motion stream without access to the distance features.

Tab. 4 varies the relation variable. A small set of pairs improves on Base under both selection rules and both evaluators, whereas larger sets fall back to the level of Base and the dense set of all non-bone pairs is worse. This decline does not necessarily mean that more pairs are harmful: $\mathcal{L}_{\text{simple}}$ averages over all channels, so adding pairs also shifts loss weight from motion to distances (Sec. 3.4), and the two effects are confounded in this sweep. The variance family performs better than the rule-based family, and we use it for AMDD on HumanML3D (Appendix F). A few well-chosen pairs thus suffice.

Could the gain come merely from generating an extra stream, or from generating any distance-like quantity? We keep the architecture and coupling and replace the generated distances with two controls, one for each explanation. An *isometric re-encoding* tests the first: it multiplies the local joint positions by a fixed random orthogonal matrix, so the extra stream carries exactly the information of these positions in redundant form. *Random virtual-point distances* test the second: they measure distances between random weighted averages of the joints, rescaled to the magnitude of joint-pair

Table 4: Relation variables on HumanML3D (no gate). Best in bold, second best underlined. ♦ All non-bone pairs. ‡ Trained with $\lambda_{\text{geo}}=0$; its joint-pair counterpart scores 0.180 / 0.186.

Relation variable	FID (T2M)↓	FID (MARDM)↓
Rule family, $K=32$	<u>0.141</u>±.002	0.197±.008
Rule family, $K=64$	0.182±.013	0.231±.005
Variance family, $K=32$	**0.132**±.004	<u>0.178</u>±.010
Variance family, $K=64$	0.152±.008	**0.152**±.006
Variance family, $K=128$	0.225±.004	0.307±.015
Dense set, $K=210$♦	0.356±.013	0.459±.019
Isometric re-encoding‡	0.272±.015	0.338±.015
Random virtual-point distances	0.258±.019	0.279±.009
Base (no second stream)	0.214±.012	0.300±.016

Table 5: Contact on HumanML3D: gap to real motions (cm; Sec. 4.4) in the per-clip minimum wrist–wrist (Hand, No contact) or wrist–head (Head) distance, over 89 / 400 / 33 text clusters. * / †: significantly smaller / larger than Base ($p<0.05$).

Model	Hand↓	No contact↓	Head↓
MDM (BERT-Dec), 263-d	3.0	1.6	1.7
MoMask	4.7	3.7†	1.6
SALAD	4.7	2.6†	3.3†
Base (67-d)	4.1	1.3	1.7
+ distance stream (var., $K=64$)	1.8*	1.9	2.3
+ text gate (AMDD)	2.3	4.3†	2.8

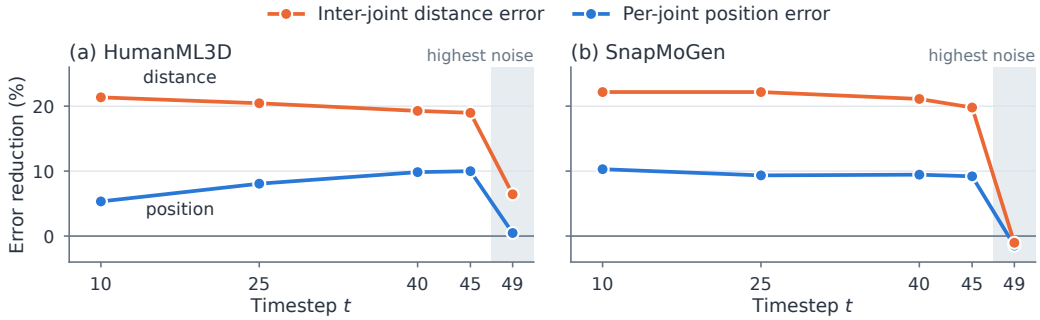

Figure 3: One-step $\hat{x}_0$ error reduction of AMDD (no gate, variance family, $K=64$) relative to Base on the same noised clips. **The distance stream reduces inter-joint distance error about twice as much as position error, or more.** The distance error covers all joint pairs.

distances, and so share the functional form, units, and rigid-motion invariance of joint-pair distances without measuring actual joints. Both controls are clearly worse than their joint-pair counterparts and, under the T2M evaluator, even worse than Base (Tab. 4). Neither a redundant extra stream nor a distance of the same form suffices; the gain depends on distances between actual joints.

4.4 Does the Distance Stream Improve Inter-Joint Relations?

Relational error during denoising. Deviations with the same position error can differ sharply in relational error (Fig. 1a), so we measure the two errors separately. We feed the same noised ground-truth clips (same noise, same timestep) to Base and to gate-free AMDD (variance family, $K=64$) and compare the errors of their one-step predictions $\hat{x}_0$. At every $t \leq 45$ and on both datasets, AMDD reduces the inter-joint distance error about twice as much as the per-joint position error, or more (Fig. 3). At the highest noise level ($t=49$), where predictions approach the text-conditional mean, the advantage shrinks. Across joints, the improvement correlates with how strongly the selected distances depend on each joint (Appendix I).

We also test whether freely generated motions realize the contacts a text requires. With fixed keyword rules, we select HumanML3D test descriptions that require the hands to meet, a hand to touch the head, or no contact (Appendix J). For each clip, we take the minimum of the relevant joint distance and measure the gap to real motions of the same texts as the Wasserstein-1 distance (cm) between the two distributions, weighting text clusters equally; differences are tested with a paired bootstrap over clusters. MoMask and SALAD use their official weights and settings.

On texts that require the hands to meet, the distance stream brings the minimum hand–hand distance closer to that of real motions: the gate-free configuration (variance family, $K=64$) reduces the gap from 4.1 cm (Base) to 1.8 cm, also significantly below 263-d MDM, MoMask, and SALAD (Tab. 5). On texts without contact, the gap does not change significantly, so the improvement appears only



Figure 4: **Qualitative comparison on HumanML3D.** For each method, five frames of one generated clip, light to dark in time, each taken within ± 0.3 s of an equally spaced instant where the described relation is most pronounced (same rule for all methods), above the corresponding distance computed from the generated joints: **(a)** knee–chest distance of the right (blue) and left (gray) knee; **(b)** wrist–wrist distance; contact: below 0.15 m for ≥ 3 frames. Dots mark the rendered frames. Meshes are SMPL fits to the generated joints, for display only.

where the text calls for contact. Hand–head contact does not improve; sparser categories are in Appendix J. The text gate does not reduce the hand–hand gap further but brings the hands significantly closer on texts without contact, a contraction that, like its FID gain, comes from its interaction with classifier-free guidance (Appendix G).

4.5 Qualitative Results

Fig. 4 shows two descriptions defined by a joint relation and plots the relevant distance from the generated joints, so that the comparison does not rest on visual inspection alone. In (a), AMDD brings the right knee closest to the chest of all methods and then lowers it, as the text asks; SALAD raises the left knee instead, MDM does not lower the knee before the clip ends, and the right knees of Base and MoMask stay farther from the chest. In (b), AMDD claps most often and reopens the hands widest between claps, matching the repeated, large movements the text describes; SALAD claps repeatedly with smaller openings, and the other methods bring the hands together only once or twice. More examples are in Appendix K.

5 Conclusion

We presented AMDD, which makes inter-joint distances an explicit state of text-to-motion diffusion: they are denoised alongside motion and read by it, one way, at every step. AMDD lowers FID on HumanML3D across three diffusion backbones and on SnapMoGen, and its distance stream reduces the inter-joint distance error about twice as much as the position error. Because generating distances improves the baseline where penalizing them does not, these results show that denoising relations as a state, rather than using them only as losses, is an effective way to improve motion generation. Two limitations remain. The distance stream generates the same pairs for every description, and the text gate, whose benefit relies on classifier-free guidance, does not make this choice text-dependent; and on HumanML3D, R-precision stays on par with the backbones. In future work, we plan to select pairs according to the description and to extend the distance state to human–object and multi-person interaction.

AI USE STATEMENT

In this work, we used generative AI tools, namely Claude models (Anthropic), mostly through the Claude Code agent, and GPT models (OpenAI), for the following tasks with required disclosure: implementing methods, including the model, training, and evaluation code and the preprocessing code that computes joint-pair distances from the data, and running training and evaluation jobs; designing and giving feedback on experiments, such as proposing control experiments and diagnostics; proposing and refining hypotheses about the observed effects, and refining the conceptual framing of the method; interpreting results; and translating our drafts from Chinese into English. The methodological ideas of the paper, namely jointly denoising inter-joint distances with motion, the one-way coupling from distances to motion, and the text-conditioned gate, were proposed by the authors. We have not used generative AI tools to generate synthetic data sets. Formulating or proving mathematical claims and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for creating and editing code, suggesting experimental parameters, creating figures (plotting and rendering scripts), identifying and summarizing related literature, formatting references, drafting and editing parts of the paper, and suggesting its structure and title candidates. We have reviewed all AI-assisted work: an author reviewed all AI-written code, and unit tests cover the model, data-processing, and evaluation code; every number in the paper was traced back to the output of our evaluation scripts; descriptions of prior work and the numbers we quote from it were checked against the original papers; interpretations suggested by AI were treated as hypotheses and kept only when our experiments supported them, and several were discarded; and the authors read the full paper, including the appendix. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Sec. 3 defines the model, its training objective, and per-stream guidance (Eq. (8)), and Appendix B.2 gives the sparse mixture form of the text gate used by AMDD and the distance auxiliary loss. Appendix A describes how the joint pairs are selected and lists the selected pairs on both datasets. The architectures, hyperparameters, and training and sampling settings of all backbones are in Sec. 4.1, Appendix B.1, and Appendix B.3, and the inference cost is in Appendix C. Appendix B.4 gives the repeated training runs; Appendix B.5 specifies the evaluators, the numbers of samples and repetitions, and the measured variability across evaluations, retraining, and checkpoints. Complete results with all metrics are in Appendix D, which marks the source of each result of prior methods, and per-checkpoint results on HumanML3D and SnapMoGen are in Appendices F and H. Both datasets, HumanML3D and SnapMoGen, are public; we use their standard splits and evaluators, plus the MARDM evaluator on HumanML3D. We will release the code, configuration files, joint-pair lists, and trained checkpoints upon publication.

REFERENCES

Ling-Hao Chen, Shunlin Lu, Wenxun Dai, Zhiyang Dou, Xuan Ju, Jingbo Wang, Taku Komura, and Lei Zhang. Pay attention and move better: Harnessing attention for interactive motion generation and training-free editing. *arXiv preprint arXiv:2410.18977*, 2024.

Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Xuhai Chen, Zhi Cen, Huaijin Pi, Sida Peng, Xiaowei Zhou, and Yong Liu. Contact Matrix: Enhancing dance motion synthesis with precise interaction modeling. *arXiv preprint arXiv:2605.04662*, 2026.

Jungbin Cho, Junwan Kim, Jisoo Kim, Minseo Kim, Mingu Kang, Sungeun Hong, Tae-Hyun Oh, and Youngjae Yu. DisCoRD: Discrete tokens to continuous motion via rectified flow decoding. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Min Jin Chong and David Forsyth. Effectively unbiased FID and Inception score and where to find them. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Wenxun Dai, Ling-Hao Chen, Yufei Huo, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Real-time controllable motion generation via latent consistency model. MotionLCM-V2 technical report, https://dai-wenxun.github.io/files/MotionLCM_V2.pdf, 2024a.

Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. MotionLCM: Real-time controllable motion generation via latent consistency model. In *European Conference on Computer Vision (ECCV)*, 2024b.

Ginger Delmas, Philippe Weinzaepfel, Thomas Lucas, Francesc Moreno-Noguer, and Grégory Rogez. PoseScript: Linking 3D human poses and natural language. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(7):5146–5159, 2025.

Christian Diller and Angela Dai. CG-HOI: Contact-guided 3D human-object interaction generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Dawei Guan, Di Yang, Chengjie Jin, and Jiangtao Wang. FlowCoMotion: Text-to-motion generation via token-latent flow modeling. *arXiv preprint arXiv:2604.11083*, 2026.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Tao Ji, Xingyu Li, and Lu Cheng. Generating diverse and natural 3D human motions from text. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sena Sen, and Lu Cheng. MoMask: Generative masked modeling of 3D human motions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Chuan Guo, Inwoo Hwang, Jian Wang, and Bing Zhou. SnapMoGen: Human motion generation from expressive texts. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025a.

Ziyan Guo, Zeyu Hu, De Wen Soh, and Na Zhao. MotionLab: Unified human motion generation and editing via the motion-condition-motion paradigm. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025b.

Wenkun He, Yun Liu, Ruitao Liu, and Li Yi. SyncDiff: Synchronized motion diffusion for multi-body human-object interaction synthesis. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Seokhyeon Hong, Chaelin Kim, Serin Yoon, Junghyun Nam, Sihun Cha, and Junyong Noh. SALAD: Skeleton-aware latent diffusion for text-driven motion generation and editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Seyed Rohollah Hosseyni, Ali Ahmad Rahmani, Seyed Jamal Seyedmohammadi, Sanaz Seyedin, and Arash Mohammadi. BAD: Bidirectional auto-regressive diffusion for text-to-motion generation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025.

Yiheng Huang, Hui Yang, Chuanchen Luo, Yuxi Wang, Shibiao Xu, Zhaoxiang Zhang, Man Zhang, and Junran Peng. StableMoFusion: Towards robust and efficient diffusion-based motion generation framework. In *ACM International Conference on Multimedia (ACM MM)*, 2024a.

Yiming Huang, Weilin Wan, Yue Yang, Chris Callison-Burch, Mark Yatskar, and Lingjie Liu. CoMo: Controllable motion generation through language guided pose code editing. In *European Conference on Computer Vision (ECCV)*, 2024b.

Inwoo Hwang, Hojun Jang, Bing Zhou, Jian Wang, Young Min Kim, and Chuan Guo. ScaleMoGen: Autoregressive next-scale prediction for human motion generation. In *European Conference on Computer Vision (ECCV)*, 2026.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. MotionGPT: Human motion as a foreign language. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Zhongyu Jiang, Wenhao Chai, Zhuoran Zhou, Cheng-Yen Yang, Hsiang-Wei Huang, and Jenq-Neng Hwang. PackDiT: Joint human motion and text generation via mutual prompting. *arXiv preprint arXiv:2501.16551*, 2025.

Theodoros Kouzelis, Efstathios Karypidis, Ioannis Kakogeorgiou, Spyridon Gidaris, and Nikos Komodakis. Boosting generative image modeling via joint image-feature synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Junlin Li, Xinhao Song, Siqi Wang, Haibin Huang, and Yili Zhao. A unified conditional flow for motion generation, editing, and intra-structural retargeting. *arXiv preprint arXiv:2604.13427*, 2026.

Zhe Li, Weihao Yuan, Yisheng He, Lingteng Qiu, Shenhao Zhu, Xiaodong Gu, Weichao Shen, Yuan Dong, Zilong Dong, and Laurence T. Yang. LaMP: Language-motion pretraining for motion generation, retrieval, and captioning. In *International Conference on Learning Representations (ICLR)*, 2025.

Han Liang, Wenqian Zhang, Wenxuan Li, Jingyi Yu, and Lan Xu. InterGen: Diffusion-based multi-human motion generation under complex interactions. *International Journal of Computer Vision*, 132(9):3463–3483, 2024.

Weixin Liang, Lili Yu, Liang Luo, Srinivasan Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-Transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*, 2025.

Hanchao Liu, Xiaohang Zhan, Shaoli Huang, Tai-Jiang Mu, and Ying Shan. Programmable motion generation for open-set motion control tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Huakun Liu, Qing Yu, Kent Fujiwara, Hideaki Uchiyama, and Kiyoshi Kiyokawa. ARMS: Anchor-relational motion streaming for seamless solo-social motion transitions. In *European Conference on Computer Vision (ECCV)*, 2026a.

Xinpeng Liu, Yong-Lu Li, Ailing Zeng, Zizheng Zhou, Yang You, and Cewu Lu. Bridging the gap between human motion and action semantics via kinematic phrases. In *European Conference on Computer Vision (ECCV)*, 2024b.

Yifei Liu, Changxing Ding, Ling Guo, Huaiguang Jiang, and Qiong Cao. Reconstruction-anchored diffusion model for text-to-motion generation. In *European Conference on Computer Vision (ECCV)*, 2026b.

G. Lyu, X. Cheng, Q. Liu, C. Xu, J. Yan, M. Yang, F. Fang, and C. Deng. COME: Advancing representation learning and generative modeling for high-quality text-to-motion generation. Unpublished submission to ICLR 2026, `https://openreview.net/forum?id=D8fW5tAHq8`, 2025.

Zichong Meng, Zeyu Han, Xiaogang Peng, Yiming Xie, and Huaizu Jiang. Absolute coordinates make motion generation easy. *arXiv preprint arXiv:2505.19377*, 2025a.

Zichong Meng, Yiming Xie, Xiaogang Peng, Zeyu Han, and Huaizu Jiang. Rethinking diffusion for text-driven human motion generation: Redundant representations, evaluation, and masked autoregression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025b.

Linzhan Mou, Jiahui Lei, Zhiyang Dou, Chenyue Cai, Chaoyue Song, Adam Finkelstein, and Szymon Rusinkiewicz. UniMate: One unified model to animate diverse skeletons. In *SIGGRAPH Asia Conference Papers*, 2026.

Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Pu Wang, Minwoo Lee, Srijan Das, and Chen Chen. BAMM: Bidirectional autoregressive motion model. In *European Conference on Computer Vision (ECCV)*, 2024a.

Ekkasit Pinyoanuntapong, Pu Wang, Minwoo Lee, and Chen Chen. MMM: Generative masked motion model. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Mingyeong Song, Jungbin Cho, Jisoo Kim, Ananya Bal, Kartik Sharma, Youngjae Yu, Laszlo A Jeni, and Junhyug Noh. JointHOI: Jointly generating contact maps enhances hand object interaction generation. *arXiv preprint arXiv:2607.01768*, 2026.

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. In *International Conference on Learning Representations (ICLR)*, 2023.

Yiran Wang, Zeyu Zhang, Ling Shao, and Hao Tang. ReMoMask-2: Latent retrieval-augmented masked motion generation. *arXiv preprint arXiv:2609.08365*, 2026.

Zhenzhi Wang, Jingbo Wang, Yixuan Li, Dahua Lin, and Bo Dai. InterControl: Zero-shot human interaction generation by controlling every joint. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Mengqing Xue, Yifei Liu, Ling Guo, Shaoli Huang, and Changxing Ding. Guiding human-object interactions with rich geometry and relations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Zhankai Ye, Yukai Jin, Bingyang Wei, Bofan Li, Yusen Wu, Fangyi Li, Shangqian Gao, and Xin Liu. MUGEN: A unified framework for efficient motion understanding and generation. *arXiv preprint arXiv:2607.27581*, 2026.

Fanghua Yu, Jinjin Gu, Jinfan Hu, Zheyuan Li, and Chao Dong. UniCon: Unidirectional information flow for effective control of large-scale diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025a.

Qing Yu, Kent Fujiwara, and Edgar Simo-Serra. Causal motion diffusion models for autoregressive motion generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *International Conference on Learning Representations (ICLR)*, 2025b.

Weihao Yuan, Yisheng He, Weichao Shen, Yuan Dong, Xiaodong Gu, Zilong Dong, Liefeng Bo, and Qixing Huang. MoGenTS: Motion generation based on spatial-temporal joint modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Jia-Qi Zhang, Jia-Jun Wang, Fang-Lue Zhang, and Miao Wang. Generating distance-aware human-to-human interaction motions from text guidance. *IEEE Transactions on Visualization and Computer Graphics*, 32(3):2615–2627, 2026.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2M-GPT: Generating human motion from textual descriptions with discrete representations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023a.

Jianrong Zhang, Hehe Fan, and Yi Yang. EnergyMoGen: Compositional human motion generation with energy-based diffusion model in latent space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. ReMoDiffuse: Retrieval-augmented motion diffusion model. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023b.

Mingyuan Zhang, Huirong Li, Zhongang Cai, Jiahe Ren, Lei Yang, and Ziwei Liu. FineMoGen: Fine-grained spatio-temporal motion generation and editing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023c.

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. MotionDiffuse: Text-driven human motion generation with diffusion model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024a.

Mingyuan Zhang, Daisheng Jin, Chenyang Gu, Fangzhou Hong, Zhongang Cai, Jingfang Huang, Chongzhi Zhang, Xinying Guo, Lei Yang, Ying He, and Ziwei Liu. Large motion model for unified multi-modal motion generation. In *European Conference on Computer Vision (ECCV)*, 2024b.

Pengfei Zhang, Pinxin Liu, Pablo Garrido, Hyeongwoo Kim, and Bindita Chaudhuri. KinMo: Kinematic-aware human motion understanding and generation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025b.

Wanyue Zhang, Rishabh Dabral, Vladislav Golyanik, Vasileios Choutas, Eduardo Alvarado, Thabo Beeler, Marc Habermann, and Christian Theobalt. BimArt: A unified approach for the synthesis of 3D bimanual interaction with articulated objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025c.

Zeyu Zhang, Yiran Wang, Wei Mao, Danning Li, Rui Zhao, Biao Wu, Zirui Song, Bohan Zhuang, Ian Reid, and Richard Hartley. Motion Anything: Any to motion generation. *arXiv preprint arXiv:2503.06955*, 2025d.

Zhiwei Zheng, Shibo Jin, Lingjie Liu, and Mingmin Zhao. Next-scale autoregressive models for text-to-motion generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Qiran Zou, Shangyuan Yuan, Shian Du, Yu Wang, Chang Liu, Yi Xu, Jie Chen, and Xiangyang Ji. ParCo: Part-coordinating text-to-motion synthesis. In *European Conference on Computer Vision (ECCV)*, 2024.

Overview of the Appendix

- Appendix A: selection rules and lists of the joint pairs (Sec. 3.2).
- Appendix B: backbones, text gate, training and sampling, checkpoint selection and repeated runs, and evaluation protocol (Sec. 4.1); Appendix C: inference cost.
- Appendix D: complete comparison tables with all metrics (Tab. 1 and Tab. 2).
- Appendix E: conditional distribution in the evaluator space (Sec. 4.2).
- Appendix F: further HumanML3D ablations and per-checkpoint results (Sec. 4.3).
- Appendix G: interaction of the text gate with classifier-free guidance (Sec. 4.3 and Sec. 4.4).
- Appendix H: SnapMoGen ablations and guidance scales (Tab. 2).
- Appendix I: setup of the relational-error measurement and per-joint analysis (Fig. 3).
- Appendix J: contact evaluation protocol and all categories (Tab. 5).
- Appendix K: more qualitative results (Sec. 4.5).

A Joint-Pair Selection

Candidate sets. The HumanML3D skeleton has 22 joints and thus 231 pairs. The variance family selects from the 210 pairs that remain after removing the 21 bones; these 210 pairs form the dense set in Tab. 4. The rule-based family considers only pairs more than two bones apart on the skeleton, because pairs one or two bones apart are nearly rigid and their distances barely change with motion. The SnapMoGen skeleton has 24 joints; the bone offset between the pelvis and the root is zero, so the two always coincide, and the pelvis is therefore not used as a pair endpoint. The remaining 23 joints form 253 pairs, and both families consider only the 210 of them that are more than two bones apart. The dense sets of the two datasets are therefore defined differently and only happen to have the same size.

Variance family. The candidate pairs are sorted by the variance of their distance in descending order, and the top K are taken. The variance is computed over all frames of all training sequences (with the same length filter as in training, including mirrored clips).

Rule-based family. We first keep the pairs whose mean temporal standard deviation is at least 2 cm and sort them by the average of two ranks: the rank of the temporal standard deviation and the rank of the periodicity prevalence, which is normalized separately over all training samples and within six training subsets defined by caption keywords (walk, run, jump, clap, wave, kick). Going down this order, we then remove every pair whose absolute correlation with an already selected pair is at least 0.85 (relaxed to 0.88 on SnapMoGen to fill all 64 slots). On both datasets, these statistics are computed from 3000 non-mirrored clips sampled from the training set. The 58th pair of the SnapMoGen rule-based family was originally left elbow–pelvis; since the pelvis coincides with the root, it is replaced by the equivalent root–left elbow.

Nesting and data split. Both families are listed in a fixed order, and the top K pairs are always a subset of the top $K+1$, so varying K in the ablations changes only the number of channels, not the selection rule. Both selections use only the training set, never the validation or test set.

Sets used and their characteristics. AMDD uses the top 64 pairs of the variance family on HumanML3D and the top 64 pairs of the rule-based family on SnapMoGen; on both datasets, the pair set $\mathcal{P}'$ of $\mathcal{L}_{\text{aux}}$ is fixed to the top 64 pairs of the respective rule-based family (Appendix B.2). On HumanML3D, $\mathcal{P}'$ is therefore not the same set as the pairs $\mathcal{P}$ generated by the distance stream; on SnapMoGen, the two coincide. The top 64 pairs of the two families share only 30 pairs on HumanML3D, and 41 on SnapMoGen. The variance family concentrates on the distal ends of the limbs: in its top 64 pairs, each wrist and each ankle appears 13 times on HumanML3D, and each hand appears 19 times on SnapMoGen. The rule-based family gives some slots to pairs with pronounced periodicity, so elbows and knees appear more often (in its top 64 pairs on HumanML3D, the left elbow and each knee appear 10 times).

The two families select pairs with clear commonalities (Fig. A.1). Almost all selected pairs cross body parts: at most 3 of the 64 pairs in each set lie within one part, and none lies within the torso, although the variance family excludes only the bones themselves. In every set, the largest

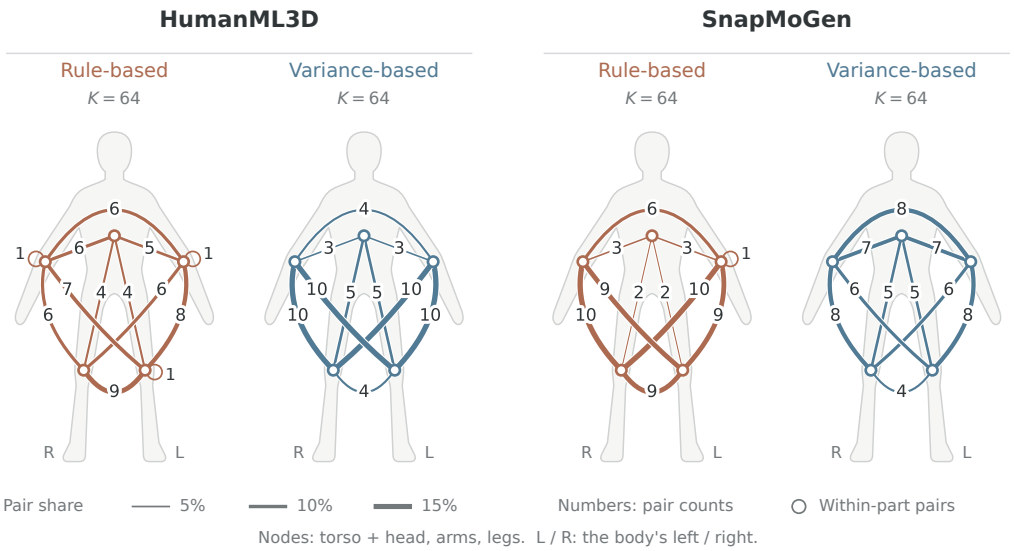

Figure A.1: **Distribution of the pairs of both families over five body parts.** The body is divided into five parts: the torso (with the head), the left and right arms (clavicle to wrist / hand), and the left and right legs (hip to foot / toe); the nodes are drawn at the torso, the elbows, and the knees. Line width is proportional to the share of pairs in the set that connect the two parts, numbers are pair counts, and small rings beside the nodes denote within-part pairs. L / R: the body's left / right. Each family is shown with its top 64 pairs. AMDD uses the top 64 pairs of the variance family on HumanML3D and the top 64 pairs of the rule-based family on SnapMoGen; on both datasets, $\mathcal{L}_{\text{aux}}$ uses the top 64 pairs of the rule-based family.

group is arm–leg pairs (42%–63%), and most endpoints are distal limb joints (elbows, wrists, knees, ankles, feet). Distances within one limb stay nearly constant or change only with the flexion of a single joint, whereas the distal joints accumulate the rotations along the kinematic chain, so their mutual distances vary the most, e.g., periodically with the gait cycle in walking and running.

Table A.1: Top 64 pairs of the two families on HumanML3D, in rank order. The top K ranks form the corresponding K-pair set; AMDD uses the top 64 of the variance family. L / R: left / right.

#	Rule-based family	Variance family	#	Rule-based family	Variance family
1	R ankle–R wrist	R ankle–R wrist	33	L knee–R foot	R elbow–L wrist
2	L wrist–R wrist	R foot–R wrist	34	L knee–L elbow	L ankle–head
3	L ankle–L wrist	L ankle–L wrist	35	L ankle–L collar	R ankle–head
4	L foot–R foot	L foot–L wrist	36	spine2–L wrist	head–R wrist
5	R ankle–L foot	L ankle–R wrist	37	L knee–R knee	head–L wrist
6	R ankle–R elbow	L foot–R wrist	38	R collar–R wrist	L foot–head
7	L ankle–R foot	R ankle–L wrist	39	L hip–L elbow	spine1–R wrist
8	R foot–L wrist	R foot–L wrist	40	spine1–R ankle	R foot–head
9	L foot–R wrist	R knee–R wrist	41	R knee–R elbow	spine1–L wrist
10	L ankle–L elbow	L knee–L wrist	42	R foot–neck	R knee–R elbow
11	pelvis–R wrist	L wrist–R wrist	43	R foot–L shoulder	L knee–L elbow
12	L knee–R wrist	L knee–R wrist	44	spine1–L ankle	L ankle–L shoulder
13	R ankle–L elbow	R knee–L wrist	45	L collar–L wrist	R ankle–R shoulder
14	L hip–L wrist	R hip–R wrist	46	spine2–L foot	L ankle–neck
15	head–L wrist	L hip–L wrist	47	L hip–R foot	R ankle–neck
16	L ankle–R elbow	L hip–R wrist	48	L knee–head	L foot–L shoulder
17	R foot–R shoulder	R hip–L wrist	49	L hip–L foot	L ankle–R shoulder
18	L elbow–R elbow	pelvis–R wrist	50	R knee–head	R ankle–L shoulder
19	head–R wrist	pelvis–L wrist	51	spine1–L elbow	R foot–R shoulder
20	R knee–L ankle	L foot–R foot	52	R knee–R shoulder	L ankle–L collar
21	L shoulder–R wrist	L ankle–L elbow	53	spine2–R elbow	L foot–neck
22	L knee–R ankle	R ankle–R elbow	54	R knee–L collar	L knee–R elbow
23	L foot–L shoulder	L ankle–R elbow	55	R shoulder–L elbow	R ankle–R collar
24	R ankle–head	L ankle–R foot	56	R knee–L shoulder	R knee–L elbow
25	L foot–head	L ankle–R ankle	57	spine3–L elbow	R foot–neck
26	R shoulder–L wrist	R ankle–L foot	58	L shoulder–R elbow	L ankle–R collar
27	L foot–R shoulder	R ankle–L elbow	59	L knee–L shoulder	L foot–R shoulder
28	R knee–L elbow	L foot–L elbow	60	head–L elbow	R ankle–L collar
29	pelvis–R elbow	R foot–R elbow	61	L knee–R shoulder	R foot–L shoulder
30	L knee–R elbow	L foot–R elbow	62	R knee–R collar	L ankle–spine3
31	R knee–L foot	R foot–L elbow	63	head–R elbow	R ankle–spine3
32	spine2–R wrist	L elbow–R wrist	64	L knee–R collar	L elbow–R elbow

Table A.2: Top 64 pairs of the two families on SnapMoGen, in rank order. AMDD uses the top 64 of the rule-based family.

#	Rule-based family	Variance family	#	Rule-based family	Variance family
1	R hand–R ankle	L hand–L ankle	33	R elbow–R toe	R elbow–R toe
2	L hand–L ankle	R hand–R ankle	34	L clavicle–L hand	L elbow–L toe
3	L hand–R hand	L hand–L toe	35	L shoulder–L ankle	R elbow–L ankle
4	R hand–L toe	R hand–R toe	36	R shoulder–R ankle	L elbow–R ankle
5	R hand–R hip	L hand–L knee	37	L shoulder–L toe	L ankle–R ankle
6	L hand–L hip	R hand–R knee	38	head–L ankle	L elbow–R toe
7	spine1–R hand	L hand–R ankle	39	L ankle–R knee	R elbow–L toe
8	L hand–L toe	R hand–L ankle	40	R elbow–L hip	neck2–L hand
9	L elbow–R hand	L hand–R hand	41	R shoulder–L ankle	neck2–R hand
10	R hand–R toe	L hand–R toe	42	R shoulder–R toe	head–R toe
11	L toe–R toe	R hand–L toe	43	L elbow–R toe	head–L toe
12	L hand–R elbow	L hand–R knee	44	L shoulder–R ankle	L elbow–L knee
13	head–L hand	R hand–L knee	45	head–R ankle	R elbow–R knee
14	L hand–R hip	L hand–L hip	46	R elbow–R knee	head–R ankle
15	head–R hand	R hand–R hip	47	R elbow–L toe	head–L ankle
16	L toe–R ankle	L hand–R hip	48	R elbow–L knee	L elbow–R elbow
17	R elbow–R ankle	R hand–L hip	49	L elbow–L knee	neck1–L hand
18	L ankle–R toe	root–L hand	50	R elbow–R hip	neck1–R hand
19	L elbow–L ankle	root–R hand	51	L knee–R ankle	neck2–R ankle
20	R elbow–L ankle	spine1–L hand	52	L toe–R knee	neck2–L ankle
21	L ankle–R ankle	spine1–R hand	53	R clavicle–L ankle	neck2–R toe
22	spine2–L hand	L toe–R toe	54	L knee–R toe	neck2–L toe
23	L elbow–R elbow	L hand–R elbow	55	L clavicle–R toe	spine3–L hand
24	spine3–R hand	L elbow–R hand	56	R clavicle–L toe	spine3–R hand
25	L hand–R ankle	head–L hand	57	L clavicle–R ankle	R shoulder–R ankle
26	L clavicle–R hand	head–R hand	58	root–L elbow	L shoulder–L ankle
27	L hand–R shoulder	L ankle–R toe	59	L elbow–R knee	L hand–R clavicle
28	L elbow–L toe	L toe–R ankle	60	R shoulder–L toe	L clavicle–R hand
29	L hand–R knee	L elbow–L ankle	61	L elbow–L hip	L hand–R shoulder
30	head–R toe	R elbow–R ankle	62	L shoulder–R toe	L shoulder–R hand
31	head–L toe	spine2–L hand	63	L knee–R knee	neck1–R ankle
32	L elbow–R ankle	spine2–R hand	64	R shoulder–R knee	neck1–L ankle

B Implementation and Evaluation Details

B.1 Backbones

MDM (BERT-Dec). All ablations, analyses, and SnapMoGen experiments are built on this backbone. They share one architecture and training recipe (the two parameter-matched single-stream baselines in Tab. 3b only double the width or the depth): an 8-layer Transformer decoder of width 512, a BERT text encoder, a 50-step cosine schedule, $\hat{x}_0$ prediction, EMA with decay 0.9999 (evaluated with the EMA weights), a text dropout rate of 0.1, a learning rate of 10^{-4}, and the same random seed for every configuration. On this backbone, AMDD has 61.0M trainable parameters and Base 26.2M.

The other two backbones. MDM (CLIP-Enc) and RAM are both retrained on the 67-d representation, and each differs architecturally from its original (see below). On both, AMDD adds the same components as on MDM (BERT-Dec) (a distance stream over the variance-family pairs with $K{=}64$ and the write-side text gate). Each backbone shares the changes below with its AMDD version; the backbone only removes the distance stream and the associated loss terms, and all other settings are identical.

MDM (CLIP-Enc). The original configuration prepends the pooled CLIP text token to the input of the global self-attention. With the distance stream added, the sequence length grows from $T{+}1$ to $2T{+}2$, diluting the attention the text token receives; in this configuration, adding the gate-free distance stream instead raises FID (variance family, $K{=}64$: T2M 0.321 $\rightarrow$ 0.633, MARDM 0.619 $\rightarrow$ 0.838; the rule-based family with $K{=}64$ also raises it; all with 5 repetitions). We therefore move the text out of the global self-attention: each stream has an additional attention layer whose queries are that stream's tokens and whose keys and values are the text token together with the stream's own tokens. The backbone includes this layer as well (its parameters grow from 17.7M to 26.1M), and its FID is also lower than that of the original configuration (T2M 0.245 vs. 0.321, MARDM 0.343 vs. 0.619; 5 repetitions). The 67-d MDM (CLIP-Enc) in Tab. D.5 is the original configuration.

RAM. It uses the same Transformer encoder and per-stream text attention as MDM (CLIP-Enc), with CLIP replaced by RAM's text encoder: the per-token features of a frozen DistilBERT are encoded by RAM's 6-layer Transformer into a single text token. Training adds RAM's reconstruction branch as in the original: at each step, the model denoises the same x_t at the same t once more, conditioned on the encoding of x_0 by RAM's motion encoder, and the denoising losses of the two branches are added with equal weights; $\mathcal{L}_{\text{geo}}$ and $\mathcal{L}_{\text{aux}}$ are applied only to the text-conditioned branch. Since the encoders are frozen, the latent alignment loss and the sentence-embedding loss that the original RAM (Liu et al., 2026b) uses to train its encoders are not used. Both encoders are taken from a RAM model that we trained on the 67-d representation (500k, EMA weights). That model has a bidirectional second stream of 64 joint-pair distances, and RAM trains its encoders jointly with the denoiser, so these two encoders were influenced by the distance stream during training. The backbone and RAM + AMDD use the same encoders.

Common settings. The two backbones and their AMDD versions are each trained once (batch size 64), and are evaluated at 300k / 400k / 500k (Tab. F.3). All three backbones use the 50-step cosine schedule, $\hat{x}_0$ prediction, and EMA weights, with guidance scale 2.5 at inference; RAM does not use REG. In the MDM (CLIP-Enc) pair, the backbone is trained with TF32 matrix multiplication and its AMDD version in fp32 (the difference between the two precisions is given in Appendix B.5).

B.2 Text Gate and Distance Auxiliary Loss

Write-side gate. Eq. (4) in the main text gives a gate built from a single linear map. AMDD uses its sparse mixture form: each layer ℓ has $N_e{=}4$ zero-initialized linear maps $W^g_{\ell,e}$ without bias, two of which are selected by the layer's router according to the text:

$$g_\ell = \mathbf{1} + \sum_{e \in \mathcal{T}_\ell(\bar{e}_c)} \pi_{\ell,e}(\bar{e}_c)\, \tanh\left(W^g_{\ell,e}\, \bar{e}_c\right), \tag{9}$$

where the router is a two-layer MLP ($512 \rightarrow 256 \rightarrow N_e$, SiLU activation), $\mathcal{T}_\ell(\bar{e}_c)$ is the set of the two experts with the highest scores, and $\pi_{\ell,e}$ are the softmax weights over these two scores.

Table B.1: Two runs of each read-side gate configuration (variance family, K=32). The runs differ only in random seed (10 / 11). Taking the lowest of the three checkpoints, run 2 reaches 0.096 / 0.138 (N_e=1) and 0.109 / 0.126 (N_e=4) (T2M / MARDM); re-evaluating the same weights on another machine changes FID by at most 0.0007. Bold: lowest of the three checkpoints within each run.

		Run 1			Run 2			Lowest of three	
Gate	Evaluator	300k	400k	500k	300k	400k	500k	Run 1	Run 2
Read-side, N_e=1	T2M	0.106	**0.080**	0.091	0.140	0.109	**0.096**	0.080	0.096
	MARDM	0.109	**0.084**	0.094	0.184	**0.138**	0.145	0.084	0.138
Read-side, N_e=4	T2M	0.137	0.101	**0.069**	0.141	0.113	**0.109**	0.069	0.109
	MARDM	0.180	0.127	**0.112**	0.189	0.129	**0.126**	0.112	0.126

With N_e=1, Eq. (9) reduces to Eq. (4). During training, we add the load-balancing loss of Switch Transformers (Fedus et al., 2022), $N_e \sum_e f_e \bar{P}_e$, where f_e is the fraction of samples routed to expert e and $\bar{P}_e$ is the mean routing probability of expert e; its coefficient is 0.01, and it is computed only on samples with text. The gate input $\bar{e}_c$ is the mean of the projected text tokens over the valid tokens. In the unconditional branch, $\bar{e}_c = \mathbf{0}$, so $g_\ell \equiv \mathbf{1}$; zero initialization makes $g_\ell \equiv \mathbf{1}$ at the start of training, equivalent to the gate-free model. In AMDD, each layer has its own router on HumanML3D, and the write-side gate adds 9.5M parameters in total (8.4M for the experts, 1.1M for the routers); on SnapMoGen, all layers share one router (0.13M parameters), and everything else is the same. Other gate variants in Tab. F.1 and Tab. H.1 differ in routing as marked there.

Read-side gate. The read-side gate in Tab. F.1 scales the distance values read by motion queries (the attention values) and leaves the distance stream unchanged. With N_e=1 it is Eq. (4) and adds 2.1M parameters (8 layers, 512×512 each); with N_e=4 it is mixed in the same way as the write-side gate.

Distance auxiliary loss (Sec. 3.4). Let $d^{\mathrm{gt}} = \mathcal{D}_{\mathcal{P}'}(\Phi(x_0^m))$. Then

$$\mathcal{L}_{\mathrm{aux}} = \mathbb{E}_{t,\epsilon}\Big[\big\|\sqrt{w} \odot \big(\mathcal{D}_{\mathcal{P}'}(\Phi(\hat{x}_0^m)) - d^{\mathrm{gt}}\big)\big\|_{\mathbf{v}}^2\Big], \qquad w = \mathbf{1} + \alpha \exp\big(-d^{\mathrm{gt}}/d_0\big), \tag{10}$$

where the weight w is computed per frame and per pair, carries no gradient, and gives the small-distance range a higher weight. In all configurations, $\mathcal{P}'$ is fixed to the top 64 pairs of the dataset's rule-based family (Appendix A); α=4 and d_0=0.1 m on both datasets (distances on SnapMoGen are converted to the body scale of HumanML3D). All configurations that use $\mathcal{L}_{\mathrm{geo}}$ and $\mathcal{L}_{\mathrm{aux}}$ set λ_{geo}=λ_{aux}=1.

B.3 Training and Sampling

Training. On HumanML3D, the batch size is 64 and each configuration is trained for 500k steps; six gated configurations are trained further to 700k steps (Tab. F.4). On SnapMoGen, the batch size is 256 and each configuration is trained for 300k–500k steps.

Sampling. Apart from the analyses that state other scales, all models use the same guidance scale s=2.5 for both streams at inference (s_m=s_r in Eq. (8)), with one exception: AMDD on SnapMoGen in Tab. 2 and Tab. D.3 uses s_m=4.5 and s_r=2.5 for the two streams, whereas Base (the MDM (BERT-Dec) row) in the same tables uses s=2.5. Results of the same AMDD weights at s=2.5 are in Tab. H.3, and a comparison of Base and AMDD at several guidance scales is in Tab. H.4. All other SnapMoGen results in Appendix H use s=2.5.

B.4 Repeated Runs

Tab. B.1 and Tab. B.2 give the per-checkpoint FID of the repeated training runs of the two read-side gate configurations and of AMDD on HumanML3D.

Table B.2: Per-checkpoint FID of the three runs of AMDD on HumanML3D (write-side MoE gate, variance family, $K=64$), with 5 repetitions. Bold: lowest entry of each run under each evaluator; –: not evaluated. Tab. 1 reports run 1 at 550k, which scores 0.064 / 0.064 with 20 repetitions. Run 2 was evaluated only at 500k.

Run	Evaluator	300k	400k	500k	550k	600k
Run 1 (TF32)	T2M	0.112±.003	0.088±.008	0.069±.003	**0.067**±.004	0.094±.003
	MARDM	0.143±.008	0.090±.009	0.084±.009	**0.063**±.002	0.083±.006
Run 2 (fp32)	T2M	–	–	**0.101**±.006	–	–
	MARDM	–	–	**0.102**±.006	–	–
Run 3 (fp32)	T2M	0.133±.006	0.127±.008	**0.104**±.007	–	–
	MARDM	0.194±.020	0.169±.012	**0.108**±.008	–	–

Table B.3: Results under the 1000-sample protocol (HumanML3D, 500k, 20 repetitions). Both two-stream configurations use the variance family with $K=32$; the read-side gate has $N_e=1$. Best FID in bold.

	T2M evaluator					MARDM evaluator				
Configuration	FID↓	R@1↑	R@3↑	MM-Dist↓	Div→	FID↓	R@1↑	R@3↑	Matching↓	Div→
Real	0.0016	0.512	0.795	2.978	9.44	0.0004	0.503	0.794	3.255	10.43
Base (67-d)	0.341±.030	0.545	0.834	2.768	10.01	0.433±.043	0.531	0.827	3.155	11.05
+ distance stream	0.213±.018	0.560	0.848	2.686	9.79	0.286±.027	0.542	0.830	3.047	10.89
+ distance stream + read-side gate	**0.180**±.020	0.552	0.841	2.714	9.65	**0.191**±.023	0.544	0.833	3.026	10.64

B.5 Evaluation Protocol

Scoring and repetitions. On HumanML3D, each repetition scores the full test set (4640 samples), and we report the mean and the 95% confidence interval. The rows we trained in Tab. 1 use 20 repetitions, as do MoMask, SALAD, and other methods; the ablations and the remaining configurations in the appendix use 5. The number of repetitions mainly affects the width of the confidence interval; the means stay within it: on the same weights, AMDD scores 0.067 / 0.063 with 5 repetitions and 0.064 / 0.064 with 20 (T2M / MARDM). The per-checkpoint results in Tabs. B.2, F.3, and F.4 all use 5 repetitions. MDM (Tevet et al., 2023) scores only 1000 samples per repetition; because of the finite-sample bias of FID (Chong & Forsyth, 2020), this protocol yields a higher FID, by +0.08 to +0.13 on the same weights (Tab. B.3), so results under it are not directly comparable to ours. On SnapMoGen, we use the official evaluator and score the full test set (4000 samples); TOP-k is 100-way cosine retrieval. Our two rows in Tab. 2 and the results in Tab. H.3 use 20 repetitions (as in the official protocol); the guidance sweep in Tab. H.4 uses a single repetition, and all other SnapMoGen results use 5.

263-d reconstruction. The 263-d representation required by the T2M evaluator on HumanML3D is reconstructed from joint positions with the dataset's standard pipeline. Its round-trip FID is 0.024; the reconstruction is the same for all 67-d models, but this value is not an upper bound on its effect.

1000-sample protocol. To show the size of this protocol difference, we evaluate the 500k weights of three configurations (the same as the 500k column of Tab. F.3) under MDM's protocol: each repetition scores only 1000 samples, with 20 repetitions (Tab. B.3). The Real row agrees with the published values entry by entry (T2M: 0.0016 / 0.512 / 0.795 / 2.978 vs. 0.002 / 0.511 / 0.797 / 2.974). Compared with the 500k entries of the same weights in Tab. F.3, this protocol gives an FID higher by 0.08–0.13 (Base +0.13 / +0.13, + distance stream +0.08 / +0.11, + read-side gate +0.09 / +0.10). This is the range observed on the three tested configurations, not a conversion rule across methods; the ranking of the three configurations is the same under both protocols.

FID reductions relative to Base within Tab. B.3 (same 500k weights) are -37% / -34% for + distance stream and -47% / -56% for + read-side gate; AMDD with the write-side gate was not evaluated under this protocol. Under the full-test-set protocol, the reductions on the same 500k weights are -39% / -42% and -57% / -69% (Tab. F.3), and -39% / -41% and -63% / -72% when each configuration takes its best evaluated checkpoint.

Variability scales. The 95% confidence intervals in the tables cover only repeated generation and evaluation at a fixed checkpoint, not training randomness or checkpoint selection. For reference, we measure three further sources of variability on HumanML3D (FID, T2M / MARDM). (i) Evaluation: repeated full evaluations of the same weights, including across machines, differ by up to about 0.01 / 0.02. (ii) Retraining: seven configurations trained twice, once with TF32 and once with fp32 matrix multiplication, differ at 500k by 0.005–0.032 / 0.005–0.019; the seed-based second runs of the two read-side gates differ by 0.005–0.051 (Tab. B.1). (iii) Checkpoints: in the six further-trained configurations, adjacent checkpoints 50k steps apart differ by 0.007–0.027 / 0.006–0.028, 1–7 times the confidence interval of an entry (Tab. F.4). We use (ii) as a reference scale, not as a significance threshold. On SnapMoGen, the confidence interval of an entry is typically about ± 0.10 for FID and ± 0.004 for TOP-1, two trainings of the same configuration differ by 0.5–0.7 FID at 300k, and FID changes over 25k steps by a median of -0.55 (range -1.6 to $+2.0$).

C Inference Cost

Relative to Base, AMDD uses 2.33× the parameters, 2.82× the time (2.34× at batch size 32), and 1.33× the memory (1.62× at batch size 32). Most of this comes from the distance stream: adding the stream alone already reaches 1.96× in parameters, 1.82× (2.24×) in time, and 1.26× (1.54×) in memory, with batch size 32 in parentheses, because the two streams share one attention operation and the sequence length grows from T to $2T$. Compared with the read-side gate, the write-side gate takes 27% more time at batch size 1, the same time at batch size 32 (0.995×), and slightly more memory (Tab. C.1). The time at batch size 1 is dominated by kernel-launch overhead and differs by 8–14% across machines; at batch size 32, it differs by at most 2.5%.

Table C.1: Inference cost on a single RTX 4090 (torch 2.5.1, fp32, TF32 disabled) with T=196 frames, 50 steps, and guidance scale 2.5 (two forward passes per step). Trainable parameters exclude the frozen text encoder. The times of the four configurations are measured consecutively in one process (with randomly initialized weights; time and memory do not depend on the weight values); peak memory is measured for each configuration in a separate process.

		Batch size 1		Batch size 32	
Configuration	Trainable params	Time / clip	Peak memory	Time / 32 clips	Peak memory
Base (67-d)	26.2M	0.65 s	0.39 GB	1.22 s	0.50 GB
+ distance stream	51.5M	1.19 s	0.49 GB	2.72 s	0.77 GB
+ distance stream + read-side gate	53.6M	1.45 s	0.50 GB	2.87 s	0.79 GB
AMDD	61.0M	1.83 s	0.52 GB	2.86 s	0.81 GB

D Complete Comparison Tables

Tab. 1 and Tab. 2 list only a subset of methods. This section gives the complete comparison: Tab. D.1 and Tab. D.2 list all methods on HumanML3D under the two evaluators, Tab. D.3 all methods on SnapMoGen, Tab. D.4 further methods not listed in Tab. D.1 (non-comparable protocol or data, preprints, decoder plug-ins, and others), and Tab. D.5 the two MDM backbones on the 263-d and 67-d representations.

263-d vs. 67-d representation. Replacing the 263-d representation with the 67-d representation (Meng et al., 2025b) by itself lowers the FID of both backbones (CLIP-Enc 0.640 → 0.321, BERT-Dec 0.412 → 0.214; Tab. D.5). The FID reductions of AMDD in Tab. 1 are all computed against the 67-d backbones and do not include this gain. On MDM (BERT-Dec), the 263-d representation gives a higher R-precision under the T2M evaluator than both Base and AMDD on the 67-d representation (R@1 0.565 vs. 0.545 / 0.547), and AMDD does not close this gap; under the MARDM evaluator, AMDD matches the 263-d model (R@1 0.542 vs. 0.541).

Table D.1: Complete results on the HumanML3D test set (T2M evaluator). Methods are grouped into discrete-token and diffusion-based methods; best and second best per group in **bold** and underlined (Div: closest to real motion), with our rows ranked among the diffusion-based methods. Literature rows include peer-reviewed text-to-motion methods trained only on HumanML3D motion data, ordered by publication date within each group; preprints, methods using extra data or a non-comparable evaluation protocol, and decoder plug-ins are in Tab. D.4. In the Ours block, all three backbones are retrained by us on the 67-d representation under the same protocol; parentheses give the FID reduction of AMDD over its backbone. These rows are scored on the full test set with 20 repetitions, as are MoMask, SALAD, and other methods. $^{\diamond}$ Reported by RAM (263-d representation): the RAM row is its main result, which uses reconstruction-error guidance (REG) at inference; the RAM w/o REG row is taken from Table 4 of RAM, without REG and with guidance scale 2.5, the same inference setting as ours. † Our implementation: RAM's frozen encoders and reconstruction branch on MDM's Transformer encoder (Appendix B.1). ‡ MARDM reports this result only in an appendix table, without $\pm$. MotionLCM uses 4-step sampling. Backbone results on the 263-d representation are in Tab. D.5. –: not evaluated; n/r: not reported.

Method	FID↓	R@1↑	R@2↑	R@3↑	MM-Dist↓	Div→	MModality
Real	0.002±.000	0.511±.003	0.703±.003	0.797±.002	2.974±.008	9.503±.065	–
Discrete-token methods							
T2M-GPT (Zhang et al., 2023a)	0.141±.005	0.492±.003	0.679±.002	0.775±.002	3.121±.009	9.722±.082	1.831±.048
MoMask (Guo et al., 2024)	0.045±.002	0.521±.002	0.713±.002	0.807±.002	2.958±.008	n/r	1.241±.040
MMM (Pinyoanuntapong et al., 2024b)	0.089±.005	0.515±.002	0.708±.002	0.804±.002	2.926±.007	9.577±.050	1.226±.035
BAMM (Pinyoanuntapong et al., 2024a)	0.055±.002	0.525±.002	0.720±.003	0.814±.003	2.919±.008	9.717±.089	1.687±.051
MoGenTS (Yuan et al., 2024)	0.033±.001	0.529±.003	0.719±.002	0.812±.002	2.867±.006	9.570±.077	n/r
LaMP (Li et al., 2025)	0.032±.002	0.557±.003	0.751±.002	0.843±.001	2.759±.007	9.571±.069	n/r
KinMo (Zhang et al., 2025b)	0.039±.003	0.532±.002	0.724±.003	0.821±.003	2.901±.010	9.674±.058	1.321±.039
MoMask++ (Guo et al., 2025a)	0.072±.003	0.528±.003	0.718±.003	0.811±.002	2.912±.008	n/r	1.227±.046
MoScale (Zheng et al., 2026)	0.046±.002	0.540±.002	0.727±.002	0.817±.002	2.830±.005	**9.525**±.090	0.873±.044
ScaleMoGen (Hwang et al., 2026)	**0.030**±.002	**0.577**±.004	**0.769**±.003	**0.856**±.002	**2.626**±.006	n/r	1.105±.057
Diffusion-based methods							
MotionDiffuse (Zhang et al., 2024a)	0.630±.001	0.491±.001	0.681±.001	0.782±.001	3.113±.001	9.410±.049	1.553±.042
MLD (Chen et al., 2023)	0.473±.013	0.481±.003	0.673±.003	0.772±.002	3.196±.010	9.724±.082	2.413±.079
ReMoDiffuse (Zhang et al., 2023b)	0.103±.004	0.510±.005	0.698±.006	0.795±.004	2.974±.016	9.018±.075	1.795±.043
FineMoGen (Zhang et al., 2023c)	0.151±.008	0.504±.002	0.690±.002	0.784±.002	2.998±.008	9.263±.094	2.696±.079
MotionLCM (Dai et al., 2024b)	0.304±.012	0.502±.003	0.698±.002	0.798±.002	3.012±.007	9.607±.066	2.259±.092
StableMoFusion (Huang et al., 2024a)	0.098±.003	0.553±.003	0.748±.002	0.841±.002	n/r	9.748±.092	1.774±.051
MARDM-SiT‡ (Meng et al., 2025b)	0.061	0.523	0.715	0.810	n/r	n/r	n/r
SALAD (Hong et al., 2025)	0.076±.002	0.581±.003	0.769±.003	0.857±.002	2.649±.009	9.696±.096	1.751±.062
CMDM (Yu et al., 2026)	0.068±.003	**0.588**±.004	**0.778**±.002	**0.860**±.003	**2.620**±.010	n/r	1.785±.074
RAM$^{\diamond}$ (Liu et al., 2026b)	**0.032**±.002	0.561±.003	0.751±.002	0.839±.002	2.716±.007	**9.487**±.084	1.142±.038
RAM w/o REG$^{\diamond}$ (Liu et al., 2026b)	0.107±.004	0.563±.002	0.753±.002	0.840±.002	2.729±.006	9.711±.066	n/r
Ours: AMDD on three backbones (67-d, retrained under the same protocol)							
MDM (CLIP-Enc)	0.236±.007	0.497±.003	0.691±.002	0.786±.002	3.069±.007	10.002±.092	–
+ AMDD (−75%)	0.059±.004	0.506±.003	0.697±.003	0.794±.003	2.982±.010	9.429±.053	–
MDM (BERT-Dec)	0.219±.006	0.547±.003	0.743±.003	0.836±.002	2.776±.011	10.042±.087	–
+ AMDD (−71%)	0.064±.003	0.547±.003	0.743±.002	0.836±.002	2.743±.007	9.580±.060	–
RAM† w/o REG	0.099±.003	0.552±.002	0.747±.002	0.836±.002	2.744±.007	9.790±.068	–
+ AMDD (−51%)	0.048±.003	0.552±.003	0.746±.003	0.838±.002	2.710±.007	9.604±.065	–

Table D.2: Complete results on the HumanML3D test set (MARDM evaluator, 67-d). Methods are grouped into discrete-token and diffusion-based methods; best and second best per group in **bold** and underlined, with our rows ranked among the diffusion-based methods. § Retrained by Meng et al. (2025b) with guidance scale 4.5. ⋆ Re-evaluated by Yu et al. (2026) with guidance scale 3.0. The Ours block (including RAM†) is as in Tab. D.1: guidance scale 2.5, full test set, 20 repetitions; parentheses give the FID reduction over each backbone. RAM does not report results under the MARDM evaluator and is therefore not listed. Preprints (arXiv only as of September 2026) are listed separately and not marked. –: not evaluated.

Method	FID↓	R@1↑	R@2↑	R@3↑	Matching↓	MModality
Real	0.000±.000	0.503±.002	0.696±.001	0.795±.002	3.244±.005	–
Discrete-token methods						
T2M-GPT§ (Zhang et al., 2023a)	0.335±.003	0.470±.003	0.659±.002	0.758±.002	3.505±.017	2.018±.053
MMM§ (Pinyoanuntapong et al., 2024b)	0.132±.004	0.487±.003	0.683±.002	0.782±.001	3.359±.009	1.241±.073
MoMask§ (Guo et al., 2024)	**0.116**±.006	**0.490**±.004	**0.687**±.003	**0.786**±.003	**3.353**±.010	1.263±.079
Diffusion-based methods						
MDM (50 steps)§ (Tevet et al., 2023)	0.518±.032	0.440±.007	0.636±.006	0.742±.004	3.640±.028	3.604±.031
MotionDiffuse§ (Zhang et al., 2024a)	0.778±.005	0.450±.006	0.641±.005	0.753±.005	3.490±.023	3.179±.046
MLD§ (Chen et al., 2023)	0.431±.014	0.461±.004	0.651±.004	0.750±.003	3.445±.019	3.506±.031
ReMoDiffuse§ (Zhang et al., 2023b)	0.883±.021	0.468±.003	0.653±.003	0.754±.005	3.414±.020	2.703±.154
MARDM-DDPM (Meng et al., 2025b)	0.116±.004	0.492±.006	0.690±.005	0.790±.005	3.349±.010	2.470±.053
MARDM-SiT (Meng et al., 2025b)	0.114±.007	0.500±.004	0.695±.003	0.795±.003	3.270±.009	2.231±.071
SALAD⋆ (Hong et al., 2025)	0.124±.005	0.552±.003	0.748±.003	0.839±.002	2.990±.010	1.833±.081
CMDM (AR) (Yu et al., 2026)	0.085±.004	0.550±.004	0.747±.004	0.838±.003	2.987±.011	1.810±.068
CMDM (Yu et al., 2026)	0.078±.003	**0.563**±.004	**0.759**±.003	**0.849**±.002	**2.920**±.007	1.827±.094
Ours: AMDD on three backbones (67-d, retrained under the same protocol)						
MDM (CLIP-Enc)	0.338±.008	0.476±.004	0.668±.003	0.769±.003	3.465±.010	–
+ AMDD (−84%)	0.054±.003	0.491±.002	0.686±.002	0.784±.002	3.301±.007	–
MDM (BERT-Dec)	0.297±.006	0.529±.002	0.731±.002	0.826±.002	3.134±.008	–
+ AMDD (−78%)	0.064±.002	0.542±.002	0.740±.003	0.833±.002	3.007±.008	–
RAM† w/o REG	0.081±.003	0.534±.002	0.731±.002	0.826±.002	3.071±.009	–
+ AMDD (−43%)	**0.046**±.004	0.541±.002	0.735±.002	0.829±.002	3.024±.006	–
Preprints (not marked)						
ACMDM-XL (Meng et al., 2025a)	0.058±.004	0.522±.002	0.713±.002	0.807±.002	3.205±.008	2.077±.083

Table D.3: Complete results on the SnapMoGen test set (official evaluator). Inclusion as in Tab. D.1; methods are grouped into discrete-token and diffusion-based methods, with best and second best per group in **bold** and underlined (our rows in the diffusion-based group). Preprints (arXiv only as of September 2026) are listed separately and not marked. § Retrained by Guo et al. (2025a) with T5-Base on the 296-d representation. We use only the MDM (BERT-Dec) backbone and compare it with and without AMDD (148-d, retrained under the same protocol). The backbone is taken at 500k, its lowest-FID checkpoint among those evaluated, and AMDD (rule-based family, K=64, write-side gate) at 300k; sampling settings are in Appendix B.3, and a per-checkpoint comparison in Fig. H.1. –: not evaluated; n/r: not reported.

Method	FID↓	TOP-1↑	TOP-2↑	TOP-3↑	Matching↑	MModality
Real	0.001±.000	0.940±.001	0.976±.001	0.985±.001	0.837±.000	–
Discrete-token methods						
T2M-GPT§ (Zhang et al., 2023a)	32.63±.09	0.618±.002	0.773±.002	0.812±.002	0.573±.001	9.17±.18
MoMask§ (Guo et al., 2024)	17.40±.05	0.777±.002	0.888±.002	0.927±.002	0.664±.001	8.18±.18
MoMask++ (Guo et al., 2025a)	**15.06**±.07	0.802±.001	0.905±.002	0.938±.001	0.685±.001	7.26±.18
ScaleMoGen (Hwang et al., 2026)	16.35±.09	**0.807**±.004	**0.908**±.003	**0.941**±.003	**0.693**±.001	9.40±.67
Diffusion-based methods						
MDM§ (Tevet et al., 2023)	57.78±.09	0.503±.002	0.653±.002	0.727±.002	0.481±.001	13.41±.23
StableMoFusion§ (Huang et al., 2024a)	27.80±.06	0.679±.002	0.823±.002	0.888±.002	0.605±.001	9.06±.14
MARDM§ (Meng et al., 2025b)	26.88±.13	0.659±.002	0.812±.002	0.860±.002	0.602±.001	9.81±.29
CMDM (AR) (Yu et al., 2026)	15.01±.07	0.824±.002	0.918±.004	0.951±.002	0.699±.001	9.74±.19
CMDM (Yu et al., 2026)	**14.45**±.07	**0.831**±.004	**0.926**±.003	**0.958**±.002	**0.702**±.001	9.52±.20
MDM (BERT-Dec)	18.86±.08	0.785±.003	0.891±.002	0.927±.001	0.673±.001	–
+ AMDD	16.56±.07	0.805±.002	0.907±.001	0.941±.001	0.687±.001	–
Preprints (not marked)						
FlowCoMotion (Guan et al., 2026)	14.68±.06	0.776±.002	0.890±.002	0.934±.001	0.672±.001	8.55±.23
Unified Cond. Flow (Li et al., 2026)	16.49±.08	0.807±.002	0.906±.002	0.939±.001	0.688±.001	9.58±.18
MUGEN (Ye et al., 2026)	21.05±.07	0.815±.003	0.916±.002	0.947±.001	0.698±.001	6.09±.32
ReMoMask-2 (Wang et al., 2026)	13.17±.06	0.788±.003	0.897±.004	0.934±.002	n/r	8.33±.17

Table D.4: Methods not listed in Tab. D.1 (HumanML3D, T2M evaluator). Values are taken from the original papers (* quoted from Table 1 of RAM (Liu et al., 2026b)); values without $\pm$ are reported only as means in the original papers. $^\circ$ MDM scores only 1000 samples per repetition and uses different evaluator weights, so it is not directly comparable with the other rows (Appendix B.5). ¶ LMM is trained jointly on MotionVerse (16 datasets, 320K sequences), i.e., on more training data than the other methods. MotionGPT uses a pretrained language model; T2M is the baseline released with the dataset. $^\diamondsuit$ COME is an ICLR 2026 submission that was not accepted and has no arXiv version (we cite its OpenReview page). $^\triangleright$ MotionLCM v2 is a technical report released by the authors (an extension of MotionLCM); the row is its 4-step MotionLCM-M. n/r: not reported.

Method	FID↓	R@1↑	R@2↑	R@3↑	MM-Dist↓	Div→	MModality
Non-comparable evaluation protocol or training data							
MDM$^\circ$ (Tevet et al., 2023)	0.544±.044	0.320±.005	0.498±.004	0.611±.007	5.566±.027	9.559±.086	2.799±.072
LMM-Large¶ (Zhang et al., 2024b)	0.040±.002	0.525±.002	0.719±.002	0.811±.002	2.943±.012	9.814±.076	2.683±.054
Preprints, technical reports, and unaccepted submissions (not peer-reviewed publications)							
COME$^{*\diamondsuit}$ (Lyu et al., 2025)	0.041±.002	0.526±.003	0.723±.002	0.816±.002	2.898±.006	9.532±.062	1.704±.059
MotionLCM v2$^{*\triangleright}$ (Dai et al., 2024a)	0.056±.003	0.553±.003	0.746±.002	0.837±.002	2.773±.009	9.598±.067	1.758±.056
Motion Anything (Zhang et al., 2025d)	0.028	0.546	0.735	0.829	2.859	9.521	2.705
FlowCoMotion (Guan et al., 2026)	0.061	0.550	0.740	0.832	2.852	9.457	2.071
ReMoMask-2 (Wang et al., 2026)	0.042	0.528	0.719	0.813	2.860	9.472	1.243
MUGEN (Ye et al., 2026)	0.087	0.579	0.772	0.859	2.629	9.510	0.83
MotionCLR (Chen et al., 2024)	0.099	0.542	0.733	0.827	2.981	n/r	2.145
Decoder plug-ins							
MoMask + DisCoRD (Cho et al., 2025)	0.032±.002	0.524±.003	0.715±.003	0.809±.002	2.938±.010	n/r	1.288±.043
Other methods							
T2M (Guo et al., 2022)	1.087±.021	0.455±.003	0.636±.003	0.736±.002	3.347±.008	9.175±.083	2.219±.074
MotionGPT (Jiang et al., 2023)	0.232±.008	0.492±.003	0.681±.003	0.778±.002	3.096±.008	9.528±.071	2.008±.084
BAD (CBS) (Hosseyni et al., 2025)	0.049	0.511	0.704	0.800	2.957	9.688	1.119
EnergyMoGen (Zhang et al., 2025a)	0.176	0.526	0.718	0.815	2.931	9.500	2.270
MotionLab (Guo et al., 2025b)	0.167	n/r	n/r	0.810	2.830	9.593	2.912

Table D.5: Two MDM backbones on the original 263-d representation and on the 67-d representation (HumanML3D, 5 repetitions). The two rows of each backbone share the architecture and differ only in the representation; the 263-d rows are trained and evaluated directly on the original representation. The MDM (CLIP-Enc) backbone in Tab. 1 instead uses per-stream text attention (Appendix B.1), so its architecture differs from that of the 67-d row here; the MDM (BERT-Dec) numbers in Tab. 1 are from 20 repetitions of the same checkpoint. The RAM backbone is not retrained on the 263-d representation.

		T2M evaluator					MARDM evaluator			
Backbone	Repr.	FID↓	R@1↑	R@3↑	MM-Dist↓	Div	FID↓	R@1↑	R@3↑	Matching↓
MDM (CLIP-Enc)	263-d	0.640±.026	0.498	0.790	3.102	10.489	0.811±.007	0.471	0.766	3.527
	67-d	0.321±.006	0.473	0.772	3.209	10.028	0.619±.021	0.469	0.771	3.541
MDM (BERT-Dec)	263-d	0.412±.007	0.565	0.851	2.716	10.123	0.456±.009	0.541	0.833	3.077
	67-d	0.214±.012	0.545	0.838	2.769	10.051	0.300±.016	0.532	0.829	3.121

E CONDITIONAL DISTRIBUTION ANALYSIS

Sec. 4.2 notes that AMDD lowers FID substantially on HumanML3D while R-precision stays on par with the backbones. This section decomposes the generated distribution in the embedding space of the evaluators to show where the FID reduction comes from and why R-precision does not rise with it. The analysis covers the two gate-free $K{=}64$ configurations, without the text gate; the effect of the gate is analyzed in Appendix G.

The FID gain is a second-moment effect. When the FID of the evaluator embeddings is split into a mean term and a covariance term, at least 94% of the improvement of the distance-stream configurations over Base comes from the covariance term in five of six comparisons, and the mean term worsens in four. Without the distance stream, the models are over-dispersed by 9–22% along the leading principal directions of the real embeddings, and the distance-stream configurations bring this dispersion back close to the real level. Across models, Diversity alone explains 49% / 63% of the variance of FID (T2M / MARDM evaluator), and R@1 only 1.4% / 17%.

Table E.1: Decomposition of the conditional distribution in the evaluator space. Each cell gives rule-based / variance family ($K{=}64$), both gate-free. D: within-caption dispersion; $d(\bar{z}, t)$: distance from the conditional mean $\bar{z}$ to its caption t. The two effects are the main effects on R@1 of the $2{\times}2$ mean–residual swap; on SnapMoGen, R@1 is the TOP-1 retrieval accuracy.

	HumanML3D, T2M	HumanML3D, MARDM	SnapMoGen
$D_{\text{dist. stream}} / D_{\text{Base}}$	0.967 / 0.896	0.916 / 0.876	0.756 / 0.724
$\Delta d(\bar{z}, t)$ vs Base	+0.049 / +0.042 (away)	+0.015 / −0.007 (≈0)	−0.0084 / −0.0041 (toward)
Conditional-residual effect (main effect)	+0.5 / +0.6 pp	+0.8 / +0.9 pp	+2.4 / +2.6 pp
Conditional-mean effect (main effect)	−0.9 / −0.5 pp	−0.4 / +0.2 pp	+1.7 / +1.3 pp
R@1: Base → distance stream	0.552 → 0.548 / 0.553	0.533 → 0.537 / 0.544	0.771 → 0.811 / 0.810

Conditional mean and conditional residual (Tab. E.1). We generate 16 samples for each test caption and decompose them in the evaluator space (the analyzed configurations are the two gate-free $K{=}64$ distance-stream configurations, not the gated AMDD). On both datasets, the distance stream narrows the within-caption dispersion (by 3–12% on HumanML3D and 24–28% on SnapMoGen), whereas the shift of the conditional mean depends on the evaluator: it moves toward the text on SnapMoGen and away from the text under the T2M evaluator on HumanML3D. On SnapMoGen, the $2{\times}2$ decomposition that swaps conditional means and residuals gives a main effect of +2.4 / +2.6 pp for the conditional residual and +1.7 / +1.3 pp for the conditional mean; under the T2M evaluator on HumanML3D, the mean effect is reversed (−0.9 / −0.5 pp) and offsets the gain from the residual. A sweep over the guidance scale (Fig. E.1) shows that the mean gap under the T2M evaluator is present at every tested scale in $s \in [1, 4]$ for the rule-based family, and from $s{=}2.5$ on for the variance family (not significant at $s{=}1.5$–2.0 and absent at $s{=}1$).

Under the T2M evaluator on HumanML3D, the retrieval gain from the narrower within-caption dispersion of the distance stream is offset by the conditional mean moving away from the text, and R@1 stays on par with Base; on SnapMoGen, the two effects point in the same direction, and TOP-1 rises accordingly. Fig. E.1 shows that this mean gap is not specific to the guidance scale 2.5.

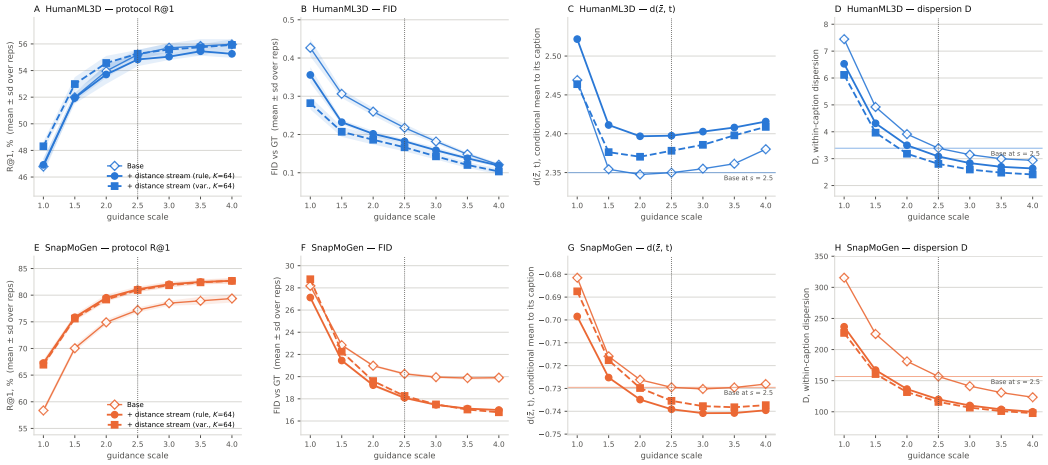

Figure E.1: Conditional dispersion, conditional mean, R@1, and FID on the same samples at guidance scales $s \in \{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$. Base and the two distance-stream configurations (rule-based / variance family, K=64; HumanML3D at 500k, SnapMoGen at 300k), with 8 samples per caption in each cell; all configurations and scales share the same noise (paired), and the s=2.5 cells use the first 8 of the 16 samples per caption behind Tab. E.1. On SnapMoGen, R@1 is the TOP-1 retrieval accuracy. **HumanML3D** (top row, T2M evaluator; under the MARDM evaluator the conclusion for the dispersion term is the same and that for the mean term differs): the $d(\bar{z}, t)$ of Base is U-shaped in s (closest at s=2), and the curves of the distance-stream configurations are the same U shifted up by +0.02 to +0.06, so the mean gap is a roughly constant offset (for the variance family, the shift is absent at s=1 and not significant at s=1.5–2.0); the dispersion ratio rises with s and then plateaus (variance family 0.86 $\rightarrow$ 0.91, rule-based family 0.92 $\rightarrow$ 0.97); the R@1 advantage of the variance family, +1.5 pp at s=1, decays monotonically and crosses zero exactly at s=2.5 (the rule-based family is never more than +0.1 pp ahead); FID on the same samples is lower for both distance-stream configurations at every scale (variance family $-0.145 \rightarrow -0.017$, rule-based family $-0.071 \rightarrow -0.002$). **SnapMoGen** (bottom row): at every scale $s \geq 2$, both distance-stream configurations give a lower dispersion (ratio 0.71–0.81) and a closer mean; the R@1 advantage decays from +8.9 / +8.6 pp (rule-based / variance family) at s=1 to a plateau of about +3.3 pp rather than to zero, and the FID advantage grows with s to -2.9 / -3.1. The mean gap under the T2M evaluator in Tab. E.1 is therefore not specific to the guidance scale 2.5.

F ADDITIONAL ABLATIONS ON HUMANML3D

This section gives the HumanML3D ablations beyond Tab. 3 and Tab. 4 (Tab. F.1), all metrics of Tab. 3a,b (Tab. F.2), and the per-checkpoint results of all configurations (Tab. F.3 and Tab. F.4).

Table F.1: Loss terms and text-gate variants. FID under both evaluators; for each configuration, the lowest FID of that evaluator over all evaluated checkpoints. Run 2 / 3: repeated training runs of the same configuration (Appendix B.4). § 20 repetitions (as in Tab. 1); all other rows use 5 repetitions, with which this row scores 0.067 / 0.063.

Configuration	FID (T2M)↓	FID (MARDM)↓
Loss terms, rule-based family, K=64 (gate-free)		
Full (λ_{geo}=λ_{aux}=1)	0.182±.013	0.231±.005
− $\mathcal{L}_{\text{geo}}$	0.167±.006	0.192±.006
− $\mathcal{L}_{\text{aux}}$	0.178±.010	0.245±.005
$\mathcal{L}_{\text{simple}}$ only	0.187±.008	0.242±.009
Loss terms, variance family, K=32 (gate-free)		
Full	0.132±.004	0.178±.010
− $\mathcal{L}_{\text{geo}}$	0.141±.005	0.198±.008
Loss terms, variance family, K=64 (gate-free)		
Full	0.152±.008	0.152±.006
− $\mathcal{L}_{\text{geo}}$	0.180±.005	0.186±.005
$\mathcal{L}_{\text{simple}}$ only	0.188±.008	0.217±.007
Gates (variance family, K=32, unless noted)		
None	0.132±.004	0.178±.010
Read-side gate, N_e=1 (+2.1M)	0.080±.006	0.084±.005
↪ run 2	0.096±.007	0.138±.006
Read-side gate, N_e=4	0.069±.006	0.095±.006
↪ run 2	0.109±.003	0.126±.010
Write-side gate, N_e=4 (rule-based family, K=64; no per-layer router)	0.094±.009	0.085±.010
Gates (variance family, K=64; gate-free: 0.152 / 0.152)		
Write-side MoE gate, N_e=4, per-layer router (AMDD, +9.5M)§	0.064±.003	0.064±.002
↪ run 2	0.101±.006	0.102±.006
↪ run 3	0.104±.007	0.108±.008

Loss terms and gate variants. The effect of the two geometric loss terms ($\mathcal{L}_{\text{geo}}$ and $\mathcal{L}_{\text{aux}}$) depends on the relation variable: removing $\mathcal{L}_{\text{geo}}$ worsens FID on the variance family (K=64: 0.152 / 0.152 → 0.180 / 0.186; K=32: 0.132 / 0.178 → 0.141 / 0.198) but improves it on the rule-based family with K=64 (0.182 / 0.231 → 0.167 / 0.192; Tab. F.1). Even with $\mathcal{L}_{\text{simple}}$ alone, the dual-stream model still improves on Base (rule-based family, K=64: MARDM FID 0.242 vs 0.300; variance family, K=64: 0.217 vs 0.300). Every gate variant improves on its gate-free counterpart in every run, but the ranking among the gate variants is not stable across repeated runs.

Table F.2: All metrics of the incremental experiments and the parameter-matched baselines on HumanML3D (full version of Tab. 3a,b). Starting from the single-stream baseline, we add the distance stream, the distance auxiliary loss $\mathcal{L}_{\text{aux}}$, the geometric consistency loss $\mathcal{L}_{\text{geo}}$, and the text gate in turn; all rows use the joint denoising loss $\mathcal{L}_{\text{simple}}$, and the relation variable is fixed to the variance family with $K{=}64$. The bottom two rows are parameter-matched single-stream baselines trained with $\mathcal{L}_{\text{simple}}$ only. R-precision, MM-Dist, and Diversity are under the T2M evaluator. Best and second best per column in **bold** and underlined (for Diversity, the value closest to the real data, 9.503, as in the Real row of Tab. 1). The text-gate row is MDM (BERT-Dec) + AMDD of Tab. 1, with 20 repetitions; all other rows use 5 repetitions, which mainly affects the width of the confidence intervals.

Dist. stream	$\mathcal{L}_{\text{aux}}$	$\mathcal{L}_{\text{geo}}$	Gate	Params	FID (T2M)↓	FID (MARDM)↓	R@1↑	R@2↑	R@3↑	MM-Dist↓	Div→
				26.2M	0.214±.012	0.300±.016	0.545±.004	0.744±.006	0.838±.004	2.769±.030	10.05±.26
✓				51.5M	0.188±.008	0.217±.007	**0.555**±.003	0.745±.002	0.837±.003	2.739±.008	10.06±.08
✓	✓			51.5M	0.180±.005	0.186±.005	0.550±.005	0.746±.004	0.837±.003	2.743±.007	9.90±.15
✓	✓	✓		51.5M	0.152±.008	0.152±.006	0.543±.008	**0.747**±.009	**0.841**±.006	**2.731**±.020	9.94±.14
✓	✓	✓	✓	61.0M	**0.064**±.003	**0.064**±.002	0.547±.003	0.743±.002	0.836±.002	2.743±.007	**9.58**±.06
Parameter-matched single-stream baselines											
Base, 2× width				51.6M	0.302±.005	0.485±.014	0.520±.005	0.719±.004	0.819±.006	2.928±.021	10.05±.10
Base, 2× depth				51.5M	0.296±.012	0.440±.020	0.527±.004	0.722±.005	0.817±.002	2.909±.015	10.08±.22

Table F.3: FID at 300k / 400k / 500k for the HumanML3D configurations of the main text and this section, except the variance-family $K{=}64$ model trained with $\mathcal{L}_{\text{simple}}$ only and the two per-layer-router gated configurations that appear only in Tab. F.4 (both evaluators; mean$_{\pm 95\%\ \text{CI}}$, full test set, 5 repetitions). Bold: the lowest FID of the three checkpoints for each evaluator; the main-text tables report the bold cells. The 550k / 600k results of the six gated configurations trained further are in Tab. F.4; where the reported value is at 550k or 600k, it is not in this table, and no cell is bold. * 300k and 400k evaluated on a different machine (cross-machine differences in Appendix B.5). † TF32 run; its reported checkpoint, 550k, is in Tab. F.4. Repeated runs of the gated configurations are in Tab. B.1, Tab. B.2, and Tab. F.1.

	T2M evaluator			MARDM evaluator		
Configuration	300k	400k	500k	300k	400k	500k
Base (67-d)	0.224±.010	0.216±.017	**0.214**±.012	0.335±.016	**0.300**±.016	0.308±.016
MDM (263-d, our retrain)	0.468±.014	0.467±.018	**0.412**±.007	0.551±.023	0.546±.009	**0.456**±.009
Aux. loss only	**0.240**±.011	0.246±.009	0.255±.016	0.345±.017	**0.325**±.014	0.339±.017
Concat	0.334±.012	**0.327**±.008	0.358±.008	0.420±.011	**0.419**±.007	0.441±.008
Bidirectional	0.345±.017	**0.303**±.013	0.319±.018	0.353±.002	**0.293**±.008	0.323±.016
Reversed	0.250±.015	**0.228**±.014	0.229±.013	0.359±.008	0.292±.015	**0.266**±.015
Base, 2× width	**0.302**±.005	0.370±.026	0.551±.028	**0.485**±.014	0.644±.025	0.835±.034
Base, 2× depth	0.302±.014	**0.296**±.012	0.305±.027	**0.440**±.020	0.466±.016	0.558±.012
Rule family, $K{=}32$	0.215±.005	0.178±.013	**0.141**±.002	0.298±.009	0.225±.011	**0.197**±.008
Rule family, $K{=}64$	0.232±.017	0.194±.012	**0.182**±.013	0.307±.008	0.243±.006	**0.231**±.005
Variance family, $K{=}32$	0.177±.010	0.162±.010	**0.132**±.004	0.213±.011	0.196±.009	**0.178**±.010
Variance family, $K{=}64^*$	0.284±.015	0.227±.008	**0.152**±.008	0.334±.012	0.268±.009	**0.152**±.006
Variance family, $K{=}128$	0.245±.005	0.228±.005	**0.225**±.004	0.328±.013	0.319±.019	**0.307**±.015
Dense set, $K{=}210$ (non-bone)	0.404±.019	0.386±.009	**0.356**±.013	0.542±.014	0.492±.015	**0.459**±.019
Control: isometric re-encoding	0.347±.018	0.308±.007	**0.272**±.015	0.429±.016	0.371±.020	**0.338**±.015
Control: random virtual-point distances	0.323±.018	**0.258**±.019	0.263±.020	0.367±.007	0.312±.011	**0.279**±.009
rule $K{=}64$, $-\,\mathcal{L}_{\text{geo}}$	0.200±.010	0.184±.008	**0.167**±.006	0.284±.009	0.259±.012	**0.192**±.006
rule $K{=}64$, $-\,\mathcal{L}_{\text{aux}}$	0.208±.007	0.187±.005	**0.178**±.010	0.296±.014	0.268±.004	**0.245**±.005
rule $K{=}64$, $\mathcal{L}_{\text{simple}}$ only	0.261±.014	0.205±.005	**0.187**±.008	0.346±.007	0.282±.012	**0.242**±.009
var. $K{=}32$, $-\,\mathcal{L}_{\text{geo}}$	0.194±.007	0.163±.008	**0.141**±.004	0.244±.013	0.220±.010	**0.198**±.008
var. $K{=}64$, $-\,\mathcal{L}_{\text{geo}}$	0.264±.006	0.201±.009	**0.180**±.005	0.311±.007	0.231±.011	**0.186**±.005
var. $K{=}32$ + read-side gate ($N_e{=}1$)	0.105±.004	**0.080**±.006	0.091±.006	0.109±.004	**0.084**±.005	0.094±.006
AMDD: var. $K{=}64$ + write-side MoE gate†	0.112±.003	0.088±.008	0.069±.003	0.143±.008	0.090±.009	0.084±.009
var. $K{=}32$ + read-side gate ($N_e{=}4$)	0.137±.011	0.101±.008	**0.069**±.006	0.180±.012	0.127±.007	0.112±.007
rule $K{=}64$ + read-side gate ($N_e{=}1$)	0.178±.006	**0.139**±.008	0.142±.007	0.274±.016	0.207±.009	**0.202**±.014
rule $K{=}64$ + write-side gate ($N_e{=}4$)	0.165±.013	0.112±.008	**0.094**±.009	0.186±.015	0.123±.012	0.098±.010

Table F.4: FID at 500k / 550k / 600k of the six gated configurations trained further. Bold: the lowest FID of that evaluator over all evaluated checkpoints of the configuration (including 300k / 400k); no bold: the lowest is at 300k or 400k (Tab. F.3 or the row label). The adjacent-checkpoint variation (iii) in Appendix B.5 is computed from the 12 series in this table.

	T2M evaluator			MARDM evaluator		
Configuration	500k	550k	600k	500k	550k	600k
AMDD: var. K=64 + write-side MoE gate	0.069	**0.067**	0.094	0.084	**0.063**	0.083
var. K=32 + read-side gate (N_e=1; best: 400k, 0.080 / 0.084)	0.091	0.091	0.111	0.094	0.093	0.111
var. K=32 + read-side gate (N_e=4)	**0.069**	0.082	0.070	0.111	0.111	**0.095**
var. K=32 + read-side gate (N_e=4, per-layer router)	0.092	0.092	**0.085**	**0.103**	0.108	0.131
rule K=64 + write-side gate (N_e=4)	**0.094**	0.103	0.100	0.098	0.096	**0.085**
rule K=64 + write-side gate (N_e=4, per-layer router; best T2M: 400k, 0.087)	0.103	0.110	0.100	0.100	0.098	**0.094**

G TEXT GATE AND CLASSIFIER-FREE GUIDANCE

Observation: the write-side gate contracts the generated motion as a whole. At guidance scale 2.5, we compare the gated model (the run used in Tab. 1, 550k) with its gate-free counterpart (variance family, K=64, 500k) on all 6880 rows of the contact manifest (Appendix J; the manifest also contains categories not reported there, such as cross-legged texts). For each of the 64 selected pairs, we take the minimum distance over the clip; the per-pair median of this minimum decreases for all 64 pairs, by 2.7 cm on average for pairs that contain a wrist and by 1.4 cm for the others. The wrists also come closer on texts unrelated to the hands, e.g., the median wrist–wrist minimum drops from 40.7 to 31.0 cm on kneeling texts and from 53.1 to 37.7 cm on cross-legged texts. The same holds for the other two runs and for the write-side gate on the rule-based family, whereas the two read-side gates change the distances by only -0.3 to -0.4 cm. This contraction accounts for both costs of the gate in Tab. 5: on texts without contact, the hands come closer together; on texts that require a hand to touch the head, the hands draw in toward each other and the torso, and the wrist–head distance increases instead.

Table G.1: Change of the gated model relative to the gate-free model under inference switches. Texts of four categories (No contact, Kneeling, Cross-legged, and Head; 2210 rows in total); the two models use the same noise. The three value columns give the mean change of the minimum distance of the 64 pairs, the mean change of the minimum wrist–wrist distance, and the threshold success rate on Head; *: $p<0.05$ (paired cluster bootstrap). This probe is a separate generation (different seed from Tab. G.3), so its success rates differ slightly from those in Tab. G.3 and Tab. J.3.

Inference setting	64 pairs (cm)	Wrist–wrist: No contact / Kneeling / Cross-legged (cm)	Head: gate-free → gated (%)
Guidance, s=2.5	-1.96^*	-3.9^* / -7.4^* / -12.8^*	86.3 → 74.8
No guidance (s=1)	-0.63^*	-0.8 / $+0.8$ / -0.3	81.7 → 84.3
Per-stream: s_m=2.5, s_r=1	-1.00^*	-1.7^* / -0.7 / -1.8	84.1 → 85.0
s=2.5, gate forced to $\mathbf{1}$	-9.84^*	-9.2^* / -10.6^* / -10.0^*	86.3 → 50.2

Table G.2: FID under different guidance settings (T2M / MARDM; full test set, 5 repetitions). The two models in each row were evaluated in the same run on the same machine; the 0.152 / 0.152 of the gate-free configuration in Tab. 3 and Tab. 4 is another evaluation of the same checkpoint on a different machine (Appendix B.5, (i)).

Inference setting	Gated (run of Tab. 1)	Gate-free (variance family, K=64)
Guidance, s=2.5	0.065±.004 / 0.064±.005	0.161±.006 / 0.169±.007
Per-stream: s_m=2.5, s_r=1	0.171±.008 / 0.166±.008	0.216±.015 / 0.178±.004
No guidance (s=1)	0.267±.010 / 0.271±.010	0.281±.022 / 0.237±.007

Other metrics without guidance. Without guidance, the other two gated runs (500k) reach an FID of 0.272 / 0.253 and 0.315 / 0.306, compared with about 0.10 / 0.10 at guidance scale 2.5 (Tab. B.2). Without guidance, the T2M FIDs of the three runs (0.267, 0.272, 0.315) have confidence intervals that overlap that of the gate-free configuration ($0.281 \pm .022$), and their MARDM FIDs (0.271, 0.253, 0.306) are all higher than that of the gate-free configuration ($0.237 \pm .007$); none of the three

runs therefore improves FID over the gate-free configuration (Tab. G.2). In the same evaluation, the gate lowers Diversity from 9.83 to 9.60 at guidance scale 2.5 (T2M evaluator; real data: 9.50), whereas without guidance the two are comparable (gated 9.83 vs. gate-free 9.77), consistent with the contraction appearing mainly under guidance. On the other hand, without guidance the gated model reaches a higher R@1 than the gate-free configuration (T2M 0.496–0.502 vs 0.473; MARDM 0.491–0.492 vs 0.463), whereas at guidance scale 2.5 the two are comparable (0.546 vs 0.548).

Interpretation. We write the gate value after a semicolon in the arguments of the denoiser. At a uniform scale s, the output of Eq. (8) is then

$$\hat{x}_0 = G_\theta(x_t, t, \varnothing;\, \mathbf{1}) + s\,\Delta, \qquad \Delta = G_\theta\big(x_t, t, c;\, g(c)\big) - G_\theta\big(x_t, t, \varnothing;\, \mathbf{1}\big). \tag{11}$$

where $g(c) = (g_\ell)_\ell$ collects the gate values of all layers for text c (Eq. (9)). In the unconditional branch, the gate input is zero and the gate is identically $\mathbf{1}$; in the conditional branch, most gate values are below 1 (87%–97% on HumanML3D). Besides the effect of the text, Δ therefore contains a component caused by the different gate values of the two branches; this component has a similar direction for almost all texts and is amplified s times under guidance, whereas at $s{=}1$ it appears only once, as in training. Three results support this interpretation (Tab. G.1 and Tab. G.2): (i) without guidance, the contraction and the FID advantage of the gate largely disappear; (ii) removing the amplification only on the distance channels ($s_r{=}1$) halves the contraction, restores the success rate on Head, and shrinks the FID advantage to -21% / -7%; (iii) forcing the gate to $\mathbf{1}$ in the conditional branch, a combination never seen in training, amplifies the contraction about fivefold. The direction of this component (contraction) is a measured result; it cannot be derived from Eq. (11). The gate-free model is still slightly over-dispersed (Diversity 9.83 vs. 9.50 for real data; Appendix E), and the contraction brings it close to real data (9.60), which matches the FID gain of the gate. The read-side gate leaves the distance stream unchanged, so it creates no difference between the distance states of the two branches, consistent with its almost absent contraction; the gate-free distance stream does not involve this mechanism, and on the same samples its FID on HumanML3D is lower than that of Base at every guidance scale in Fig. E.1.

Replication of the contact effects across three runs. Tab. G.3 compares the three gated runs with the gate-free configuration by the threshold success rate (Appendix J).

Table G.3: Contact effects of the text gate in three independent runs (500k, the same number of steps as the gate-free configuration); threshold success rate (%). The three runs share the configuration and differ only in numerical precision and machine. Parentheses: paired difference from the gate-free configuration (variance family, $K{=}64$); *: $p{<}0.05$.

Category	Base	Gate-free, $K{=}64$	Run of Tab. 1	Repeated run A	Repeated run B
Hand	48.2	60.1	70.3 ($+10.2^*$)	69.2 ($+9.2^*$)	69.6 ($+9.5^*$)
Head	87.5	89.4	81.5 (-7.9)	84.5 (-4.9)	78.7 (-10.7^*)
Hands on hips	87.2	84.2	96.0 ($+11.8$)	90.0 ($+5.8$)	95.2 ($+11.0^*$)
Hand on ground (control)	90.2	72.6	87.0 ($+14.3^*$)	82.6 ($+9.9^*$)	88.7 ($+16.1^*$)
False-contact rate	8.4	7.0	10.1 ($+3.1$)	8.9 ($+1.9$)	8.4 ($+1.4$)

In all three runs, the text gate raises the success rate on Hand by 9–10 pp, significantly in each, and the three runs differ by only 1.1 pp. The success rate on Head decreases in all three runs (by 4.9–10.7 pp; significant in one, $p \approx 0.06$ in the other two), so we regard it as a cost of the text gate rather than random variation across training runs. The false-contact rate rises by 1.4–3.1 pp over the gate-free configuration, but two of the runs are on par with Base (8.9% / 8.4% vs 8.4%). The gate also brings the success rate on Hand on ground back close to that of Base.

H ABLATIONS ON SNAPMOGEN

This section reports the coupling, loss, gate, and relation-variable ablations on SnapMoGen, and a comparison of guidance scales. Unless stated otherwise, results use a uniform guidance scale $s{=}2.5$ and 5 repetitions; Base denotes MDM (BERT-Dec) on the 148-d representation (the MDM (BERT-Dec) row of Tab. 2). The AMDD of Tab. 2 is the model of the highlighted row of Tab. H.1 (rule-based family, $K{=}64$, write-side gate, 300k).

Per-stream guidance. Per-stream guidance lowers FID and raises Matching in all 20 evaluations (Tab. H.3); TOP-1 rises only slightly (+0.40 pp, higher in 14 of 20), and Diversity moves slightly farther from that of real data. This comparison changes only the sampling setting; the weights are the same.

Guidance-scale sweep. Tab. H.4 compares the two models at the same checkpoint (300k; Base in Tab. 2 is at 500k, where we did not sweep). At every swept setting, AMDD has a lower FID than Base with the same motion-stream scale: by 1.7–2.9 under uniform scales and by 1.0–3.9 under per-stream guidance; their respective minima differ by 3.2 (16.59 vs. 19.75). TOP-1, in contrast, depends on the guidance scale. At $s{=}2.5$, AMDD is 2.5 pp higher (0.811 vs. 0.786), but Base gains about 2 pp as the scale increases (0.806 at $s{=}4.0$–4.5), whereas AMDD gains at most 0.2 pp; at the same motion-stream scale of 4.5, the two are on par (0.808 vs. 0.806), and their respective maxima differ by 0.6 pp (0.812 vs. 0.806). Hence, a substantial part of the TOP-1 gain in Tab. 2 comes from the different guidance scales of the two models, whereas the FID reduction holds at every guidance setting swept.

Text gate. The AMDD of Tab. 2 uses the write-side gate, as on HumanML3D. At the same checkpoint and guidance, compared with its gate-free counterpart (rule-based family, $K{=}64$, 300k, 20 repetitions), AMDD has a lower FID ($s{=}2.5$: 17.67 vs. 18.02; $s_m/s_r{=}4.5/2.5$: 16.56 vs. 17.30) but also a lower TOP-1 (0.801 vs. 0.815; 0.805 vs. 0.828). Pairing each of the four write-side gate configurations (both families at $K{=}64$, each with a shared router and with per-layer routers) with its gate-free counterpart at the same steps and averaging over steps, all four lower TOP-1 by 0.9–1.2 pp, whereas only AMDD lowers FID (-0.36; the other three $+0.17$ to $+0.25$); the read-side gate does not lower TOP-1 (Tab. H.1 and Tab. H.2). Unlike on HumanML3D (Appendix G), the write-side gate barely changes Diversity here (19.93 vs. 19.94 at $s{=}2.5$), so the contraction seen there does not occur. The cause of the TOP-1 loss remains open: the gate varies less across descriptions than on HumanML3D (standard deviation 0.13–0.15 in layers 2–7 vs. 0.24), but replacing its mean-pooled input with attention pooling restores this variation (0.21 at 100k) and lowers TOP-1 even more (2.0–2.5 pp below the gate-free configurations at 300k, vs. 1.3–1.6 pp with mean pooling).

Table H.1: Same-backbone comparison on SnapMoGen (official evaluator, s=2.5, 4000 samples $\times$ 5 repetitions). Unless marked otherwise, two-stream configurations use the variance family with K=32 and are reported at 300k. TOP-k is 100-way cosine retrieval; Matching is cosine similarity. Both write-side gates are MoE gates with N_e=4 and differ in routing (per-layer routers or one shared router). In the position-stream control, the second stream generates 72-d joint positions instead of distances, all else being equal; the italic row is Base at the end of training. * The SnapMoGen counterpart of the HumanML3D AMDD configuration (variance family, K=64, write-side MoE gate, per-layer routing); it is worse than its gate-free counterpart (variance family, K=64: 18.30 / 0.815). The highlighted AMDD row is the model of Tab. 2; its results over 20 repetitions under per-stream and uniform guidance are in Tab. H.3. It differs from the HumanML3D configuration in two ways: the relation variable is the rule-based family, and all layers share one router (Appendix B.2). Its TOP-1 is about 1.3 percentage points (pp) below that of its gate-free counterpart (rule-based family, K=64; 0.815 over 20 repetitions), and the neighboring checkpoints score 18.45 (275k) and 19.04 (325k) (Fig. H.1).

Method	FID↓	TOP-1↑	TOP-2↑	TOP-3↑	Matching↑	Div→
Real	–	0.940	0.975	0.984	0.836	19.74
Base (148-d)	20.19±.10	0.781±.005	0.889±.005	0.926±.004	0.669±.002	19.97
Concat	20.48±.09	0.789±.004	0.897±.003	0.931±.003	0.674±.001	19.92
Bidirectional	21.20±.10	0.788±.004	0.896±.005	0.933±.003	0.677±.002	19.91
One-way, motion → distance	19.55±.11	0.775±.005	0.882±.005	0.923±.004	0.667±.002	20.06
Position stream (72-d, control)	19.01±.14	0.807±.006	0.908±.003	0.942±.002	0.683±.001	19.99
+ distance stream (var., K=32)	19.10±.10	0.814±.006	0.909±.003	0.942±.004	0.687±.002	19.93
+ read-side gate (300k)	18.39±.04	0.820±.004	0.914±.003	0.945±.002	0.691±.001	19.92
+ read-side gate (250k)	17.81±.05	0.811±.003	0.908±.003	0.942±.003	0.688±.001	19.92
Var., K=64 + write-side gate (per-layer)*	18.94±.09	0.801±.001	0.904±.003	0.938±.003	0.682±.001	19.99
AMDD: rule, K=64 + write-side gate (shared)	17.72±.04	0.803±.006	0.901±.002	0.936±.003	0.685±.001	19.90
Base @ 500k (end of training)	*18.82±.12*	*0.786±.008*	*0.894±.003*	*0.930±.003*	*0.674±.002*	*19.98*

Table H.2: Loss, gate, and relation-variable ablations on SnapMoGen (300k). † At 250k; Base at the same step scores 20.80 / 0.770. ° Generated on a different GPU type (RTX 4090) from the other cells (A800). ♦ On SnapMoGen, the dense set is the 253 pairs of the 23 non-degenerate joints minus the near-rigid pairs at most two bones apart on the skeleton; it happens to contain 210 pairs as well, but its definition differs from that on HumanML3D. At K=210, the distance channels take about 59% of the channel weight in the joint denoising loss.

Configuration	FID↓	TOP-1↑	Configuration	FID↓	TOP-1↑
Losses (variance family, K=32, gate-free)			*Relation variables (gate-free)*		
Full	19.10±.10	0.814±.006	Rule family, K=32	18.71±.13	0.817±.004
$-\mathcal{L}_{\text{geo}}$	18.92±.03	0.816±.004	Rule family, K=64°	18.04±.15	0.816±.003
$-\mathcal{L}_{\text{aux}}$	18.57±.15	0.811±.004	Variance family, K=16	18.36±.10	0.814±.005
$\mathcal{L}_{\text{simple}}$ only	18.66±.08	0.811±.006	Variance family, K=32	19.10±.10	0.814±.006
Gates (variance family, K=32)			Variance family, K=64	18.30±.13	0.815±.005
Read-side gate, N_e=1	18.39±.04	0.820±.004	Variance family, K=128†	19.86±.06	0.800±.002
Read-side gate, N_e=4	18.73±.08	0.813±.004	Dense set, K=210♦†	21.61±.15	0.773±.006
Write-side gate, N_e=4	19.18±.17	0.803±.002			
Base	20.19±.10	0.781±.005			

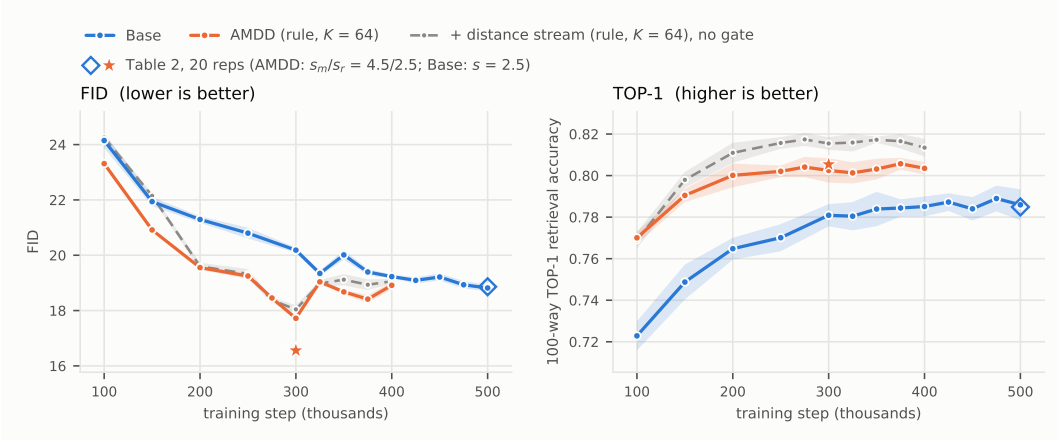

Figure H.1: FID and TOP-1 over training steps on SnapMoGen (uniform guidance scale s=2.5, 5 repetitions). Base, AMDD (rule-based family, K=64, write-side gate), and its gate-free counterpart (rule-based family, K=64; dashed); shaded bands are 95% confidence intervals. The markers are the two points of Tab. 2 (20 repetitions): AMDD at 300k with s_m/s_r=4.5/2.5 (star) and Base at 500k with s=2.5 (diamond). At every checkpoint evaluated for both, AMDD has a lower FID than Base (by 0.3–2.5) and a higher TOP-1 (by 1.8–4.7 pp), with non-overlapping confidence intervals; the TOP-1 lead narrows over training. Compared with its gate-free counterpart, the gate lowers TOP-1 by 0.8–1.5 pp from 150k on, and lowers FID only at some checkpoints. The checkpoint with the lowest FID of AMDD at s=2.5 is 300k (17.72); the neighboring 275k and 325k score 18.45 and 19.04. The cells of the gate-free counterpart from 275k onward were generated on a different type of GPU and are not fully comparable with the other two curves.

Table H.3: Paired comparison of per-stream and uniform guidance on SnapMoGen (AMDD, 300k; official evaluator, 4000 samples × 20 repetitions). The same weights are evaluated 20 times under each sampling setting; Paired diff. gives the mean and 95% CI of the per-repetition paired differences over the 20 repetitions. The better setting is in bold.

Metric	s_m/s_r=4.5/2.5	s=2.5	Paired diff.	Per-stream better (of 20)
FID↓	**16.558**±.071	17.671±.072	−1.113±.028	20/20
TOP-1↑	**0.8054**±.0021	0.8014±.0024	+0.40±.22 pp	14/20
TOP-2↑	**0.9070**±.0012	0.9022±.0014	+0.47±.13 pp	19/20
TOP-3↑	**0.9407**±.0012	0.9373±.0012	+0.34±.10 pp	18/20
Matching↑	**0.6871**±.0005	0.6851±.0006	+0.0020±.0002	20/20
Diversity (real 19.75)	19.989±.031	19.932±.033	+0.057±.009	farther from real

Table H.4: Guidance-scale sweep on SnapMoGen (300k, single repetition, official evaluator, 4000 samples). s is a uniform scale, and s_m/s_r are the separate scales of the two streams; Base has only one stream, so only uniform scales are swept. Each cell is paired with the s=2.5 cell of the same model (same descriptions, crops, noise, and retrieval groups); the FID spread of a single repetition is about 0.17. Best per model and metric in bold.

	Base		AMDD			AMDD	
Uniform	FID↓	TOP-1↑	FID↓	TOP-1↑	Per-stream	FID↓	TOP-1↑
s=2.0	20.80	0.768	19.06	0.794	s_m/s_r=2.5/1.5	19.07	0.785
s=2.5	20.02	0.786	17.73	0.811	s_m/s_r=2.5/3.5	17.38	0.810
s=3.0	**19.75**	0.800	17.02	**0.812**	s_m/s_r=3.5/2.5	16.90	**0.812**
s=3.5	19.78	0.803	16.88	0.809	s_m/s_r=4.5/1.5	17.40	0.797
s=4.0	19.86	**0.806**	17.21	0.802	s_m/s_r=4.5/2.5	**16.59**	0.808
s=4.5	19.99	**0.806**	–	–	s_m/s_r=4.5/3.5	16.84	0.805
s=5.5	20.40	0.801	–	–	s_m/s_r=5.5/2.5	16.68	0.808
s=6.5	21.04	0.795	–	–	s_m/s_r=5.5/3.5	17.20	0.801
					s_m/s_r=6.5/2.5	17.11	0.806

I Relational Error during Denoising

Setup (Fig. 3). We compare Base with gate-free AMDD (variance family, $K{=}64$), both with EMA weights, at the 500k checkpoint on HumanML3D and the 300k checkpoint on SnapMoGen. The measurement uses the full test set of each dataset (4640 clips on HumanML3D, 4000 clips on SnapMoGen). For each test clip, we noise it with fixed noise at $t \in \{10, 25, 40, 45, 49\}$ (of 50 steps) to obtain x_t, feed x_t to the model together with the clip's description, and read the one-step prediction $\hat{x}_0$, without iterative sampling. The two models receive the same clip crops, descriptions, and noise, and are paired clip by clip. No classifier-free guidance is used ($s{=}1$), so we measure the model's conditional prediction itself, unaffected by the amplification of guidance (see Appendix G). Both errors are computed on joint positions in the root frame, excluding the global trajectory, and averaged over valid frames. The position error is the mean per-joint Euclidean error. The inter-joint distance error is the mean absolute distance error over all joint pairs: 231 pairs of the 22 joints on HumanML3D and 276 pairs of the 24 joints on SnapMoGen. The distance stream generates only 64 of these pairs, so the two models are scored on the same pairs, most of which the distance stream never generates (167 pairs, 72%, on HumanML3D). On SnapMoGen, joint positions are obtained by forward kinematics and converted to the HumanML3D body scale by the ratio of the leg lengths of the two datasets, so that centimeters on both datasets correspond to the same body proportions; the position error of Base grows with t from 1.5 cm to 12.0 cm on HumanML3D and from 1.7 cm to 13.2 cm on SnapMoGen. Fig. 3 reports the relative reduction $(E_{\text{Base}} - E_{\text{AMDD}})/E_{\text{Base}}$, and differences are tested with a Wilcoxon signed-rank test paired over clips. At $t \leq 45$, both errors are reduced significantly on both datasets ($p < 10^{-16}$). At $t{=}49$, AMDD still reduces the distance error by 6.4% and the position error by 0.5% on HumanML3D; on SnapMoGen, the two reductions are -1.0% and -1.5% (both $p < 10^{-4}$), that is, AMDD is slightly worse. This measurement is a one-step prediction under teacher forcing: it describes the model's denoising error on noised real clips and does not directly correspond to free sampling; joint relations in freely generated motions are tested by the contact evaluation (Sec. 4.4, Appendix J).

Per-joint improvement and leverage. The conclusion of Sec. 4.4 is averaged over all joints: AMDD improves the inter-joint distance error more than the position error. If this improvement indeed comes from the distance stream, joints on which the selected distances depend more should improve more. We measure this dependence by the Frobenius norm of the Jacobian of the selected distances with respect to the coordinates of joint j, $\|\partial \mathcal{D}_{\mathcal{P}}/\partial p_j\|_F$, and call it the leverage of joint j. It depends only on the selected pair set, not on the model: joints that appear in several selected pairs (such as the wrists) have larger leverage, and joints that affect no selected distance have zero leverage. On SnapMoGen, joints are represented by rotations, and the leverage (with respect to the joint rotation r_j) is computed by automatic differentiation through forward kinematics.

The per-joint analysis uses the same paired protocol and checkpoints as Fig. 3, but only a subset of clips and the three lower noise levels: 400 equally spaced test clips on HumanML3D, the first 400 test clips on SnapMoGen, and $t \in \{10, 25, 40\}$; the compared configurations are the two gate-free configurations with $K{=}64$, one per family. The error of each joint is first averaged over the three timesteps, and its reduction relative to Base is then computed: for the position error on HumanML3D, and for the geodesic angle between the predicted and ground-truth rotations on SnapMoGen. We compare the reductions with the leverage (Fig. I.1). On both datasets, the per-joint error reduction correlates positively with the leverage: Spearman ρ is $+0.88$ / $+0.90$ (rule-based / variance family) on HumanML3D and $+0.66$ / $+0.78$ on SnapMoGen. On SnapMoGen, the five end joints with zero leverage barely improve (on average $+0.15\%$ / -0.37% for the rule-based / variance family), whereas the five joints with the largest leverage improve by 7.5% / 8.6% on average.

This correlation should be interpreted with caution. First, on HumanML3D, the leverage is ordered like the error of Base itself; the partial correlation controlling for the Base error remains $+0.58$ for the rule-based family but drops to $+0.18$ for the variance family. Second, the only joint with zero leverage on HumanML3D still improves (6.0% / 2.5% for the rule-based / variance family), showing that the shared network also benefits joints not constrained by the selected distances. This result therefore indicates that the improvement tends to concentrate on the joints that the distance stream constrains, but it is an association, not an exact attribution.

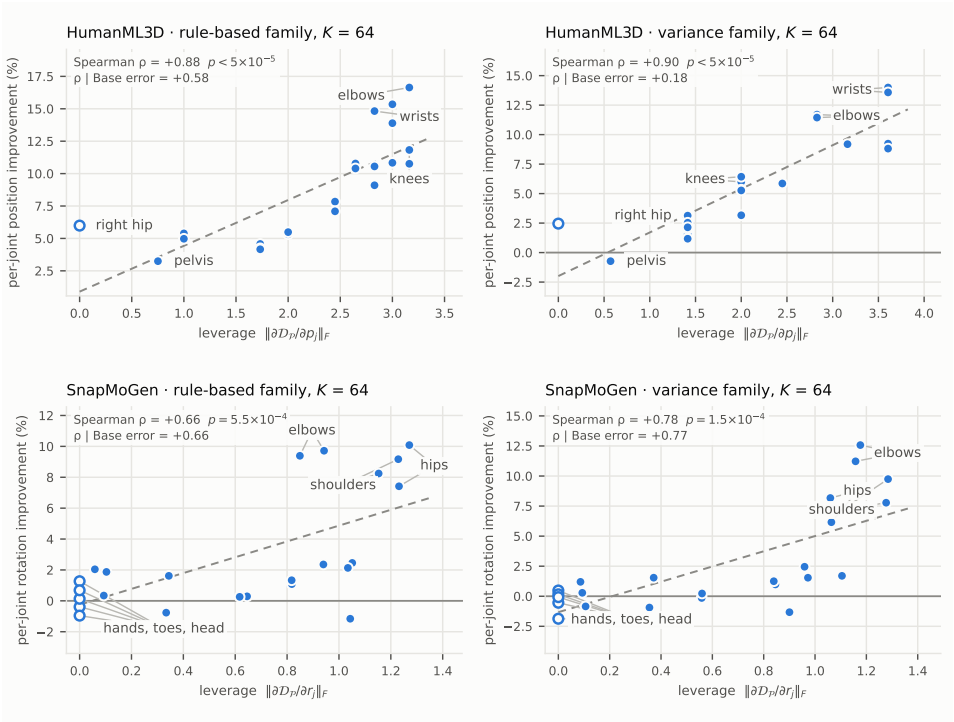

Figure I.1: **Per-joint error reduction versus leverage.** Left: rule-based family, $K{=}64$; right: variance family, $K{=}64$ (both gate-free AMDD). Top: HumanML3D (position error); bottom: SnapMoGen (rotation error). Hollow markers denote joints with zero leverage; ρ is the Spearman correlation.

J CONTACT EVALUATION

Protocol. The contact manifest is fixed before the experiments: texts are selected from the HumanML3D test set with pre-written regular expressions, and each category specifies the joint pair (or the height of a joint above the ground) to check and a threshold (Tab. J.1); the criterion is met when this quantity stays within the threshold for at least 3 consecutive frames. All models generate from the same manifest. Mirrored versions and different texts of the same motion form one cluster, the unit of pairing and bootstrap. The Real row applies the same criterion to the motions of the dataset itself. The false-contact category consists of texts that match no contact category; since the exclusion rules are written as keywords, a few variants with a contact meaning may still slip in, and the 5.0% of the Real row reflects this residue.

Table J.1: Categories of the contact manifest.

Category	Criterion	Threshold	Texts	Clusters	Samples per text
Hand	Wrist–wrist distance	0.15 m	282	89	10
Head	Wrist–head distance	0.25 m	92	33	5
Hands on hips	Wrist–hip distance	0.25 m	58	20	5
Hand on ground (control)	Hand height above the ground	0.10 m	182	43	5
Kneeling (control)	Knee height above the ground	0.10 m	144	49	5
False-contact rate	Wrist–wrist distance	0.15 m	400	400	2

Table J.2: Gap per category (cm, clusters weighted equally). Parentheses: paired difference from Base where computed; *: $p<0.05$. No contact uses the texts of the false-contact category.

Category	Clusters	MDM (263-d)	MoMask	SALAD	Base	+ distance stream (var., $K=32$)	+ distance stream (var., $K=64$)	AMDD
Hand	89	3.0	4.7	4.7	4.1	2.1 (-2.0)	1.8 (-2.3^*)	2.3 (-1.8)
No contact	400	1.6	3.7 ($+2.4^*$)	2.6 ($+1.3^*$)	1.3	2.1 ($+0.8$)	1.9 ($+0.6$)	4.3 ($+3.1^*$)
Head	33	1.7	1.6	3.3 ($+1.6^*$)	1.7	3.9 ($+2.2^*$)	2.3 ($+0.6$)	2.8 ($+1.1$)
Hand on ground (control)	43	5.7	7.8	18.7 ($+11.7^*$)	7.0	4.2 (-2.8^*)	3.7 (-3.3^*)	4.6 (-2.4^*)

Gap (main readout of Tab. 5). From the same generations, we take for each row the minimum over the clip of the distance (or height) the category checks and compute the Wasserstein-1 distance (cm) between generated and real motions (Tab. J.2); clusters are weighted equally (split evenly over texts and samples, as for the success rates), with 2000 cluster-bootstrap resamples shared by all models. We prefer the gap to the success rate, which depends on the threshold and is not bounded by real motions: SALAD exceeds real motions on Hand (70.0% vs. 60.7%; Tab. J.4) because it brings the hands too close overall, and on hand-on-ground texts its hands pass through the ground (median minimum hand height -14.0 cm vs. $+3.6$ cm for real motions).

On Hand, $K=64$ also has a smaller gap than 263-d MDM, MoMask, and SALAD (by 1.2, 2.9, and 2.9 cm; all $p<0.05$). $K=32$ and AMDD improve in the same direction but not significantly against Base; they become significant when texts rather than clusters are weighted equally, so their gains concentrate in clusters with more texts, and the main text draws the conclusion only for $K=64$. $K=32$ is significantly worse on Head. On Hand on ground, the distance stream moves the gap toward real motions ($K=64$ succeeds on 72.6% vs. 72.1% for real motions, whereas Base overshoots at 90.2%), so its effect is not limited to the selected pairs; specificity is supported instead by the texts without contact, on which AMDD widens the gap by bringing the hands too close (generated minus real: -3.9 cm; Appendix G). Hands on hips (20 clusters) and Kneeling are reported only as success rates (Tab. J.3).

On Hands on hips (Tab. J.3), Base already exceeds real data, and the category has only 20 clusters, so we draw no conclusion from it. The gate-free configuration of the rule-based family ($K=64$) scores $+2.0$ percentage points over Base on Hand, which is not significant; the gain of the distance stream on Hand is thus significant at both sizes of the variance family and weaker for the rule-based family.

Generation with MoMask and SALAD. Both models generate in their own codebases, with their official HumanML3D weights and test settings, from the same manifest (same texts, same ground-truth lengths, and the same number of samples per text): MoMask with a condition scale of 4, 10

Table J.3: Threshold success rates for all categories (%, averaged over clusters). Parentheses: paired difference from Base where computed; *: $p<0.05$. AMDD uses the 550k checkpoint, as in Tab. 1 and Tab. 5; the 500k checkpoint matches the training steps of the gate-free configurations, and its readings agree with those at 550k within noise.

Category	Real	MDM (263-d)	Base	+ distance stream (var., $K=32$)	+ distance stream (var., $K=64$)	AMDD @500k	AMDD @550k
Hand	60.7	60.2 ($+11.9^*$)	48.2	67.2 ($+19.0^*$)	60.1 ($+11.8^*$)	70.3 ($+22.1^*$)	67.7 ($+19.4^*$)
Hands on hips	80.0	81.1 (-6.2)	87.2	92.0 ($+4.8$)	84.2 (-3.0)	96.0 ($+8.8^*$)	95.8 ($+8.5$)
Head	84.8	84.4	87.5	89.6	89.4	81.5 (-6.0)	78.0 (-9.5^*)
Hand on ground (control)	72.1	90.4	90.2	88.6 (-1.6)	72.6 (-17.6^*)	87.0 (-3.2)	86.4 (-3.8)
Kneeling (control)	69.4	65.1	65.5	61.2	61.9	57.8 (-7.7)	61.8
False-contact rate	5.0	7.5	8.4	8.1	7.0	10.1	9.1

Table J.4: Threshold success rates of MoMask and SALAD (%). Criteria as in Tab. J.3.

Category	Real	MoMask	SALAD
Hand	60.7	52.7	70.0
Head	84.8	81.5	70.0
Hands on hips	80.0	79.5	51.5
Hand on ground (control)	72.1	91.0	88.9
False-contact rate	5.0	11.5	8.1

iterations, and 5 residual layers; SALAD with 50 DDIM steps and a guidance scale of 7.5. We take the first 67 dimensions of their 263-d outputs and score them with the same code as the MDM configurations; the real reference is identical, row by row, to that of the MDM configurations. To verify the generation settings, we re-run their official test pipelines: MoMask obtains an FID / R@1 of 0.044 / 0.525 (reported: 0.045 / 0.521), and SALAD 0.074 / 0.579 (reported: 0.076 / 0.581).

K More Qualitative Results

HumanML3D. Beyond Fig. 4, Fig. K.1 adds a description that involves both arms and both legs: "A person takes a large step to the left while stretching both arms straight out to the sides." AMDD keeps both arms stretched out to the sides for about 2.8 s (wrist–wrist distance above 1.2 m), and its ankles are at most 0.80 m apart during the step. SALAD also achieves both. Base takes a large step but does not raise its arms, and MDM and MoMask keep their arms stretched out for only 0.2 s and 0.7 s, respectively.

SnapMoGen. Fig. K.2 shows four examples on SnapMoGen, compared with MoMask++ and Base. SnapMoGen descriptions contain several actions in sequence rather than a single joint relation, so the curves of each row are fixed in advance by the key phrases of that row, and all four rows are plotted. (a) AMDD extends the left hand first and then the right, in the order of the description; Base barely extends its hands. (b) MoMask++ does not kneel; Base and AMDD both put the left knee on the ground and then stand up. (c) Base and AMDD row while seated throughout, whereas MoMask++ stands up midway. (d) None of the three methods performs the small jump in the description; only AMDD raises both hands above its head at the end. Each description has a single sample, so this figure is illustrative only.



Figure K.1: **Qualitative comparison across methods on HumanML3D.** Rendered as in Fig. 4. Below: wrist–wrist (blue) and ankle–ankle (orange) distances; the upper-right corner gives the net pelvis displacement along the initial left–right direction; in the strip, the pelvis of each frame is moved back to its original position, so the direction of displacement cannot be seen there. Meshes are SMPL fits to the generated joints, for display only.

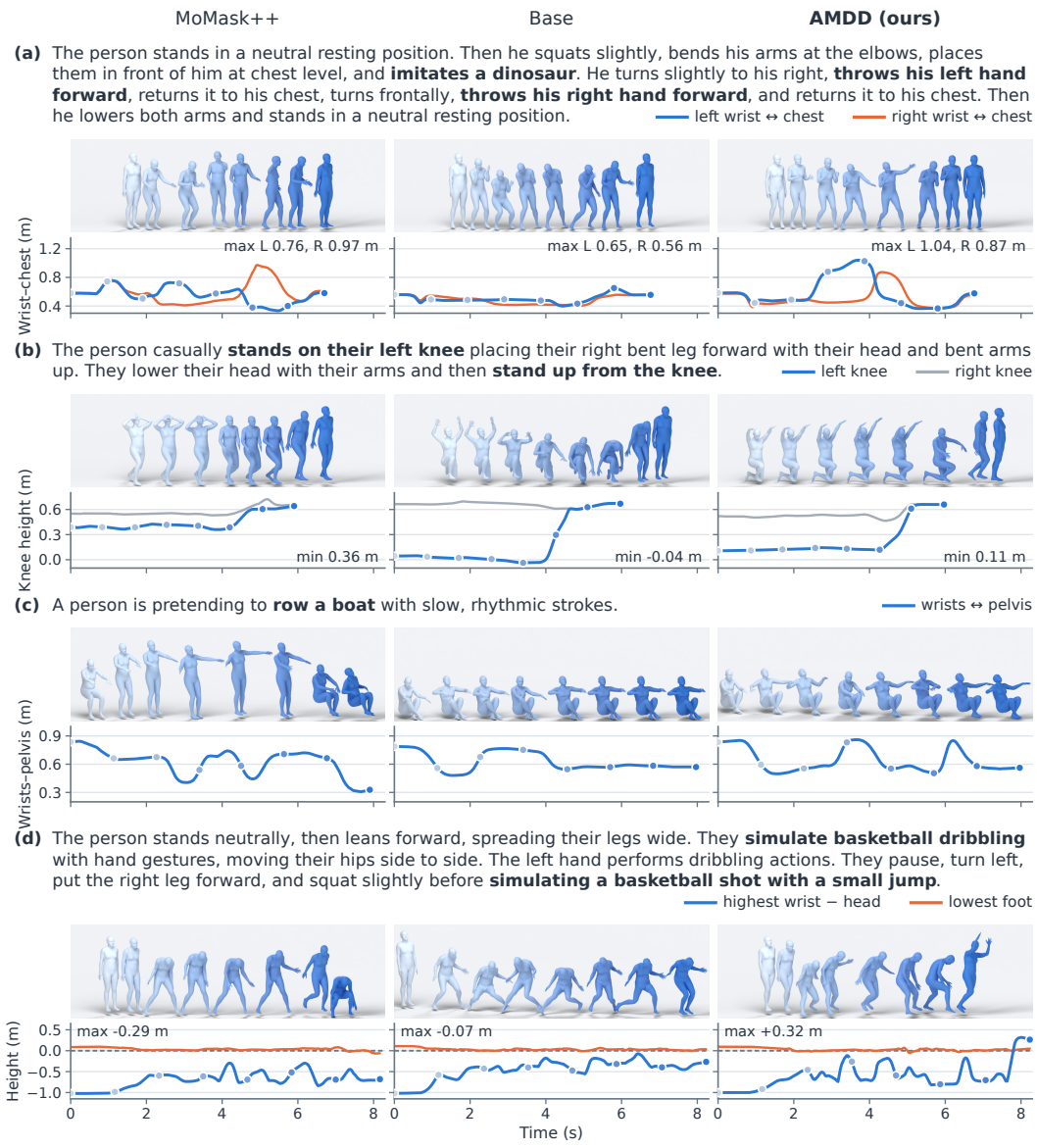

Figure K.2: **Qualitative comparison across methods on SnapMoGen.** Each row is one test description (printed in full, key phrases in bold); each cell shows 8 equally spaced frames over the whole generated clip, light to dark in time, with the same rule for all methods; root displacement is not shown. The curves below are computed from the generated joints: **(a)** left wrist–chest (blue) and right wrist–chest (orange) distances; **(b)** heights above the ground of the left (blue) and right (gray) knees; **(c)** distance from the midpoint of the two wrists to the pelvis; **(d)** height of the highest wrist minus that of the head (blue; the dashed line marks the head height) and height above the ground of the lowest foot joint (orange). AMDD and Base are sampled with a uniform guidance scale s=2.5 (AMDD in Tab. 2 uses s_m=4.5, s_r=2.5), and MoMask++ with its official default settings. Meshes are fits after retargeting the generated joints to SMPL proportions, for display only.