PLAN, DON'T POSE: LONG COMPOSITE MOTION GENERATION WITH TEXT-ALIGNED BFM

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-motion (T2M) generation has broad applications in character animation, virtual avatars, and human-robot interaction. Existing methods typically generate pose trajectories or motion tokens directly from language, forcing a single model to handle semantic interpretation, long-horizon structure, and low-level physical realization. This coupling makes them costly and often unreliable for long, compositional, or semantically dense prompts. We propose Text2BFM, a framework that performs T2M generation directly in the latent policy space of a frozen, pretrained Behavioral Foundation Model (BFM), trained only on paired text–motion data and without video generation or heavy end-to-end motion generators. A text-aligned variational behavioral bottleneck compresses BFM policy-latent sequences into compact continuous latents that are compatible with language and preserve long-horizon behavioral structure. A lightweight conditional flow model generates in this compact space, and the result is decoded into policy latents that drive the frozen BFM in a single closed-loop rollout. Under identical clause boundaries for all methods, Text2BFM raises compositional order accuracy from 0.42 (best kinematic baseline) to 0.67 and reduces boundary discontinuity by 60%.



A person does a cartwheel, a spin, then two twirls,
and finishes with a low jump landing using ballet movements.



A person runs forward, then turns left, does a quick, sharp punch, and
follows it with a kick to the right. After that, he walks forward,
raises his arms in victory, and finishes by waving happily.

Figure 1: Two rollouts of Text2BFM following multi-stage prompts.

1 INTRODUCTION

Natural language describes human motion at the level of intentions, phases, and behaviors: a person walks toward an object, reaches for it, picks it up, and then performs another action. Most text-to-motion methods, however, generate pose trajectories or discrete motion tokens directly from language embeddings (Guo et al., 2022a; Petrovich et al., 2022; Zhang et al., 2024; Tevet et al., 2023; Chen et al., 2023; Zhang et al., 2023a; Guo et al., 2024). This requires a single generative model to simultaneously solve semantic interpretation, long-horizon temporal organization, and low-level physical realization. As a result, visually plausible motions can still exhibit foot sliding, unstable

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.

contacts, unnatural transitions, or poor consistency under compositional prompts (Rempe et al., 2021; Yuan et al., 2023; Tessler et al., 2023). We argue that this stems from a representational mismatch: language specifies behavior, while most text-to-motion models synthesize positions.

A more natural interface is to expose motion to the text model as executable behavior. *Behavioral Foundation Models* (BFMs) provide such an interface by learning structured spaces of reusable behaviors from offline data (Tirinzoni et al., 2025; Sikchi et al., 2025; Li et al., 2026; Bobrin et al., 2026). In particular, Forward-Backward representations (Blier et al., 2021; Touati & Ollivier, 2021) enable promptable behavior inference through policy latents. A single global policy latent can represent short behaviors such as walking forward or turning, but long text prompts typically require temporally localized phases and transitions (Petrovich et al., 2022; Athanasiou et al., 2022; Barquero et al., 2024; Fan et al., 2025). Representing a motion as a trajectory of local policy latents $z_{1:T}$ is therefore more expressive, but directly generating such trajectories creates another long, weakly text-aligned sequence-modeling problem.

We address this problem with a text-aligned variational behavioral bottleneck. Given a trajectory of Forward-Backward policy latents $z_{1:T}$, the bottleneck compresses it into a shorter continuous latent sequence that reconstructs executable local policy latents while being aligned with textual descriptions. Unlike a purely reconstructive autoencoder (Kingma & Welling, 2014; Rezende et al., 2014), the bottleneck is additionally regularized by a motion–text contrastive objective, following cross-modal representation learning (Radford et al., 2021; Tevet et al., 2022; Petrovich et al., 2023). A conditional flow-matching generator (Lipman et al., 2023; Liu et al., 2023; Albergo & Vanden-Eijnden, 2023) samples in this compact space, and the decoded policy latents are executed by the frozen BFM. The generator is therefore responsible for semantics and temporal structure, while physical realization is delegated to the executor.

The contribution of this work is the *generation interface*, not new building blocks: we deliberately use standard components (a VAE, flow matching, contrastive alignment) so that measured effects can be attributed to the change of generation space. Our contributions are:

- We formulate text-to-motion generation as generation in the policy-latent space of a frozen BFM, and introduce a text-aligned variational bottleneck that makes this space compact and language-discriminative.
- We show that composing clauses as consecutive latent programs inside one continuous BFM rollout yields markedly better stage ordering and smoother transitions than kinematic baselines given the *same* clause boundaries.
- We quantify the overlap between the BFM pretraining data and the benchmarks, evaluate on test motions unseen by the BFM, and isolate the role of the prior with a BFM-free variant of the same bottleneck and generator; we also characterize the distributional (FID) shift that the frozen BFM imposes.

2 Related Work

Zero-shot RL and behavioral foundation models. Successor representations and successor features (Dayan, 1993; Barreto et al., 2017) decompose value functions into dynamics- and reward-dependent components. Forward-Backward (FB) representations learn low-rank successor-measure factorizations whose latent codes index executable behaviors (Touati & Ollivier, 2021; Blier et al., 2021; Touati et al., 2023; Agarwal et al., 2025); Meta Motivo and BFM-Zero scale this to humanoid control (Tirinzoni et al., 2025; Li et al., 2026). Standard FB policies are prompted with a single latent or a short-horizon command, which is too limited for multi-phase language prompts; we instead generate sequences of local policy latents through a compact text-aligned bottleneck. RL Zero (Sikchi et al., 2025) also maps language to BFM behaviors, but does so without supervision by imagining the behavior with video generation models, which makes it expensive at inference time.

Physics-based and language-conditioned character control. AMP and ASE learn motion priors and reusable skill embeddings through adversarial imitation (Peng et al., 2021; 2022), CALM learns directable latent controllers (Tessler et al., 2023), and PHC tracks motions robustly (Luo et al., 2023). PULSE distills such an imitator into a low-dimensional latent space that can be sampled for downstream generation (Luo et al., 2024), and MoConVQ learns a discrete latent space of a

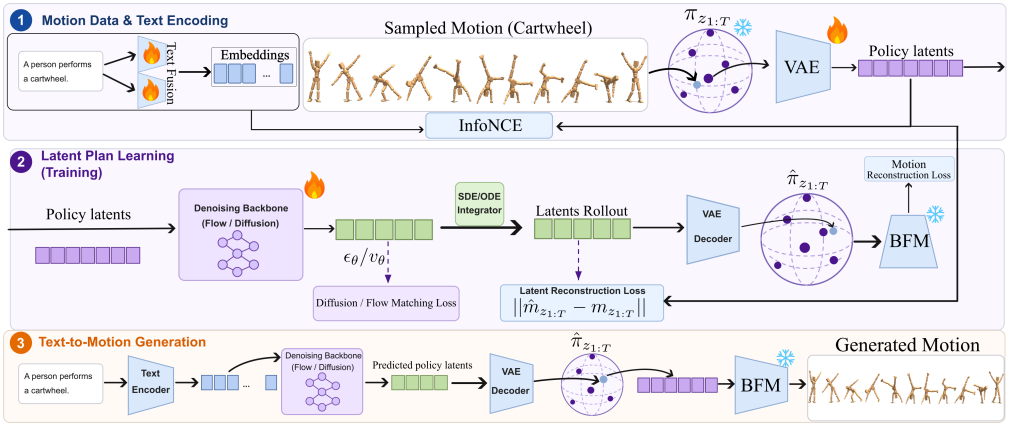


Figure 2: Text2BFM. Training (steps 1 and 2) and generation (3).

physics-based controller in which text-conditioned motion is generated as token sequences (Yao et al., 2024). Several works condition such controllers on language: PADL grounds language commands in a learned skill space (Juravsky et al., 2022) and SuperPADL scales it to thousands of skills through progressive distillation (Juravsky et al., 2024), InsActor pairs an instruction-conditioned diffusion planner with a latent skill space (Ren et al., 2023), MaskedMimic trains a unified controller that accepts partial constraints including text (Tessler et al., 2024), and CLoSD closes the loop between a text-conditioned motion diffusion planner and an RL tracker (Tevet et al., 2025). These methods either plan in kinematic space and track the plan, or learn a latent or language-conditioned space jointly with their own controller. Text2BFM instead generates continuous latent programs for a frozen, task-agnostic BFM that was trained without any text or task-specific objective, composes clauses inside one rollout, and executes the latents without an intermediate kinematic plan. As each is tied to its own controller and simulator, we leave a numerical comparison under a shared executor to future work.

Text-to-motion generation and composition. T2M has progressed from recurrent and VAE-based models (Ahuja & Morency, 2019; Ghosh et al., 2021; Petrovich et al., 2022) to diffusion, token, and latent generative approaches trained on HumanML3D (Guo et al., 2022a): MotionDiffuse and MDM denoise poses (Zhang et al., 2024; Tevet et al., 2023), MLD denoises in a motion latent space (Chen et al., 2023), and T2M-GPT and MoMask generate discrete tokens (Zhang et al., 2023a; Guo et al., 2024); recent models scale data and model size (Fan et al., 2025; Wen et al., 2025; Rempe et al., 2026). For multi-action prompts, TEACH generates actions autoregressively (Athanasiou et al., 2022) and FlowMDM composes segments with blended positional encodings (Barquero et al., 2024). All of these operate in pose, token, or kinematic latent spaces, so transitions must be synthesized kinematically; in Text2BFM, consecutive clauses are executed by one policy from the state the previous clause reached.

Language-aligned latent spaces. CLIP-style learning (Radford et al., 2021), MotionCLIP (Tevet et al., 2022), and TMR (Petrovich et al., 2023) show that language can organize motion embeddings. Our bottleneck similarly aligns latents with text, but keeps them tied to executable policy commands rather than using them only for retrieval or conditioning.

3 Method

Given a natural-language instruction Y, our goal is to generate a motion trajectory $\tau = \{s_t\}_{t=1}^{T}$. Rather than synthesizing joint poses directly, we generate motion through a frozen BFM, which provides a latent-conditioned policy

$$a_t \sim \pi_{\mathrm{BFM}}(a_t \mid s_t, z_t), \qquad s_{t+1} \sim p_{\mathrm{env}}(s_{t+1} \mid s_t, a_t), \tag{1}$$

where s_t is the humanoid state, a_t is the control action, and z_t is a local behavioral latent. Since actions are applied between consecutive states, the policy-latent sequence has length $T_z = T - 1$.

For each training motion $\tau^i = \{s_t^i\}_{t=1}^T$, we infer a sequence of BFM policy latents with the frozen Meta Motivo tracking procedure (Tirinzoni et al., 2025): we apply the backward map B to future states, average over a short lookahead window, and project to the normalized policy-latent space,

$$z_t^i = \mathrm{Proj}_z\Big(\frac{1}{H_t}\sum_{k=0}^{H_t-1} B(s_{t+1+k}^i)\Big), \qquad H_t = \min(L, T-t), \qquad t = 1, \dots, T_z. \tag{2}$$

Thus, z_t^i summarizes the future-conditioned behavior to be executed from state s_t^i.

Variational behavioral bottleneck (VBB). Directly generating $z_{1:T_z}$ remains a long-horizon sequence-modeling problem. Because BFM latents encode behavioral intent rather than frame-level pose details, and language-described motions consist of few semantic phases, we expect the latent trajectory to be redundant over time (see the motivation in Appendix F and the compression ablation in Appendix C). We therefore compress $z_{1:T_z}$ into a shorter sequence $m_{1:T_m}$, $T_m < T_z$, with a posterior $q_\phi(m_{1:T_m} \mid z_{1:T_z}) = \mathcal{N}\big(\mu_\phi(z_{1:T_z}), \mathrm{diag}(\sigma_\phi^2(z_{1:T_z}))\big)$ and a decoder $\hat{z}_{1:T_z} = D_\theta(m_{1:T_m})$. We call $m_{1:T_m}$ a *behavioral program*: a compact continuous latent sequence that decodes into an executable policy-latent sequence. The term refers to executability and modularity only; we make no claim of symbolic structure or interpretability. The reconstruction loss includes a policy-level term,

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{T_z}\sum_{t=1}^{T_z}\Big(\|z_t - \hat{z}_t\|_2^2 + \lambda_\pi D_{\mathrm{KL}}\big(\pi_{\mathrm{BFM}}(\cdot \mid s_t, \hat{z}_t) \,\|\, \pi_{\mathrm{BFM}}(\cdot \mid s_t, z_t)\big)\Big), \tag{3}$$

where s_t are the dataset states from which $z_{1:T_z}$ was extracted, so the policy term is evaluated open-loop, without rolling out the BFM during training, and the program space is regularized towards $\mathcal{N}(0, I)$ with $\mathcal{L}_{\mathrm{KL}} = D_{\mathrm{KL}}(q_\phi(m \mid z) \,\|\, \mathcal{N}(0, I))$.

Motion–text alignment. To make the bottleneck language-discriminative, program tokens $m_{i,t}$ and text tokens $y_{j,k}$ are projected to a shared normalized space, and a similarity R_{ij} between program i and text j is computed by log-sum-exp pooling over text tokens followed by a softmax-weighted average over program frames (full definition in Appendix B). Text tokens are obtained by combining the token embeddings of two pretrained encoders, the CLIP ViT-L/14 text encoder (Radford et al., 2021) and a Qwen3-Embedding-8B (Zhang et al., 2025), followed by a learned text adapter (Table 7); the same text representation conditions the flow generator. Both encoders are distinct from the GloVe/GRU text encoder of the benchmark evaluator (Guo et al., 2022a). With a learnable logit scale γ and batch size B, the semantic loss is the symmetric InfoNCE objective

$$\mathcal{L}_{\mathrm{sem}} = -\frac{1}{2B}\sum_{i=1}^{B}\Big(\log\frac{\exp(\gamma R_{ii})}{\sum_j \exp(\gamma R_{ij})} + \log\frac{\exp(\gamma R_{ii})}{\sum_j \exp(\gamma R_{ji})}\Big), \tag{4}$$

which aligns different program frames with the relevant text tokens rather than relying only on a single global similarity. The full bottleneck objective is $\mathcal{L}_{\mathrm{VBB}} = \mathcal{L}_{\mathrm{rec}} + \beta\mathcal{L}_{\mathrm{KL}} + \lambda_{\mathrm{sem}}\mathcal{L}_{\mathrm{sem}}$.

Text-conditioned flow matching. We then train a conditional generator in the program space. For each training pair we sample $m \sim q_\phi(m \mid z_{1:T_z})$ and $\epsilon \sim \mathcal{N}(0, I)$, and use the linear path $m(r) = (1-r)\epsilon + rm$, $r \sim \mathcal{U}(0, 1)$:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{\epsilon, m, r, Y}\big\|v_\eta(m(r), r, Y) - (m - \epsilon)\big\|_2^2. \tag{5}$$

Training proceeds in three stages: the pretrained BFM is kept frozen; the encoder, decoder, and projection modules are trained with $\mathcal{L}_{\mathrm{VBB}}$; and, with the bottleneck frozen, the flow model is trained with $\mathcal{L}_{\mathrm{FM}}$.

Inference and duration. Given Y and an initial state s_1, we sample $m(0) \sim \mathcal{N}(0, I)$, integrate $\frac{dm}{dr} = v_\eta(m(r), r, Y)$ from $r = 0$ to 1, decode $\hat{z}_{1:T_z} = D_\theta(m(1))$, and roll out the frozen BFM with $\hat{z}_t$ at each step. The model predicts the sequence duration from the text input and generates a corresponding sequence of policy latents, which are executed at a control rate of 30 Hz. In our experiments, sequences are limited to 256 control steps (approximately 8.5 s). The bottleneck compresses time by $4\times$, so the program length is fixed at 64 latents regardless of the executed duration. The cost of the generator is therefore independent of motion length, while the cost of the BFM rollout grows linearly with it.

Table 1: Training data of each component. Only the frozen BFM, which we use as released, has seen benchmark test motions, as motion trajectories without text.

Component	Training data	Text	Test motions seen
BFM (public Meta Motivo S-1/M-1)	AMASS (Mahmood et al., 2019)	no	69% of HML3D test
Bottleneck (VBB)	MotionHub (no AMASS)	yes	none
Flow generator	HML3D- / KIT-train	yes	none
Baselines (Table 2)	HML3D- / KIT-train	yes	none

Composition. For a prompt decomposed into clauses $Y^{(1)}, \ldots, Y^{(N)}$, **Text2BFM-Compose** samples one program per clause, $\hat{z}^{(n)} = D_\theta(\hat{m}^{(n)})$, $\hat{m}^{(n)} \sim p_\eta(m \mid Y^{(n)})$, and concatenates the decoded latent sequences. At each boundary, O overlapping latents are linearly blended,

$$\tilde{z}_o = (1-\rho_o)\hat{z}^{(n)}_{T_n-O+o} + \rho_o \hat{z}^{(n+1)}_o, \qquad \rho_o = \tfrac{o}{O+1}, \qquad o = 1, \ldots, O. \tag{6}$$

The composed sequence is executed in a single continuous BFM rollout: each clause starts from the simulator state reached by the preceding clause and inherits its pose, position, heading, and velocity. No pose-space stitching, root alignment, or offset correction is performed.

4 Experiments

4.1 Experimental Setup

We evaluate on HumanML3D (Guo et al., 2022a) (14,616 motions, 44,970 descriptions) and KIT-ML (Plappert et al., 2016) (3,911 motions, 6,278 descriptions) with the standard splits, evaluators, and metrics (Guo et al., 2022a): R-Precision, FID, MultiModal Distance (MM-Dist), and MultiModality. For our method we report the mean and 95% confidence interval over 20 evaluation runs with different seeds. Rollouts (30 Hz) are resampled to 20 FPS and converted to the standard 263-dimensional HumanML3D features, so all methods are scored by the same evaluator (Appendix A). Unlike the baselines, Text2BFM predicts the duration instead of receiving the ground-truth length. The executor is a public Meta Motivo checkpoint (Tirinzoni et al., 2025) (S-1 unless stated otherwise; M-1 in Sec. 4.6), used as released and kept frozen: we do not train or fine-tune the BFM. Table 1 lists the training data of every component. On one NVIDIA A100 80GB GPU, latent extraction takes 3 h (HumanML3D) and 1 h (KIT-ML), bottleneck training 17 h, and generator training 28 h; hyperparameters are in Appendix A.

4.2 Main Results

Table 2 compares Text2BFM with representative T2M methods. Text2BFM obtains the highest R-Precision and the lowest MM-Dist on both datasets, while its FID is higher than that of kinematic baselines.

R-Precision above real motions. The R-Precision and MM-Dist of Text2BFM are better than those of real test motions (0.876 vs. 0.797 and 2.498 vs. 2.974 on HumanML3D), and so is the BFM-free reconstruction of real motions (0.946, Table 3); the effect therefore comes from the text-aligned bottleneck, not from generation. We attribute it to the contrastive objective (Eq. 4), which keeps the factors that discriminate between descriptions and discards idiosyncratic detail, making decoded motions more prototypical of their text than recorded performances. These metrics thus measure text discriminability, not fidelity or quality. Our text encoders differ from the evaluator's (Sec. 3), so its features are not optimized directly.

FID and the executor floor. The FID gap is largely induced by the executor: passing ground-truth test motions through the frozen BFM alone yields FID 1.13 (HumanML3D) and 1.38 (KIT-ML), because the BFM imposes a fixed humanoid morphology and simulator dynamics that a kinematic evaluator registers as a distribution shift. Against this reconstruction distribution, our generations reach FID 0.28 and 0.29. The same floor applies to the alternative design of planning kinematically and tracking the plan with the same BFM: tracking even the ground-truth motions yields 1.13, so such a pipeline cannot be expected to fall below it, while Text2BFM reaches 1.172 without an intermediate kinematic plan. Whether the shift is perceptually harmless is left to a user study.

Table 2: Comparison on HumanML3D and KIT-ML. Baseline numbers are from the original papers; *Real motion* scores the ground-truth test motions with the same evaluator (Guo et al., 2022a). R-Precision and MM-Dist of Text2BFM are better than those of real motions; they therefore measure text discriminability rather than fidelity (Sec. 4.2). The last row of each block passes ground-truth test motions through the frozen BFM (states → latents → rollout) and is the FID floor of any method executed by this BFM. The BFM was pretrained on AMASS motions that include sources of most test motions (Sec. 4.3); this is therefore not an equal-data comparison. Best generated result in blue.

Dataset	Method	R-Prec. Top-3 ↑	FID ↓	MM-Dist ↓	MultiModality ↑
HumanML3D	*Real motion*	$0.797_{\pm.002}$	$0.002_{\pm.000}$	$2.974_{\pm.008}$	–
	TM2T (Guo et al., 2022b)	$0.729_{\pm.002}$	$1.501_{\pm.017}$	$3.467_{\pm.011}$	$2.424_{\pm.093}$
	T2M (Guo et al., 2022a)	$0.736_{\pm.002}$	$1.087_{\pm.021}$	$3.347_{\pm.008}$	$2.219_{\pm.074}$
	MDM (Tevet et al., 2023)	$0.611_{\pm.007}$	$0.544_{\pm.044}$	$5.566_{\pm.027}$	$2.799_{\pm.072}$
	MLD (Chen et al., 2023)	$0.772_{\pm.002}$	$0.473_{\pm.013}$	$3.196_{\pm.010}$	$2.413_{\pm.079}$
	MotionDiffuse (Zhang et al., 2024)	$0.782_{\pm.001}$	$0.630_{\pm.001}$	$3.113_{\pm.001}$	$1.553_{\pm.042}$
	T2M-GPT (Zhang et al., 2023a)	$0.775_{\pm.002}$	$0.141_{\pm.005}$	$3.121_{\pm.009}$	$1.831_{\pm.048}$
	ReMoDiffuse (Zhang et al., 2023b)	$0.795_{\pm.004}$	$0.103_{\pm.004}$	$2.974_{\pm.016}$	$1.795_{\pm.043}$
	MoMask (Guo et al., 2024)	$0.807_{\pm.002}$	$0.045_{\pm.002}$	$2.958_{\pm.008}$	$1.241_{\pm.040}$
	Text2BFM	$0.876_{\pm.005}$	$1.172_{\pm.013}$	$2.498_{\pm.061}$	$1.233_{\pm.085}$
KIT-ML	*Real motion*	$0.779_{\pm.006}$	$0.031_{\pm.004}$	$2.788_{\pm.012}$	–
	TM2T (Guo et al., 2022b)	$0.587_{\pm.005}$	$3.599_{\pm.153}$	$4.591_{\pm.026}$	$3.292_{\pm.081}$
	T2M (Guo et al., 2022a)	$0.681_{\pm.007}$	$3.022_{\pm.107}$	$3.488_{\pm.028}$	$2.052_{\pm.107}$
	MDM (Tevet et al., 2023)	$0.396_{\pm.004}$	$0.497_{\pm.021}$	$9.191_{\pm.022}$	$1.907_{\pm.214}$
	MLD (Chen et al., 2023)	$0.734_{\pm.007}$	$0.404_{\pm.027}$	$3.204_{\pm.027}$	$2.192_{\pm.071}$
	MotionDiffuse (Zhang et al., 2024)	$0.739_{\pm.004}$	$1.954_{\pm.062}$	$2.958_{\pm.005}$	$0.730_{\pm.013}$
	T2M-GPT (Zhang et al., 2023a)	$0.745_{\pm.006}$	$0.514_{\pm.029}$	$3.007_{\pm.023}$	$1.570_{\pm.039}$
	ReMoDiffuse (Zhang et al., 2023b)	$0.765_{\pm.055}$	$0.155_{\pm.006}$	$2.814_{\pm.012}$	$1.239_{\pm.028}$
	MoMask (Guo et al., 2024)	$0.781_{\pm.005}$	$0.204_{\pm.011}$	$2.779_{\pm.022}$	$1.131_{\pm.043}$
	Text2BFM	$0.901_{\pm.008}$	$1.482_{\pm.098}$	$2.658_{\pm.074}$	$1.367_{\pm.103}$

4.3 Data Exposure and the Role of the BFM

HumanML3D is built from AMASS and HumanAct12 (Guo et al., 2020). Matching source files, **69%** of HumanML3D test motions have a source sequence in the AMASS data on which the public Meta Motivo checkpoints were pretrained; KIT-ML is itself part of AMASS, so a comparable or larger overlap should be assumed there. No text annotation and no test text–motion pair enters any component (Table 1): we do not train the BFM, the bottleneck is trained on MotionHub (Ling et al., 2024) motions that are not derived from AMASS, and the generator is trained only on the benchmark training splits. The overlap is therefore *prior exposure of the executor to motion trajectories*, not label leakage, but it makes Table 2 an unequal-data comparison. We examine it in two ways.

Motions unseen by the BFM. On the 31% of HumanML3D test motions whose sources were not seen by the BFM, Text2BFM obtains R-Precision Top-3 0.83, FID 1.306, MM-Dist 2.645, and MultiModality 2.536 (full test set: 0.876, 1.172, 2.498, 1.233). Semantic alignment thus remains high on motions the executor has never seen. These numbers are not directly comparable with the full test set or published baselines, since R-Precision depends on the retrieval pools, FID is biased by sample size, and the subset differs in content; we read them only as evidence that alignment does not collapse without exposure. Subset IDs are released with the code.

Removing the BFM. To test whether the text alignment comes from the pretrained prior, we run the same bottleneck and flow generator without any BFM: both operate directly on motion features instead of policy latents, are trained on the same data as their BFM-based counterparts (MotionHub without AMASS for the VAE, the training split for the generator), and are evaluated with the same evaluator (Table 3, upper block). Without ever interacting with a model pretrained on AMASS, generation reaches R-Precision 0.912, MM-Dist 2.409, and FID 0.234, against a reconstruction ceiling of 0.947, 2.388, and 0.107. The text alignment of our pipeline is therefore explained by the text-aligned bottleneck and generator rather than by the BFM's exposure to test motions. Executing through the BFM costs distributional fidelity (FID 0.234→1.172, R-Precision 0.912→0.876), largely through the executor's floor (1.13). In exchange, motions are produced by a physics-based policy, show fewer contact artifacts (Sec. 4.5), and can be composed in one rollout (Sec. 4.4). The lower

Table 3: Role of the BFM and of compression on HumanML3D. Upper block: the same bottleneck and generator on motion features, without any BFM. Middle and lower blocks: execution by the frozen BFM (Meta Motivo S-1); the lower block ablates compression of the policy-latent sequence.

	Method	FID ↓	MM-Dist ↓	R-Prec. ↑
No BFM	VAE reconstruction	0.107	2.388	0.947
	VAE + generator	0.234	2.409	0.912
	Text2BFM (text-aligned VBB)	$1.172_{\pm.013}$	$2.498_{\pm.061}$	$0.876_{\pm.005}$
Frozen BFM, ablation	Average pooling	$2.373_{\pm.010}$	$3.573_{\pm.006}$	$0.786_{\pm.070}$
	No compression	$1.689_{\pm.007}$	$2.869_{\pm.006}$	$0.743_{\pm.004}$
	VBB w/o $\mathcal{L}_{\text{sem}}$	$1.272_{\pm.007}$	$2.832_{\pm.025}$	$0.863_{\pm.003}$

Table 4: Compositional prompts with identical clause boundaries for all methods. Decomp.: clauses generated separately and composed. Chance level of Order Accuracy is $1/N^N$: 0.037 for $N = 3$ and 0.004 for $N = 4$. $N = 4$ was evaluated for the Text2BFM variants only.

		$N = 3$			$N = 4$		
Method	Decomp.	Order ↑	MM-Dist ↓	Trans. ↓	Order ↑	MM-Dist ↓	Trans. ↓
MoMask	–	0.076	6.279	0.129	–	–	–
MoMask	✓	0.127	5.324	0.124	–	–	–
Kimodo	–	0.324	5.237	0.125	–	–	–
Kimodo	✓	0.423	3.517	0.118	–	–	–
Text2BFM	–	0.395	4.476	0.222	0.322	6.579	0.237
Text2BFM-Compose	✓	**0.671**	**3.421**	**0.047**	**0.509**	**4.61**	**0.037**

block of Table 3 ablates how $z_{1:T}$ is compressed. Average pooling ($m = \frac{1}{T}\sum_t z_t$, the conventional single-latent BFM prompt) removes temporal order and performs worst; generating the full sequence without compression improves quality but does not organize the space by language; the variational bottleneck helps, and the semantic loss further improves text–motion matching.

4.4 Compositional Motion Generation

We build N-stage prompts $Y^{(1)}$ then $\cdots$ then $Y^{(N)}$, $N \in \{3, 4\}$, from HumanML3D test descriptions (Appendix D). All methods receive the *same* clause boundaries: kinematic baselines generate clauses independently and are stitched with root alignment, offset correction, and boundary blending, whereas Text2BFM-Compose executes the clauses in one rollout (Sec. 3). Every method is also run on the full prompt without decomposition. All outputs are segmented identically, using the same clause boundaries. **Order Accuracy** embeds each segment and each clause with a frozen TMR embedder, assigns every segment independently to its most similar clause, and counts a prompt as correct if the i-th segment is assigned to the i-th clause for every i. A uniformly random assignment is correct with probability $1/N^N$, which gives a chance level of 0.037 for $N = 3$ and 0.004 for $N = 4$. **Transition** measures the discontinuity at clause boundaries,

$$\text{Transition} = \frac{1}{N-1}\sum_{i=1}^{N-1}\left(\|p_i^+ - p_i^-\|_2 + \|v_i^+ - v_i^-\|_2\right), \tag{7}$$

where p_i^-, v_i^- and p_i^+, v_i^+ are the joint positions and velocities in the last frame before and the first frame after the i-th boundary.

Table 4 compares Text2BFM with MoMask (Guo et al., 2024) and Kimodo (Rempe et al., 2026). Without any decomposition, the policy-space model already orders stages best (0.395 vs. 0.324 and 0.076), so its advantage is not an artifact of the composition procedure. Full prompts exceed MoMask's training length (9.8 s), which partly explains its non-decomposed score, close to the chance level of 0.037. Decomposition helps every method (+0.051, +0.099, +0.276) but helps policy-space generation 2.8× more than Kimodo. It also changes transitions very differently: for kinematic baselines stitching barely reduces boundary discontinuity (0.129→0.124, 0.125→0.118), whereas sequential execution reduces it by 79% (0.222→0.047). The single-program variant has the roughest boundaries of all rows; we attribute this to phase changes being realized by abrupt switches in the



Figure 3: Compositional generation for the prompt of Fig. 1 (bottom), with identical clause boundaries.

Table 5: Physical-plausibility proxies (means; definitions and 95% CIs in Appendix E).

Method	Fall (%) ↓	Skate (m/s) ↓	Pen. (cm) ↓	Cons. (%) ↑	Switch ↓	>2 cm (%) ↓
MoMask	0.7	0.115	0.02	64.1	0.015	0.2
Text2BFM	**0.2**	**0.034**	**0.00**	**94.7**	**0.011**	**0.0**
VAE+FM	0.24	0.113	0.00	68.2	0.014	0.0
Reference	0.0	0.135	0.00	56.6	0.021	0.0

decoded latent sequence, without the blending of equation 6. The gain therefore comes from the composition operator acting on executable latents, not from the representation in isolation. Long $N = 4$ prompts remain challenging, although Text2BFM-Compose stays far above the chance level of 0.004; the kinematic baselines were evaluated at $N = 3$ only. Figure 3 shows a qualitative example.

4.5 Physical Plausibility

Executing motion through a physics-based policy is not by itself evidence of physical quality, so we measure it directly on an AMASS/HumanAct12 subset excluded from BFM pretraining. All outputs are represented in the same global joint space and evaluated with identical thresholds and a shared ground plane (definitions and 95% bootstrap intervals in Appendix E). Text2BFM shows much lower foot skating and higher contact consistency than MoMask and no ground penetration (Table 5). Because MoMask and the reference are kinematic, these are kinematic proxies, not simulator measurements; the reference's higher skating likely reflects residual mocap sliding, 20 FPS sampling, and contact-inference noise. Scoring better than the reference therefore does not mean more realistic than mocap: the proxies measure the absence of specific artifacts and may also favor conservative motion with steadier contacts. Simulator-level measurements would require executing all methods through a shared controller, which we leave to future work.

4.6 Prior Capacity and Ablations

Table 6 varies only the BFM checkpoint. All metrics improve with the larger prior, and MultiModality most of all (2.7× on HumanML3D, 2.5× on KIT-ML): a larger BFM covers more executable behaviors, so one prompt admits more distinct realizations without loss of alignment. Conversely, the low MultiModality of S-1 (1.233, among the lowest in Table 2) indicates that the smaller prior, not the generator, is the main limit on diversity. The lower FID of M-1 is consistent with the FID being dominated by the executor's floor, which a stronger BFM narrows. Appendix C shows that compression up to 8× preserves policy-level behavior and that flow matching beats diffusion in FID and MM-Dist with 16 instead of 50 steps.

Table 6: Effect of the frozen prior (S-1: 24.5M parameters, M-1: 288M; all else identical).

	HumanML3D				KIT-ML			
BFM	R-Prec. ↑	FID ↓	MM-Dist ↓	MModality ↑	R-Prec. ↑	FID ↓	MM-Dist ↓	MModality ↑
S-1	0.876	1.172	2.498	1.233	0.901	1.482	2.658	1.367
M-1	0.882	1.093	2.477	3.371	0.911	1.376	2.427	3.431

5 LIMITATIONS

Prior exposure. Table 2 is not an equal-data comparison (Sec. 4.3); baselines were not re-evaluated on the unseen subset, and a BFM pretrained without test-motion sources remains to be evaluated. **Dependence on the executor.** Text2BFM inherits the BFM's morphology, dynamics, and data coverage; behaviors outside its latent space cannot be generated. The bottleneck is trained open-loop on dataset states (Eq. 3), so drift of the closed-loop rollout from these states is not modeled during training. **Controlled baselines.** We do not evaluate kinematic generators tracked by the same frozen BFM, nor physics-based text-conditioned controllers under a shared executor. The GT→BFM floor bounds the FID of the former (Sec. 4.2); its compositional behavior remains to be measured, and part of the transition gain may come from physical execution itself. **Composition.** Clause boundaries are an input, not inferred; Order Accuracy and Transition rely on a frozen embedder and inherit its biases. Baselines were evaluated for $N = 3$ only, and composition-specific methods such as TEACH and FlowMDM (Athanasiou et al., 2022; Barquero et al., 2024) are not included. **Theory.** Appendix F relies on unverified stability assumptions and does not describe the learned bottleneck. **Physical and perceptual quality.** Our physical metrics are kinematic proxies (Sec. 4.5), and no user study verifies that the BFM-induced FID shift is perceptually harmless. Rare, acrobatic, and object-dependent motions remain challenging.

6 CONCLUSION

We introduced **Text2BFM**, a framework for text-to-motion generation that operates directly in the policy-latent space of a frozen Behavioral Foundation Model. A text-aligned variational behavioral bottleneck compresses trajectories of local policy latents into compact behavioral programs, which are generated from language and executed in a single closed-loop BFM rollout. This separates semantic and temporal planning from low-level physical realization, without requiring an intermediate kinematic trajectory or fine-tuning the executor.

This representation is particularly effective for compositional motion. Under identical clause boundaries, Text2BFM-Compose improves three-stage order accuracy over the strongest evaluated kinematic baseline while reducing boundary discontinuity. Experiments on motions unseen during BFM pretraining and with a BFM-free variant further show that text alignment is learned by the proposed bottleneck and generator rather than inherited from the pretrained behavioral prior.

More broadly, our results suggest an alternative interface for language-conditioned generation: rather than generating poses directly, generate compact representations over reusable, executable behaviors, instantiated via pretrained behavioral model. Extending this interface to automatically discovered temporal structure, richer interactions, and broader behavioral foundation models is a promising direction for long-horizon, language-conditioned behavior generation.

AI USE STATEMENT

AI tools were used only to polish the manuscript text and improve grammar and readability. All technical content, experiments, and final wording were reviewed and verified by the authors.

ETHICS STATEMENT

Text2BFM may have positive societal impacts by making high-quality character animation more accessible for applications such as virtual avatars, games, education, assistive interfaces, and simulation-based robotics research. By generating motion through executable behavioral policies, it may also support safer prototyping of humanoid behaviors in simulation before physical deployment. Potential

negative impacts include deceptive avatar animation and unsafe transfer of simulated behaviors to real robots without validation.

We use HumanML3D (Guo et al., 2022a), KIT-ML (Plappert et al., 2016), AMASS (Mahmood et al., 2019), MotionHub (Ling et al., 2024), and the public Meta Motivo checkpoints (Tirinzoni et al., 2025) according to their terms of use and cite their creators. We do not redistribute third-party datasets, baseline code, or checkpoints. Baseline numbers are taken from the corresponding papers. The processing of the training data is described in Appendix G.

Reproducibility statement

We provide our code in the supplementary material, including the implementations used for all experiments, the overlap / non-overlap test-subset IDs (Sec. 4.3), and the compositional prompt lists (Appendix D). Hyperparameters and compute are listed in Appendix A.

References

Siddhant Agarwal, Harshit Sikchi, Peter Stone, and Amy Zhang. Proto successor measure: Representing the behavior space of an RL agent. In *International Conference on Machine Learning (ICML)*, 2025. URL https://openreview.net/forum?id=mUDnPzopZF.

Chaitanya Ahuja and Louis-Philippe Morency. Language2Pose: Natural language grounded pose forecasting. In *International Conference on 3D Vision (3DV)*, pp. 719–728, 2019.

Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations (ICLR)*, 2023.

Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. TEACH: Temporal action composition for 3D humans. In *International Conference on 3D Vision (3DV)*, 2022.

German Barquero, Sergio Escalera, and Cristina Palmero. Seamless human motion composition with blended positional encodings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 457–469, 2024.

André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, Hado van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Léonard Blier, Corentin Tallec, and Yann Ollivier. Learning successor states and goal-dependent values: A mathematical viewpoint. *arXiv preprint arXiv:2101.07123*, 2021.

Maksim Bobrin, Ilya Zisman, Alexander Nikulin, Vladislav Kurenkov, and Dmitry V. Dylov. Zero-shot adaptation of behavioral foundation models to unseen dynamics. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=dBDBg4WF4F.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18000–18010, 2023.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993.

Ke Fan, Shunlin Lu, Minyue Dai, Runyi Yu, Lixing Xiao, Zhiyang Dou, Junting Dong, Lizhuang Ma, and Jingbo Wang. Go to zero: Towards zero-shot motion generation with million-scale data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13336–13348, 2025.

Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1396–1406, 2021.

Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2Motion: Conditioned generation of 3D human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2021–2029, 2020. doi: 10.1145/3394171.3413635.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3D human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5152–5161, 2022a.

Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. TM2T: Stochastic and tokenized modeling for the reciprocal generation of 3D human motions and texts. In *European Conference on Computer Vision (ECCV)*, pp. 580–597, 2022b.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. MoMask: Generative masked modeling of 3D human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1900–1910, 2024.

Jordan Juravsky, Yunrong Guo, Sanja Fidler, and Xue Bin Peng. PADL: Language-directed physics-based character control. In *SIGGRAPH Asia 2022 Conference Papers*, 2022. doi: 10.1145/3550469.3555391.

Jordan Juravsky, Yunrong Guo, Sanja Fidler, and Xue Bin Peng. Superpadl: Scaling language-directed physics-based control with progressive supervised distillation. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Yitang Li, Zhengyi Luo, Tonghe Zhang, Cunxi Dai, Anssi Kanervisto, Andrea Tirinzoni, Haoyang Weng, Kris Kitani, Mateusz Guzek, Ahmed Touati, Alessandro Lazaric, Matteo Pirotta, and Guanya Shi. BFM-zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=jkhl2oI0g5.

Zeyu Ling, Shunlin Lu, Yuhong Zhang, et al. VersatileMotion: A unified framework for motion synthesis and comprehension. *arXiv preprint arXiv:2411.17335*, 2024. First version titled "MotionLLaMA: A Unified Framework for Motion Synthesis and Comprehension".

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.

Zhengyi Luo, Jinkun Cao, Alexander Winkler, Kris Kitani, and Weipeng Xu. Perpetual humanoid control for real-time simulated avatars. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10895–10904, 2023.

Zhengyi Luo, Jinkun Cao, Josh Merel, Alexander Winkler, Jing Huang, Kris Kitani, and Weipeng Xu. Universal humanoid motion representations for physics-based control. In *International Conference on Learning Representations*, volume 2024, pp. 56766–56782, 2024.

Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5442–5451, 2019.

Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. AMP: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics*, 40(4): 144:1–144:20, 2021.

Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. ASE: Large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions on Graphics*, 41(4):94:1–94:17, 2022.

Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, pp. 480–497, 2022.

Mathis Petrovich, Michael J. Black, and Gül Varol. TMR: Text-to-motion retrieval using contrastive 3D human motion synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9488–9497, 2023.

Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. *Big Data*, 4(4):236–252, 2016. doi: 10.1089/big.2016.0028.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. HuMoR: 3D human motion model for robust pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11488–11499, 2021.

Davis Rempe, Mathis Petrovich, Ye Yuan, Haotian Zhang, Xue Bin Peng, Yifeng Jiang, Tingwu Wang, Umar Iqbal, David Minor, Michael de Ruyter, Jiefeng Li, Chen Tessler, Edy Lim, Eugene Jeong, Sam Wu, Ehsan Hassani, Michael Huang, Jin-Bey Yu, Chaeyeon Chung, Lina Song, Olivier Dionne, Jan Kautz, Simon Yuen, and Sanja Fidler. Kimodo: Scaling controllable human motion generation. *arXiv preprint arXiv:2603.15546*, 2026.

Jiawei Ren, Mingyuan Zhang, Cunjun Yu, Xiao Ma, Liang Pan, and Ziwei Liu. InsActor: Instruction-driven physics-based characters. In *Advances in Neural Information Processing Systems*, volume 36, pp. 59911–59923, 2023.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning (ICML)*, pp. 1278–1286, 2014.

Harshit Sikchi, Siddhant Agarwal, Pranaya Jajoo, Samyak Parajuli, Caleb Chuck, Max Rudolph, Peter Stone, Amy Zhang, and Scott Niekum. RL zero: Zero-shot language to behaviors without any supervision. In *7th Robot Learning Workshop: Towards Robots with Human-Level Abilities (ICLR 2025 Workshop)*, 2025. URL https://openreview.net/forum?id=wNvuk13MnP.

Chen Tessler, Yoni Kasten, Yunrong Guo, Shie Mannor, Gal Chechik, and Xue Bin Peng. CALM: Conditional adversarial latent models for directable virtual characters. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

Chen Tessler, Yunrong Guo, Ofir Nabati, Gal Chechik, and Xue Bin Peng. MaskedMimic: Unified physics-based character control through masked motion inpainting. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 43(6), 2024. doi: 10.1145/3687951.

Guy Tevet, Brian Gordon, Amir Hertz, Amit H. Bermano, and Daniel Cohen-Or. MotionCLIP: Exposing human motion generation to CLIP space. In *European Conference on Computer Vision (ECCV)*, pp. 358–374, 2022.

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.

Guy Tevet, Sigal Raab, Setareh Cohan, Daniele Reda, Zhengyi Luo, Xue Bin Peng, Amit H. Bermano, and Michiel van de Panne. CLoSD: Closing the loop between simulation and diffusion for multi-task character control. In *International Conference on Learning Representations (ICLR)*, 2025.

Andrea Tirinzoni, Ahmed Touati, Jesse Farebrother, Mateusz Guzek, Anssi Kanervisto, Yingchen Xu, Alessandro Lazaric, and Matteo Pirotta. Zero-shot whole-body humanoid control via behavioral foundation models. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=9sOR0nYLtz.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. In *Advances in Neural Information Processing Systems*, volume 34, pp. 13–24, 2021.

Ahmed Touati, Jérémy Rapin, and Yann Ollivier. Does zero-shot reinforcement learning exist? In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=MYEap_OcQI.

Yuxin Wen, Qing Shuai, Di Kang, Jing Li, Cheng Wen, Yue Qian, Ningxin Jiao, Changhai Chen, Weijie Chen, Yiran Wang, et al. HY-Motion 1.0: Scaling flow matching models for text-to-motion generation. *arXiv preprint arXiv:2512.23464*, 2025.

Heyuan Yao, Zhenhua Song, Yuyang Zhou, Tenglong Ao, Baoquan Chen, and Libin Liu. Moconvq: Unified physics-based motion control via scalable discrete representations. *ACM Transactions on Graphics (TOG)*, 43(4):1–21, 2024.

Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. PhysDiff: Physics-guided human motion diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16010–16021, 2023.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14730–14740, 2023a.

Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. ReMoDiffuse: Retrieval-augmented motion diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 364–373, 2023b.

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*, 46(6):4115–4128, 2024. doi: 10.1109/TPAMI.2024.3355414.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

A Technical Details and Hyperparameters

Training follows Sec. 3. Policy latents are first extracted offline with the frozen BFM (Eq. 2); the variational behavioral bottleneck is then trained with $\mathcal{L}_{\mathrm{VBB}}$ (Stage 1); finally, the text-conditioned flow generator is trained with $\mathcal{L}_{\mathrm{FM}}$ while the bottleneck is frozen (Stage 2). The BFM is never trained or fine-tuned.

Stage 1: Variational behavioral bottleneck. The backbone is a causal 1D convolutional encoder-decoder with residual temporal blocks and hierarchical downsampling. The temporal axis is compressed by a factor of 4 (256 policy-latent steps $\rightarrow$ 64 program latents). The latent sequence m has stochastic posterior parameterization $(\mu, \log \sigma^2)$ and reparameterized sampling.

Stage 2: Text-conditioned flow generator. The generator is a transformer trained from scratch with the flow-matching objective only (Eq. 5); no other loss is used and the bottleneck stays frozen. It generates a program m, the frozen decoder maps it to policy latents $\hat{z}_{1:T_z}$, and the frozen BFM policy executes $\hat{z}_{1:T_z}$ in the simulator.

Evaluation in the HumanML3D feature space. The simulator and the BFM policy run at 30 Hz. For evaluation, we record the SMPL joint positions of each rollout, resample them to 20 FPS, and compute the 263-dimensional HumanML3D features with the official preprocessing script (Guo et al., 2022a). The same conversion is applied to the GT$\rightarrow$BFM rollouts.

Compute resources. All experiments were run on a single NVIDIA A100 80GB GPU with 32 CPU cores and 256GB system RAM. Offline BFM policy-latent extraction took approximately 3 hours for HumanML3D and 1 hour for KIT-ML. Training the behavioral bottleneck and text-to-plan flow generator took 17 and 28 hours, respectively, and evaluation took approximately 2–3 hours across HumanML3D and KIT-ML, including repeated sampling for confidence intervals. A full reproduction of the main Text2BFM results requires approximately 50 A100 GPU-hours, excluding dataset download and environment setup.

B Motion–Text Alignment Details

Let $m_i = \{m_{i,t}\}_{t=1}^{T_m}$ denote the program for sample i and $Y_i = \{y_{i,k}\}_{k=1}^{K_i}$ its token-level text representation. Program and text tokens are projected into a shared normalized space:

$$\tilde{m}_{i,t} = \frac{P_m(m_{i,t})}{\|P_m(m_{i,t})\|_2}, \qquad \tilde{y}_{j,k} = \frac{P_y(y_{j,k})}{\|P_y(y_{j,k})\|_2}. \tag{8}$$

For each pair (i, j) in a batch, a frame-level score pools over text tokens,

$$F_{ijt} = \lambda_{\mathrm{tok}} \log \Big(\frac{1}{K_j} \sum_{k=1}^{K_j} \exp \Big(\frac{\tilde{m}_{i,t}^{\top} \tilde{y}_{j,k}}{\lambda_{\mathrm{tok}}} \Big) \Big), \tag{9}$$

where λ_{tok} controls the sharpness of token pooling; the normalization by K_j prevents longer descriptions from receiving systematically larger scores. Frame-importance weights and the final similarity are

$$w_{ijt} = \frac{\exp(F_{ijt}/\lambda_{\mathrm{frm}})}{\sum_{u=1}^{T_m} \exp(F_{iju}/\lambda_{\mathrm{frm}})}, \qquad R_{ij} = \sum_{t=1}^{T_m} w_{ijt} F_{ijt}, \tag{10}$$

where λ_{frm} controls how concentrated the frame weighting is. With $\gamma = \exp(\alpha)$ a positive learnable logit scale, the two directions of the contrastive loss are

$$\mathcal{L}_{m \to Y} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\exp(\gamma R_{ii})}{\sum_{j=1}^{B} \exp(\gamma R_{ij})}, \qquad \mathcal{L}_{Y \to m} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\exp(\gamma R_{ii})}{\sum_{j=1}^{B} \exp(\gamma R_{ji})}, \tag{11}$$

and $\mathcal{L}_{\mathrm{sem}} = \frac{1}{2}(\mathcal{L}_{m \to Y} + \mathcal{L}_{Y \to m})$ (Eq. 4).

Table 7: Core architecture and hyperparameters.

Category	Hyperparameter	Value
VAE	Batch size	256
VAE	Input width	256
VAE	Latent dim d_m	48
VAE	Latent ratio / min / max	0.25 / 32 / 48
VAE	Stride per level	2
VAE	Channel width / residual depth	256 / 3
VAE	Dilation growth rate	3
VAE	Activation / norm	ReLU / GroupNorm
VAE	Residual dropout	0.1
VAE	Padding mode	replicate
VAE	Learning rate / weight decay	5×10^{-5} / 5×10^{-4}
VAE	Loss coef. λ_π	0.1
VAE	Loss coef. λ_{sem}	0.35
VAE	KL regularization β	10^{-4}
Generator	Batch size	256
Generator	Learning rate / weight decay	3×10^{-5} / 5×10^{-4}
Generator	LR warmup / start factor	2000 / 0.1
Generator	Min LR ratio / LR decay steps	0.1 / 15000
Generator	Early stopping patience / min delta	12 / 10^{-4}
Generator	Feature width / depth / heads	512 / 8 / 8
Generator	Dropout / MLP ratio	0.1 / 4.0
Generator	MLP activation	`gelu_tanh`
Generator	QK normalization / QKV bias	`rms` / True
Generator	Start token / long skip connection	False / False
Conditioning	Text token dim / context dim	768 / 1024
Conditioning	Text adapter (width / depth / heads)	768 / 2 / 8
Conditioning	Text adapter dropout	0.1
Conditioning	Temporal mask type / span	`narrowband` / 2.0
Conditioning	Condition dropout probability	0.2
Conditioning	RoPE on single branch / time factor	True / 1.0
Conditioning	Null context length	1
Sampling	Guidance scale	1.5
Sampling	Sampling steps / solver	16 / `euler`

C Additional Ablations

Compression factor. Given $z_{1:T}$, the encoder produces a shorter program and the decoder reconstructs $\hat{z}_{1:T}$. We compare compression factors $\{4, 8, 16\}$ by latent reconstruction error and by the policy-level Action KL, which measures whether the reconstructed latents induce similar action distributions under the frozen BFM:

$$\text{ActionKL} = \frac{1}{T} \sum_{t=1}^{T} D_{\text{KL}} \left(\pi_{\text{BFM}}(\cdot \mid s_t, \hat{z}_t) \,\|\, \pi_{\text{BFM}}(\cdot \mid s_t, z_t) \right). \tag{12}$$

Table 8 shows that $4\times$ and $8\times$ preserve both latent reconstruction and induced policy behavior, while $16\times$ substantially increases Action KL. We use $4\times$ (64 program latents) in all other experiments.

Table 8: Effect of temporal compression on BFM latent reconstruction and policy-level behavior preservation. Lower is better.

Compression	**Reconstruction** $\downarrow$	**Action KL** $\downarrow$
$4\times$	0.187	1.04
$8\times$	0.194	1.05
$16\times$	0.342	2.02

Flow matching vs. diffusion. Table 9 compares the generation backbone in the same bottleneck space. Flow matching improves FID and MM-Dist and requires fewer sampling steps, while diffusion obtains slightly higher R-Precision.

Table 9: Choice of generator. Both variants operate in the learned behavioral bottleneck space.

Generator	FID ↓	MM-Dist ↓	R-Prec. ↑	Sampling Steps ↓
Diffusion	$1.672_{\pm.011}$	$2.617_{\pm.064}$	$0.885_{\pm.003}$	50
Flow matching	$1.172_{\pm.013}$	$2.498_{\pm.061}$	$0.876_{\pm.005}$	**16**

D Compositional Evaluation Protocol

Prompt construction. We construct multi-stage prompts by concatenating $N \in 3, 4$ atomic descriptions from the HumanML3D test split with "then". Each clause specifies a successive action, and the resulting sequence is executed in a single continuous rollout. The full prompt lists are released with the code.

Clause boundaries and stitching. All methods receive the same clause decomposition. For MoMask and Kimodo with decomposition, each clause is generated independently with the clause duration from ground-truth data, and consecutive clauses are stitched by aligning the root position and heading of the next clause to the last frame of the previous one, correcting the vertical offset, and linearly blending 10 frames at the boundary. Text2BFM-Compose uses the latent blending of equation 6 and no pose-space stitching.

Segmentation and metrics. Every output is segmented at the given clause boundaries (identically for all methods). Order Accuracy uses a frozen TMR (Petrovich et al., 2023) text–motion embedder and assigns each segment independently to its most similar clause; a prompt counts as correct only if every segment is assigned to its own clause (chance level $1/N^N$).

E Physical Plausibility Metrics

Definitions. *Fall/collapse proxy*: percentage of sequences containing a sustained low-pelvis or low-head configuration together with excessive torso tilt. *Foot skating*: mean tangential foot velocity during inferred ground contact. *Joint–ground penetration*: mean per-frame maximum joint depth below the ground plane. *Contact consistency*: percentage of inferred contact frames whose tangential foot velocity is below 0.1 m/s. *Contact switch rate*: frequency of changes in the inferred foot-contact state. *Penetration-frame rate*: percentage of frames with more than 2 cm of joint–ground penetration. Contacts are inferred when foot height above the ground plane is at most $0.04\,\mathrm{m}$ and absolute vertical foot velocity is at most $0.20\,\mathrm{m/s}$ for at least two consecutive frames, using identical thresholds for all methods. All values are means with 95% bootstrap confidence intervals over matched sequences.

Table 10: Physical-plausibility proxies with 95% bootstrap confidence intervals.

Method	Fall (%) ↓	Skate (m/s) ↓	Pen. (cm) ↓	Cons. (%) ↑	Switch ↓	>2 cm (%) ↓
MoMask	0.7 [0.3, 1.2]	0.115 [0.108, 0.125]	0.02 [0.01, 0.02]	64.1 [62.4, 65.7]	0.015 [0.014, 0.016]	0.2 [0.1, 0.3]
Text2BFM	0.2 [0.0, 0.4]	0.034 [0.032, 0.036]	0.00 [0.00, 0.00]	94.7 [94.3, 95.0]	0.011 [0.011, 0.012]	0.0 [0.0, 0.0]
Reference	0.0 [0.0, 0.0]	0.135 [0.130, 0.141]	0.00 [0.00, 0.00]	56.6 [55.1, 58.1]	0.021 [0.019, 0.022]	0.0 [0.0, 0.0]

F Motivation for Text-Aligned Behavioral Compression

This section gives an informal motivation for compressing BFM policy-latent trajectories into compact text-aligned behavioral programs. The statements below hold under idealized assumptions and are not guarantees for the trained system; the empirical support for compression is the ablation in Appendix C. The key point is that MetaMotivo-style latents are not arbitrary framewise codes. They are future-conditioned behavioral contexts obtained from a frozen backward representation and then projected to a normalized policy-latent space. Thus, a latent z_t should be interpreted as a local

command describing what behavior should unfold from state s_t, rather than as a direct encoding of the current pose.

For notational simplicity, we denote the policy-latent sequence length by T; in the main text, this is $T_z = T_{\text{motion}} - 1$. Given a motion trajectory $s_{1:T+1}$, the tracked BFM pseudo-labels are constructed for $t = 1, \dots, T$ as

$$\bar{z}_t = \frac{1}{H_t} \sum_{k=0}^{H_t-1} B_\psi(s_{t+1+k}), \qquad H_t = \min(L, T+1-t),$$

which is Eq. 2 with $T_{\text{motion}} = T + 1$; in particular, $H_t \geq 1$ for every t. The averages are then projected to the normalized latent sphere:

$$z_t = \text{Proj}_z(\bar{z}_t), \qquad \text{Proj}_z(u) = \sqrt{d_z} \frac{u}{\|u\|_2}.$$

Hence, $z_{1:T}$ is a temporally ordered curve on the sphere of radius $\sqrt{d_z}$. Its complexity is better described by its path length than by the ambient latent dimension. We measure this path length using total variation:

$$\text{TV}(z_{1:T}) = \sum_{t=1}^{T-1} \|z_{t+1} - z_t\|_2.$$

Low total variation means that behavioral intent changes gradually. Such trajectories are naturally compressible: neighboring latents often communicate nearly identical control information to the frozen BFM policy.

The future-conditioned construction provides an explicit smoothing mechanism. When both windows are full, $H_t = H_{t+1} = L$, which holds for $t \leq T - L$, adjacent averages satisfy a telescoping identity:

$$\bar{z}_{t+1} - \bar{z}_t = \frac{1}{L} \left(B_\psi(s_{t+L+1}) - B_\psi(s_{t+1}) \right).$$

Thus, most intermediate future terms cancel. Assume that B_ψ is L_B-Lipschitz and that $\|\bar{z}_t\|_2 \geq \rho > 0$ for all t. For $\|u\|_2, \|v\|_2 \geq \rho$,

$$\left\| \frac{u}{\|u\|_2} - \frac{v}{\|v\|_2} \right\|_2 \leq \frac{\|u - v\|_2}{\|u\|_2} + \|v\|_2 \left| \frac{1}{\|u\|_2} - \frac{1}{\|v\|_2} \right| = \frac{\|u - v\|_2 + \left| \|v\|_2 - \|u\|_2 \right|}{\|u\|_2} \leq \frac{2\|u - v\|_2}{\rho},$$

so the projection map is $\frac{2\sqrt{d_z}}{\rho}$-Lipschitz on this region. For $t \leq T - L$ this yields

$$\|z_{t+1} - z_t\|_2 \leq \frac{2\sqrt{d_z} L_B}{\rho L} \|s_{t+L+1} - s_{t+1}\|_2.$$

The remaining $L - 1$ steps $t > T - L$ have shrinking windows, and each of them is bounded by the diameter $2\sqrt{d_z}$ of the sphere. Consequently, if the motion has bounded future-window displacement,

$$\sum_{t=1}^{T-L} \|s_{t+L+1} - s_{t+1}\|_2 \leq C_{\text{motion}},$$

then

$$\text{TV}(z_{1:T}) \leq \frac{2\sqrt{d_z} L_B}{\rho L} C_{\text{motion}} + 2(L-1)\sqrt{d_z}.$$

Since every window displacement is at most the path length of the states inside the window, $C_{\text{motion}} \leq L\, \text{TV}(s_{1:T+1})$, so the first term never exceeds $\frac{2\sqrt{d_z} L_B}{\rho}\, \text{TV}(s_{1:T+1})$. It is much smaller when the motion oscillates within the window, because then $\|s_{t+L+1} - s_{t+1}\|_2$ is small although the path inside the window is long. BFM latents therefore inherit a smoothing bias from future averaging: fluctuations faster than the window are averaged out, while persistent changes in behavioral intent remain.

We now connect latent compression to rollout quality. Let the closed-loop BFM dynamics be

$$s_{t+1} = F(s_t, z_t), \qquad F(s, z) = f(s, \pi_{\text{BFM}}(s, z)),$$

where f is the environment dynamics and $\pi_{\mathrm{BFM}}(s, z)$ denotes the deterministic action of the frozen policy (e.g., its mean), so that the closed loop is deterministic, as for a deterministic simulator. Assume F is Lipschitz:

$$\|F(s, z) - F(s', z')\|_2 \leq L_s\|s - s'\|_2 + L_z\|z - z'\|_2.$$

Here, L_s measures closed-loop sensitivity to state perturbations, while L_z measures sensitivity to latent reconstruction error.

Proposition F.1 (Compact latent plans yield bounded rollout error)**.** *Let $z_{1:T}$ be a BFM policy-latent trajectory with total variation*

$$V = \mathrm{TV}(z_{1:T}).$$

For any $m \geq 1$, there exists a piecewise-constant approximation $\tilde{z}_{1:T}$ with at most m contiguous segments such that

$$\max_t \|z_t - \tilde{z}_t\|_2 \leq \frac{V}{m}.$$

Let $s_{1:T}$ and $\tilde{s}_{1:T}$ be rollouts from the same initial state using $z_{1:T}$ and $\tilde{z}_{1:T}$, respectively. Then

$$\|\tilde{s}_t - s_t\|_2 \leq L_z \frac{V}{m} \sum_{j=1}^{t-1} L_s^{t-1-j}.$$

In particular, if $L_s < 1$, then

$$\|\tilde{s}_t - s_t\|_2 \leq \frac{L_z V}{m(1 - L_s)}.$$

Proof. Let $\delta = V/m$. If $V = 0$, all latents coincide and one segment suffices, so assume $V > 0$. Partition the sequence greedily: start a segment at index 1 and extend it while its internal total variation stays at most δ; when adding the next step would exceed δ, start a new segment at the next index. For every segment except the last, its internal total variation plus the length of the step that closes it exceeds δ. These quantities are disjoint parts of V, so a partition with K segments satisfies $(K - 1)\delta < V$, i.e., $K - 1 < m$ and $K \leq m$. Define $\tilde{z}_t$ as the first latent vector in the segment containing t. If the segment begins at index a, then

$$\|z_t - \tilde{z}_t\|_2 = \|z_t - z_a\|_2 \leq \sum_{j=a}^{t-1} \|z_{j+1} - z_j\|_2 \leq \delta.$$

Thus,

$$\max_t \|z_t - \tilde{z}_t\|_2 \leq \frac{V}{m}.$$

Let

$$e_t = \|\tilde{s}_t - s_t\|_2.$$

Since both rollouts begin from the same initial state, $e_1 = 0$. By Lipschitz continuity,

$$e_{t+1} = \|F(\tilde{s}_t, \tilde{z}_t) - F(s_t, z_t)\|_2 \leq L_s e_t + L_z\|\tilde{z}_t - z_t\|_2.$$

Unrolling the recursion gives

$$e_t \leq L_z \sum_{j=1}^{t-1} L_s^{t-1-j}\|\tilde{z}_j - z_j\|_2.$$

Using the uniform approximation bound yields

$$e_t \leq L_z \frac{V}{m} \sum_{j=1}^{t-1} L_s^{t-1-j}.$$

If $L_s < 1$, the geometric sum is bounded by $\frac{1}{1-L_s}$, giving

$$e_t \leq \frac{L_z V}{m(1 - L_s)}.$$

□

The results in this section are a motivation for the design, not a guarantee for the trained system (see the scope paragraph at the end of this section). In particular, the uniform bound in Proposition F.1 requires a contractive closed loop ($L_s < 1$), a strong assumption for contact-rich humanoid dynamics; without it the bound grows geometrically with the horizon and should be read only as a statement about short segments. The proposition gives a conservative explanation for why short latent programs can remain executable. The relevant quantity is not the dimensionality of the latent vector, but the temporal variation of the latent curve. If behavioral intent changes slowly and the closed loop is contractive, then a compressed latent program induces only bounded rollout error. A learned decoder is more expressive than the piecewise-constant construction in the proof: it can generate smooth curves, reuse motifs, model phase structure, and allocate capacity to transitions such as contacts or direction changes.

The variational behavioral bottleneck can be interpreted as a rate–distortion relaxation. Let $Z = z_{1:T}$ be the tracked latent trajectory and let M be the compact behavioral program. The encoder $q_\phi(M|Z)$ and the decoder $\hat{Z} = D_\theta(M)$ optimize the reconstruction and KL terms of $\mathcal{L}_{\mathrm{VBB}}$, a tradeoff of the form

$$\mathcal{L}_{\mathrm{rec}} + \beta\mathcal{L}_{\mathrm{KL}} = \mathbb{E}_{q_\phi(M|Z)}\left[D(Z,\hat{Z})\right] + \beta D_{\mathrm{KL}}\left(q_\phi(M|Z)\|p(M)\right),$$

where D is the reconstruction loss of Eq. 3 and $p(M) = \mathcal{N}(0, I)$. The reconstruction term preserves the information needed to recover executable BFM latents, while the KL term prevents M from becoming a framewise copy of Z. The bottleneck therefore encourages the model to store reusable behavioral factors such as speed, direction, gait, contact structure, and style.

However, reconstruction alone does not guarantee that the compressed program preserves the aspects of motion that matter for language. A purely reconstructive code may spend capacity on details that improve latent reconstruction but are weakly discriminative under text. Semantic alignment biases the bottleneck toward language-relevant behavioral factors.

The following statement idealizes the alignment as a cosine similarity between global unit-normalized embeddings. Our similarity R_{ij} (Eq. 10) pools over tokens and frames and is not a cosine, so the statement describes the mechanism rather than the exact loss.

Proposition F.2 (Semantic alignment preserves text discrimination under a margin)**.** *Let e_Y be a unit-normalized embedding of the correct text prompt and let e_M be a unit-normalized embedding of the corresponding compact behavioral program. Suppose*

$$1 - e_Y^\top e_M \le \eta.$$

Let e_Y^- be the unit-normalized embedding of an incorrect prompt. If

$$e_Y^\top e_Y^- \le 1 - \Delta$$

and

$$\Delta > \eta + \sqrt{2\eta},$$

then

$$e_M^\top e_Y > e_M^\top e_Y^-.$$

Proof. Since e_Y and e_M are unit-normalized,

$$\|e_M - e_Y\|_2^2 = 2 - 2e_M^\top e_Y.$$

The alignment assumption gives

$$\|e_M - e_Y\|_2 \le \sqrt{2\eta}.$$

For the negative prompt,

$$e_M^\top e_Y^- = e_Y^\top e_Y^- + (e_M - e_Y)^\top e_Y^-.$$

By Cauchy–Schwarz,

$$(e_M - e_Y)^\top e_Y^- \le \sqrt{2\eta}.$$

Therefore,

$$e_M^\top e_Y^- \le 1 - \Delta + \sqrt{2\eta}.$$

Meanwhile,

$$e_M^\top e_Y \ge 1 - \eta.$$

Hence,

$$e_M^\top e_Y - e_M^\top e_Y^- \geq \Delta - \eta - \sqrt{2\eta}.$$

If $\Delta > \eta + \sqrt{2\eta}$, the right-hand side is positive, proving the claim. □

For multiple negatives $\{e_{Y_i^-}\}_{i=1}^N$ satisfying

$$e_Y^\top e_{Y_i^-} \leq 1 - \Delta,$$

the same argument gives

$$e_M^\top e_Y - e_M^\top e_{Y_i^-} \geq \Delta - \eta - \sqrt{2\eta}.$$

Let $p(Y|M)$ be the softmax over the correct prompt and the N negatives with logits $e_M^\top e_{Y'}/\tau$, where the temperature τ corresponds to $1/\gamma$ in Eq. 4. Then

$$p(Y|M) = \frac{1}{1 + \sum_{i=1}^N \exp\left((e_M^\top e_{Y_i^-} - e_M^\top e_Y)/\tau\right)} \geq \frac{1}{1 + N\exp\left(-\frac{\Delta - \eta - \sqrt{2\eta}}{\tau}\right)}.$$

Thus, reducing alignment error increases the contrastive separability of the compact behavioral program. This is an elementary consequence of the geometry of unit vectors and is included only to make the role of $\mathcal{L}_{\text{sem}}$ explicit.

Together, these arguments motivate text-aligned behavioral compression. Future-conditioned BFM latents form smooth behavioral curves; bounded-variation curves admit compact approximations; a contractive closed loop ($L_s < 1$) converts small latent error into bounded rollout error; and semantic alignment encourages the compressed program to preserve the factors of motion that are discriminative under language. The intended representation is therefore a compact semantic control program: low-rate enough to compress motion, executable enough to drive the frozen BFM policy, and aligned enough to support text-conditioned generation.

Scope of the analysis. The arguments above rest on four assumptions: (i) the backward map B_ψ is L_B-Lipschitz; (ii) the averaged latents stay away from the origin, $\|\bar{z}_t\|_2 \geq \rho > 0$; (iii) the closed loop F is deterministic and Lipschitz with constants L_s, L_z; and (iv) for the horizon-independent bound, the closed loop is contractive, $L_s < 1$. We do not estimate L_B, ρ, L_s, or L_z for Meta Motivo, and we do not verify (iv). Global contractivity is unlikely for contact-rich humanoid dynamics: contacts and balance make the closed loop locally expansive around some states, so Proposition F.1 should be read as a statement about short segments, with an error that can grow geometrically otherwise. The BFM policy acts on the current simulator state, so it can correct deviations instead of accumulating them, but we have not measured this.

Nor do the propositions describe the learned bottleneck. Proposition F.1 concerns a piecewise-constant approximation, not the trained encoder and decoder, and Proposition F.2 concerns an idealized cosine similarity, not the pooled score R_{ij}. The behavior of the learned bottleneck is established only empirically. Table 8 measures its latent reconstruction error and the policy-level Action KL of the reconstructed latents under the frozen BFM. If the number of segments is identified with the program length, the piecewise-constant bound doubles with each doubling of the compression factor; the learned decoder is almost unaffected from $4\times$ to $8\times$ (reconstruction $0.187 \to 0.194$, Action KL $1.04 \to 1.05$) and degrades only at $16\times$ (0.342, 2.02). This is consistent with the qualitative prediction that compression fails once the program becomes too short for the variation of the latent curve, while at moderate compression the learned decoder degrades much more slowly than the constructive bound would suggest. The end-to-end effect of compression on executed motion is measured in Table 3. Estimating the constants above, for example by perturbing the initial state or the latents of BFM rollouts and measuring the divergence of the resulting trajectories over the horizon, is left to future work.

G Dataset Structure

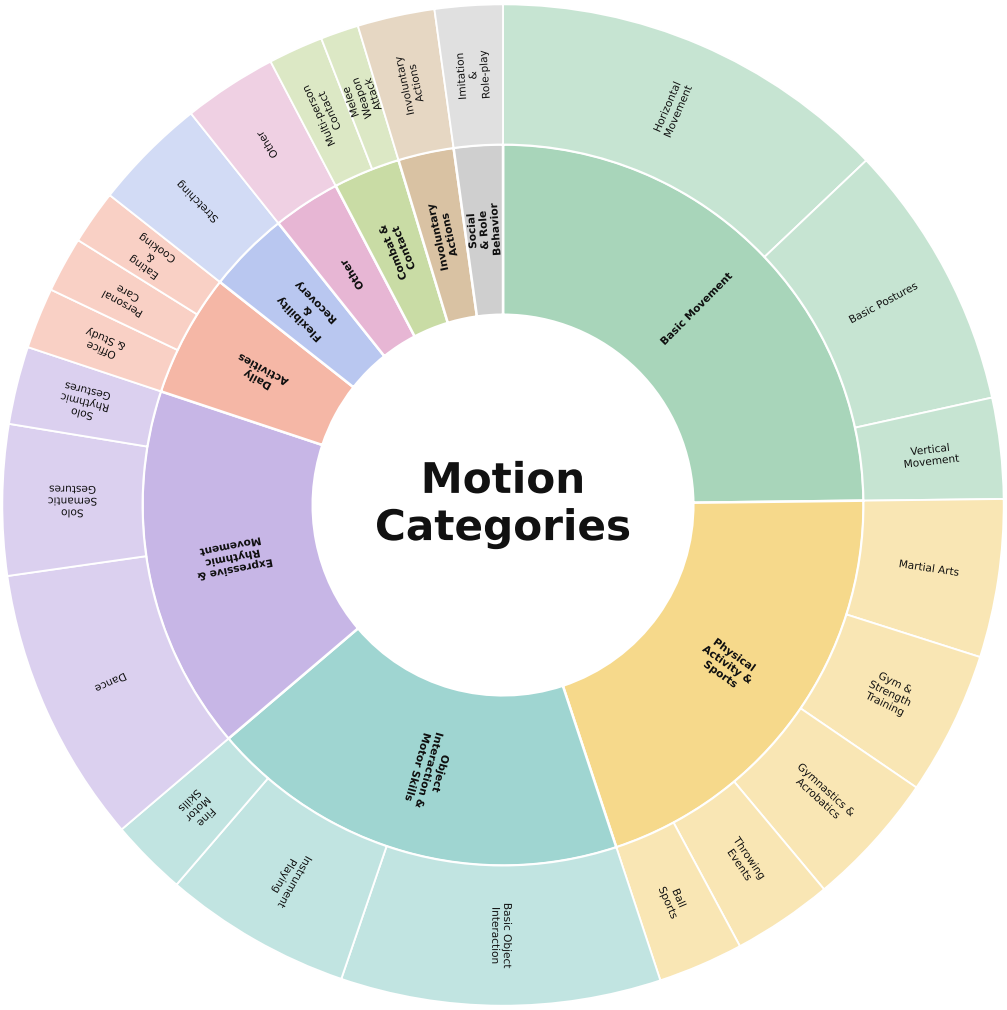
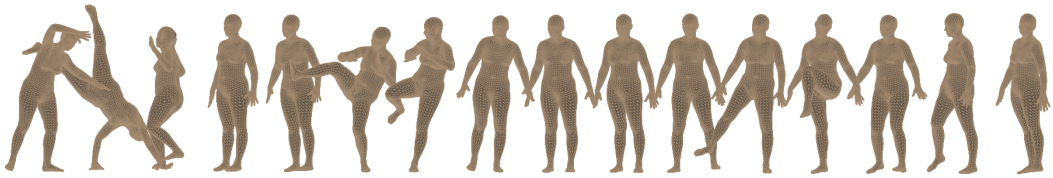


Figure 4: Hierarchical visualization of motion categories in the dataset. This corpus is used to train the variational behavioral bottleneck; it is drawn from MotionHub (Ling et al., 2024) and contains no AMASS-derived sequences, so it shares no motions with the HumanML3D or KIT-ML test sets. The inner ring represents high-level motion domains, while the outer ring shows their corresponding subcategories. Segment sizes reflect the relative number of motion classes within each subgroup

H ADDITIONAL EXAMPLES OF MOTION GENERATION USING OUR METHOD



A person gets into a stance and kicks with his right leg. Then the person does a kick spin to the left, then kicks slowly with their left leg.



A person performing a clean cartwheel. Then a man bends his right leg then kicks it out in the air. Then the person kicking/swinging legs from side to side.



A standing person performs a cartwheel starting with their right hand, then a person is performing a cartwheels. Then the figure tilts and then kicks the air above knee level with its left leg, it then does a lower kick with its right leg.



A person throws something with their left hand then kicks with their left foot before catching the object with both hands then a man jumps then kicks the air whilst moving to the opposite end of the room. Then the person takes a few steps forward kicks something with their left foot.

Figure 5: Additional Text2BFM rollouts for multi-stage prompts. Each row is one continuous rollout of the frozen BFM; the prompt is shown below the frames.