Synthesizing Physically Plausible Human Motions in 3D Scenes

Liang Pan[1] Jingbo Wang[2] Buzhen Huang[1] Junyu Zhang[1] Haofan Wang[3] Xu Tang[3] Yangang Wang[1*]
[1]Southeast University [2]Shanghai AI Laboratory [3]Xiaohongshu Inc.

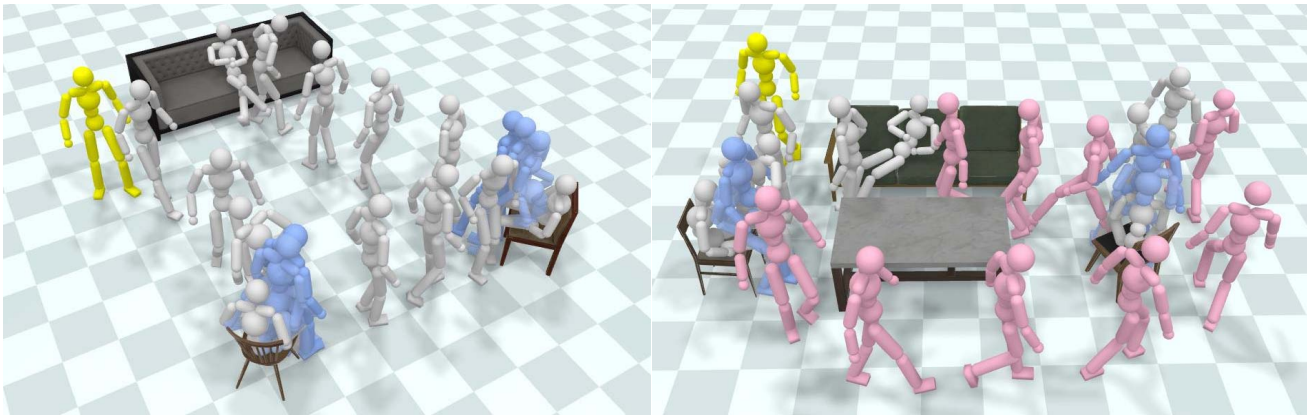

Figure 1. We propose ***InterScene***, a novel method that generates physically plausible long-term motion sequences in 3D indoor scenes. Our approach enables physics-based characters to exhibit natural interaction-involved behaviors, such as sitting down (**gray**), getting up (**blue**), and walking while avoiding obstacles (**pink**).

Abstract

We present a physics-based character control framework for synthesizing human-scene interactions. Recent advances adopt physics simulation to mitigate artifacts produced by data-driven kinematic approaches. However, existing physics-based methods mainly focus on single-object environments, resulting in limited applicability in realistic 3D scenes with multi-objects. To address such challenges, we propose a framework that enables physically simulated characters to perform long-term interaction tasks in diverse, cluttered, and unseen 3D scenes. The key idea is to decouple human-scene interactions into two fundamental processes, ***Inter****acting and* ***Nav****igating, which motivates us to construct two reusable* ***Con****trollers, namely* ***InterCon*** *and* ***NavCon****. Specifically, InterCon uses two complementary policies to enable characters to enter or leave the interacting state with a particular object (*e.g.*, sitting on a chair or getting up). To realize navigation in cluttered environments, we introduce NavCon, where a trajectory following policy enables characters to track pre-planned collision-free paths. Benefiting from the divide and conquer strategy, we can train all policies in simple environments and directly apply them in complex multi-object scenes through coordination from a rule-based scheduler. Video and code are available at https://liangpan99.github.io/InterScene/.*

1. Introduction

Creating virtual humans with various motion patterns in daily living scenarios remains a fundamental undertaking within computer vision and graphics. While previous kinematics-based works [7, 12, 14, 23, 28, 30, 37, 38, 47, 49] have achieved long-term human motion generation in 3D indoor scenes, their models are challenging to avoid inherently physical artifacts like penetration, floating, and foot sliding. Recent physics-based works [1, 8] began leveraging physics simulation and motion control techniques to enhance the physical realism of generated results. However, their frameworks remain constrained to simulated environments containing one isolated object since the policy trained with reinforcement learning (RL) has limited modeling capacity and thus results in gaps in synthesizing long-term se-

*Corresponding author. E-mail: yangangwang@seu.edu.cn. All the authors from Southeast University are affiliated with the Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Nanjing, China. This work was supported in part by the National Natural Science Foundation of China (No. 62076061), the Natural Science Foundation of Jiangsu Province (No. BK20220127).

2475-7888/24/$31.00 ©2024 IEEE
DOI 10.1109/3DV62453.2024.00149

quences in complex multi-object scenes.

To bridge the gaps, our work endeavors to enable physically simulated characters to perform long-term interaction tasks in diverse, cluttered, and unseen 3D scenes. The key insight of our framework is to construct two reusable controllers, *i.e*., **InterCon** and **NavCon**, for learning comprehensive motor skills referring to the fundamental processes in human-scene interactions. InterCon learns skills that require environmental affordances, such as sitting on a chair and rising from seated positions. NavCon considers environmental constraints to control characters' locomotion following collision-free paths. The strengths lie in 1) our framework decouples long-term interaction tasks into a scheduling problem of two controllers; 2) both controllers can be trained in relatively simple environments without relying on costly 3D scene data; 3) trained controllers can directly generalize to complex multi-object scenes without additional training.

Although existing works [1, 8] presented controllers for executing interaction tasks, they cannot be applied in multi-object scenarios due to incomplete interaction modeling. To address this issue, our InterCon employs two complementary control policies to learn complete interaction skills. Different from previous works [1, 8], InterCon not only involves reaching and interacting with the object but also includes leaving the interacting object. As illustrated in Fig 1 (left), leveraging the two policies guarantees InterCon is a closed-loop controller that realizes interactions with multiple objects. In cluttered environments with obstacles like Fig 1 (right), we introduce NavCon to enable characters to navigate and avoid obstacles. InterCon and NavCon provide complete interaction skills, including sitting, getting up, and obstacle-free trajectory following. Thus, we can leverage a finite state machine to schedule the two controllers, allowing characters to perform long-term interaction tasks in complex 3D scenes without additional training.

We train all policies by goal-conditioned reinforcement learning and adversarial motion priors (AMP) [26]. It is challenging to train InterCon stably because the policy needs to coordinate the character's fine-grained movement in relation to the object, and the reward is sparse. Previous work [8] conditions the discriminator on the scene context to construct a dense reward. In this work, we propose interaction early termination to achieve stable training via balancing data distribution in training samples. To ensure that our InterCon can generalize to unseen objects, various objects with diverse shapes are required for the training [8]. However, the get-up policy relies on various seated poses in plausible contact with objects (*e.g*., without floating and penetration), which is difficult to obtain. To tackle this issue, we introduce seated pose sampling, where a trained sit policy will generate plausible sitting poses without incurring additional costs of motion capture.

In summary, our main contributions are:

1. We propose a system that enables physically simulated characters to perform long-term interaction tasks in diverse, cluttered, and unseen 3D scenes.
2. We propose two reusable controllers for modeling interaction and navigation to decouple challenging human-scene interactions.
3. We leverage a rule-based scheduler that enables users to create human-scene interactions through intuitive instructions without additional training.

2. Related Work

Human-Scene Interaction Creating virtual characters capable of interacting with surrounding environments is widely explored in computer vision and graphics. One of the streams of methods is to build data-driven kinematic models leveraging large-scale motion capture datasets [21]. Phase-based neural networks [10, 30–32] are widely used in generating natural and realistic human motions. Holden *et al*. [10] propose PFNNs to produce motions where characters adapt to uneven terrain. Strake *et al*. [30] extend the idea of phase variables to generate motions in human-scene interaction scenarios such as sitting on chairs and carrying boxes. Strake *et al*. [31] propose a local motion phase based model for synthesizing contact-rich interactions. An alternative approach uses generative models like conditional variance autoencoder (cVAE) and motion diffusion model (MDM) [15, 35, 43, 50] to model human-scene interaction behaviors. Wang *et al*. [37] present a hierarchical generative framework to synthesize long-term 3D motion conditioning on the 3D scene structure. Hassan *et al*. [7] present a stochastic scene-aware motion generation framework using two cVAE models for learning target goal position and human motion manifolds. Wang *et al*. [38] present a framework to synthesize diverse scene-aware human motions. Taheri *et al*. [33] and Wu *et al*. [40] design similar frameworks to generate whole-body interaction motions. Most recent works adopt reinforcement learning (RL) to develop control policies for motion generation. MotionVAE [16] is a most representative work that proposes a new paradigm for motion generation based on RL and generative models. Zhang *et al*. [48] extend MotionVAE to synthesize diverse digital humans in 3D scenes. Zhao *et al*. [49] present interaction and locomotion policies to synthesize human motions in 3D indoor scenes. Lee *et al*. [14] also use reinforcement learning and motion matching to solve the locomotion, action, and manipulation task in 3D scenes. In this work, we aim to synthesize 3D motions of interacting with everyday indoor objects (*e.g*., chairs, sofas, and stools). Most relevant previous works are phase-based neural networks [30], cVAE-based generative models [7, 37, 38], and RL-based methods [14, 49].

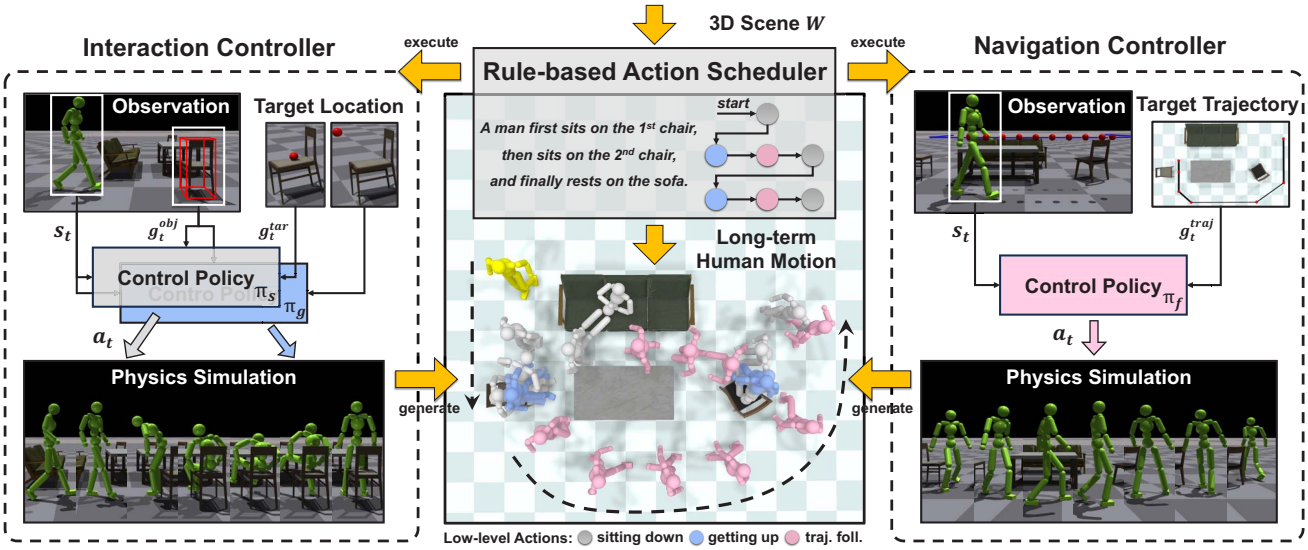

Figure 2. **System overview.** Given a multi-object 3D scene, our goal is to synthesize long-term motion sequences by controlling a physics-based character to perform a series of scene interaction tasks. First, our system employs an interaction controller to provide two primary actions, *i.e.*, sitting down and getting up. Second, we introduce a navigation controller to acquire another action, *i.e.*, collision-free trajectory following. Finally, a rule-based action scheduler is exploited to obtain outputs by organizing reusable low-level actions according to user-designed instructions.

Physics-based Character Control Physics-based methods focus on developing motion control techniques to animate characters in physics simulators [22, 36]. In recent years, various motion imitation (or motion tracking) based methods have been established to enable simulated characters to imitate diverse, challenging, and natural motor skills. Liu *et al*. [18, 19] propose sampling-based methods equipped with Covariance Matrix Adaptation (CMA) [5]. DeepMimic [24] adopts deep reinforcement learning (DRL) to train policy neural networks. Model-based methods [4, 39, 44] use supervised learning to train policies efficiently. Tracking-based methods have trained control policies to animate simulated characters for carrying box [1, 42], sports [17, 45], learning skills from videos [11, 25, 46]. However, motion tracking needs reference motions to implement an imitation objective. It can be challenging to obtain desired reference motions when policies are applied to perform new tasks that require diverse skills. Recently, Peng *et al*. [26] introduce Generative Adversarial Imitation Learning (GAIL) [9] to the character animation field and present Adversarial Motion Priors (AMP) [26]. AMP replaces the complex tracking-based objectives with a motion discriminator trained on large unstructured datasets. AMP-based works have achieved impressive results in both motion imitation and motion generation. A series of papers [2, 13, 27, 34] use AMP to learn latent skill embeddings from large motion datasets and then train a high-level policy to reuse the embeddings to solve downstream tasks. Luo *et al*. [20] propose a motion imitator that can track a large-scale motion sequences. Rempe *et al*. [29] build a pedestrian animation system using controllers trained with AMP to generate pedestrian locomotion. InterPhys [8] first extends the AMP framework to human-scene interaction tasks, such as sitting on chairs, lying on sofas, and carrying boxes. UniHSI [41] involves Large Language Models to drive simulated characters to perform scene interactions. In this work, we focus on synthesizing long-term interactions in cluttered 3D scenes.

3. Method

3.1. Preliminaries

Motion Synthesis Paradigm We achieve motion synthesis through physics-based character control. To train policies that enable characters to perform tasks in a life-like manner, we leverage the goal-conditioned RL framework of Adversarial Motion Priors (AMP) [26]. At each time step t, the policy $\pi(a_t|s_t, g_t)$ predicts an action a_t based on the character state s_t and the task-specific goal state g_t. Applied that action, the environment transitions to the next state s_{t+1} based on its dynamics $p(s_{t+1}|s_t, a_t)$. Then it receives a scalar reward r_t computed by $r = w^G r^G + w^S r^S$, where r^G is a task reward designed using human knowledge, and r^S is a naturalness reward modeled by a motion discriminator. The policy is trained to maximize the expected discounted return $J(\pi) = \mathbb{E}_{p(\tau|\pi)}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right]$, where T is the horizontal length and $\gamma \in [0, 1]$ defines the discount factor.

Character Model and State Representation The character has 15 rigid bodies, 12 movable internal joints, and 28 DoF actuators. We exploit proportional derivative (PD) controllers to convert the action $a \in \mathbb{R}^{28}$ into torques to actuate the internal joints. The root joint is not controllable. The character state $s \in \mathbb{R}^{223}$ input to the policy network is in the maximal coordinate system, including:

- Root height $s^{rh} \in \mathbb{R}^1$
- Root rotation $s^{rr} \in \mathbb{R}^6$
- Root linear velocity $s^{rv} \in \mathbb{R}^3$
- Root angular velocity $s^{ra} \in \mathbb{R}^3$
- Positions of other bodies $s^{jp} \in \mathbb{R}^{14\times3}$
- Rotations of other bodies $s^{jr} \in \mathbb{R}^{14\times6}$
- Linear velocities of other bodies $s^{jv} \in \mathbb{R}^{14\times3}$
- Angular velocities of other bodies $s^{ja} \in \mathbb{R}^{14\times3}$

The height and rotation of the root are recorded in the world coordinate frame, and other terms are recorded in the character's local coordinate frame. We use 6D rotation representations [51]. The state input to the discriminator $s \in \mathbb{R}^{125}$ is in the reduced coordinate system. Please refer to our publicly released code for more details.

Location Task As illustrated in Fig 2, our system contains three control policies: sit policy π_s, get-up policy π_g, and trajectory following policy π_f. Although they are trained for different scene interaction tasks, *i.e.*, sitting on chairs, getting up from seated states, and following trajectories, we can implement the policy training in a generic *location task*. The task objective is for the character to move its root to a target location. For instance, the target location can be formulated as a point on a chair seat that we expect the character to sit on or a point in a trajectory that the character should approach. Then, we carefully design the task settings for training each policy. In the subsequent Sec 3.3 and Sec 3.4, we will describe them in detail. Besides, we can detect whether a task is completed by measuring the distance between the root and target location, which the rule-based action scheduler uses to perform action transition.

3.2. System Overview

As illustrated in Fig 2, our complete system integrates two reusable controllers, *i.e.*, **Inter**action **Con**troller (InterCon) and **Nav**igation **Con**troller (NavCon), serving as low-level executors, and a Rule-based Action Scheduler, serving as a high-level planner to schedule two executors to synthesize human motions according to user instructions. InterCon consists of two control policies: sit policy π_s and get-up policy π_g. Each policy is conditioned on the target object features g_t^{obj} and the target location g_t^{tar} of the character's root. NavCon contains a trajectory following policy π_f conditioned on the target trajectory features g_t^{traj}. These input conditions can be seen as explicit control signals.

We use the cluttered 3D scene W shown in Fig 2, which contains three interactable objects $W = \{w_1, w_2, w_3\}$, as an example to describe the workflow of using our system to generate long-term interactions. Based on three control policies, we provide users with three reusable actions: sitting down k_s, getting up k_g, and trajectory following k_f. To synthesize human motions described by "A man first sits on the 1st chair, then sits on the 2nd chair, and finally rests on the sofa", the user needs to construct the following instruction:

$$\begin{aligned} I = \{&(k_s, w_1), \\ &(k_g, w_1), (k_f, h_1), (k_s, w_2), \\ &(k_g, w_2), (k_f, h_2), (k_s, w_3)\}, \end{aligned} \tag{1}$$

where $(k_s/k_g, w)$ denotes that the character performs the action k_s to sit on the object w or performs the action k_g to get up from the object w, and (k_f, h) denotes that the character walks along an obstacle-free trajectory h that can be either generated by the A* path planning algorithm [6] or defined by the user. The rule-based action scheduler translates this instruction into a sequence of explicit control signals and then schedules low-level policies to execute the instruction. It is also responsible for performing action transitions. For instance, an action will be terminated when the overlapping time between the character's root and its current target outperforms a fixed time.

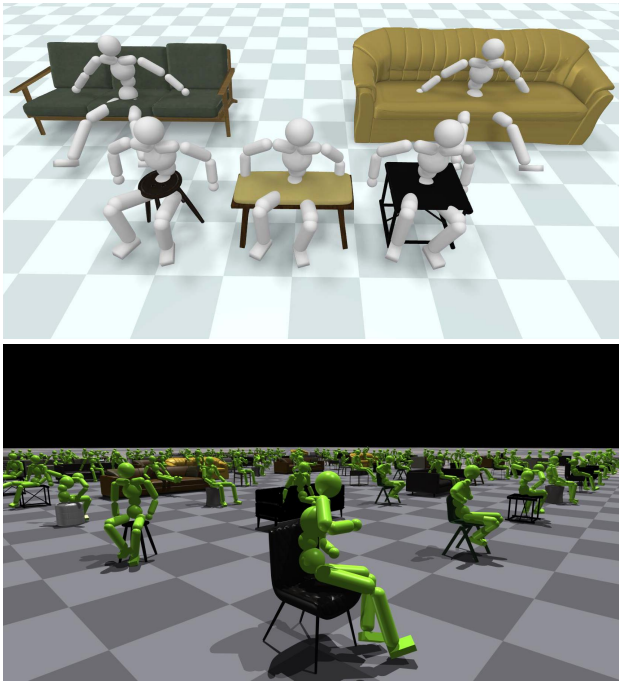

Figure 3. Seated poses sampled from the reference dataset (upper row) and generated by pre-trained sit policy (lower row).

3.3. Interaction Controller

Given a target object $w_i \in W$, our InterCon aims to enable characters to enter or leave the interacting state with the object. Previous methods [1, 8] mainly focus on developing the former ability for the character, overlooking the latter's importance for long-term interaction tasks. By incorporating a new skill of getting up, the controller allows the character to transition from a seated state to a standing state, which provides an opportunity to perform the next new interaction task.

Task-specific Goal State As illustrated in Fig 2, we construct InterCon using two separate policies. They share the same character state s_t, task-specific goal state $g_t = \{g_t^{obj}, g_t^{tar}\}$, and network structure. We input the target object's features $g_t^{obj} \in \mathbb{R}^{3+6+2+24}$ that contain the object's position and rotation, the horizontal vector of its facing direction, and 8 vertices of its bounding box to the policy to enable it to be aware of the target object state. We also follow [8] to condition the discriminator on the object state g_t^{obj}, which is crucial for policy to effectively learn how to coordinate the movement of a character concerning the target. In addition, we add explicit target location $g_t^{tar} \in \mathbb{R}^3$ into the goal features g_t. All these goal features are recorded in the character's local frame. The target location of the sit task is generated before training and placed 10 cm above the center of the top surface of the object seat. The target location of the get-up task is computing online according to the states of feet bodies. Please refer to our publicly released code for more details.

Motion and Object Datasets We use 111 motion sequences from the SAMP dataset [7] containing multiple behaviors (walking, sitting, and getting up) and diverse human-object configurations. We manually split the motion dataset into two subsets used to train the sit and get-up policies, respectively. To construct the object dataset, we select 40 straight chairs, 40 low stools, and 40 sofas from the 3D-Front object dataset [3] and randomly divide 30 as the training set and 10 as the testing set. We coarsely fit objects to human motions since we condition the discriminator on the object state.

Initialization At the beginning of each episode, we need to initialize them appropriately to facilitate the training process. For the sit task, we utilize Reference State Init (RSI) [24] to sample interaction states from the previously fitted dataset randomly. As shown in Fig 3, partial frames in a seated state will penetrate with objects, which will hurt physics simulation. Therefore, we pass these frames during RSI. We also follow [8] to encourage the character to execute the sit task from a wide range of initial configurations by randomizing the position and rotation of objects. For the get-up task, we should start training from

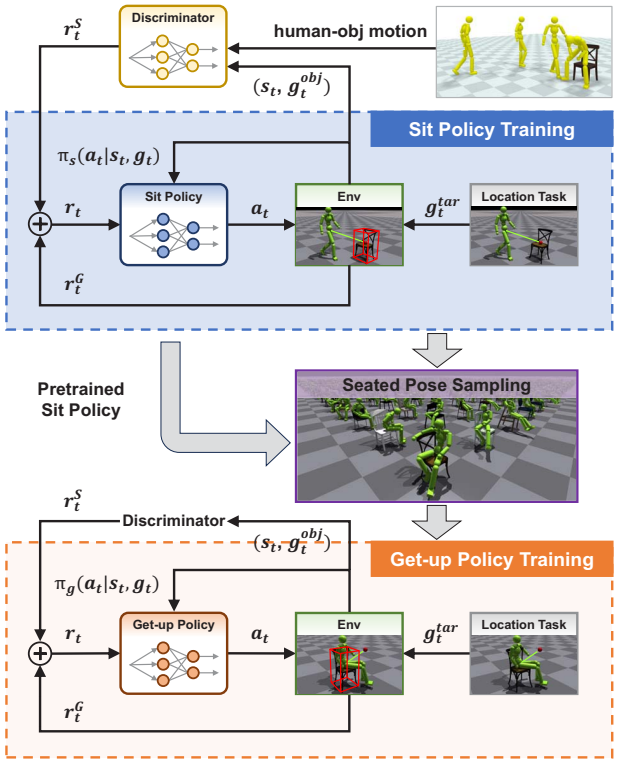

Figure 4. **The process for training the sit and get-up policies** consists of three steps. **1) Sit Policy Training:** Inspired by [8], we first extend the standard AMP framework with several improvements to train a robust sit policy. **2) Seated Pose Sampling:** We tackle the issue of lacking high-quality seated human poses adapted to various object shapes by using the pre-trained sit policy to generate numerous seated poses randomly. **3) Get-up Policy Training:** We adopt a similar method to train the get-up policy. At the beginning of each training episode, the character will be initialized to a seated state sampled from the previously synthesized database.

seated states. However, it is difficult to obtain reasonable states from reference. Moreover, constructing an accurately aligned dataset is also costly. Thus, we build a synthetic database, illustrated in Fig 3, consisting of plenty of seated poses with high-quality contact. It will be further discussed in subsequent paragraphs.

Reset and Early Termination An episode terminates after a fixed episode length or when early termination conditions have been triggered. The episode length is set to 10 seconds. We use fall detection as a basic condition. In order to improve sampling efficiency, we introduce interaction early termination (IET). IET triggers when the accumulative overlapping time between the character's root x_t^{root} and the target location g_t^{tar} exceeds a fixed threshold. Such a simple mechanism can effectively assist the RL training process.

Training Scheme of Two Policies The training process is detailed in Fig 4. There is a dependency graph among the sitting and getting up behaviors. We need to initialize the character to a plausible seated state at the beginning of each episode to train the get-up policy. However, collecting such a diverse set of initial states is hard and costly. Thus, our training scheme starts upstream in the dependency graph to train a sit policy first. Subsequently, we directly employ it to perform seated pose sampling to obtain high-quality interaction states with no floating and penetration. We collect 300 poses for each training object. In total, we generate 300×90 poses to form a synthetic database. A snapshot of partial data is shown in Fig 3. Then, we train the get-up policy from scratch and keep most of the settings. Such a database is the key to successfully training the get-up policy.

Task Rewards Given a target location g_t^{tar} and a target velocity $g_t^{vel} = 1.5$m/s, the task reward for sit is defined as:

$$r_t^G = \begin{cases} 0.7\, r_t^{near} + 0.3\, r_t^{far}, \left\| x_t^* - x_t^{root} \right\|^2 > 0.5 \\ 0.7\, r_t^{near} + 0.3, \text{otherwise} \end{cases} \tag{2}$$

$$r_t^{far} = \exp\big(-2.0 \left\| g_t^{vel} - d_t^* \cdot \dot{x}_t^{root} \right\|^2 \big) \tag{3}$$

$$r_t^{near} = \exp\big(-10.0 \left\| g_t^{tar} - x_t^{root} \right\|^2 \big) \tag{4}$$

where x_t^{root} is the position of character's root, $\dot{x}_t^{root}$ is the linear velocity of character's root, x_t^* is the position of the object, d_t^* is a horizontal unit vector pointing from x_t^{root} to x_t^*, $a \cdot b$ represents vector dot product. We only use the r_t^{near} as the task reward for the get-up task.

3.4. Navigation Controller

While InterCon has been able to generate long-term interactions, it cannot avoid obstacles. To learn a necessary skill in cluttered environments, *i.e.*, collision-free navigation, we introduce NavCon to our system, which contains a trajectory following policy $\pi_f(a_t|s_t, g_t^{traj})$ [29] and off-the-shelf path planning algorithms, as shown in Fig 2. Incorporating a module for obtaining collision-free locomotion has been widely used in previous works [7, 38, 49]. However, kinematics-based methods produce artifacts like foot skating and penetration. Physics-based Pacer [29] can generate physically plausible gaits and contacts. In this work, we apply this technique in the character-scene interaction field to solve the navigation problem.

Task-specific Goal State We construct the goal features using a short future path $g_t^{traj} \in \mathbb{R}^{10\times2}$ consisting of a sequence of 2D target positions of the character's root for the next 1.0 seconds sampled at 0.1s intervals.

Motion and Trajectory Datasets We use ~ 200 sequences from the AMASS dataset [21]. Target trajectories used for training are procedurally generated. A complete trajectory $\tau = \{p_0^\tau, ..., p_{T-1}^\tau, p_T^\tau\}$ is modeled as a set of 2D points with a fixed 0.1 seconds time interval. At each simulation time step t, we query 10 points $\{p_t^\tau, ..., p_{t+9}^\tau\}$ in the future 1.0 seconds from the complete trajectory τ by interpolating.

Other Settings We terminate the training episode when the character falls or deviates too far from the established trajectory. Characters are initialized using RSI. Trajectories will be re-generated when environments reset. The task reward r_t^G measures how far away the root x_t^{root} is on the horizontal plane from the desired location $p_t^\tau \in \tau$:

$$r_t^G = \exp\big(-2.0 \left\| x_t^{root} - p_t^\tau \right\|^2 \big). \tag{5}$$

Method	Success Rate (%)	Execution Time (s)	Error (mm)
Chao *et al.* [1]	17.0	–	–
AMP [26]	87.5	4.0	**36.5**
InterPhys [8]	93.7	3.7	90.0
Ours	**98.8**	**2.5**	36.8

Table 1. Comparisons with physics-based methods on sit task.

IET Step	Success Rate (%)	Execution Time (s)	Error (mm)
30	**98.8**	**2.5**	36.8
60	93.9	3.1	**34.9**
90	92.3	3.3	36.3
×	87.5	4.0	36.5

Table 2. Metrics of sit policies trained with various early termination settings. × means that the model is trained without IET.

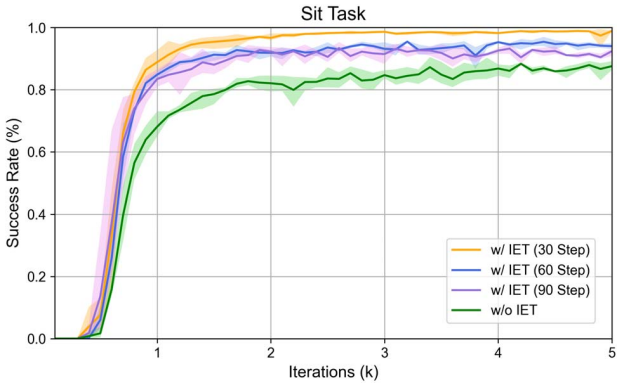

Figure 5. Performance curves of sit policies trained with various early termination settings. Colored regions denote the fluctuation range over 3 models.

4. Experiment

4.1. Individual Tasks

The interaction controller has two policies responsible for performing sit and get-up tasks. We first quantitatively demonstrate the effectiveness of each policy in its corresponding task. Then, we conduct ablation studies on the

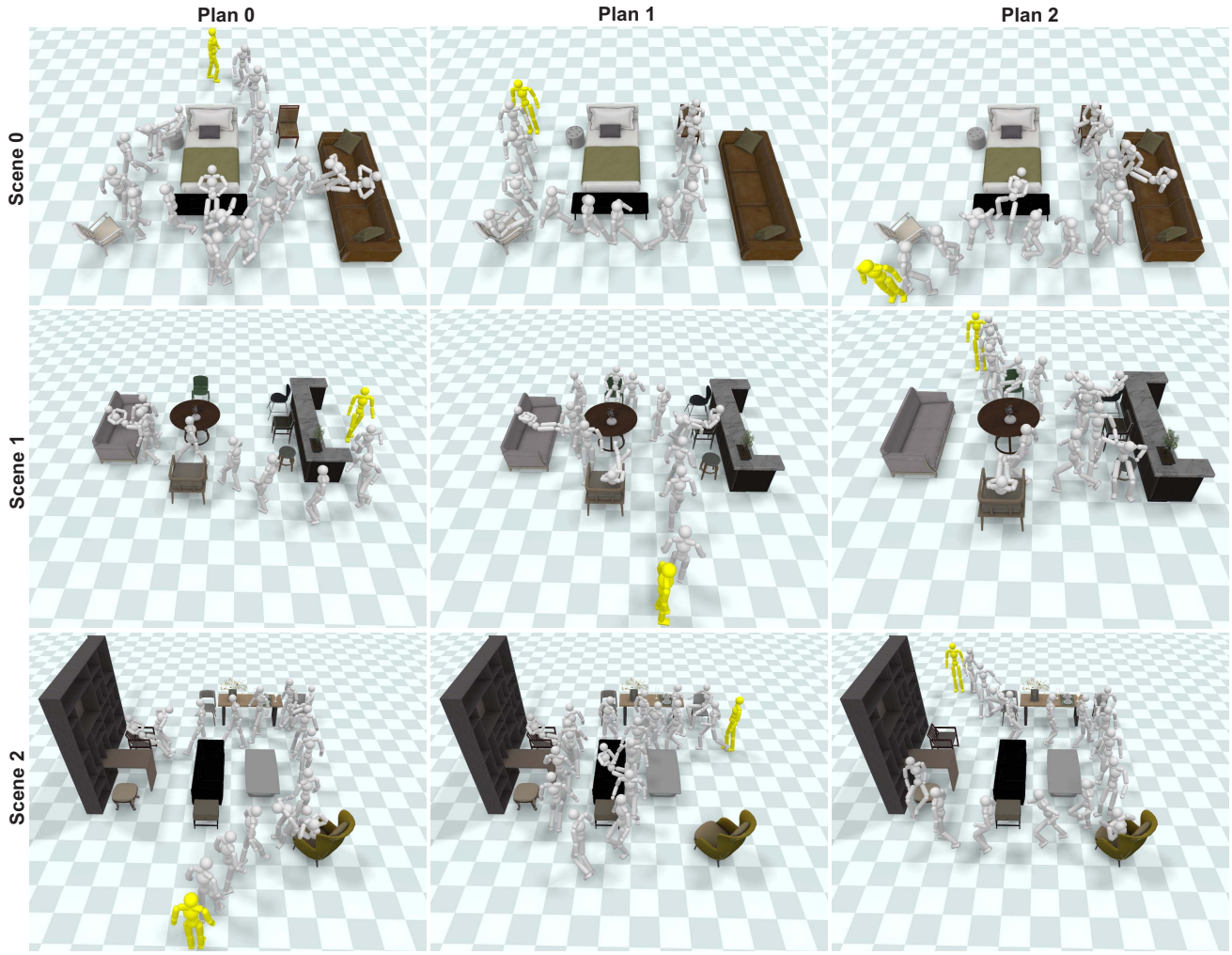

Figure 6. Our complete system successfully generates long-term motion sequences in three challenging 3D indoor scenes.

interaction early termination (IET). All experimental results are collected in single-object environments in which the object is randomly sampled from the object testing set. We follow [8] to comprehensively evaluate the task performance by measuring success rate, execution time, and error. A trial will be determined to be successful if the Euclidean distance between the character's root and its target location is less than 20 cm. To mitigate randomness, we train 3 models initialized with different random seeds and evaluate each model using 4096 trials for each experiment setting. Thus, all metrics are averaged over 3×4096 trials.

Sit Task Table 1 summarizes the quantitative comparisons between our sit policy and existing physics-based methods. We use AMP [26] as baseline to train a policy where IET is not employed, and the discriminator's obs does not include object states. Compared with InterPhys [8], our training involves more detailed object information and a new termination strategy (the IET step parameter is set to 30). During testing, the object is placed $[1, 5]$ m away from the character with a random orientation. Experimental results show that our method achieves a high success rate of 98.8% and significantly outperforms others. Using IET can focus the distribution of samples collected during RL training on approaching the object and sitting down. We demonstrate that IET can improve performance effectively.

Get-up Task We first use a pre-trained sit policy to collect 300 seated states of a given testing object. These states contain various facing directions and body poses. When testing, the character is initialized to a random seated state. Our get-up policy can achieve a success rate of 94.6%, which demonstrates its effectiveness.

Ablation Studies on IET For a more in-depth analysis of the impact of IET, we construct 4 variants equipped with various early termination settings, including *w/o IET* and

3D Scene	Plan 0	Plan 1	Plan 2
0	92.1	99.2	82.0
1	98.4	88.2	90.6
2	57.0	71.8	57.8

Table 3. Success rate metrics of the complete system in final application environments. Each synthesis plan is tested by 128 trials.

w/ IET with varied step parameters. Metrics in Table 2 suggest that task performance gradually improves as the step parameter decreases. Fig 5 also illustrates that using a small step parameter can effectively reduce the fluctuation margin to stabilize the training process.

4.2. Long-term Motion Synthesis

The ultimate goal of this work is to synthesize long-term motions involving multiple actions in diverse and cluttered 3D scenes. We quantitatively and qualitatively evaluate the effectiveness of our complete system in multi-object environments. 3 synthetic scenes are constructed using unseen objects from the 3D-Front dataset [3]. Each scene is populated with $5 \sim 6$ interactable objects and some obstacles. For each scene, we design 3 plans manually, each of which contains a sequence of actions for the character to perform.

Experimental Results As shown in Fig 6, our complete system can successfully synthesize desired long-term motions. It validates the proposed divide and conquer idea that the policies trained in simple environments can generalize to unseen complex 3D scenes. We also provide quantitative metrics of success rate for all plans in Table 3, which shows that performance is unstable and highly relevant to the complexity of the scene and plan.

Ablation Studies on NavCon Our system relies on NavCon to avoid obstacles. We validate its importance by qualitatively comparing motions generated by the complete system and a variant without NavCon. As illustrated in Fig 7, given the same target, the former can successfully control the character to interact with the object. The latter gets the character stuck by obstacles because InterCon tends to follow the shortest path to the target.

5. Limitations and Future Work

This work aims to design a physics-based animation system to synthesize motions in complex 3D scenes and lets the system come true. However, many valuable problems remain that should be investigated in future work. Firstly, since the system is built on AMP, policies are prone to master a small subset of behaviors depicted in the motion dataset. To tackle this issue, we can explore how to introduce conditional generation capabilities [2, 27] into the system to improve skill diversity. Secondly, the system cannot handle with abnormal situations that would occur when executing tasks in complex scenes. For example, NavCon tracks a pre-planned and fixed path. Due to tracking errors, it is inevitable for the character to deviate from the path and collide with obstacles. This is also the main reason why the performance is unstable in Table 3. Creating a closed-loop system with periodic re-planning features will be necessary to improve the robustness. Thirdly, our policy only observes the state of the isolated target object and is unaware of the surroundings. Future work should explore incorporating more global representations of the cluttered environment. In addition, the real world has more complex environments, such as dynamic scenes, uneven terrains, multi-floor buildings, and narrow spaces, which are not considered in this work. It would be exciting to create autonomous agents living in a physically simulated environment with the above characteristics.

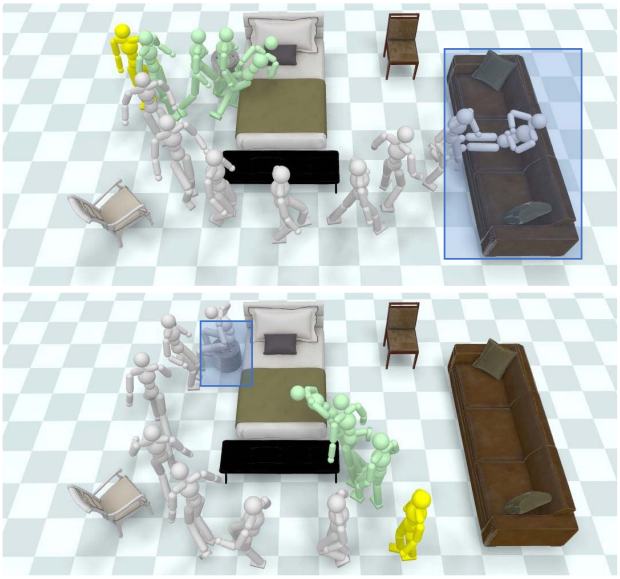

Figure 7. Comparisons of the complete system (gray) and a variant without NavCon (green). Blue rectangles denote target objects.

6. Conclusion

In this paper, we presented a physics-based animation system to synthesize long-term human motions in diverse, cluttered, and unseen 3D scenes. This is achieved by jointly using two reusable controllers, *i.e.*, InterCon and NavCon, as well as a rule-based action scheduler, to decompose the human-scene interactions. We introduced some training techniques to train the policies of InterCon successfully. We qualitatively and quantitatively evaluated our complete system's long-term motion generation ability. Our physically simulated characters realistically and naturally presented interaction and locomotion behaviors.

References

[1] Yu-Wei Chao, Jimei Yang, Weifeng Chen, and Jia Deng. Learning to sit: Synthesizing human-chair interactions via hierarchical control. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5887–5895, 2021. 1, 2, 3, 5, 6

[2] Zhiyang Dou, Xuelin Chen, Qingnan Fan, Taku Komura, and Wenping Wang. C· ase: Learning conditional adversarial skill embeddings for physics-based characters. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3, 8

[3] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10933–10942, 2021. 5, 8

[4] Levi Fussell, Kevin Bergamin, and Daniel Holden. Supertrack: Motion tracking for physically simulated characters using supervised learning. *ACM Transactions on Graphics (TOG)*, 40(6):1–13, 2021. 3

[5] Nikolaus Hansen. The cma evolution strategy: a comparing review. *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, pages 75–102, 2006. 3

[6] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4 (2):100–107, 1968. 4

[7] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael J Black. Stochastic scene-aware motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11374–11384, 2021. 1, 2, 5, 6

[8] Mohamed Hassan, Yunrong Guo, Tingwu Wang, Michael Black, Sanja Fidler, and Xue Bin Peng. Synthesizing physical character-scene interactions. In *ACM SIGGRAPH 2023 Conference Proceedings*, New York, NY, USA, 2023. Association for Computing Machinery. 1, 2, 3, 5, 6, 7

[9] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, page 4572–4580, Red Hook, NY, USA, 2016. Curran Associates Inc. 3

[10] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 36(4):1–13, 2017. 2

[11] Buzhen Huang, Liang Pan, Yuan Yang, Jingyi Ju, and Yangang Wang. Neural mocon: Neural motion control for physically plausible human motion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022. 3

[12] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16750–16761, 2023. 1

[13] Jordan Juravsky, Yunrong Guo, Sanja Fidler, and Xue Bin Peng. Padl: Language-directed physics-based character control. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 3

[14] Jiye Lee and Hanbyul Joo. Locomotion-action-manipulation: Synthesizing human-scene interactions in complex 3d environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9663–9674, 2023. 1, 2

[15] Jiaman Li, Jiajun Wu, and C Karen Liu. Object motion guided human motion synthesis. *ACM Transactions on Graphics (TOG)*, 42(6):1–11, 2023. 2

[16] Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. Character controllers using motion vaes. *ACM Transactions on Graphics (TOG)*, 39(4):40–1, 2020. 2

[17] Libin Liu and Jessica Hodgins. Learning basketball dribbling skills using trajectory optimization and deep reinforcement learning. *ACM Trans. Graph.*, 37(4), 2018. 3

[18] Libin Liu, KangKang Yin, Michiel Van de Panne, Tianjia Shao, and Weiwei Xu. Sampling-based contact-rich motion control. In *ACM SIGGRAPH 2010 papers*, pages 1–10. 2010. 3

[19] Libin Liu, KangKang Yin, and Baining Guo. Improving sampling-based motion control. In *Computer Graphics Forum*, pages 415–423. Wiley Online Library, 2015. 3

[20] Zhengyi Luo, Jinkun Cao, Alexander Winkler, Kris Kitani, and Weipeng Xu. Perpetual humanoid control for real-time simulated avatars. *arXiv preprint arXiv:2305.06456*, 2023. 3

[21] 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*, pages 5442–5451, 2019. 2, 6

[22] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021. 3

[23] Aymen Mir, Xavier Puig, Angjoo Kanazawa, and Gerard Pons-Moll. Generating continual human motion in diverse 3d scenes. In *International Conference on 3D Vision (3DV)*, 2024. 1

[24] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel Van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions On Graphics (TOG)*, 37(4):1–14, 2018. 3, 5

[25] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. Sfv: reinforcement learning of physical skills from videos. *ACM Trans. Graph.*, 37(6), 2018. 3

[26] 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 (ToG)*, 40(4):1–20, 2021. 2, 3, 6, 7

[27] 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 (TOG)*, 41(4):1–17, 2022. 3, 8

[28] Huaijin Pi, Sida Peng, Minghui Yang, Xiaowei Zhou, and Hujun Bao. Hierarchical generation of human-object interactions with diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15061–15073, 2023. 1

[29] Davis Rempe, Zhengyi Luo, Xue Bin Peng, Ye Yuan, Kris Kitani, Karsten Kreis, Sanja Fidler, and Or Litany. Trace and pace: Controllable pedestrian animation via guided trajectory diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13756–13766, 2023. 3, 6

[30] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Trans. Graph.*, 38(6):209–1, 2019. 1, 2

[31] Sebastian Starke, Yiwei Zhao, Taku Komura, and Kazi Zaman. Local motion phases for learning multi-contact character movements. *ACM Transactions on Graphics (TOG)*, 39 (4):54–1, 2020. 2

[32] Sebastian Starke, Ian Mason, and Taku Komura. Deepphase: Periodic autoencoders for learning motion phase manifolds. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 2

[33] Omid Taheri, Vasileios Choutas, Michael J Black, and Dimitrios Tzionas. Goal: Generating 4d whole-body motion for hand-object grasping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13263–13273, 2022. 2

[34] 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*, pages 1–9, 2023. 3

[35] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 2

[36] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012. 3

[37] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9401–9411, 2021. 1, 2

[38] Jingbo Wang, Yu Rong, Jingyuan Liu, Sijie Yan, Dahua Lin, and Bo Dai. Towards diverse and natural scene-aware 3d human motion synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20460–20469, 2022. 1, 2, 6

[39] Jungdam Won, Deepak Gopinath, and Jessica Hodgins. Physics-based character controllers using conditional vaes. *ACM Transactions on Graphics (TOG)*, 41(4):1–12, 2022. 3

[40] Yan Wu, Jiahao Wang, Yan Zhang, Siwei Zhang, Otmar Hilliges, Fisher Yu, and Siyu Tang. Saga: Stochastic whole-body grasping with contact. In *European Conference on Computer Vision*, pages 257–274. Springer, 2022. 2

[41] Zeqi Xiao, Tai Wang, Jingbo Wang, Jinkun Cao, Bo Dai, Dahua Lin, and Jiangmiao Pang. Unified human-scene interaction via prompted chain-of-contacts. *Arxiv*, 2023. 3

[42] Zhaoming Xie, Jonathan Tseng, Sebastian Starke, Michiel van de Panne, and C Karen Liu. Hierarchical planning and control for box loco-manipulation. *arXiv preprint arXiv:2306.09532*, 2023. 3

[43] Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14928–14940, 2023. 2

[44] Heyuan Yao, Zhenhua Song, Baoquan Chen, and Libin Liu. Controlvae: Model-based learning of generative controllers for physics-based characters. *ACM Transactions on Graphics (TOG)*, 41(6):1–16, 2022. 3

[45] Zhiqi Yin, Zeshi Yang, Michiel Van De Panne, and Kangkang Yin. Discovering diverse athletic jumping strategies. *ACM Trans. Graph.*, 40(4), 2021. 3

[46] Haotian Zhang, Ye Yuan, Viktor Makoviychuk, Yunrong Guo, Sanja Fidler, Xue Bin Peng, and Kayvon Fatahalian. Learning physically simulated tennis skills from broadcast videos. *ACM Trans. Graph.*, 42(4), 2023. 3

[47] Xiaohan Zhang, Bharat Lal Bhatnagar, Sebastian Starke, Vladimir Guzov, and Gerard Pons-Moll. Couch: Towards controllable human-chair interactions. In *European Conference on Computer Vision*, pages 518–535. Springer, 2022. 1

[48] Yan Zhang and Siyu Tang. The wanderings of odysseus in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20481–20491, 2022. 2

[49] Kaifeng Zhao, Yan Zhang, Shaofei Wang, Thabo Beeler, and Siyu Tang. Synthesizing diverse human motions in 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14738–14749, 2023. 1, 2, 6

[50] Wenyang Zhou, Zhiyang Dou, Zeyu Cao, Zhouyingcheng Liao, Jingbo Wang, Wenjia Wang, Yuan Liu, Taku Komura, Wenping Wang, and Lingjie Liu. Emdm: Efficient motion diffusion model for fast, high-quality motion generation. *arXiv preprint arXiv:2312.02256*, 2023. 2

[51] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. 4