Parameterized Quasi-Physical Simulators for Dexterous Manipulations Transfer

Anonymous ECCV 2024 Submission

Paper ID #10032

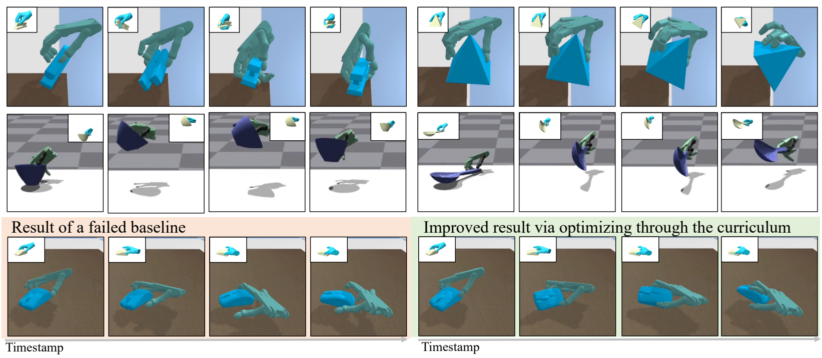

Fig. 1: By optimizing through a **quasi-physical simulator curriculum**, we successfully transfer human demonstrations to dexterous robot hand simulations. We enable accurate tracking of complex manipulations with changing contacts (*Fig. (a)*), non-trivial object motions (*Fig. (b)*) and intricate tool-using (*Fig. (c,d)*). Besides, our physics curriculum can substantially improve the performance of a failed baseline as well (*Fig. (e,f)*).

Abstract. We explore the dexterous manipulation transfer problem by designing simulators. The task wishes to transfer human manipulations to dexterous robot hand simulations and is inherently difficult due to its intricate, highly-constrained, and discontinuous dynamics and the need to control a dexterous hand with a DoF to accurately replicate human manipulations. Previous approaches that optimize in high-fidelity black-box simulators or a modified one with relaxed constraints only demonstrate limited capabilities or are restricted by insufficient simulation fidelity. We introduce **parameterized quasi-physical simulators** and a **physics curriculum** to overcome these limitations. The key ideas are 1) balancing between fidelity and optimizability of the simulation via a curriculum of parameterized simulators, and 2) solving the problem in each of the simulators from the curriculum, with properties ranging from high task optimizability to high fidelity. We successfully enable a dexterous hand to track complex and diverse manipulations in high-fidelity simulated environments, boosting the success rate by 11%+ from the best-performed baseline. We include a website to introduce the work.

Keywords: Dexterous Manipulation Transfer · Hybrid Simulation

1 Introduction

Advancing an embodied agent's capacity to interact with the world represents a significant stride toward achieving general artificial intelligence. Due to the substantial costs and potential hazards of setting up real robots to do trial and error, the standard approach for developing embodied algorithms involves learning in physical simulators [9, 15, 23, 25, 33, 56, 59] before transitioning to real-world deployment. In most cases, physical simulators are treated as black boxes, and extensive efforts have been devoted to developing learning and optimization methods for embodied skills within these black boxes. Despite the considerable progress [2,6–8,16,20,21,31,36,39,43,46,60,62,66,68], the question like whether the simulators used are the most suitable ones is rarely discussed. In this work, we investigate this issue and illustrate how optimizing the simulator concurrently with skill acquisition can benefit a popular yet challenging task in robot manipulation – dexterous manipulation transfer.

The task aims at transferring human-object manipulations to a dexterous robot hand, enabling it to physically track the reference motion of both the hand and the object (see Fig. 1). It is challenged by 1) the complex, highly constrained, non-smooth, and discontinuous dynamics with frequent contact establishment and breaking involved in the robot manipulation, 2) the requirement of precisely controlling a dexterous hand with a high DoF to densely track the manipulation at each frame, and 3) the morphology difference. Some existing works rely on high-fidelity black-box simulators, where a small difference in robot control can result in dramatically different manipulation outcomes due to abrupt contact changes, making the tracking objective highly non-smooth and hard to optimize [4,6,8,43,46]. In this way, their tasks are restricted to relatively simple goal-driven manipulations such as pouring and re-locating [8,43,46,68], in-hand re-orientation, flipping and spinning [4,6] with a fixed-root robot hand, or manipulating objects with simple geometry such as balls [36]. Other approaches attempt to improve optimization by relaxing physical constraints, with a primary focus on smoothing out contact responses [3,24,38,55,56]. However, their dynamics models may significantly deviate from real physics [38], hindering skill deployment. Consequently, we ask how to address the optimization challenge while preserving the high fidelity of the simulator.

Our key insight is that a single simulator can hardly provide both high fidelity and excellent optimizability for contact-rich dexterous manipulations. Inspired by the line of homotopy methods [12, 28, 29, 61], we propose a curriculum of simulators to realize this. We start by utilizing a quasi-physical simulator to initially relax physical constraints and warm up the optimization. Subsequently, we transfer the optimization outcomes to simulators with gradually tightened physical constraints. Finally, we transition to a physically realistic simulator for skill deployment in realistic dynamics.

To realize this vision, we propose **a family of parameterized quasi-physical simulators** for contact-rich dexterous manipulation tasks. These simulators can be customized to enhance task optimizability while can also be tailored to approximate realistic physics. The parameterized simulator represents an articulated multi rigid body as a parameterized point set, models contact using an unconstrained parameterized spring-damper, and compensates for unmodeled effects via parameterized residual physics. Specifically, the articulated multi-body dynamics model is relaxed as the point set dynamics model. An articulated object is relaxed into a set of points, sampled from the ambient space surrounding each body's surface mesh. The resulting dynamics model combines the original articulated dynamics with the mass-point dynamics of each individual point. Parameters are introduced to control the point set construction and the dynamics model. The contact model is softened as a parameterized spring-damper model [3, 19, 35, 38, 51] with parameters introduced to control when to calculate contacts and contact spring stiffness. The residual physics network compensate for unmodeled effects from the analytical modeling [22]. The parameterized simulator can be programmed for high optimizability by relaxing constraints in the analytical model and can be tailored to approximate realistic physics by learning excellent residual physics. We demonstrate that the challenging dexterous manipulation transfer task can be effectively addressed through curriculum optimization using a series of parameterized physical simulators. Initially, both articulated rigid constraints and the contact model stiffness are relaxed in the simulator. It may not reflect physical realism but provides a good environment where the manipulation transfer problem can be solved easily. Subsequently, the articulated rigid constraints and the contact model are gradually tightened. Task-solving proceeds iteratively within each simulator in the curriculum. Finally, the parameterized simulator is optimized to approximate realistic physics. Task optimization continues, yielding a dexterous hand trajectory capable of executing the manipulation in environments with realistic physics.

We demonstrate the superiority of our method and compare it with previous model-free and model-based methods on challenging manipulation sequences from three datasets, describing single-hand or bimanual manipulations with daily objects or using tools. We conduct dexterous manipulation transfer on two widely used simulators, namely Bullet [9] and Isaac Gym [33] to demonstrate the generality and the efficacy of our method and the capability of our quasi-physical simulator to approximate the unknown black-box physics model in the contact-rich manipulation scenario (Fig. 1). We can track complex manipulations involving non-trivial object motions such as large rotations and complicated tool-using such as using a spoon to bring the water back and forth. Our approach successfully surpasses the previous best-performed method both quantitatively and qualitatively, achieving more than 11% success rate than the previous best-performed method. Besides, optimizing through the physics curriculum can significantly enhance the performance of previously under-performed RL-based methods, almost completing the tracking problem from failure, as demonstrated in Fig. 1. This indicates the universality of our approach to embodied AI through optimization via a physics curriculum. Thorough ablations are conducted to validate the efficacy of our designs.

Our contributions are three-fold:

- We introduce a family of parameterized quasi-physical simulators that can be configured to relax various physical constraints, facilitating skill optimization, and can also be tailored to achieve high simulation fidelity.
- We present a quasi-physics curriculum along with a corresponding optimization method to address the challenging dexterous manipulation transfer problem.
- Extensive experiments demonstrate the effectiveness of our method in transferring complex manipulations, including non-trivial object motions and changing contacts, to a dexterous robot hand in simulation.

2 Related Works

Dexterous manipulation transfer. Transferring human manipulations to dexterous robot-hand simulations is an important topic in robot skill acquisition [8,21,31,43,60,62,68,70]. Most approaches treat the simulator as black-box physics models and try to learn skills directly from that [4,6,8,43,46]. However, their demonstrated capabilities are restricted to relatively simple tasks. Another trend of work tries to relax the physics model [37,38] to create a better environment for task optimization. However, due to the disparity between their modeling approach and realistic physics, successful trials are typically demonstrated only in their simulators, which can hardly complete the task under physically realistic dynamics. In this work, we introduce various parameterized analytical relaxations to improve the task optimizability while compensating for unmodeled effects via residual physics networks so the fidelity would not be sacrificed.
Learning for simulation. Analytical methods can hardly approximate an extremely realistic physical world despite lots of smart and tremendous efforts made in developing numerical algorithms [19, 23, 26, 27]. Recently, data-driven approaches have attracted lots of interest for their high efficiency and strong approximation ability [10, 11, 22, 40, 41, 50, 63]. Special network designs are proposed to learn the contact behaviour [22,41]. We in this work propose to leverage an analytical-neural hybrid approach and carefully design network modules for approximating residual contact forces in the contact-rich manipulation scenario.
Sim-to-Sim and Sim-to-Real transfer. The field of robot manipulation continues to face challenges in the areas of Sim2Sim and Sim2Real transferability [71]. Considering the modeling gaps, the optimal strategy learned in a specific simulator is difficult to transfer to a different simulator or the real world. Therefore, many techniques for solving the problem have been proposed, including imitation learning [34,42,43,45,46,48], transfer learning [72], distillation [47,57], residual physics [17,67], and efforts on bridging the gap from the dynamics model aspect [22,69]. Our parameterized simulators learn residual physics involved in contact-rich robot manipulations. By combining an analytical base with residual networks, we showcase their ability to approximate realistic physics.

3 Method

Given a human manipulation demonstration, composed of a human hand mesh trajectory and an object pose trajectory $\{\mathcal{H}, \mathcal{O}\}$, the goal is transferring the

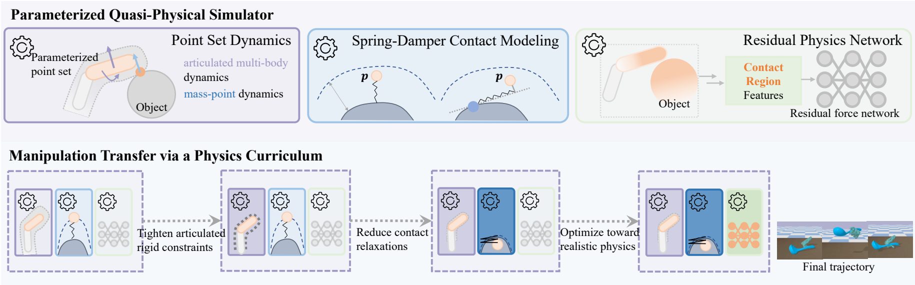

Fig. 2: The **parameterized quasi-physical simulator** relaxes the articulated multi rigid body dynamics as the *parameterized point set dynamics*, controls the contact behavior via an unconstrained *parameterized spring-damper contact model*, and compensates for unmodeled effects via *parameterized residual physics networks*. We tackle the difficult dexterous manipulation transfer problem via **a physics curriculum**.

demonstration to a dexterous robot hand in simulation. Formally, we aim to optimize a control trajectory $\mathcal{A}$ that drives the dexterous hand to manipulate the object in a realistic simulated environment so that the resulting hand trajectory $\hat{\mathcal{H}}$ and the object trajectory $\hat{\mathcal{O}}$ are close to the reference motion $\{\mathcal{H}, \mathcal{O}\}$. The problem is challenged by difficulties from the highly constrained, discontinuous, and non-smooth dynamics, the requirement of controlling a high DoF dexterous hand for tracking, and the morphology difference.

Our method comprises two key designs to tackle the challenges: 1) a family of parameterized quasi-physical simulators, which can be programmed to enhance the optimizability of contact-rich dexterous manipulation tasks and can also be tailored to approximate realistic physics (Section 3.1), and 2) a physics curriculum that carefully adjusts the parameters of a line of quasi-physical simulators and a strategy that solves the difficult dexterous manipulation transfer task by addressing it within each simulator in the curriculum (Section 3.2).

3.1 Parameterized Quasi-Physical Simulators

Our quasi-physical simulator represents an articulated multi-body, *i.e.*, the robotic dexterous hand, as a point set. The object is represented as a signed distance field. The base of the simulator is in an analytical form leveraging an unconstrained spring-damper contact model. Parameters are introduced to control the analytical relaxations on the articulated rigid constraints and the softness of the contact model. Additionally, neural networks are introduced to compensate for unmodeled effects beyond the analytical framework. We will elaborate on each of these design aspects below.

Parameterized point set dynamics. Articulated multi-body represented in the reduced coordinate system [19,59] may require a large change in joint states to achieve a small adjustment in the Euclidean space. Moving the end effector from one point to a nearby point may require adjusting all joint states (Fig. 3). Besides, transferring the hand trajectory to a morphologically different hand requires correspondences to make the resulting trajectory close to the original one. Defining correspondences in the reduced coordinate or via sparse correspondences will make the result suffer from noise in the data, leading to unwanted results finally (Fig. 3). Hence, we propose relaxing an articulated multi-rigid body into a mass-point set sampled from the ambient space surrounding each body. Each point is considered attached to the body from which it is sampled and is capable of both self-actuation and actuation via joint motors. We introduce a parameter α to control the point set construction and the dynamics. This representation allows an articulated rigid object to behave similarly to a deformable object, providing a larger action space to adjust its state and thereby easing the control optimization problem.

Specifically, for each body of the articulated object, we sample a set of points from the ambient space near the body mesh. The point set $\mathcal{Q}$ is constructed by concatenating all sampled points together. Each point $\mathbf{p}_i \in \mathcal{Q}$ is treated as a mass point with a finite mass $\mathbf{m}_i$ and infinitesimal volume. The dynamics of the point set consist of articulated multi-body dynamics [14, 30], as well as the mass point dynamics of each point $\mathbf{p}_i$. For each $\mathbf{p}_i$, we have:

$$m_i \ddot{\mathbf{x}}_i = \mathbf{J}_i \mathbf{u} + \alpha \mathbf{f}_i + \alpha \mathbf{a}_i, \tag{1}$$

where $\mathbf{J}_i$ represents the Jacobian mapping from the generalized velocity to the point velocity $\dot{\mathbf{x}}_i$, $\mathbf{u}$ denotes the generalized joint force, $\mathbf{f}_i$ accounts for external forces acting on $\mathbf{p}_i$, and $\mathbf{a}_i \in \mathbb{R}^3$ represents the actuation force applied to the point $\mathbf{p}_i$. Consequently, the point set is controlled by a shared control in the reduced coordinate space $\mathbf{u}$ and per-point actuation force $\mathbf{a}_i$.

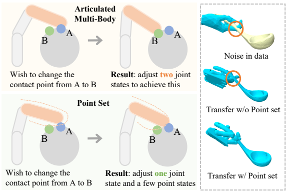

Fig. 3: Point Set can flexibly adjust its states, avoid overfitting to data noise, and ease the difficulty brought by the morphology difference.

Parameterized spring-damper contact modeling. To ease the optimization challenges posed by contact-rich manipulations, which arise from contact constraints such as the non-penetration requirement and Coulomb friction law [3, 5], as well as discontinuous dynamics involving frequent contact establishment and breaking, we propose a parameterized contact model for relaxing constraints and controlling the contact behavior. Specifically, we leverage a classical unconstrained spring-damper model [19, 35, 51, 59, 64] to model the contacts. This model allows us to flexibly adjust the contact behavior by tuning the contact threshold and the spring stiffness coefficients. Intuitively, a contact model with a high threshold and low spring stiffness presents "soft" behaviors, resulting in a continuous and smooth optimization space. This makes optimization through such a contact model relatively easy. Conversely, a model with a low threshold and large stiffness coefficients will produce "stiff" behaviors, increasing the discontinuity of the optimization space due to frequent contact establishment and breaking. However, it also becomes more physically realistic, meaning contact forces are calculated only when two objects collide, and a large force is applied to separate them if penetrations are observed, thus better satisfying the non-penetration condition. Therefore, by adjusting the contact distance threshold and spring stiffness coefficients, we can modulate the optimizability and fidelity of the contact model. The parameter set of the contact model comprises a distance threshold d^c and spring stiffness coefficients. Next, we will delve into the details of the contact establishment, breaking, and force calculations processes.

Contacts are established between points in the manipulator's point set $\mathcal{Q}$ and the object. A point $\mathbf{p} \in \mathcal{Q}$ is considered to be in "contact" with the object if its signed distance to the object $\mathrm{sd}(\mathbf{p})$ is smaller than the contact distance threshold d^c. Subsequently, the object surface point nearest to $\mathbf{p}$ is identified as the corresponding contact point on the object, denoted as $\mathbf{p}^o$. The normal direction of the object point $\mathbf{p}^o$ is then determined as the contact normal direction, denoted as $\mathbf{n}^o$. The contact force $\mathbf{f}^c$ applied from the manipulator point $\mathbf{p}$ to $\mathbf{p}^o$ is calculated as follows:

$$\mathbf{f}^c = -(k^n d - k^d d\dot{d})\mathbf{n}^o, \tag{2}$$

where, k^n represents the spring stiffness coefficient, k^d denotes the damping coefficient, and $d = d^c - \mathrm{sd}(\mathbf{p})$ is always positive. To enhance the continuity of $\mathbf{f}^c$ [64], $k^d d\dot{d}$ is used as the magnitude of the damping force, rather than $k^d\dot{d}$.

Friction forces are modeled as penalty-based spring forces [3,65]. Once a point $\mathbf{p}$ is identified as in contact with the object, with the object contact point denoted as $\mathbf{p}^o$, the contact pair is stored. Contact forces between them are continually calculated until the contact breaking conditions are met. In more detail, the static friction force from $\mathbf{p}$ to $\mathbf{p}^o$ is calculated using a spring model:

$$\mathbf{f}_s^f = k^f \mathbf{T}_n(\mathbf{p} - \mathbf{p}^o), \tag{3}$$

where k^f is the friction spring stiffness coefficient, $\mathbf{T}_n = \mathbf{I} - \mathbf{n}^o\mathbf{n}^{oT}$ is a tangential projection operator. When the static friction satisfies $\|\mathbf{f}_s^f\| \leq \mu\|\mathbf{f}^c\|$, $\mathbf{f}_s^f$ is applied to the object point $\mathbf{p}^o$. Otherwise, the dynamic friction force is applied, and the contact breaks:

$$\mathbf{f}_d^f = -\mu\|\mathbf{f}_s^f\| \frac{\mathbf{T}_n \mathbf{v}_{\mathbf{p}\leftarrow\mathbf{p}^o}}{\|\mathbf{T}_n \mathbf{v}_{\mathbf{p}\leftarrow\mathbf{p}^o}\|}, \tag{4}$$

where $\mathbf{v}_{\mathbf{p}\leftarrow\mathbf{p}^o}$ is the relative velocity between $\mathbf{p}$ and $\mathbf{p}^o$.

Parameterized residual physics. The analytical designs facilitate relaxation but may limit the use of highly sophisticated and realistic dynamics models, deviating from real physics. To address this, the final component of our quasi-physical simulator is a flexible neural residual physics model [1, 22, 41].

Specifically, we propose to use networks to learn to predict residual contact forces and friction forces from contact-related information. For fine-grained residual contact force prediction, we design a local contact network $f_{\psi_{\text{local}}}$ that inherits contact information identified in the parameterized contact model and predicts residual forces between each contact pair. To close the gap caused by contact region identification between the parameterized contact model and real contact region, we further include a global residual network $f_{\psi_{\text{global}}}$ that predicts residual forces and torques added directly to the object's center of mass. In more detail, given a contact pair $(\mathbf{p}, \mathbf{p}^o)$, the local contact network maps the contact-related features of the local contact region, consisting of geometry, per-point velocity, and per object point normal, to the residual contact force and residual friction force between such two points in the contact pair. The global residual network additionally takes contact-related information of the global contact region, including the geometry, per-point velocity, and per-object point normal, as input and predicts a residual force and residual torque added to the object's center of mass. Details such as contact region identification and network architectures are deferred to the Supp. We denote the optimizable parameters in the residual physics network as $\psi = (\psi_{\text{global}}, \psi_{\text{local}})$. By optimizing the residual physics network, we unlock the possibility of introducing highly non-linear dynamics to align our parametrized quasi-physical simulator with any realistic black-box physical simulator.

Semi-implicit time-stepping is leveraged to make the simulation auto differentiable and easy to combine with neural networks [22].

3.2 Dexterous Manipulation Transfer via a Physics Curriculum

Based on the family of parameterized quasi-physical simulators, we propose to solve the challenging dexterous manipulation transfer problem via a physics curriculum. It is constructed by a series of parameterized simulators varying from the one with few constraints and the softest contact behavior, gradually to a realistic simulator. We then solve the problem by transferring the demonstration to the dexterous hand in each simulator across the curriculum gradually. In more detail, the optimization starts from the parameterized simulator with both the articulated rigid constraints removed and the contact model tuned to the softest level. The residual physics networks are deactivated. It provides a friendly environment for optimization, and we can easily arrive at a workable hand trajectory here. Then the physics is gradually tightened and we solve the task in each simulator. After reaching the most tightened analytical model, the analytical part is fixed and residual networks are activated. The simulator is gradually optimized to approximate the dynamics in a realistic physical environment. At the same time, the control trajectory $\mathcal{A}$ continues to be refined in the quasi-physical simulator. Finally, we arrive at a simulator optimized to be with high fidelity and a trajectory $\mathcal{A}$ that can drive the dexterous hand to physically track the demonstration in a realistic simulated physical environment. Additionally, since object properties as well as system parameters like linear and angular velocity damping coefficients are unknown from the kinematics-only demonstration, we set them optimizable and identify them (denoted $\mathcal{S}$) together with optimizing the hand control trajectory. Next we'll illustrate this in detail.

Transferring human demonstration via point set dynamics. To robustly transfer the human demonstration to a morphologically different dexterous robot hand in simulation and to overcome noise in the kinematic trajectory, we initially relax the articulated rigid constraints and transfer the kinematics human demonstration to the control trajectory of the point set. Specifically, the point set representation with the relaxation parameter α for the dynamic human hand [8] is constructed. The shared control trajectory $\mathcal{A}$ and per-point per-frame actions are optimized so that the resulting trajectory of the point set can manipulate the object according to the demonstration. After that, a point set with the same parameter α is constructed to represent the dexterous robot hand. Subsequently, the shared control trajectory $\mathcal{A}$ and per-point per-frame actions are optimized to track the manipulation accordingly.

Transferring through a contact model curriculum. After that, the articulated rigid constraint is tightened by freezing the point set parameter α to zero. The following optimization starts from a parameterized simulator with the softest contact model. We then gradually tighten the contact model by adjusting its distance threshold, contact force spring stiffness, etc. By curriculum optimizing the trajectory $\mathcal{A}$ and parameters $\mathcal{S}$ in each of the quasi-physical simulators, we finally arrive at the control trajectory that can drive a dexterous hand to accomplish the tracking task in the parameterized simulator with the most tightened analytical model.

Optimizing towards a realistic physical environment. Subsequently, the residual physics network is activated and the parameterized simulator is optimized to approximate the dynamics in a realistic physical environment. We continue to optimize the hand trajectory in the quasi-physical simulator. Specifically, we leverage the successful trial in model-based human tracking literature [16,66] and iteratively optimize the control trajectory $\mathcal{A}$ and the parameterized simulator. In more detail, the following two subproblems are iteratively solved: 1) optimizing the quasi-physical simulator to approximate the realistic dynamics, and 2) optimizing the control trajectory $\mathcal{A}$ to complete the manipulation in the quasi-physical simulator. Gradient-based optimization is leveraged taking advantage of the differentiability of the parameterized simulator.

After completing the optimization, the final control trajectory is yielded by model predictive control (MPC) [18] based on the optimized parameterized simulator and the hand trajectory $\mathcal{A}$. Specifically, in each step, the current and the following controls in several subsequent frames are optimized to reduce the tracking error. More details are deferred to the Supp.

4 Experiments

We conduct extensive experiments to demonstrate the effectiveness of our method. The evaluation dataset is constructed from three HOI datasets with both single-hand and bimanual manipulations (with rigid objects), with complex manipulations with non-trivial object movements, and rich and changing contacts involved (see Section 4.1). We use Shadow hand [49] and test in two simulators widely used in the embodied AI community: Bullet [9] and Isaac Gym [33]. We compare our method with both model-free approaches and model-based strategies and demonstrate the superiority of our method both quantitatively and qualitatively. We can track complex contact-rich manipulations with large object rotations, back-and-forth object movements, and changing contacts successfully in both of the two simulators, while the best-performed baseline fails (see Section 4.2, Fig. 4). On average, we boost the tracking success rate by 11%+ from the previous best-perfomed (see Section 4.2). We make further analysis and discussions and show that the core philosophy of our work, optimizing through a quasi-physics curriculum, is potentially general and can help improve the performance of a model-free baseline (see Section 4.3).

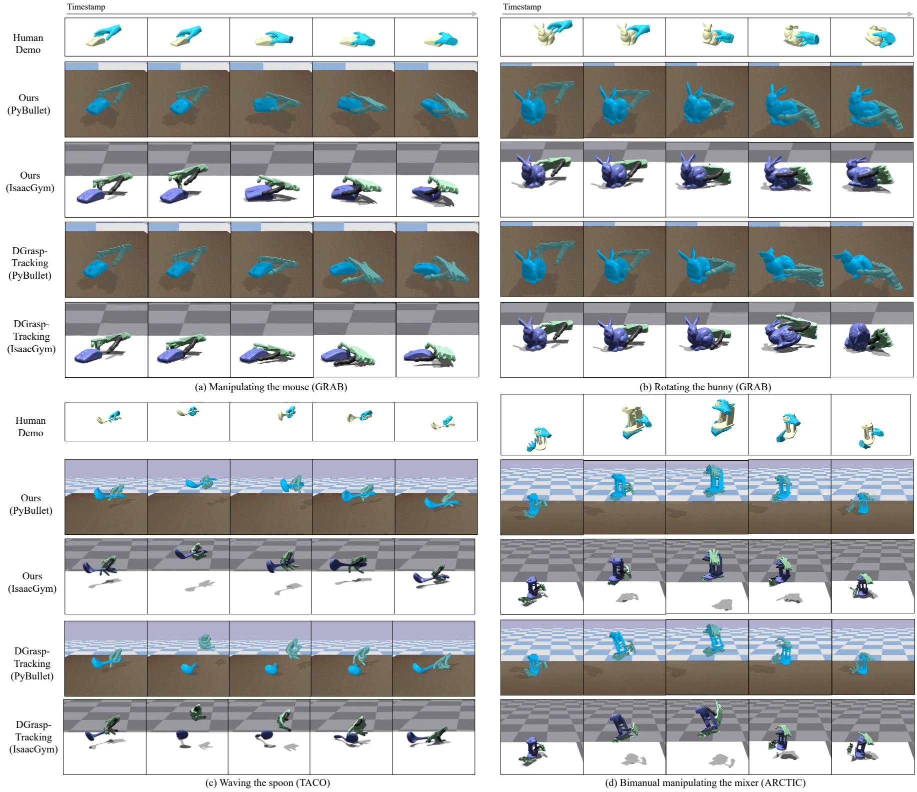

Fig. 4: Qualitative comparisons. Please refer to **our website and the supplementary video** for animated results.

4.1 Experimental Settings

Datasets. Our evaluation dataset is compiled from three distinct sources, namely GRAB [53], containing single-hand interactions with daily objects, TACO [32], containing humans manipulating tools, and ARCTIC [13] with bimanual manipulations. For GRAB, we randomly sample a manipulation trajectory for each object. If its manipulation is extremely simple, we additionally sample one trajectory for it. The object is not considered if its corresponding manipulation is bimanual such as `binoculars`, involves other body parts such as `bowl`, or with detailed part movements such as the `game controller`. The number of manipulation sequences from GRAB is 27. For TACO [32], we acquire data by contacting authors. We randomly select one sequence for each right-hand tool object. Sequences with very low quality like erroneous object motions are excluded. 14 trajectories in total are selected finally. For ARCTIC [13], we randomly select one sequence for each object from its available manipulation trajectories, resulting in 10 sequences in total. More details are deferred to the Supp.

Metrics. We introduce three distinct metrics to assess the quality of object tracking, the accuracy of hand tracking, and the overall success of the tracking task: 1) Per-frame average object rotation error: $R_{\text{err}} = \frac{1}{N}\sum_{n=1}^{N}(1 - (\mathbf{q}_n \cdot \hat{\mathbf{q}}_n))$, where $\mathbf{q}_n$ is the ground-truth orientation and $\hat{\mathbf{q}}_n$ is the tracked result, represented in quaternion. 2) Per-frame average object translation error: $T_{\text{err}} = \frac{1}{N}\sum_{n=1}^{N}\|\mathbf{t}_n - \hat{\mathbf{t}}_n\|$, where $\mathbf{t}$ and $\mathbf{t}_n$ are ground-truth and tracked translations respectively. 3) Mean Per-Joint Position Error (MPJPE) $= \frac{1}{N}\sum_{n=1}^{N}\|\mathbf{J}_n - \hat{\mathbf{J}}_n\|$ [20, 44, 58], where $\mathbf{J}_n$ and $\hat{\mathbf{J}}_n$ are keypoints of GT human hand and the simulated robot hand respectively. We manually define the keypoints and the correspondences to the human hand keypoints for the Shadow hand. 4) Per-frame average hand Chamfer Distance: CD $= \frac{1}{N}\sum_{n=1}^{N}$ Chamfer-Distance$(\mathbf{H}_n - \hat{\mathbf{H}}_n)$, for evaluating whether the Shadow hand can "densely" track the demonstration. 5) Success rate: a tracking is regarded as successful if the object rotation error R_{err}, object translation error T_{err}, and the hand tracking error MPJPE are smaller than their corresponding threshold. Three success rates are calculated using three different thresholds, namely $10° - 10cm - 10cm$, $15° - 15cm - 15cm$.

Baselines. We compare with two trends of baselines. For model-free approaches, since there is no prior work with exactly the same problem setting as us, we try to modify and improve a goal-driven rigid object manipulation method DGrasp [8] into two methods for tracking: 1) DGrasp-Base, where the method is almost kept with same with the original DGrasp. We use the first frame where the hand and the object are in contact with each other as the reference frame. Then the policy is trained to grasp the object according to the reference hand and object goal at first. After that, only the root is guided to complete the task. 2) DGrasp-Tracking, where we divide the whole sequence into several subsequences, each of which has 10 frames, and define the end frame of the subsequence as the reference frame. Then the grasping policy is used to guide the hand and gradually track the object according to the hand and the object pose of each reference frame. We improve the DGrasp-Tracking by optimizing the policy through the quasi-physical curriculum and creating "DGrasp-Tracking (w/ Curriculum)" trying to improve its performance. For model-based methods, we compare with Control-VAE [66] and traditional MPC approaches. For Control-VAE, we modify its implementation for the manipulation tracking task. We additionally consider three differentiable physics models to conduct model-predictive control for solving the task. Taking the analytical model with the most tightened contact model as the base model ("MPC (w/ base sim.)"), we further augment it with a general state-of-the-art contact smoothing for robot manipulation [52] and create "MPC (w/ base sim. w/ soften)". Details of baseline models are deferred to the Supp.

Training and evaluation settings. The physics curriculum is composed of three stages. In the first stage, the parameter α varies from 0.1 to 0.0 and the contact model stiffness is relaxed to the softest level. In the second stage, α is fixed and the contact model stiffness varies from the softest version to the most tightened level gradually through eight stages. Details w.r.t. parameter settings are deferred to the Supp. In the first two stages, we alternately optimize the trajectory $\mathcal{A}$ and parameters $\mathcal{S}$. In each optimization iteration, the $\mathcal{A}$ is optimized for 100 steps while $\mathcal{S}$ is optimized for 1000 steps. In the third stage, $\mathcal{A}$ and ψ are optimized for 256 steps in each iteration. For time-stepping, $\mathrm{d}t$ is set to 5×10^{-4} in the parameterized and the target simulators. The articulated multi-body is

Table 1: Quantitative evaluations and comparisons to baselines. **Bold red** numbers for best values and *italic blue* values for the second best-performed ones.

Simulator		Method	R_{err} ($°, \downarrow$)	T_{err} ($cm, \downarrow$)	MPJPE ($mm, \downarrow$)	CD ($mm, \downarrow$)	Success Rate ($\%, \uparrow$)
Bullet	Model Free	DGrasp-Base	44.24	5.82	40.55	16.37	0/13.73/15.69
		DGrasp-Tracking	44.45	5.04	37.56	14.72	0/15.69/15.69
		DGrasp-Tracking (w/ curric.)	33.86	4.60	30.47	13.53	7.84/*23.53*/*37.25*
	Model Based	Control-VAE	42.45	*2.73*	25.21	10.94	0/15.68/23.53
		MPC (w/ base sim.)	32.56	3.67	*24.62*	*10.80*	0/15.68/31.37
		MPC (w/ base sim. w/ soften)	*31.89*	3.63	28.26	11.31	0/21.57/*37.25*
		Ours	**24.21**	**1.97**	**24.40**	**9.85**	**27.45**/**37.25**/**58.82**
Isaac Gym	Model Free	DGrasp-Base	36.41	4.56	50.97	18.78	0/7.84 /7.84
		DGrasp-Tracking	44.71	5.57	41.53	16.72	0/0/7.84
		DGrasp-Tracking (w/ curric,)	38.75	5.13	40.09	16.26	0/*23.53*/*31.37*
	Model Based	Control-VAE	*35.40*	4.61	27.63	13.17	0/13.73/29.41
		MPC (w/ base sim.)	37.23	4.73	**23.19**	**9.75**	0/15.69/*31.37*
		MPC (w/ base sim. w/ soften)	36.40	*4.46*	*23.27*	10.34	0/9.80/23.53
		Ours	**25.97**	**2.08**	25.33	*10.31*	**21.57**/**43.14**/**56.86**

controlled by joint motors and root velocities in the parameterized quasi-physical simulator while PD control [54] is leveraged in the target simulators.

4.2 Dexterous Manipulating Tracking

We conducted thorough experiments in two widely used simulators [9, 33]. We treat them as realistic simulated physical environments with high fidelity and wish to track the manipulation in them. In summary, we can control a dexterous hand to complete a wide range of the manipulation tracking tasks with non-trivial object movements and changing contacts. As presented in Table 1, we can achieve significantly higher success rates calculated under three thresholds than the best-performed baseline in both tested simulators. Fig. 4 showcases qualitative examples and comparisons. Please refer to our website and supplementary video for animated results.

Complex manipulations. For examples shown in Fig. 4, we can complete the tracking task on examples with large object re-orientations and complicated tool-using (*Fig. (a,b,c)*). However, DGrasp-Tracking fails to establish sufficient contact for correctly manipulating the object. In more detail, in Fig. 4(b), the bunny gradually bounced out from its hand in Bullet, while our method does not suffer from this difficulty. In Fig. 4(c), the spoon can be successfully picked up and waved back-and-forth in our method, while DGrasp-Tracking loses the track right from the start.

Bimanual manipulations. We are also capable of tracking bimanual manipulations. As shown in the example in Fig. 4(d), where two hands collaborate to relocate the object, DGrasp-Tracking fails to accurately track the object, while our method significantly outperforms it.

4.3 Further Analysis and Discussions

Could model-free methods benefit from the physics curriculum? In addition to the demonstrated merits of our quasi-physical simulators, we further explore whether model-free strategies can benefit from them. We introduce the "DGrasp-Tracking (w/ Curriculum)" method and compare its performance with

Table 2: Ablation studies. **Bold red** numbers for best values and *italic blue* values for the second best-performed ones. The simulation environment is Bullet.

Method	R_{err} $(°, \downarrow)$	T_{err} $(cm, \downarrow)$	MPJPE $(mm, \downarrow)$	CD $(mm, \downarrow)$	Success Rate $(\%, \uparrow)$
Ours w/o Analytical Sim.	44.27	4.39	29.84	12.91	0/13.73/25.49
Ours w/o Residual Physics	33.69	3.81	*26.57*	10.34	5.88/23.53/41.18
Ours w/o Local Force NN	35.98	2.90	32.87	12.44	0/19.61/35.29
Ours w/o Curriculum	42.40	4.87	32.61	13.37	0/17.64/29.41
Ours w/ Curriculum II	*29.58*	*2.33*	31.61	*10.29*	*11.76*/*27.45*/*50.98*
Ours	**24.21**	**1.97**	**24.40**	**9.85**	**27.45**/**37.25**/**58.82**

the original DGrasp-Tracking model. As shown in Table 1 and the visual comparisons in Fig. 6, the DGrasp-Tracking model indeed benefits from a well-designed physics curriculum. For example, as illustrated in Fig. 6, the curriculum can significantly improve its performance, enabling it to nearly complete challenging tracking tasks where the original version struggles.

5 Ablation Study

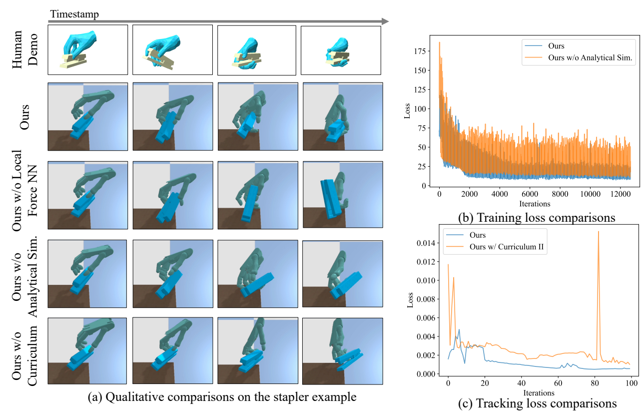

Fig. 5: (a) Qualitative comparisons between our full method and the ablated models; (b) Training loss curve comparisons; (c) Tracking loss curve comparisons.

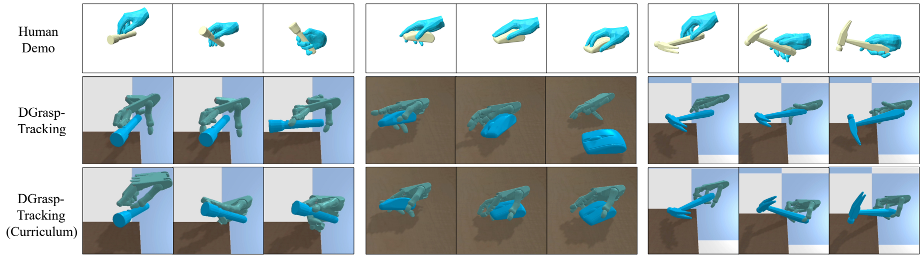

Fig. 6: Visual evidence on boosting DGrasp-Tracking's performance via optimizing it through a physics curriculum.

We conduct a wide range of ablation studies to validate the effectiveness of some of our crucial designs, including the parameterized analytical physics model, the parameterized residual physics, the role of the local force network, the necessity of introducing a physics curriculum into the optimization, and how the design on the curriculum stages affects the result.

Parameterized analytical model. The skeleton of the quasi-physical simulator is an analytical physics model. The intuition is that the parameterized simulator with such physical bias can be optimized towards a realistic simulator more easily than training pure neural networks for approximating. To validate this, we ablate the analytical model and use neural networks to approximate physics in Bullet directly (denoted as "Ours w/o Analytical Sim."). The quantitative (Table 2) and qualitative (Fig. 5) results indicate that the physical biases brought by the analytical model could help the parameterized simulator to learn better physics in the contact-rich scenario. For instance, in the example demonstrated in Fig. 5, the ablated version fails to guide the robot hand to successfully pinch the object in the second figure.

Parameterized residual physics. To validate the necessity of introducing residual force networks to close the gap between the physics modeled in the parameterized analytical simulator and that of a realistic simulator, we ablate the parameterized force network and create a version named "Ours w/o Residual Physics". Table 2 demonstrated its role in enabling the parameterized simulator to approximate realistic physics models.

Local residual force network. To adequately leverage state and contact-related information for predicting residual contact forces, we propose to use two types of networks: 1) a local force network for per contact pair residual forces and 2) a global network for additionally compensating. The local network is introduced for fine-grained approximation. We ablate this design and compare the result with our full model to validate this (see Fig. 5 and Table 1).

Optimizing through an analytical physics curriculum. We further investigate the effectiveness of the analytical curriculum design and how its design influences the result. Specifically, we create two ablated versions: 1) "Ours w/o Curriculum", where the optimization starts directly from the parameterized analytical model with articulated rigid constraints tightened and the stiffest contact model, and 2) "Ours w/ Curriculum II", where we move some stages out from the original curriculum. Table 2 and Fig. 5 demonstrate that both the curriculum and the optimization path will affect the model's performance.

6 Conclusion and Limitations

In this work, we investigate creating better simulators for solving complex robotic tasks involving complicated dynamics where the previous best-performed optimization strategy fails. We present a family of parameterized quasi-physical simulators that can be both programmed to relax various constraints for task optimization and can be tailored to approximate realistic physics. We tackle the difficult manipulation transfer task via a physics curriculum.

Limitations. The method is limited by the relatively simple spring-damper model for contact constraint relaxation. Introducing delicate analytical contact models to parameterized simulators is an interesting research direction.

References

1. Ajay, A., Wu, J., Fazeli, N., Bauza, M., Kaelbling, L.P., Tenenbaum, J.B., Rodriguez, A.: Augmenting physical simulators with stochastic neural networks: Case study of planar pushing and bouncing. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3066–3073. IEEE (2018) 7
2. Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R., et al.: Solving rubik's cube with a robot hand. arXiv preprint arXiv:1910.07113 (2019) 2
3. Andrews, S., Erleben, K., Ferguson, Z.: Contact and friction simulation for computer graphics. In: ACM SIGGRAPH 2022 Courses, pp. 1–172 (2022) 2, 3, 6, 7
4. Andrychowicz, O.M., Baker, B., Chociej, M., Jozefowicz, R., McGrew, B., Pachocki, J., Petron, A., Plappert, M., Powell, G., Ray, A., et al.: Learning dexterous in-hand manipulation. The International Journal of Robotics Research **39**(1), 3–20 (2020) 2, 4
5. Baraff, D.: An introduction to physically based modeling: rigid body simulation ii—nonpenetration constraints. SIGGRAPH course notes pp. D31–D68 (1997) 6
6. Chen, T., Tippur, M., Wu, S., Kumar, V., Adelson, E., Agrawal, P.: Visual dexterity: In-hand reorientation of novel and complex object shapes. Science Robotics **8**(84), eadc9244 (2023). https://doi.org/10.1126/scirobotics.adc9244, https://www.science.org/doi/abs/10.1126/scirobotics.adc9244 2, 4
7. Chen, T., Xu, J., Agrawal, P.: A system for general in-hand object re-orientation. Conference on Robot Learning (2021) 2
8. Christen, S., Kocabas, M., Aksan, E., Hwangbo, J., Song, J., Hilliges, O.: D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20577–20586 (2022) 2, 4, 8, 11
9. Coumans, E., Bai, Y.: Pybullet, a python module for physics simulation for games, robotics and machine learning (2016) 2, 3, 9, 12
10. Deng, Y., Yu, H.X., Wu, J., Zhu, B.: Learning vortex dynamics for fluid inference and prediction. arXiv preprint arXiv:2301.11494 (2023) 4
11. Du, T.: Deep learning for physics simulation. In: ACM SIGGRAPH 2023 Courses, pp. 1–25 (2023) 4
12. Dunlavy, D.M., O'Leary, D.P.: Homotopy optimization methods for global optimization. Tech. rep., Sandia National Laboratories (SNL), Albuquerque, NM, and Livermore, CA ... (2005) 2
13. Fan, Z., Taheri, O., Tzionas, D., Kocabas, M., Kaufmann, M., Black, M.J., Hilliges, O.: ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023) 10
14. Featherstone, R.: Rigid body dynamics algorithms (2007), https://api.semanticscholar.org/CorpusID:58437819 6
15. Freeman, C.D., Frey, E., Raichuk, A., Girgin, S., Mordatch, I., Bachem, O.: Brax–a differentiable physics engine for large scale rigid body simulation. arXiv preprint arXiv:2106.13281 (2021) 2
16. Fussell, L., Bergamin, K., Holden, D.: Supertrack: Motion tracking for physically simulated characters using supervised learning. ACM Transactions on Graphics (TOG) **40**(6), 1–13 (2021) 2, 9

17. Gao, J., Michelis, M.Y., Spielberg, A., Katzschmann, R.K.: Sim-to-real of soft robots with learned residual physics. arXiv preprint arXiv:2402.01086 (2024) 4
18. Garcia, C.E., Prett, D.M., Morari, M.: Model predictive control: Theory and practice—a survey. Automatica **25**(3), 335–348 (1989) 9
19. Geilinger, M., Hahn, D., Zehnder, J., Bächer, M., Thomaszewski, B., Coros, S.: Add: Analytically differentiable dynamics for multi-body systems with frictional contact. ACM Transactions on Graphics (TOG) **39**(6), 1–15 (2020) 3, 4, 5, 6
20. Grandia, R., Farshidian, F., Knoop, E., Schumacher, C., Hutter, M., Bächer, M.: Doc: Differentiable optimal control for retargeting motions onto legged robots. ACM Transactions on Graphics (TOG) **42**(4), 1–14 (2023) 2, 11
21. Gupta, A., Eppner, C., Levine, S., Abbeel, P.: Learning dexterous manipulation for a soft robotic hand from human demonstrations. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3786–3793. IEEE (2016) 2, 4
22. Heiden, E., Millard, D., Coumans, E., Sheng, Y., Sukhatme, G.S.: Neuralsim: Augmenting differentiable simulators with neural networks. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 9474–9481. IEEE (2021) 3, 4, 7, 8
23. Howell, T.A., Le Cleac'h, S., Kolter, J.Z., Schwager, M., Manchester, Z.: Dojo: A differentiable simulator for robotics. arXiv preprint arXiv:2203.00806 **9** (2022) 2, 4
24. Howell, T.A., Le Cleac'h, S., Singh, S., Florence, P., Manchester, Z., Sindhwani, V.: Trajectory optimization with optimization-based dynamics. IEEE Robotics and Automation Letters **7**(3), 6750–6757 (2022) 2
25. Hwangbo, J., Lee, J., Hutter, M.: Per-contact iteration method for solving contact dynamics. IEEE Robotics and Automation Letters **3**(2), 895–902 (2018) 2
26. Lan, L., Yang, Y., Kaufman, D., Yao, J., Li, M., Jiang, C.: Medial ipc: accelerated incremental potential contact with medial elastics. ACM Transactions on Graphics **40**(4) (2021) 4
27. Li, M., Ferguson, Z., Schneider, T., Langlois, T.R., Zorin, D., Panozzo, D., Jiang, C., Kaufman, D.M.: Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. ACM Trans. Graph. **39**(4), 49 (2020) 4
28. Liao, S.: On the homotopy analysis method for nonlinear problems. Applied mathematics and computation **147**(2), 499–513 (2004) 2
29. Lin, X., Yang, Z., Zhang, X., Zhang, Q.: Continuation path learning for homotopy optimization (2023) 2
30. Liu, C.K., Jain, S.: A quick tutorial on multibody dynamics. Online tutorial, June p. 7 (2012) 6
31. Liu, X., Pathak, D., Kitani, K.M.: Herd: Continuous human-to-robot evolution for learning from human demonstration. arXiv preprint arXiv:2212.04359 (2022) 2, 4
32. Liu, Y., Yang, H., Si, X., Liu, L., Li, Z., Zhang, Y., Liu, Y., Yi, L.: Taco: Benchmarking generalizable bimanual tool-action-object understanding. arXiv preprint arXiv:2401.08399 (2024) 10
33. Makoviychuk, V., Wawrzyniak, L., Guo, Y., Lu, M., Storey, K., Macklin, M., Hoeller, D., Rudin, N., Allshire, A., Handa, A., et al.: Isaac gym: High performance gpu-based physics simulation for robot learning. arXiv preprint arXiv:2108.10470 (2021) 2, 3, 9, 12
34. Mandikal, P., Grauman, K.: Dexvip: Learning dexterous grasping with human hand pose priors from video. In: Conference on Robot Learning. pp. 651–661. PMLR (2022) 4

35. Marcucci, T., Gabiccini, M., Artoni, A.: A two-stage trajectory optimization strategy for articulated bodies with unscheduled contact sequences. IEEE Robotics and Automation Letters **2**(1), 104–111 (2016) 3, 6
36. Mordatch, I., Popović, Z., Todorov, E.: Contact-invariant optimization for hand manipulation. In: Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation. pp. 137–144 (2012) 2
37. Pang, T., Suh, H.T., Yang, L., Tedrake, R.: Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models. IEEE Transactions on Robotics (2023) 4
38. Pang, T., Tedrake, R.: A convex quasistatic time-stepping scheme for rigid multibody systems with contact and friction. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 6614–6620. IEEE (2021) 2, 3, 4
39. Peng, X.B., Abbeel, P., Levine, S., Van de Panne, M.: Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. ACM Transactions On Graphics (TOG) **37**(4), 1–14 (2018) 2
40. Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., Battaglia, P.W.: Learning mesh-based simulation with graph networks. arXiv preprint arXiv:2010.03409 (2020) 4
41. Pfrommer, S., Halm, M., Posa, M.: Contactnets: Learning discontinuous contact dynamics with smooth, implicit representations. In: Conference on Robot Learning. pp. 2279–2291. PMLR (2021) 4, 7
42. Qin, Y., Su, H., Wang, X.: From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation. IEEE Robotics and Automation Letters **7**(4), 10873–10881 (2022) 4
43. Qin, Y., Wu, Y.H., Liu, S., Jiang, H., Yang, R., Fu, Y., Wang, X.: Dexmv: Imitation learning for dexterous manipulation from human videos. In: European Conference on Computer Vision. pp. 570–587. Springer (2022) 2, 4
44. Qin, Y., Yang, W., Huang, B., Van Wyk, K., Su, H., Wang, X., Chao, Y.W., Fox, D.: Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. arXiv preprint arXiv:2307.04577 (2023) 11
45. Radosavovic, I., Wang, X., Pinto, L., Malik, J.: State-only imitation learning for dexterous manipulation. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 7865–7871. IEEE (2021) 4
46. Rajeswaran, A., Kumar, V., Gupta, A., Vezzani, G., Schulman, J., Todorov, E., Levine, S.: Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. arXiv preprint arXiv:1709.10087 (2017) 2, 4
47. Rusu, A.A., Colmenarejo, S.G., Gulcehre, C., Desjardins, G., Kirkpatrick, J., Pascanu, R., Mnih, V., Kavukcuoglu, K., Hadsell, R.: Policy distillation. arXiv preprint arXiv:1511.06295 (2015) 4
48. Schmeckpeper, K., Rybkin, O., Daniilidis, K., Levine, S., Finn, C.: Reinforcement learning with videos: Combining offline observations with interaction. arXiv preprint arXiv:2011.06507 (2020) 4
49. ShadowRobot: Shadowrobot dexterous hand (2005), https://www.shadowrobot.com/dexterous-hand-series/ 9
50. Siahkoohi, A., Louboutin, M., Herrmann, F.J.: Neural network augmented wave-equation simulation. arXiv preprint arXiv:1910.00925 (2019) 4
51. Suh, H., Wang, Y.: Comparing effectiveness of relaxation methods for warm starting trajectory optimization through soft contact (2019) 3, 6
52. Suh, H.J.T., Pang, T., Tedrake, R.: Bundled gradients through contact via randomized smoothing. IEEE Robotics and Automation Letters **7**(2), 4000–4007 (2022) 11

53. Taheri, O., Ghorbani, N., Black, M.J., Tzionas, D.: Grab: A dataset of whole-body human grasping of objects. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16. pp. 581–600. Springer (2020) 10
54. Tan, K.K., Wang, Q.G., Hang, C.C.: Advances in PID control. Springer Science & Business Media (2012) 12
55. Tedrake, R., the Drake Development Team: Drake: Model-based design and verification for robotics (2019), https://drake.mit.edu 2
56. Todorov, E., Erez, T., Tassa, Y.: Mujoco: A physics engine for model-based control. In: 2012 IEEE/RSJ international conference on intelligent robots and systems. pp. 5026–5033. IEEE (2012) 2
57. Traoré, R., Caselles-Dupré, H., Lesort, T., Sun, T., Díaz-Rodríguez, N., Filliat, D.: Continual reinforcement learning deployed in real-life using policy distillation and sim2real transfer. arXiv preprint arXiv:1906.04452 (2019) 4
58. Villegas, R., Ceylan, D., Hertzmann, A., Yang, J., Saito, J.: Contact-aware retargeting of skinned motion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9720–9729 (2021) 11
59. Wang, Y., Weidner, N.J., Baxter, M.A., Hwang, Y., Kaufman, D.M., Sueda, S.: Redmax: Efficient & flexible approach for articulated dynamics. ACM Transactions on Graphics (TOG) **38**(4), 1–10 (2019) 2, 5, 6
60. Wang, Y., Lin, J., Zeng, A., Luo, Z., Zhang, J., Zhang, L.: Physhoi: Physics-based imitation of dynamic human-object interaction. arXiv preprint arXiv:2312.04393 (2023) 2, 4
61. Watson, L.T., Haftka, R.T.: Modern homotopy methods in optimization. Computer Methods in Applied Mechanics and Engineering **74**(3), 289–305 (1989) 2
62. Wu, Y.H., Wang, J., Wang, X.: Learning generalizable dexterous manipulation from human grasp affordance. In: Conference on Robot Learning. pp. 618–629. PMLR (2023) 2, 4
63. Xiong, S., He, X., Tong, Y., Deng, Y., Zhu, B.: Neural vortex method: from finite lagrangian particles to infinite dimensional eulerian dynamics. Computers & Fluids **258**, 105811 (2023) 4
64. Xu, J., Chen, T., Zlokapa, L., Foshey, M., Matusik, W., Sueda, S., Agrawal, P.: An end-to-end differentiable framework for contact-aware robot design. arXiv preprint arXiv:2107.07501 (2021) 6, 7
65. Yamane, K., Nakamura, Y.: Stable penalty-based model of frictional contacts. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006. pp. 1904–1909. IEEE (2006) 7
66. Yao, H., Song, Z., Chen, B., Liu, L.: Controlvae: Model-based learning of generative controllers for physics-based characters. ACM Transactions on Graphics (TOG) **41**(6), 1–16 (2022) 2, 9, 11
67. Zeng, A., Song, S., Lee, J., Rodriguez, A., Funkhouser, T.: Tossingbot: Learning to throw arbitrary objects with residual physics. IEEE Transactions on Robotics **36**(4), 1307–1319 (2020) 4
68. Zhang, H., Christen, S., Fan, Z., Zheng, L., Hwangbo, J., Song, J., Hilliges, O.: Artigrasp: Physically plausible synthesis of bi-manual dexterous grasping and articulation. arXiv preprint arXiv:2309.03891 (2023) 2, 4
69. Zhang, S., Liu, B., Wang, Z., Zhao, T.: Model-based reparameterization policy gradient methods: Theory and practical algorithms. Advances in Neural Information Processing Systems **36** (2024) 4

70. Zhang, Y., Clegg, A., Ha, S., Turk, G., Ye, Y.: Learning to transfer in-hand manipulations using a greedy shape curriculum. In: Computer Graphics Forum. vol. 42, pp. 25–36. Wiley Online Library (2023) 4
71. Zhao, W., Queralta, J.P., Westerlund, T.: Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In: 2020 IEEE symposium series on computational intelligence (SSCI). pp. 737–744. IEEE (2020) 4
72. Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q.: A comprehensive survey on transfer learning. Proceedings of the IEEE **109**(1), 43–76 (2020) 4