GUIDING WITHOUT PRESCRIBING: A DIFFERENTIABLE TRAJECTORY LEARNING FRAMEWORK FOR ROBOT-ASSISTED DRESSING

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot-assisted dressing requires generating effective manipulation trajectories under complex cloth dynamics and sustained garment-body interactions. Learning a single trajectory solver across diverse arm configurations is further challenging because effective trajectories can vary substantially with pose. Differentiable physics provides first-order gradients for trajectory optimization, but obtaining these gradients requires complete garment states unavailable to the solver at inference. We propose **Guiding without Prescribing (GwoP)**, a differentiable trajectory learning framework that integrates physical and body-relative guidance during training. GwoP separates observation-conditioned trajectory prediction from full-state physical evaluation within a differentiable computation graph, allowing physical gradients to train the solver without requiring complete garment states at inference. To support shared learning across arm poses, GwoP further predicts trajectories and auxiliary body-surface relations in parallel, providing structural guidance without prescribing trajectory geometry. Across unseen arm configurations, GwoP achieves high dressing progress with limited garment deformation. We further validate the learned trajectories through cross-simulator transfer and real-robot execution across diverse arm poses.

1 INTRODUCTION

Robot-assisted dressing aims to enable assistive robots to help people with limited mobility accomplish daily dressing tasks (Gao et al., 2015; Zhang et al., 2017). Generating effective dressing trajectories remains difficult because clothing is a high-dimensional deformable object whose behavior depends on complex and changing interactions with the human body (Erickson et al., 2018). These interactions make it difficult to generate trajectories that achieve sufficient dressing progress while avoiding excessive deformation or unfavorable contact. The difficulty further increases when a single solver must cover diverse arm configurations: trajectories effective for different poses can differ substantially in their geometry, making them difficult to learn jointly with a shared predictor (Wang et al., 2023).

Differentiable simulation provides first-order gradients for optimizing dressing trajectories through cloth dynamics (Liang et al., 2019; Li et al., 2022b). These gradients require complete simulated garment states for physical evaluation, whereas a deployable solver should predict trajectories from task observations. More importantly, learning a single solver across diverse arm configurations raises a distinct structural question: trajectories that solve the same dressing task can differ substantially across poses, so directly supervising or constraining their geometry may interfere with shared learning. This motivates asking how pose-shared structure should be introduced without prescribing the trajectory itself.

To address these challenges, we propose **Guiding without Prescribing (GwoP)**, a framework for learning a shared dressing trajectory solver with differentiable physical and body-relative guidance. First, differentiable physical guidance separates training-time physical evaluation from inference-time trajectory prediction. During training, predicted trajectories are rolled out with complete garment states in a differentiable simulator, allowing physical gradients to guide the solver according to the resulting cloth-body interactions; at inference, the solver predicts trajectories directly from task

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

observations without requiring complete garment states. Second, body-relative guidance provides a shared structural reference for learning trajectories across arm poses. The solver predicts trajectories and their body-relative relations in parallel from a shared representation, and the physical and relation objectives jointly optimize the same solver. This allows body-relative structure to guide trajectory learning without prescribing pose-specific trajectory geometry.

We evaluate GwoP across $1{,}041$ unseen arm configurations against geometric planning, model-free policy learning, and learned-dynamics planning. GwoP achieves the highest mean dressing progress of 74.7% while maintaining low garment deformation and requiring less training time than the learned baselines. A network forward pass and single-trajectory decoding require only 2.60 ms on average. Ablation studies show that physical gradients are critical for learning trajectories that effectively advance the garment under cloth-body dynamics, while body-relative guidance produces a broader and smoother local low-loss region in parameter space, indicating reduced sensitivity to parameter perturbations. Across nine arm poses, GwoP attains the highest fraction of positively aligned pose pairs at 66.7% and the lowest mean gradient magnitude within the evaluated two-dimensional parameter subspace. Finally, trajectories generated in DiffCloth remain effective when transferred to CLO without further optimization, and real-robot experiments across 12 arm poses provide additional evidence of execution feasibility.

Our contributions are as follows:

- We introduce a differentiable physical guidance framework that uses full-state differentiable rollouts to guide the training of an observation-conditioned trajectory solver, while requiring only task observations at inference.
- We introduce body-relative guidance that jointly learns trajectories and body-surface relations, providing a more stable shared representation for trajectory learning across diverse arm poses without prescribing specific trajectory geometry.
- We evaluate GwoP across $1{,}041$ unseen arm configurations, complemented by ablation studies, cross-simulator transfer, and real-robot validation.

2 Related Work

2.1 Robot-Assisted Dressing

A central challenge in robot-assisted dressing is to generate manipulation strategies that remain effective across different task configurations. Early approaches handled configuration variation explicitly by constructing paths from anatomical landmarks or intermediate waypoints (Gao et al., 2015; Zhang et al., 2019), and adapting them using force feedback (Gao et al., 2016), visual observations (Zhang et al., 2017), or user-specific mobility constraints (Zhang et al., 2017; Yamazaki et al., 2016; Erickson et al., 2019). Demonstration-based methods instead represented dressing as reusable motion primitives or state-dependent segments (Pignat & Calinon, 2017; Canal et al., 2018). While interpretable and easy to adapt, these approaches rely heavily on predefined trajectory structure.

Another line of work learns interaction models and uses them for online optimization. Early models estimate dressing states or spatial force distributions (Kapusta et al., 2016; Yu et al., 2017; Erickson et al., 2017), while later methods predict garment-body interaction forces (Erickson et al., 2018) or the dynamics of task-critical garment structures (Kotsovolis & Demiris, 2024; 2025) for model-predictive control. Such methods can adapt to changing observations, but actions are obtained through prediction and online optimization rather than directly generated by a reusable policy.

More recent methods learn reusable dressing policies. Early learned approaches were largely task-specific, with generalization limited to constrained transfer settings such as mirrored configurations (Tamei et al., 2011; Twardon & Ritter, 2018; Ildefonso et al., 2021). More recent partial-observation policies have expanded this generalization to diverse garments and arm configurations, with visual and force feedback further enabling adaptation to human motion (Wang et al., 2023; Hao et al., 2025). However, jointly learning diverse arm configurations remains difficult due to uneven learning progress and conflicting optimization signals across configurations, motivating pose-specific teachers before distillation (Wang et al., 2023). We instead exploit body-relative structural relations shared across arm configurations to guide trajectory learning.

2.2 Differentiable Simulation and Optimization

Differentiable simulation enables physical objectives to be optimized through first-order gradients of system dynamics. Early work established differentiable simulation for rigid-body dynamics (de Avila Belbute-Peres et al., 2018), while general-purpose differentiable programming frameworks support a broader range of physical systems, including deformable objects (Hu et al., 2020). Specialized simulators further developed differentiable deformable-body dynamics (Hu et al., 2019; Du et al., 2021; Chen et al., 2023b). Cloth introduces additional challenges due to thin-shell dynamics and frequent frictional contact. Differentiable cloth simulation has progressed from inverse problems and motion control (Liang et al., 2019) to dry-frictional contact and robot-assisted dressing (Li et al., 2022b). Subsequent systems further improve intersection handling and cloth-rigid coupling (Yu et al., 2023), and extend differentiable simulation to soft-rigid and soft-cloth two-way coupling (Liu et al., 2024).

Differentiable simulation has been widely used to optimize trajectories and other task variables (Huang et al., 2021; Li et al., 2022a; Wang et al., 2024; Zhang et al., 2024). Beyond direct trajectory optimization, DiffSkill uses full-state gradient-based trajectory optimization to generate demonstrations for observation-conditioned skills (Lin et al., 2022). ILD instead directly backpropagates expert state-trajectory matching losses through differentiable dynamics to optimize policy parameters (Chen et al., 2023a). More generally, differentiable simulation has also been used to optimize neural controllers or policies directly from task objectives (Li et al., 2022b; Xu et al., 2022; Xing et al., 2025). These approaches extend differentiable physics from direct trajectory optimization to reusable policy learning, but typically rely on trajectory supervision or simulator-derived policy states. We instead retain complete garment states only for physical evaluation and gradient propagation, while learning a shared trajectory solver directly from task observations.

3 Preliminaries and Problem Formulation

3.1 Differentiable Cloth Simulation

Differentiable cloth simulation models garment dynamics while providing derivatives of simulated cloth states with respect to inputs and controls (Liang et al., 2019; Li et al., 2022b). Let $\mathbf{x}_t$ denote the complete set of cloth vertex positions at time step t, $\mathbf{u}_t$ the prescribed garment attachment positions, and $\mathcal{B}$ the body and environmental geometry. For compactness, we express the dynamic dependence on velocity through consecutive cloth configurations and abstract one simulation step as:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{u}_t; \mathcal{B}), \qquad t = 0, \ldots, T-1. \tag{1}$$

Differentiating the implicit simulation step yields local sensitivities with respect to the preceding cloth configurations, $\partial \mathbf{x}_{t+1}/\partial \mathbf{x}_t$ and $\partial \mathbf{x}_{t+1}/\partial \mathbf{x}_{t-1}$, as well as the prescribed attachment positions, $\partial \mathbf{x}_{t+1}/\partial \mathbf{u}_t$. These derivatives allow changes in the attachment positions to propagate through subsequent cloth states over the rollout.

3.2 Problem Formulation

We consider robot-assisted dressing across a family of task configurations with different arm poses. A dressing configuration c is defined by $c = (\mathbf{x}_0^c, \mathbf{o}^c, \mathcal{B}^c, \mathbf{x}_{\text{goal}}^c)$, where $\mathbf{x}_0^c$ denotes the initial garment state, $\mathbf{o}^c$ contains the task information available to the trajectory solver, $\mathcal{B}^c$ describes the corresponding body and interaction geometry, and $\mathbf{x}_{\text{goal}}^c$ specifies the desired garment configuration. In our setting, $\mathbf{o}^c$ includes a partial garment point-cloud observation from the dressed-side view, together with the arm configuration and the initial gripper states. The arm pose is fixed during each dressing rollout and varies across configurations.

Our objective is to learn a shared trajectory solver $\pi_\theta(\cdot)$, where $\boldsymbol{\tau}^c = \pi_\theta(\mathbf{o}^c)$ parameterizes the dual-gripper dressing trajectory predicted for configuration c. For each configuration, the predicted trajectory is evaluated through differentiable cloth simulation under $\mathbf{x}_0^c$ and $\mathcal{B}^c$. We use $\mathcal{L}_{\text{phys}}^c(\boldsymbol{\tau}^c)$ to denote the complete physics-guided task objective, including target matching, dressing progress, garment deformation, and body-clearance requirements.

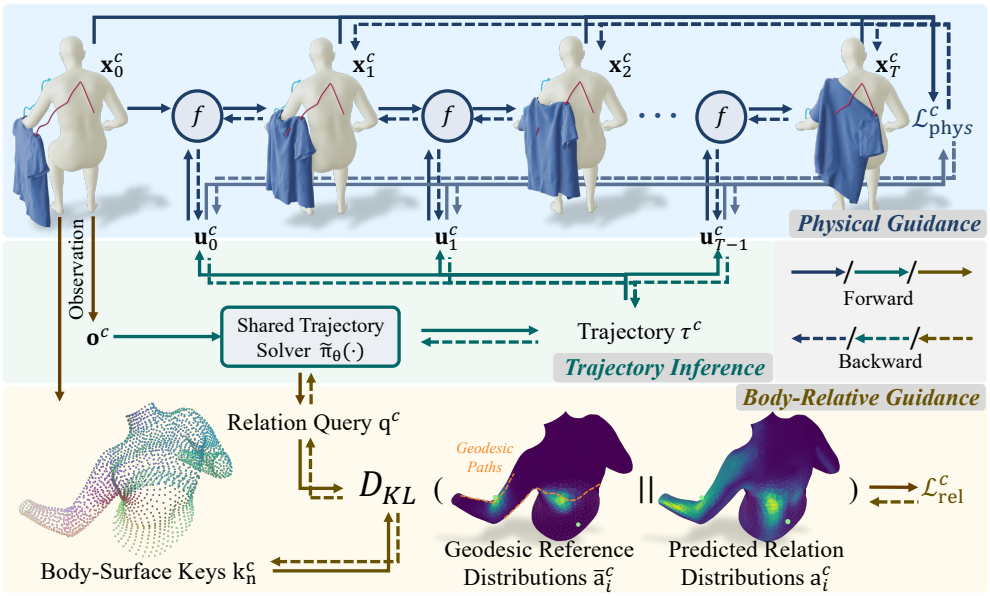

Figure 1: **Overview of GwoP for robot-assisted dressing. Top and center:** Differentiable physical guidance back-propagates physical objectives from full-state cloth rollouts to the observation-conditioned trajectory solver. **Bottom:** Body-relative guidance aligns jointly predicted body-surface relations with geodesic references, providing shared structural supervision without prescribing trajectory geometry.

The problem is therefore to learn a shared solver whose predictions minimize the expected physical dressing objective across arm configurations:

$$\min_{\theta} \; \mathbb{E}_c \left[\mathcal{L}^c_{\text{phys}} \left(\pi_\theta(\mathbf{o}^c) \right) \right]. \tag{2}$$

4 Guiding without Prescribing

Realizing the shared solver in Eq. 2 requires both physical guidance from complete simulated states and structure shared across geometrically diverse arm poses. GwoP addresses these requirements through differentiable physical guidance and auxiliary body-relative guidance.

4.1 Differentiable Physical Guidance Framework

Trajectory prediction and physical evaluation use different information in GwoP. The solver predicts $\boldsymbol{\tau}^c$ only from the task observation $\mathbf{o}^c$. The predicted trajectory is differentiably discretized into the prescribed garment attachment positions $\mathbf{u}^c_{0:T}$ over the simulation horizon, preserving the gradient path from the controls back to $\boldsymbol{\tau}^c$. Starting from $\mathbf{x}^c_0$, the simulator applies these controls sequentially according to Eq. 1, producing the rollout $\mathbf{x}^c_{1:T}$. The physical objective $\mathcal{L}^c_{\text{phys}}$ evaluates the simulated rollout and controls with respect to $\mathbf{x}^c_{\text{goal}}$ and the physical interaction requirements of the task. Hence, the simulator and physical objective can access the additional state information required for evaluation without introducing it into the forward input of the trajectory solver.

Although the forward information pathways are separated, physical evaluation remains differentiably connected to the solver through the predicted trajectory. The gradient of the physical objective with respect to the solver parameters can therefore be written as:

$$\nabla_\theta \mathcal{L}^c_{\text{phys}} = \left(\frac{\partial \boldsymbol{\tau}^c}{\partial \theta} \right)^\top \left(\frac{\partial \mathbf{u}^c_{0:T}}{\partial \boldsymbol{\tau}^c} \right)^\top \left[\left(\frac{\partial \mathbf{x}^c_{1:T}}{\partial \mathbf{u}^c_{0:T}} \right)^\top \nabla_{\mathbf{x}^c_{1:T}} \mathcal{L}^c_{\text{phys}} + \nabla_{\mathbf{u}^c_{0:T}} \mathcal{L}^c_{\text{phys}} \right]. \tag{3}$$

The first term propagates objectives defined on the simulated garment states through the differentiable cloth dynamics, where $\partial \mathbf{x}^c_{1:T} / \partial \mathbf{u}^c_{0:T}$ is obtained by chaining the local sensitivities in Eq. 1

over the rollout. The second term accounts for objectives that act directly on the attachment trajectory. Together, they provide first-order guidance on how the current predicted trajectory should change according to its physical consequences.

During training, the observation-conditioned solver is therefore followed by differentiable cloth simulation and physical evaluation, allowing the additional physical information to shape the solver through gradient updates. At inference, differentiable cloth rollouts and training-objective optimization are not required, and the learned solver directly predicts a dressing trajectory from $\mathbf{o}^c$.

4.2 Body-Relative Guidance

The physical objective provides configuration-dependent guidance for trajectory generation, but sharing a single solver across arm poses requires additional structure for organizing geometrically diverse solutions. Effective dressing trajectories may be nearly direct for some configurations, while others require substantial detours and intermediate direction changes. We therefore seek a pose-adaptive structural reference that captures common task structure across arm configurations despite substantial variation in trajectory geometry.

We use the relation between the manipulation process and the body surface as this reference. However, body-relative relations capture only structural information about the manipulation process and are not sufficient to determine the trajectory geometry. Using them as an intermediate representation for trajectory generation could unnecessarily restrict the trajectory solution space. Instead of treating a body-surface path as a trajectory target, GwoP uses body-relative relation prediction as an auxiliary task on the representation used for trajectory generation. During training, we augment the trajectory solver to output both the trajectory and auxiliary body-relative relations $\widetilde{\pi}_\theta(\mathbf{o}^c) = (\boldsymbol{\tau}^c, \mathbf{q}^c)$, where $\mathbf{q}^c = \{\mathbf{q}_i^c\}_{i=1}^{2M}$ contains M relation queries for each of the two grippers. Under this formulation, the physical objective retains a direct optimization path to the trajectory output $\boldsymbol{\tau}^c$, while the auxiliary relation objective acts on the representation shared with trajectory prediction.

To define the body-relative relation, we sample N body-surface points and encode them into normalized body-surface keys $\mathbf{k}^c = \{\mathbf{k}_n^c\}_{n=1}^N$. Each relation query induces a predicted relation distribution over the body surface:

$$a_{i,n}^c = \frac{\exp\big((\mathbf{q}_i^c)^\top \mathbf{k}_n^c / T_a\big)}{\sum_{j=1}^N \exp\big((\mathbf{q}_i^c)^\top \mathbf{k}_j^c / T_a\big)}, \tag{4}$$

where T_a is a temperature parameter. We construct the corresponding reference from a geodesic path on the SMPL body surface (Loper et al., 2023) between the initial and target regions, and convert its ordered locations into geodesic reference distributions $\bar{\mathbf{a}}_i^c$. The relation objective is:

$$\mathcal{L}_{\text{rel}}^c = \frac{1}{2M} \sum_{i=1}^{2M} D_{\text{KL}}\left(\bar{\mathbf{a}}_i^c \,\|\, \mathbf{a}_i^c\right). \tag{5}$$

This reference indicates which body-surface regions are relevant at different stages of manipulation, but does not prescribe the gripper trajectory. The trajectory direction, curvature, and detours remain optimized through physical objectives.

The benefit of this auxiliary formulation follows from how the two objectives jointly shape the trainable representation. Since the trajectory and relation outputs share an internal representation, the gradient received by the trainable parameters θ contains contributions through both relation branches:

$$\nabla_\theta \mathcal{L}^c = \left(\frac{\partial \boldsymbol{\tau}^c}{\partial \theta}\right)^\top \nabla_{\boldsymbol{\tau}^c} \mathcal{L}_{\text{phys}}^c + \lambda_{\text{rel}} \left[\left(\frac{\partial \mathbf{q}^c}{\partial \theta}\right)^\top \nabla_{\mathbf{q}^c} \mathcal{L}_{\text{rel}}^c + \left(\frac{\partial \mathbf{k}^c}{\partial \theta}\right)^\top \nabla_{\mathbf{k}^c} \mathcal{L}_{\text{rel}}^c\right]. \tag{6}$$

The first term learns representations that support physically effective trajectory generation, while the two relation-gradient terms introduce body-relative structure through the query and body-key branches. Thus, relation supervision can influence trajectory learning without directly constraining trajectory coordinates or replacing the physical optimization signal. We therefore augment the physical objective with the auxiliary body-relative objective during training:

$$\mathcal{L}(\theta) = \mathbb{E}_c\left[\mathcal{L}_{\text{phys}}^c + \lambda_{\text{rel}} \mathcal{L}_{\text{rel}}^c\right]. \tag{7}$$

Both objectives are jointly back-propagated through the trainable network. The relation objective updates both the query branch and the body-key encoder, while the query branch shares the representation used for trajectory prediction. Together with the physical guidance framework in Sec. 4.1, the two objectives guide different aspects of trajectory learning: the physical objective evaluates whether the predicted trajectory is effective for the current arm configuration, while the relation objective provides shared structural supervision across arm configurations.

5 Experiments

5.1 Experimental Setup

Dataset and task configurations. We construct a robot-assisted dressing benchmark with different poses of the dressed-side arm. The human body is represented by an SMPL model, with capsule and sphere primitives used for garment-body collision. Arm poses are parameterized by the shoulder angle ϕ_s and elbow angle ϕ_e. We use $\phi_s \in [0°, 50°]$ and $\phi_e \in [0°, 80°]$ at $2°$ intervals, yielding $1{,}066$ arm configurations. A smaller ϕ_s raises the upper arm toward the horizontal, whereas a larger ϕ_e corresponds to greater elbow flexion. We uniformly select 25 configurations for training and use the remaining $1{,}041$ unseen configurations for evaluation. The evaluation set contains both interpolated and extrapolated configurations relative to the training subset.

Implementation details. We use DiffCloth (Li et al., 2022b) as the differentiable cloth simulator. The garment is a short-sleeved open-front shirt represented by a triangular mesh with 4,455 vertices and 8,710 faces, with two fixed attachment points near the shoulder region manipulated by the grippers. The simulation time step is $1/160\,\mathrm{s}$ and each rollout contains 800 steps. Cloth density, stretching stiffness, and bending stiffness are set to 0.4, 600, and 1.0, respectively. Cloth-body collision and self-collision are enabled with a body friction coefficient of 0.1. The task observation $\mathbf{o}^c \in \mathbb{R}^{320}$ concatenates 256D body and 32D partial-garment PointNet (Qi et al., 2017) features, 6D initial grasping-point positions, and 26D joint-based arm-configuration features. The fused condition is processed by two AdaLN blocks (Peebles & Xie, 2023) to produce the trajectory and relation outputs. The gripper trajectories are parameterized by Bézier curves, with the network predicting 72D bounded trajectory parameters and 24 32D relation queries, resulting in 840 output dimensions in total. We pretrain relation prediction for 10 epochs before joint differentiable training. Adam is used with an initial learning rate of 5×10^{-4}, cosine-decayed to 2.5×10^{-4}.

Evaluation metrics. We evaluate dressing completion, garment deformation, and computational efficiency using six metrics. **Elbow-Passage Success** measures whether the garment progresses beyond the elbow. We use elbow passage as the success threshold because the elbow is a common bottleneck at which garments can catch or dressing progress can stall (Erickson et al., 2018; Wang et al., 2023). **Dressing Progress** measures the maximum normalized advancement of the garment along the wrist-elbow-shoulder path. **Edge Violation** and **Area Violation** measure the maximum excessive edge-length and triangle-area growth over the execution, respectively. **Training Time** reports the wall-clock time required to train each learning-based method. **Target Input** indicates whether the target garment state is required at inference.

5.2 Comparison with Baselines

Baselines. We compare GwoP against three representative paradigms for generating dressing trajectories, covering geometric planning, model-free policy learning, and learned dynamics with online planning. **Heuristic** represents rule-based trajectory construction from body geometry, providing a reference for what can be achieved using geometric priors alone. **OnePolicy** (Wang et al., 2023) represents model-free policy learning for dressing across diverse arm poses, directly learning actions from interaction and combining pose-specific policies through distillation. **TD-MPC2** (Hansen et al., 2024) represents model-based control, learning latent dynamics and repeatedly planning short-horizon actions from the current observation. Together, these baselines contrast GwoP with methods that rely respectively on hand-designed geometry, sampled policy learning, and learned dynamics for planning. All methods are evaluated from the same garment initial states and arm configurations.

Table 1: Quantitative comparison of dressing methods across arm configurations. We compare elbow passage, dressing progress, garment deformation violations, and training time. Best and second-best quantitative results are shown in bold and underlined, respectively.

Method	Elbow Pass ↑	Progress ↑	Edge Viol. ↓	Area Viol. ↓	Train. Time (h) ↓	Target Input
Heuristic	**98.0%**	<u>59.3%</u>	0.0149	0.0348	N/A	Yes
OnePolicy (Wang et al., 2023)	92.7%	48.3%	0.0228	0.0464	332.29	No
TD-MPC2 (Hansen et al., 2024)	19.8%	18.0%	<u>0.0118</u>	**0.0228**	<u>149.35</u>	No
Ours	<u>97.1%</u>	**74.7%**	**0.0115**	<u>0.0243</u>	**88.56**	No

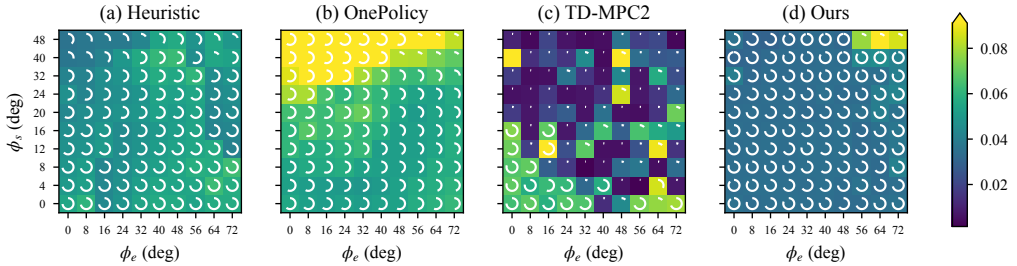

Figure 2: **Pose-wise dressing performance across arm configurations.** Each cell aggregates all evaluated sequences within the corresponding (ϕ_e, ϕ_s) range. The white circular arc represents mean dressing progress, while the background represents mean combined deformation violation, defined as the sum of Edge Violation and Area Violation for each execution. All methods share a global linear color scale, with the upper color limit set to the global 95th percentile for visual contrast. Values above this threshold are saturated.

Quantitative results. Table 1 and Fig. 2 jointly show that GwoP maintains effective garment advancement with limited deformation across a broad range of arm configurations. GwoP achieves the highest mean dressing progress of 0.747, with a 95th percentile of 0.042 for the combined deformation violation, while requiring the least training time among the learned methods. OnePolicy achieves relatively high progress in easier configurations, but its progress decreases as the elbow becomes more flexed and garment stretch increases as the upper arm is lowered, indicating reduced robustness across challenging poses. Heuristic remains comparatively smooth but advances the garment less overall. TD-MPC2 exhibits low deformation in many configurations, but this is largely associated with early termination before sufficient garment advancement, yielding a mean progress of only 0.180. In contrast, GwoP preserves consistently high progress over most of the evaluated arm-pose space, with degradation concentrated in only a few extreme boundary configurations.

Sim-to-sim qualitative results. To examine whether the generated trajectories remain effective under different cloth dynamics, we additionally transfer trajectories generated in DiffCloth to CLO (CLO Virtual Fashion Inc., 2026) without further optimization. As shown in Fig. 3, across representative arm configurations, the transferred GwoP trajectories continue to advance the garment along the arm and preserve the overall dressing behavior observed in DiffCloth. Differences in local deformation and contact evolution arise after changing simulators, but the GwoP trajectories remain effective in driving the garment toward the dressed configuration. These results provide qualitative evidence of cross-simulator transfer beyond the differentiable training environment.

5.3 Ablation Study

Design choices for training. We evaluate the roles of the two guidance sources and alternative ways of providing structural supervision. *w/o Physics Gradients* disables backpropagation through cloth dynamics while retaining forward simulation, relation supervision, and geometric trajectory supervision, testing the contribution of physical gradients beyond geometric guidance. *w/o Body-Relative Guidance* removes the body-relative guidance and trains the trajectory predictor without relation pretraining. *Trajectory Supervision* uses the same configuration but adds 10 epochs of

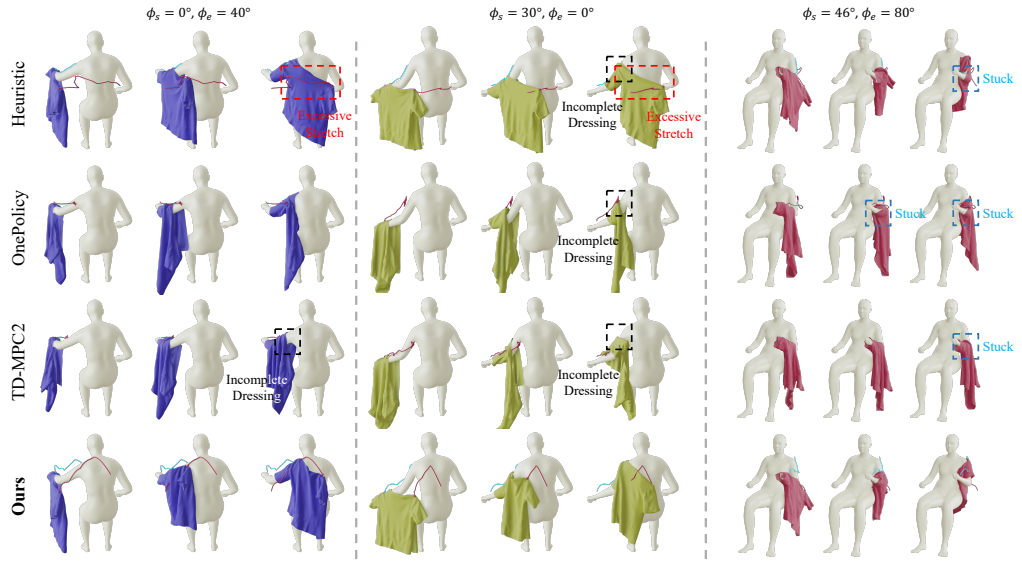

Figure 3: **Sim-to-sim transfer from DiffCloth to CLO.** Representative dressing trajectories generated in DiffCloth are directly executed in CLO without further optimization. GwoP maintains effective garment advancement after transfer, while the baselines exhibit incomplete dressing, excessive stretching, or stalled progress.

pretraining on heuristic gripper trajectories, matching the 10-epoch relation-pretraining budget of *Full* and testing direct trajectory supervision as an alternative to relation-level guidance. *Query-to-Trajectory* retains body-surface relation modeling but generates trajectory increments from the predicted queries, testing the benefit of using relations as auxiliary guidance rather than as a required intermediate representation.

Ablation results. Table 2 and Fig. 4 reveal the different roles of physical and body-relative guidance. Removing physical gradients substantially reduces dressing progress, indicating that geometric guidance alone is less effective at learning trajectories that account for the downstream consequences of cloth–body dynamics. *w/o Body-Relative Guidance* and *Trajectory Supervision* both remove relation-level guidance, differing only in whether 10 epochs of heuristic trajectory pretraining are applied. Without this pretraining, the solver can discover trajectories with relatively high progress, but the resulting solutions are less stable and incur larger deformation violations. Heuristic trajectory pretraining provides a more constrained initialization, improving elbow passing and reducing deformation, but directly prescribing trajectory geometry still yields limited progress and a less regular local $\mathcal{L}_{\text{phys}}$ landscape than *Full*. *Query-to-Trajectory* retains body-surface relation modeling but forces trajectory increments to be decoded from the predicted queries. Although this preserves relational information, the resulting coupling restricts trajectory generation and leads to lower task performance. In contrast, *Full* predicts relations and trajectories in parallel, using body-relative relations as auxiliary guidance while allowing physical gradients to optimize the trajectory directly. This combination yields the highest elbow-pass rate and dressing progress, together with a broader and smoother local low-loss region.

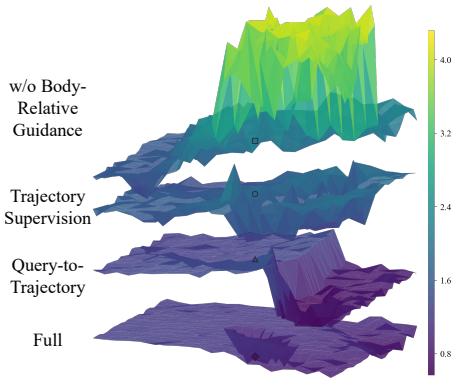

Figure 4: Local $\mathcal{L}_{\text{phys}}$ landscapes of four body-relative-guidance settings under two-dimensional perturbations of the learned solver parameters θ. All landscapes share the same color scale, with markers denoting the unperturbed solutions at $(\alpha, \beta) = (0, 0)$.

Table 2: Ablation study of the key training designs in GwoP. Best and second-best results are shown in bold and underlined, respectively.

Variant	Elbow Pass↑	Progress↑	Edge Viol.↓	Area Viol.↓
w/o Physics Gradients	82.0%	46.1%	0.0141	0.0326
w/o Body-Relative Guidance	87.3%	<u>61.2%</u>	0.0143	0.0272
Trajectory Supervision	<u>93.0%</u>	57.1%	**0.0107**	**0.0221**
Query-to-Trajectory	75.0%	54.4%	0.0128	0.0288
Full	**97.1%**	**74.7%**	<u>0.0115</u>	<u>0.0243</u>

5.4 Real-Robot Validation



Figure 5: **Real-robot execution of predicted dressing trajectories.** Representative trajectories are executed on two AgileX PiPER robot arms, progressively advancing the garment along the mannequin arm.

We further evaluate execution feasibility on two AgileX PiPER robot arms. Processed point-cloud observations are provided to the solver to predict the dressing trajectory, which is converted into robot joint commands through inverse kinematics. We evaluate 12 arm poses with 5 repeated executions per pose. Across the resulting 60 trials, 46 satisfy the elbow-passage success criterion. These results provide real-robot evidence that the learned solver can predict executable dressing trajectories from processed point-cloud observations, successfully advancing the garment across diverse arm poses.

6 Conclusion and Limitations

We presented **Guiding without Prescribing (GwoP)**, a differentiable trajectory learning framework for robot-assisted dressing. By separating full-state physical evaluation from observation-conditioned trajectory prediction, GwoP turns differentiable physical gradients into training guidance for a shared solver without requiring complete garment states at inference. Body-relative guidance further provides shared structure across arm poses without prescribing specific trajectory geometry, yielding a representation that is less sensitive to local parameter perturbations when learning diverse dressing trajectories. Experiments across unseen arm configurations show that these two forms of guidance support effective garment advancement with limited deformation. Cross-simulator transfer and real-robot experiments further provide evidence that the generated trajectories remain applicable beyond the differentiable training simulator.

Several limitations remain. First, real-world execution is open-loop after trajectory generation, making the system sensitive to initial-state errors and sim-to-real mismatch and preventing online correction when the garment deviates unexpectedly. Second, the learned solver is currently applicable to garments with structural characteristics similar to those used during training. Garments with substantially different structures require retraining. Extending the framework to broader garment structures and incorporating closed-loop feedback are important directions for future work.

AI USE STATEMENT

In this work, we used generative AI tools to assist with translation. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate or prove mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement the core methods, or interpret experimental results, and tasks involving qualitative or thematic data analysis are not applicable to this work.

Additionally, we used generative AI tools to aid in drafting and editing parts of the manuscript for clarity and readability, and as a research assistant for information retrieval, literature discovery and summarization, and terminology clarification. We also used generative AI tools for auxiliary programming and presentation tasks, including drafting and debugging visualization code, batch-processing and job submission scripts, and other supporting engineering utilities. The core algorithms and their implementations were developed by the authors.

We have reviewed all AI-assisted work. Literature-related information and scientific claims were verified against the original sources, and AI-assisted code was inspected and tested by the authors before use. Mathematical formulations, experimental results, and scientific claims were independently checked by the authors for correctness. We take responsibility for the final content of this work, including text, claims, code, or other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the implementation and evaluation details required to reproduce the main results in the Appendix and supplementary material. The Appendix specifies the trajectory parameterization, physical objectives, target and initial garment construction, network architecture, body-relative reference construction, simulator settings, pose configurations, training procedures, baseline adaptations, ablation settings, evaluation metrics, cross-simulator protocol, and real-robot setup. The supplementary material provides additional qualitative results and real-robot videos. Together, these materials document the experimental setup and implementation choices used to obtain the results reported in the main paper.

REFERENCES

Gerard Canal, Emmanuel Pignat, Guillem Alenyà, Sylvain Calinon, and Carme Torras. Joining high-level symbolic planning with low-level motion primitives in adaptive hri: Application to dressing assistance. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3273–3278, 2018. doi: 10.1109/ICRA.2018.8460606.

Siwei Chen, Xiao Ma, and Zhongwen Xu. Imitation learning as state matching via differentiable physics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7846–7855, June 2023a.

Siwei Chen, Yiqing Xu, Cunjun Yu, Linfeng Li, Xiao Ma, Zhongwen Xu, and David Hsu. Daxbench: Benchmarking deformable object manipulation with differentiable physics. In *ICLR*, 2023b.

CLO Virtual Fashion Inc. CLO: 3d fashion design software. https://www.clo3d.com/, 2026. Accessed: 2026-09-24.

Filipe de Avila Belbute-Peres, Kevin Smith, Kelsey Allen, Josh Tenenbaum, and J. Zico Kolter. End-to-end differentiable physics for learning and control. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/842424a1d0595b76ec4fa03c46e8d755-Paper.pdf.

Tao Du, Kui Wu, Pingchuan Ma, Sebastien Wah, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. Diffpd: Differentiable projective dynamics. *ACM Transactions on Graphics*, 41(2): 1–21, November 2021. ISSN 1557-7368. doi: 10.1145/3490168. URL http://dx.doi.org/10.1145/3490168.

Zackory Erickson, Alexander Clegg, Wenhao Yu, Greg Turk, C. Karen Liu, and Charles C. Kemp. What does the person feel? learning to infer applied forces during robot-assisted dressing. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6058–6065, 2017. doi: 10.1109/ICRA.2017.7989718.

Zackory Erickson, Henry M. Clever, Greg Turk, C. Karen Liu, and Charles C. Kemp. Deep haptic model predictive control for robot-assisted dressing. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4437–4444, 2018. doi: 10.1109/ICRA.2018.8460656.

Zackory Erickson, Henry M. Clever, Vamsee Gangaram, Greg Turk, C. Karen Liu, and Charles C. Kemp. Multidimensional capacitive sensing for robot-assisted dressing and bathing. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pp. 224–231, 2019. doi: 10.1109/ICORR.2019.8779542.

Yixing Gao, Hyung Jin Chang, and Yiannis Demiris. User modelling for personalised dressing assistance by humanoid robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1840–1845, 2015. doi: 10.1109/IROS.2015.7353617.

Yixing Gao, Hyung Jin Chang, and Yiannis Demiris. Iterative path optimisation for personalised dressing assistance using vision and force information. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4398–4403, 2016. doi: 10.1109/IROS.2016.7759647.

Artur Grigorev, Michael J Black, and Otmar Hilliges. Hood: Hierarchical graphs for generalized modelling of clothing dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16965–16974, 2023.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations (ICLR)*, 2024.

Alexis Yihong Hao, Yufei Wang, Navin Sriram Ravie, Bharath Hegde, David Held, and Zackory Erickson. Force-modulated visual policy for robot-assisted dressing with arm motions. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 5483–5505. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/hao25b.html.

Yuanming Hu, Jiancheng Liu, Andrew Spielberg, Joshua B. Tenenbaum, William T. Freeman, Jiajun Wu, Daniela Rus, and Wojciech Matusik. Chainqueen: A real-time differentiable physical simulator for soft robotics. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6265–6271. IEEE, May 2019. doi: 10.1109/icra.2019.8794333. URL http://dx.doi.org/10.1109/ICRA.2019.8794333.

Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. Difftaichi: Differentiable programming for physical simulation. *ICLR*, 2020.

Zhiao Huang, Yuanming Hu, Tao Du, Siyuan Zhou, Hao Su, Joshua B. Tenenbaum, and Chuang Gan. Plasticinelab: A soft-body manipulation benchmark with differentiable physics. *ICLR*, 2021.

Pedro Ildefonso, Pedro Remédios, Rui Silva, Miguel Vasco, Francisco S. Melo, Ana Paiva, and Manuela Veloso. Exploiting symmetry in human robot-assisted dressing using reinforcement learning. pp. 405–417, Berlin, Heidelberg, 2021. Springer-Verlag. ISBN 978-3-030-86229-9. doi: 10.1007/978-3-030-86230-5_32. URL https://doi.org/10.1007/978-3-030-86230-5_32.

Ariel Kapusta, Wenhao Yu, Tapomayukh Bhattacharjee, C. Karen Liu, Greg Turk, and Charles C. Kemp. Data-driven haptic perception for robot-assisted dressing. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 451–458, 2016. doi: 10.1109/ROMAN.2016.7745158.

Stelios Kotsovolis and Yiannis Demiris. Model predictive control with graph dynamics for garment opening insertion during robot-assisted dressing. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 883–890, 2024. doi: 10.1109/ICRA57147.2024.10611478.

Stelios Kotsovolis and Yiannis Demiris. Garment diffusion models for robot-assisted dressing. *IEEE Robotics and Automation Letters*, 10(2):1217–1224, 2025. doi: 10.1109/LRA.2024.3518104.

Sizhe Li, Zhiao Huang, Tao Du, Hao Su, Joshua B. Tenenbaum, and Chuang Gan. Contact points discovery for soft-body manipulations with differentiable physics. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=mmUA7_O9mjY.

Yifei Li, Tao Du, Kui Wu, Jie Xu, and Wojciech Matusik. Diffcloth: Differentiable cloth simulation with dry frictional contact. *ACM Transactions on Graphics*, 42(1):1–20, October 2022b. ISSN 1557-7368. doi: 10.1145/3527660. URL http://dx.doi.org/10.1145/3527660.

Junbang Liang, Ming Lin, and Vladlen Koltun. Differentiable cloth simulation for inverse problems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/28f0b864598a1291557bed248a998d4e-Paper.pdf.

Xingyu Lin, Zhiao Huang, Yunzhu Li, Joshua B. Tenenbaum, David Held, and Chuang Gan. Diffskill: Skill abstraction from differentiable physics for deformable object manipulations with tools. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=Kef8cKdHWpP.

Min Liu, Gang Yang, Siyuan Luo, and Lin Shao. Softmac: Differentiable soft body simulation with forecast-based contact model and two-way coupling with articulated rigid bodies and clothes. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 12008–12015, 2024. doi: 10.1109/IROS58592.2024.10801308.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. *SMPL: A Skinned Multi-Person Linear Model*. Association for Computing Machinery, New York, NY, USA, 1 edition, 2023. ISBN 9798400708978. URL https://doi.org/10.1145/3596711.3596800.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, October 2023.

Emmanuel Pignat and Sylvain Calinon. Learning adaptive dressing assistance from human demonstration. *Robotics and Autonomous Systems*, 93:61–75, 2017. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2017.03.017. URL https://www.sciencedirect.com/science/article/pii/S0921889016303669.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Tomoya Tamei, Takamitsu Matsubara, Akshara Rai, and Tomohiro Shibata. Reinforcement learning of clothing assistance with a dual-arm robot. In *2011 11th IEEE-RAS International Conference on Humanoid Robots*, pp. 733–738, 2011. doi: 10.1109/Humanoids.2011.6100915.

Lukas Twardon and Helge Ritter. Learning to put on a knit cap in a head-centric policy space. *IEEE Robotics and Automation Letters*, 3(2):764–771, 2018. doi: 10.1109/LRA.2018.2792153.

Yian Wang, Juntian Zheng, Zhehuan Chen, Zhou Xian, Gu Zhang, Chao Liu, and Chuang Gan. Thin-shell object manipulations with differentiable physics simulations. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 42647–42667, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/bb01ae312268181758299e4ca75837e5-Paper-Conference.pdf.

Yufei Wang, Zhanyi Sun, Zackory Erickson, and David Held. One policy to dress them all: Learning to dress people with diverse poses and garments. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023. doi: 10.15607/RSS.2023.XIX.008.

Eliot Xing, Vernon Luk, and Jean Oh. Stabilizing reinforcement learning in differentiable multiphysics simulation. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 91165–91198, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/e33a77969c26e5efdf3989ac46e7a036-Paper-Conference.pdf.

J. Xu, V. Makoviychuk, Y. Narang, F. Ramos, W. Matusik, A. Garg, and M. Macklin. Accelerated Policy Learning with Parallel Differentiable Simulation. In *ICLR*, 2022.

Kimitoshi Yamazaki, Ryosuke Oya, Kotaro Nagahama, Kei Okada, and Masayuki Inaba. Bottom dressing by a dual-arm robot using a clothing state estimation based on dynamic shape changes. *International Journal of Advanced Robotic Systems*, 13(1):5, 2016. doi: 10.5772/61930. URL https://doi.org/10.5772/61930.

Wenhao Yu, Ariel Kapusta, Jie Tan, Charles C. Kemp, Greg Turk, and C. Karen Liu. Haptic simulation for robot-assisted dressing. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6044–6051, 2017. doi: 10.1109/ICRA.2017.7989716.

Xinyuan Yu, Siheng Zhao, Siyuan Luo, Gang Yang, and Lin Shao. Diffclothai: Differentiable cloth simulation with intersection-free frictional contact and differentiable two-way coupling with articulated rigid bodies. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 400–407. IEEE, October 2023. doi: 10.1109/iros55552.2023.10341573. URL http://dx.doi.org/10.1109/IROS55552.2023.10341573.

Fan Zhang, Antoine Cully, and Yiannis Demiris. Personalized robot-assisted dressing using user modeling in latent spaces. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3603–3610, 2017. doi: 10.1109/IROS.2017.8206206.

Fan Zhang, Antoine Cully, and Yiannis Demiris. Probabilistic real-time user posture tracking for personalized robot-assisted dressing. *IEEE Transactions on Robotics*, 35(4):873–888, 2019. doi: 10.1109/TRO.2019.2904461.

Yutong Zhang, Fei Liu, Xiao Liang, and Michael Yip. Achieving autonomous cloth manipulation with optimal control via differentiable physics-aware regularization and safety constraints. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9931–9938. IEEE, May 2024. doi: 10.1109/icra57147.2024.10611111. URL http://dx.doi.org/10.1109/ICRA57147.2024.10611111.

A Implementation Details

A.1 Trajectory Parameterization

Each gripper follows a piecewise cubic Bézier trajectory with four segments. For gripper $g \in \{1, 2\}$ and segment $s \in \{1, \ldots, 4\}$,

$$\mathbf{b}_s^g(u) = \sum_{k=0}^{3} \binom{3}{k} (1-u)^{3-k} u^k \mathbf{P}_{s,k}^g, \qquad u \in [0, 1]. \tag{8}$$

Each segment contains one starting point and three learned control points. The first segment starts from the initial attachment position, $\mathbf{P}_{1,0}^g = \mathbf{u}_{\text{init}}^g$, while consecutive segments share endpoints, i.e., $\mathbf{P}_{s,0}^g = \mathbf{P}_{s-1,3}^g$ for $s = 2, 3, 4$. This guarantees C^0 continuity between segments.

For each learned control point, the controller predicts a 3D displacement, giving $2 \times 4 \times 3 \times 3 = 72$ trajectory parameters in total. For a raw prediction $\mathbf{r}_{s,k}^g$, the bounded displacement is $\mathbf{\Delta}_{s,k}^g = 2\tanh(\mathbf{r}_{s,k}^g)$, and control points are constructed recursively as $\mathbf{P}_{s,k}^g = \mathbf{P}_{s,k-1}^g + \mathbf{\Delta}_{s,k}^g$.

For an 800-step rollout, each Bézier segment spans 200 simulation steps. Sampling the four segments and removing duplicated boundary samples produces 801 paired attachment configurations, including the initial configuration. At simulation step $t = 0, \ldots, 799$, the two gripper positions at sample $t + 1$ are concatenated as the attachment positions.

A.2 PHYSICAL LOSS

Let $\mathbf{x}_i^t$ denote garment vertex i at simulation step t, with $t = 0$ denoting the initial state and T the rollout length. We use $[z]_+ = \max(z, 0)$ throughout.

Target and deformation objectives. For a garment with $|\mathcal{V}|$ vertices, the target loss measures the final discrepancy from the target garment:

$$L_{\text{target}} = \frac{1}{3|\mathcal{V}|} \sum_{i \in \mathcal{V}} \left\| \mathbf{x}_i^T - \mathbf{x}_i^{\text{target}} \right\|_2^2 .$$

To penalize excessive deformation, let $l_e^t = \|\mathbf{x}_j^t - \mathbf{x}_i^t\|_2$ for edge $e = (i, j)$ and $A_f^t = \frac{1}{2}\|(\mathbf{x}_b^t - \mathbf{x}_a^t) \times (\mathbf{x}_c^t - \mathbf{x}_a^t)\|_2$ for face $f = (a, b, c)$. With tolerance $\epsilon = 0.05$,

$$L_{\text{stretch}} = \frac{1}{T} \sum_{t=1}^{T} \left[\frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \frac{[l_e^t - (1+\epsilon) l_e^0]_+}{l_e^0} + \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} \frac{[A_f^t - (1+\epsilon) A_f^0]_+}{A_f^0} \right]. \tag{9}$$

Progress objective. Let $\mathcal{H}$ denote the sleeve-opening vertices and $\mathbf{h}_t = |\mathcal{H}|^{-1} \sum_{i \in \mathcal{H}} \mathbf{x}_i^t$ their center. The arm guidance path consists of the forearm and upper-arm segments $(\mathbf{a}_k, \mathbf{b}_k)$, $k \in \{1, 2\}$, with lengths $d_k = \|\mathbf{b}_k - \mathbf{a}_k\|_2$, cumulative lengths $C_k = \sum_{j=1}^{k} d_j$, and total length $D = C_2$. At step t, the desired normalized progress is $\bar{P}_t = (t-1)/(T-1)$. We project $\mathbf{h}_t$ onto the corresponding scheduled arm segment to obtain its normalized progress P_t, and minimize $L_{\text{progress}} = T^{-1} \sum_{t=1}^{T} (P_t - \bar{P}_t)^2$.

Collision objective. We penalize grasp-point collisions with the body's sphere and capsule primitives. For a sphere s with center $\mathbf{o}_s$ and radius r_s, $v_s(\mathbf{g}) = [r_s + \delta - \|\mathbf{g} - \mathbf{o}_s\|_2]_+$, where $\delta = 0.1$. For a capsule k with endpoints $\mathbf{a}_k, \mathbf{b}_k$ and radius r_k, we use the simulator's piecewise capsule-distance convention and denote the corresponding margin violation by $v_k(\mathbf{g})$. The collision loss is

$$L_{\text{collision}} = \frac{1}{|\mathcal{G}|} \sum_{\mathbf{g} \in \mathcal{G}} \left[\left(\max_{s \in \mathcal{S}} v_s(\mathbf{g}) \right)^2 + \left(\max_{k \in \mathcal{C}} v_k(\mathbf{g}) \right)^2 \right],$$

where $\mathcal{G}$ contains all grasp positions over the rollout.

Complete physical objective. $\mathcal{L}_{\text{phys}}$ denotes the complete physics-guided task objective:

$$\mathcal{L}_{\text{phys}} = \lambda_{\text{target}} L_{\text{target}} + \lambda_{\text{stretch}} L_{\text{stretch}} + \lambda_{\text{progress}} L_{\text{progress}} + \lambda_{\text{collision}} L_{\text{collision}}. \tag{10}$$

We use $\lambda_{\text{target}} = 0.4$, $\lambda_{\text{stretch}} = 80$, $\lambda_{\text{progress}} = 1$, and $\lambda_{\text{collision}} = 1000$. During joint training, the total objective is $\mathcal{L} = \mathcal{L}_{\text{phys}} + \lambda_{\text{rel}} \mathcal{L}_{\text{rel}}$.

A.3 TARGET AND INITIAL GARMENT CONSTRUCTION

For each arm pose, we construct the one-sided dressed target independently. We first obtain a dressed garment configuration with HOOD (Grigorev et al., 2023) using the corresponding posed SMPL body. We then approximate the same SMPL configuration with a set of collision spheres and capsules. The collision primitives on the undressed side are removed, allowing the corresponding side of the garment to hang naturally. The resulting garment is simulated forward in DiffCloth for 300 steps until reaching a stable configuration, which is used as the target garment for that pose.

The initial garment state is constructed separately for the same pose. The garment is positioned such that the sleeve opening is placed near the hand, after which DiffCloth is again simulated forward for 300 steps to obtain a stable initial configuration. The initial and target garments use the same garment topology and vertex ordering, so vertex-wise target supervision is directly defined.

The target garment is used to compute the terminal garment objective and the body-relative reference paths. The learned solver does not receive target garment coordinates as input. The geometric Heuristic additionally uses the target positions of the two attachment vertices to define its terminal surface projections and normal offsets, and therefore requires target input at inference, as reported in Table 1.

A.4 Network Architecture

The task observation consists of a 32D partial-garment descriptor, a 256D body descriptor, the 6D initial attachment positions, and 26D arm-configuration features, giving $\mathbf{o}^c \in \mathbb{R}^{320}$.

The partial-garment descriptor is produced by a pretrained and frozen PointNet-style encoder from 768 visible sleeve and shoulder points. Pointwise features are extracted with layer widths $[3, 64, 128, 256]$, followed by mean and max pooling and fully connected layers with widths $[512, 256, 128]$. Two projection heads produce a 3D center descriptor and a 29D shape descriptor, forming the final 32D garment feature.

The body encoder processes a fixed active SMPL patch containing $N = 2530$ vertices. Non-interacting peripheral regions, such as the fingers, face, and feet, are excluded from this representation. Each point contains its normalized 3D position and surface normal. A shared MLP with layer widths $[6, 64, 128, 256]$ produces local features, whose max-pooled representation forms the 256D body descriptor. The local features are also projected to normalized 32D body-surface keys used by the relation branch.

The fused 320D condition is mapped to a 128D embedding and used to modulate two AdaLN blocks. A 16D Gaussian latent vector is first projected to 128D; both AdaLN blocks apply condition-dependent scale and shift, with a residual connection in the second block. The resulting 128D shared representation is mapped by one linear output layer to 840 values. For each of the $2 \times 4 \times 3 = 24$ learned control points, the network predicts a 3D trajectory displacement and a 32D relation query, giving 72 trajectory parameters and 24×32 relation-query values in total. Trajectory and relation predictions therefore arise in parallel from the same shared representation.

A.5 Body-Relative Reference Construction

The body surface is represented by a fixed task-relevant subset of $N = 2530$ SMPL vertices shared across all poses. The retained vertices preserve consistent identities across poses, while their positions and normals vary with the body configuration.

For each gripper, the reference path connects the active body vertices nearest to its initial and target attachment positions. We construct an undirected graph from the triangle edges of the active SMPL patch, use posed edge lengths as weights, and compute the shortest path with Dijkstra's algorithm. Each gripper has $M = 12$ relation stages corresponding to the three learned control points of its four Bézier segments. The geodesic path is therefore divided into 12 consecutive subpaths of equal arc length.

Across the two grippers, the resulting $2M$ stages are indexed consistently with Eq. 5.

We use $T_a = 0.08$ for the relation softmax. For body vertex $\mathbf{v}_j$, let $d^c_{i,j}$ be its minimum Euclidean distance to the i-th geodesic subpath $\mathcal{P}^c_i$. The reference distribution is defined by:

$$\bar{a}^c_{i,j} = \frac{\exp[-(d^c_{i,j})^2/(2\sigma^2)]}{\sum_{n=1}^{N} \exp[-(d^c_{i,n})^2/(2\sigma^2)]}, \tag{11}$$

where $\sigma = 0.45$. Reference paths and distributions are constructed separately for every training pose and are used during relation pretraining and joint training.

B Simulation and Task Setup

B.1 Cloth Simulator

We use DiffCloth with time step $\Delta t = 1/160$ and 800 simulation steps per rollout, corresponding to 5 s of simulated time. Cloth density, stretching stiffness, and bending stiffness are set to 0.4, 600, and 1.0, respectively. Cloth-body collision and cloth self-collision are enabled, with body friction coefficient 0.1. Wind and floor interaction are disabled. The garment is initialized with zero velocity, and no additional stabilization is performed during each rollout.

B.2 Body and Garment Models

The human body is represented by a female SMPL model. Each scene specifies the body pose, global orientation, translation, and shape parameters. The SMPL surface is used for body-relative references, while physical cloth-body collision is handled by the corresponding sphere and capsule primitives.

The short-sleeved open-front garment contains 4,455 vertices and 8,710 triangular faces. Two predefined garment vertices near the shoulder region, with zero-based indices 859 and 2291, are manipulated by the two grippers. Their initial positions are taken from the initial garment mesh, and their prescribed positions are updated at every simulation step according to the predicted trajectories.

B.3 Pose Configurations

Arm poses are parameterized by shoulder angle ϕ_s and elbow angle ϕ_e. We discretize $\phi_s \in \{0^\circ, 2^\circ, \ldots, 50^\circ\}$ and $\phi_e \in \{0^\circ, 2^\circ, \ldots, 80^\circ\}$, yielding $26 \times 41 = 1{,}066$ arm configurations. The 25 training configurations are the Cartesian product:

$$\mathcal{C}_{\text{train}} = \{0^\circ, 12^\circ, 24^\circ, 36^\circ, 48^\circ\} \times \{0^\circ, 16^\circ, 32^\circ, 48^\circ, 64^\circ\}.$$

The remaining 1,041 configurations are held out from GwoP training and used for evaluation. Configurations with $\phi_s \leq 48^\circ$ and $\phi_e \leq 64^\circ$ lie within the rectangular range spanned by the training grid and are treated as interpolation poses; after excluding the 25 training poses, this gives 800 interpolation configurations. The remaining 241 configurations, with $\phi_s = 50^\circ$ or $\phi_e > 64^\circ$, form the extrapolation set.

Each evaluation configuration is repeated five times, resulting in $5{,}205$ dressing rollouts. The 25 training configurations are excluded from all evaluation statistics. In the pose-wise visualization, all evaluated configurations falling into the corresponding plotted pose bin are aggregated, together with their five repeated trials.

C Training and Baseline Details

C.1 GwoP Training

GwoP is first pretrained for 10 epochs using only body-relative relation supervision with unit weight, followed by joint differentiable training with $\mathcal{L}_{\text{phys}} + \lambda_{\text{rel}}\mathcal{L}_{\text{rel}}$, where $\lambda_{\text{rel}} = 0.1$. Relation pretraining uses the same pose-conditioned training samples as joint training and does not execute cloth simulation. The trajectory output receives no direct trajectory supervision during this stage.

Both stages use Adam with initial learning rate 5×10^{-4}, zero weight decay, gradient-norm clipping at 1, and cosine learning-rate decay. Gradients are accumulated over up to 32 training samples before each optimizer update. During joint training, one predicted trajectory per pose is evaluated through an 800-step differentiable cloth rollout.

The full model and relation-based variants use 10 epochs of relation pretraining followed by 22 epochs of joint training. *Trajectory Supervision* instead uses 10 epochs of heuristic trajectory pretraining, while *w/o Body-Relative Guidance* omits pretraining.

C.2 Heuristic

The Heuristic constructs each gripper trajectory from a shortest path on the SMPL surface together with a signed normal offset. For gripper $g \in \{1, 2\}$, let $\mathbf{u}^g_{\text{init}}$ and $\mathbf{u}^g_\star$ denote its attachment positions in the initial and target garments, respectively. Both positions are projected onto the active SMPL surface, yielding projected points $\mathbf{p}^g_0, \mathbf{p}^g_\star$ and corresponding interpolated surface normals $\mathbf{n}^g_0, \mathbf{n}^g_\star$. Their signed normal offsets are $h^g_0 = (\mathbf{u}^g_{\text{init}} - \mathbf{p}^g_0)^\top \mathbf{n}^g_0$ and $h^g_\star = (\mathbf{u}^g_\star - \mathbf{p}^g_\star)^\top \mathbf{n}^g_\star$.

The active body mesh is treated as a weighted graph whose edge weights are surface-edge lengths, and Dijkstra's algorithm computes the shortest path between the two projected endpoints. After uniform arc-length sampling, the gripper trajectory at normalized progress $\rho \in [0, 1]$ is

$$\boldsymbol{\eta}^g(\rho) = \mathbf{p}^g(\rho) + [(1-\rho)h^g_0 + \rho h^g_\star]\, \mathbf{n}^g(\rho), \qquad g \in \{1, 2\}. \tag{12}$$

The two gripper paths are sampled at the same 801 progress values and directly supplied to the simulator. Because the target attachment positions are used to construct the trajectory endpoint and its normal offset, this baseline requires target garment information at inference.

C.3 ONEPOLICY

We adapt OnePolicy (Wang et al., 2023) as the model-free policy-learning baseline. The policy receives a 1024×6 point cloud, where each point contains 3D position and a three-way semantic label for arm, garment, or robot attachment geometry. The 30 active attachment points are retained explicitly, while the remaining points are sampled from the arm and garment geometry. The policy predicts a 3D translational displacement, which is applied to the fixed 30-vertex attachment set. The displacement norm is limited to 0.75 simulation units (5 cm), and each action is executed over 20 DiffCloth substeps.

The actor and two critics use PointNet++ encoders. SAC is trained with learning rate 10^{-4}, discount factor $\gamma = 0.99$, batch size 16, replay capacity 4×10^5, and 1,000 random warm-up steps. Each episode contains at most 40 policy steps, corresponding to 800 DiffCloth simulation steps.

Following OnePolicy, each of the 2 pose dimensions is divided into 3 intervals, producing 9 pose subregions. 5 poses are randomly sampled within each subregion to train one corresponding teacher policy, after which the nine teachers are distilled into a shared student. Distillation is optimized jointly with SAC by matching the Gaussian means and standard deviations of the student and frozen teachers, with distillation weight $\lambda_{\text{distill}} = 0.01$. The shared student is trained over 45 discrete arm-pose scenes, and the checkpoint used for evaluation is obtained after approximately 3.43×10^5 environment steps.

For the final comparison, OnePolicy is evaluated on the same 1,041 arm configurations as GwoP, using the same initial garment states and five repeated trials for each configuration.

C.4 TD-MPC2

We adapt TD-MPC2 (Hansen et al., 2024) to the same DiffCloth dressing environment. It receives the same 1024×6 point-cloud representation as the OnePolicy baseline. The PointNet++ encoder contains two set-abstraction layers: a 5 cm neighborhood with 32 samples and MLP $[6, 64, 128]$, followed by a 10 cm neighborhood with 64 samples and MLP $[131, 128, 256]$. Global pooling and a projection network produce a 512D latent state. Encoder micro-batches of 16 and gradient checkpointing are used to reduce memory consumption.

The agent predicts a normalized 3D action $\mathbf{a}_t \in [-1, 1]^3$, mapped to attachment displacement $\Delta\mathbf{p}_t = (0.75/\sqrt{3})\,\text{clip}(\mathbf{a}_t, -1, 1)$. The displacement is broadcast to the fixed 30-vertex attachment set and executed over 20 DiffCloth substeps.

TD-MPC2 uses a planning horizon of 3 actions and replans at every environment step from the latest point-cloud observation. Each planning step uses 6 iterations, 512 candidate trajectories, 64 elites, and 24 policy-prior trajectories; only the first optimized action is executed before replanning. The latent model and policy use 512D hidden representations and an ensemble of five Q-functions.

Training uses batch size 256, replay capacity 10^6, 1,000 initial random steps, learning rate 3×10^{-4} for the world model and policy, and 9×10^{-5} for the PointNet++ encoder. The checkpoint used for evaluation corresponds to 118,440 accumulated environment steps. TD-MPC2 is evaluated on the same 1,041 arm configurations, with the same initial garment states and five repeated trials used for GwoP and OnePolicy.

C.5 ABLATION CONFIGURATIONS

w/o Physics Gradients disables backpropagation through the differentiable cloth simulator while retaining forward simulation, relation supervision, and geometric trajectory supervision, isolating the contribution of physical gradients.

w/o Body-Relative Guidance removes body-relative guidance and trains the trajectory predictor without relation pretraining. *Trajectory Supervision* uses the same configuration but adds 10 epochs

of direct pretraining on heuristic gripper trajectories, matching the 10-epoch relation-pretraining budget of the full model and comparing direct trajectory guidance with body-relative guidance.

Query-to-Trajectory retains body-relative relation modeling but decodes trajectory increments from the predicted queries, testing the difference between using relations as auxiliary guidance and using them as an intermediate representation for trajectory generation.

Figure 6 further illustrates the different behaviors induced by these training designs. Without physical gradients, the solver tends to favor nearly straight trajectories. This can be sufficient for poses where direct advancement is feasible, but becomes ineffective when the garment must bend around the body, leading to insufficient progress or excessive stretching. Without body-relative guidance, the solver can occasionally discover effective routes but exhibits less stable trajectory geometry across poses. Heuristic trajectory pretraining in *Trajectory Supervision* provides a more constrained initialization, but can still introduce unnecessarily twisted trajectory geometry. Query-to-Trajectory preserves body-relative information, but forcing the trajectory to pass through the relation representation can bias the predicted route toward the inner side of the body or cause regression near the target. In contrast, *Full* predicts relations and trajectories in parallel, allowing body-relative structure to guide the shared representation while physical gradients directly optimize the trajectory output, resulting in more consistent behavior across different arm configurations.

C.6 Compute and Timing Protocol

Training time is measured as wall-clock time under each method's corresponding training protocol. For GwoP inference, a network forward pass followed by single-trajectory decoding takes 2.60 ms on average. Generating and decoding 10 candidate trajectories takes 7.58 ms, while generating 10 candidates and applying capsule safety filtering takes 236.79 ms.

D Evaluation Details

D.1 Evaluation Metrics

Let $\mathbf{p}_w, \mathbf{p}_e, \mathbf{p}_s$ denote wrist, elbow, and shoulder positions. Dressing progress is measured along the wrist-elbow-shoulder path. For garment progress point $\mathbf{p}_t$,

$$P_t = \begin{cases} \dfrac{\|\mathbf{p}_t - \mathbf{p}_w\|}{\|\mathbf{p}_e - \mathbf{p}_w\| + \|\mathbf{p}_s - \mathbf{p}_e\|}, & \mathbf{p}_t \text{ on the forearm,} \\ \dfrac{\|\mathbf{p}_e - \mathbf{p}_w\| + \|\mathbf{p}_t - \mathbf{p}_e\|}{\|\mathbf{p}_e - \mathbf{p}_w\| + \|\mathbf{p}_s - \mathbf{p}_e\|}, & \mathbf{p}_t \text{ on the upper arm.} \end{cases} \tag{13}$$

We report $P_{\max} = \max_t P_t$. A trial satisfies **Elbow-Passage Success** when $P_{\max} > \|\mathbf{p}_e - \mathbf{p}_w\|/(\|\mathbf{p}_e - \mathbf{p}_w\| + \|\mathbf{p}_s - \mathbf{p}_e\|)$. The criterion indicates successful passage of the elbow bottleneck rather than complete dressing completion.

For edge $e = (i, j)$, define $L_e^t = \|\mathbf{x}_j^t - \mathbf{x}_i^t\|_2$ and $\ell_e^t = \max(L_e^t - (1+\epsilon)L_e^0, 0)/L_e^0$. For face f, let A_f^t denote its area and $a_f^t = \max(A_f^t - (1+\epsilon)A_f^0, 0)/A_f^0$, with $\epsilon = 0.05$. We report

$$V_{\text{edge}} = \max_t \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \ell_e^t, \qquad V_{\text{area}} = \max_t \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} a_f^t. \tag{14}$$

Training Time is the wall-clock training duration under each method's training protocol. **Target Input** indicates whether target garment information is required to generate the trajectory at inference.

D.2 Cross-Pose Gradient Consistency

We analyze whether different arm poses induce compatible local optimization directions. For pose c, let $\mathbf{g}^c = \nabla_\theta \mathcal{L}_{\text{phys}}^c$, and project it onto the same two-dimensional local parameter subspace used for the landscape analysis, yielding $\widetilde{\mathbf{g}}^c$. For K evaluated poses, we use the mean pairwise cosine

Table 3: Cross-pose gradient consistency across different model variants. Higher $A_{\cos}$ and A_+ indicate more consistent optimization directions across poses, while lower G_{proj} indicates lower local sensitivity.

Method	$\mathbf{A}_{\cos} \uparrow$	$\mathbf{A}_{+} \uparrow$	$\mathbf{G}_{\text{proj}} \downarrow$
w/o Body-Relative Guidance	-0.104	41.7%	16.43
Trajectory Supervision	$\mathbf{0.206}$	$\underline{61.1\%}$	$\underline{8.19}$
Query-to-Trajectory	-0.039	47.2%	8.28
Full	$\underline{0.188}$	$\mathbf{66.7\%}$	$\mathbf{2.86}$

similarity, the fraction of positively aligned pose pairs, and the mean projected-gradient magnitude:

$$
\begin{aligned}
A_{\cos} &= \frac{2}{K(K-1)} \sum_{i<j} \frac{(\widetilde{\mathbf{g}}^{c_i})^\top \widetilde{\mathbf{g}}^{c_j}}{\|\widetilde{\mathbf{g}}^{c_i}\|_2 \|\widetilde{\mathbf{g}}^{c_j}\|_2}, \\
A_{+} &= \frac{2}{K(K-1)} \sum_{i<j} \mathbb{I}\left[(\widetilde{\mathbf{g}}^{c_i})^\top \widetilde{\mathbf{g}}^{c_j} > 0\right], \\
G_{\text{proj}} &= \frac{1}{K} \sum_{i=1}^{K} \|\widetilde{\mathbf{g}}^{c_i}\|_2.
\end{aligned}
\tag{15}
$$

Higher $A_{\cos}$ and A_+ indicate more consistent optimization directions across poses, while a smaller G_{proj} indicates lower local sensitivity.

The results show that body-relative guidance improves local projected-gradient alignment across the evaluated poses. Without it, $A_{\cos}$ becomes negative and only 41.7% of pose pairs are positively aligned, accompanied by a markedly larger projected-gradient magnitude. Query-to-Trajectory exhibits a similar loss of directional consistency, suggesting that forcing the relation representation to directly determine the trajectory does not provide a stable shared optimization structure. Trajectory Supervision achieves the highest mean cosine similarity, but with a lower positive-alignment ratio and substantially larger gradient magnitude than Full. In contrast, Full attains the highest A_+ while reducing G_{proj} by a large margin, indicating that the auxiliary body-relative guidance promotes broadly compatible optimization directions across poses without making the local landscape excessively sensitive.

D.3 Cross-Simulator Protocol

For cross-simulator evaluation, trajectories predicted in DiffCloth are executed directly in CLO without further trajectory optimization. CLO uses the same initial garment-body configuration and the same two garment attachment locations as the DiffCloth rollout. The corresponding garment points are pinned and driven by the predicted trajectory.

The CLO garment uses a 20 mm particle distance and a total garment weight of 205 g. The avatar surface offset is 3 mm, with static and dynamic friction coefficients of 0.10 and 0.05, respectively. Simulation uses a time interval of 0.033333 s with five simulation iterations per frame. Gravity is set to -9800 in CLO coordinates, and garment-avatar collision, garment self-collision, and proximity-based collision handling are enabled. These settings introduce cloth and contact dynamics that differ substantially from the DiffCloth training environment while preserving the same initial configuration and commanded dressing trajectory.

E Real-Robot Details

E.1 Robot Setup and Calibration

Our real-robot platform consists of two AgileX PiPER manipulators, a Gemini 336L camera, and a mannequin for dual-arm dressing. Since human-pose estimation is not the focus of this work, we manually annotate the shoulder, elbow, and hand keypoints for each mannequin pose to obtain the corresponding arm configuration, following the real-world setup of OnePolicy (Wang et al., 2023).

Before each trial, the two end effectors grasp the garment regions corresponding to the two simulated attachment points, and the garment is initialized with the sleeve opening near the mannequin hand. We first record the visible garment point cloud from the dressed-side view, together with the current gripper positions and the pose of the robot-side calibration board. We then observe the calibration board mounted on the mannequin and use the relative transformation between the two boards to express the recorded garment point cloud and gripper positions in a mannequin-centered task frame, whose root is fixed at the mannequin base.

The resulting partial garment point cloud, annotated arm configuration, and transformed initial attachment positions are provided to GwoP for dual-gripper trajectory prediction.

E.2 Trajectory Execution

At the beginning of each trial, GwoP predicts the complete dual-gripper dressing trajectory from the initial observation. The predicted attachment positions, expressed in the mannequin-centered task frame, are transformed into the corresponding robot frames and converted to joint-space commands through inverse kinematics. The two PiPER manipulators then execute the resulting trajectories while maintaining their garment grasps, progressively advancing the garment along the mannequin arm.

Execution is open-loop after the initial trajectory prediction, with no additional garment observations or replanning during a trial. We evaluate 12 mannequin arm configurations with 5 repeated trials per configuration, resulting in 60 real-robot executions. Among them, 46 satisfy the elbow-passage success criterion. Of the remaining 14 trials, 5 fail because inverse-kinematics or execution constraints lead to collisions between the robot, particularly the relatively wide gripper geometry, and the surrounding environment, mannequin, or mannequin base. The other 9 failures occur when the sleeve becomes caught around the elbow before passing the elbow bottleneck. The supplementary video presents representative executions from eight of the evaluated arm configurations.

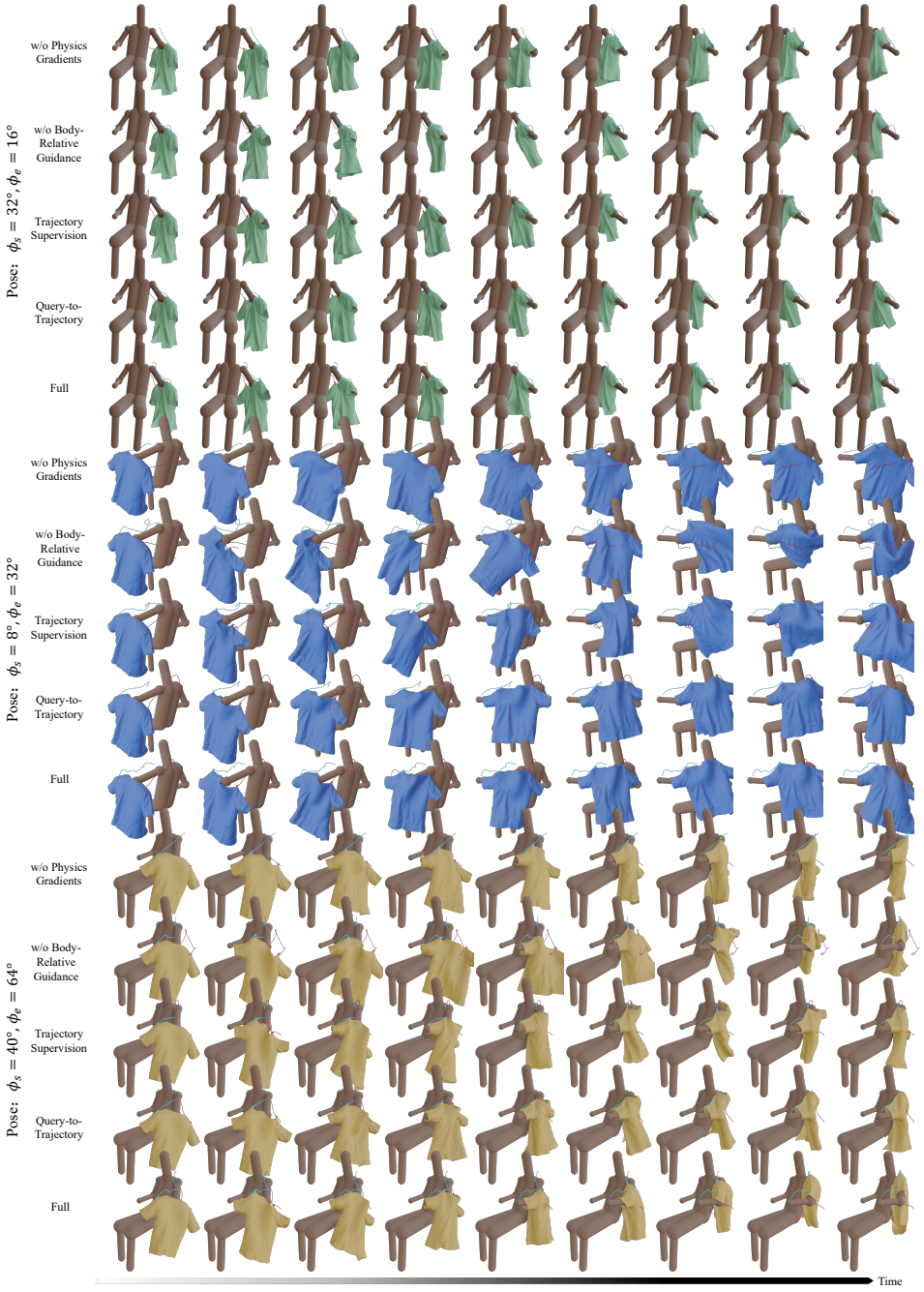

Figure 6: **Qualitative comparison of ablation variants across representative arm configurations.** From top to bottom, the 3 groups correspond to $(\phi_s, \phi_e) = (32°, 16°)$, $(8°, 32°)$, and $(40°, 64°)$, with time progressing from left to right. For configurations requiring the garment to move around the outer side of the arm, *w/o Physics Gradients* tends to produce overly straight trajectories, while *Query-to-Trajectory* may approach the torso from an unfavorable direction. *w/o Body-Relative Guidance* and *Trajectory Supervision* produce more irregular trajectory geometry. For the more direct $(8°, 32°)$ configuration, the nearly straight trajectory remains feasible, whereas *Query-to-Trajectory* loses forward progress near the end of execution. *Full* produces more consistent dressing trajectories across the three configurations.