FABRICBENCH: A STANDARDIZED POLICY BENCHMARK FOR BIMANUAL FABRIC MANIPULATION WITH A CONTACT-DRIVEN DEFORMABLE SIMULATOR

Anonymous authors
Paper under double-blind review

ABSTRACT

Fabric manipulation challenges robot policies with changing geometry, self-occlusion, and contact, yet standardized evaluation remains focused on rigid objects. We introduce FabricBench, a benchmark for bimanual fabric manipulation with six tasks spanning garments, silk, and bags, 10,000 simulation-ready assets, and controlled tests of configuration and appearance generalization. Built on an open-source solver in Newton, our simulator combines redesigned cloth–rigid contact with bidirectional force coupling, enabling grasping, release, and load transfer without artificial attachment. It succeeds in all 45 grasp–hold–release validation trials, compared with none for the tested baseline configurations. Evaluating six representative policies, including vision-language-action models and world action models, reveals substantial remaining challenges: the best policy achieves 77% average success, while no policy exceeds 36% on long-horizon T-shirt folding. Under the most deformed initial T-shirt configurations, success falls to 6–14% across policies, with larger drops than under the evaluated appearance shifts. Zero-shot deployment of $\pi_{0.5}$ on a real robot largely preserves task difficulty ordering across six tasks (Spearman $\rho = 0.94$). These results establish FabricBench as a reproducible testbed for measuring fabric-manipulation capabilities and exposing limitations in configuration generalization.

1 INTRODUCTION

Fabric manipulation is central to everyday tasks such as folding laundry and carrying bags, yet folds, self-occlusion, and changing contacts make it difficult to evaluate reliably (Lin et al., 2021; Lu et al., 2024). Vision-language-action (VLA) models and world action models (WAMs) show progress on fabric tasks, including laundry folding with π_0 (Black et al., 2025b) and towel folding with Fast-WAM (Yuan et al., 2026), while standardized manipulation benchmarks remain focused on rigid objects (Mu et al., 2025; Liu et al., 2023). We introduce FabricBench, a standardized benchmark for evaluating learned policies on bimanual fabric manipulation.

Simulation can make fabric evaluation scalable and reproducible, but reliable cloth contact remains challenging. Existing environments report penetration, jitter, and instability (Wang et al., 2025b; Zhu et al., 2026); others use attachment rules to bind cloth to the gripper (Lu et al., 2024; Luo et al., 2026). Under learned policies, small fluctuations in continuous gripper commands can repeatedly trigger attachment and release, making outcomes depend on simulator rules rather than policy capability.

To address this limitation, we develop an improved open-source cloth solver based on the Style3D cloth solver in the Newton physics engine. Redesigned cloth–rigid contact and bidirectional force coupling among the arm, fabric, and free rigid objects allow grasping, release, and load transfer to arise from contact and remain stable under learned policies. Six tasks cover T-shirt folding and flattening, silk flattening and lifting, bag carrying, and packing, spanning multilayer self-contact, dynamic manipulation, self-occlusion, and load-bearing fabric–object interactions.

Beyond task success, FabricBench separates generalization across initial fabric configurations from generalization across appearance. Folding or crumpling the same garment changes visibility, contacts, and feasible actions. For the two flattening tasks, seeded configuration levels (C0–C2) vary initial deformation while sharing assets and placement distributions. Held-out appearance variants

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

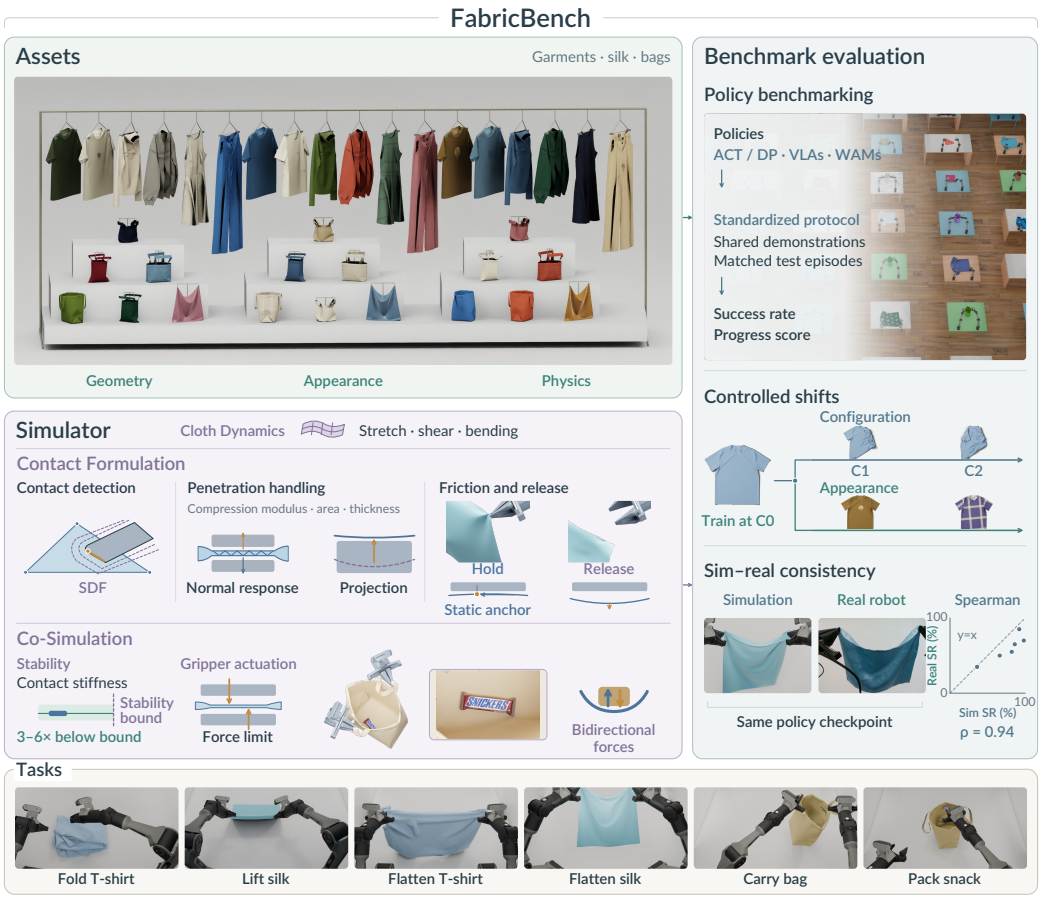

Figure 1: **Overview of FabricBench. Assets (top left):** garments, silk, and bags with geometry, appearance, and physical properties. **Simulator (bottom left):** contact detection, penetration handling, frictional grasp and release, force-limited grippers, and bidirectional coupling make grasping, release, and load transfer arise from contact forces without artificial attachment. **Benchmark evaluation (right):** shared demonstrations and matched test episodes; separate configuration and appearance tests, with geometry and physics fixed for appearance shifts; sim–real task difficulty comparison using the same checkpoint per task. **Tasks (bottom):** six tasks spanning long-horizon folding, configuration recovery, friction-limited grasping, and load-bearing fabric–object interactions.

instead change color and texture while keeping geometry and physics fixed. These controlled axes test whether policies cope with changes in fabric state as well as visual appearance.

This paper makes the following contributions:

- **A fabric manipulation benchmark.** FabricBench: six tasks over garments, silk cloth, and fabric bags; held-out variations in initial deformation and appearance on the flattening tasks; a unified training and evaluation protocol with demonstration generation.
- **An open-source simulator.** An improved cloth solver based on the Style3D solver in the Newton physics engine, with redesigned cloth–rigid contact and bidirectional force coupling among the arm, the fabric, and free rigid objects, so that grasping, release, and load transfer arise from contact forces and remain stable under learned policies.
- **A systematic assessment of policy limitations.** Our evaluation shows that current policies remain unreliable on fabric manipulation, especially long-horizon folding. Appearance shifts reduce performance, although some policies show partial robustness to these changes. On the flattening tasks, performance degrades systematically with increasing initial deformation, with drops far exceeding those under appearance shift.

2 Related Work

2.1 Manipulation Policies and Benchmarks

Manipulation policies have progressed rapidly from classical imitation learning (Zhao et al., 2023; Chi et al., 2025) to large-scale generalist models. Vision-language-action models such as OpenVLA (Kim et al., 2025), π_0, $\pi_{0.5}$ (Black et al., 2025b;a), and GR00T (NVIDIA et al., 2025) are pretrained on large multi-robot datasets and improve instruction following and cross-embodiment transfer, and world action models such as LingBot-VA (Li et al., 2026a) and Fast-WAM (Yuan et al., 2026) further predict future observations jointly with actions.

Standardized simulation benchmarks such as RoboTwin (Mu et al., 2025; Chen et al., 2026b), LIBERO (Liu et al., 2023), RoboDojo (Chen et al., 2026a), and RoboLab (Yang et al., 2026) have driven this progress on rigid bodies through fixed task suites, public demonstrations, and shared protocols. Studies of policy degradation under object-layout, camera-viewpoint, and initial-pose perturbations (Zhou et al., 2025; Fei et al., 2026) have further exposed where these policies break and guided their improvement. Deformable objects are absent from these suites or appear as a single task (Chen et al., 2026a), and the deformation state of the object lies outside their perturbation axes.

Early benchmarks for deformable objects, SoftGym (Lin et al., 2021), DaXBench (Chen et al., 2023), and Deformable Ravens (Seita et al., 2021), introduced simulated task suites for cloth, rope, and bags, but observe through low-resolution renders, heightmaps, or particle states, whereas simulated scores track real-robot performance once the visual gap is controlled (Li et al., 2025). Recent Isaac Sim suites such as LeHome (Li et al., 2026b) and FlatLab (Zhu et al., 2026) narrow this visual gap for specific categories of deformable objects. FabricBench further narrows the physics gap, bringing simulated evaluation of deformable manipulation closer to the real world.

2.2 Cloth Simulation for Robotic Manipulation

Position-based dynamics (PBD/XPBD) (Müller et al., 2007; Macklin et al., 2016) underlies many cloth-manipulation environments: SoftGym builds on the PBD-based FleX engine (Macklin et al., 2014), and GarmentLab (Lu et al., 2024), DexGarmentLab (Wang et al., 2025b), LeHome, and FlatLab use the PBD cloth of Isaac Sim (NVIDIA, n.d.). PBD is stable and easy to implement, but it approximates continuum mechanics with geometric constraint projections, producing overstretching and self-penetration, and its gap to real garment dynamics has been measured (Blanco-Mulero et al., 2024; Hu et al., 2026a). Finite-element and intersection-free solvers (Faure et al., 2012; Li et al., 2020; 2021), including the cloth solver of Genesis (Genesis Authors, 2024), are more accurate but about an order of magnitude slower, too slow for policy-scale rollouts (Appendix B.7), whereas recent fast FEM cloth solvers reach continuum-level accuracy at practical rates (Hu et al., 2026a; Luo et al., 2026). A simplified open-source version of the Style3D solver following this principle is available in the Newton physics engine (The Newton Contributors, 2025; Mohajerani et al., 2025), and we build on it.

Contact between cloth and rigid bodies, such as a gripper or objects in a bag, is typically handled by rules or artificial forces rather than simulated physically: SoftGym grasps with pickers that attach nearby particles, DaXBench with pick-and-place macro actions, and Deformable Ravens with fixed constraints; GarmentLab (Lu et al., 2024) and FLASH (Luo et al., 2026) bind cloth vertices within a distance threshold when the gripper closes, and DexGarmentLab (Wang et al., 2025b) adds artificial adhesion between the fingers and the fabric; the released Style3D solver (The Newton Contributors, 2025) couples cloth and rigid bodies only one way, so objects in a bag follow it only if a rule binds them to it. In all these cases, whether the cloth or the bag's contents follow the gripper is decided by these mechanisms rather than by contact forces, and, as discussed in Section 1, end-to-end policies have no access to them. In FabricBench, grasp, release, and load transfer arise from bidirectional contact forces alone, so task outcomes reflect the policy's actions under real physical contact.

3 Deformable Simulation and Contact Formulation

We state the cloth dynamics, the contact energy between cloth and rigid bodies, and the force exchange with the rigid-body solver; implementation details are in Appendix B.

3.1 CLOTH DYNAMICS

Material model. A garment is a triangle mesh. Its in-plane energy is the anisotropic membrane model of Baraff & Witkin (1998), with warp, weft, and shear stiffness specified per triangle; its out-of-plane energy is the discrete bending model of Bergou et al. (2006) on mesh edges. These stiffnesses and the areal density correspond directly to standard fabric tests, so they are loaded from measured values and carry no non-physical tuning parameters.

Time integration. Positions advance by implicit Euler with a fixed substep Δt. The positions at the end of a substep minimise an incremental potential,

$$\mathbf{x}_{n+1} = \arg\min_{\mathbf{x}} \; \frac{1}{2\Delta t^2} \|\mathbf{x} - \tilde{\mathbf{x}}\|^2_{\mathbf{M}} + E_{\mathrm{el}}(\mathbf{x}) + E_{\mathrm{c}}(\mathbf{x}), \tag{1}$$

where $\tilde{\mathbf{x}}$ is the inertial prediction, E_{el} the membrane and bending energy above, and E_{c} the contact energy of Sec. 3.2. The minimisation uses projective-dynamics iterations (Bouaziz et al., 2014): the Hessian of E_{el} is replaced by a constant approximation assembled once, the gradient and diagonal Hessian of E_{c} are re-evaluated at every iteration, and each linear system is solved by Jacobi-preconditioned conjugate gradients; self-contact repulsion and untangling terms (Volino & Magnenat-Thalmann, 2006) are part of E_{c}. The whole substep runs as one replayed CUDA graph with fused conjugate-gradient kernels and table-based distance queries (Appendix B.5).

3.2 CONTACT FORMULATION

The contact energy between cloth and a rigid body (gripper finger, table, or free object) is

$$E_{\mathrm{c}}(\mathbf{x}) = \sum_c \frac{k_c}{2} \big(h - d_c(\mathbf{x})\big)^2_+ + \sum_c \frac{k_t}{2} \big\|\mathbf{P}_c\big(\mathbf{x}_c - \mathbf{p}_c\big)\big\|^2, \qquad d_c(\mathbf{x}) = \min_{\mathbf{y} \in \triangle_c} \phi(\mathbf{y}), \tag{2}$$

a normal penalty and a tangential anchor potential over the active contacts c: a penalty formulation over a compliant shell, with no barrier term and no continuous collision detection. Relative to released cloth stacks it changes three things.

Contact detection. The cloth is a shell of half-thickness h (Appendix B.3). In Eq. (2), d_c is the minimum over the *whole triangle* of the rigid body's precomputed signed distance field ϕ, not a vertex distance; a contact is active when $d_c < h$. The minimisation is seeded at the triangle's vertices and centroid and at the intersections of the rigid mesh's edges with the triangle, so that a sharp finger edge passing through the interior of a triangle is detected. With the triangle as the contact primitive, h can be chosen below the mesh edge length, whereas vertex-based contact must set its collision radius above it to stop a finger from passing between vertices, tying the effective thickness to mesh resolution. The forces of Eq. (2) also act on the rigid body (Sec. 3.3).

Penetration handling. The normal stiffness is derived from material and geometry, $k_c = E_t A_c / t_0$, with E_t the fabric's transverse compression modulus, A_c the triangle area, and t_0 the fabric thickness. The normal load carried by a patch of fabric, $\int (E_t/t_0)\,(h - d)\,\mathrm{d}A$, is then independent of the triangulation, and mesh refinement does not change the grasp load. Triangles that transiently cross the shell boundary by more than a threshold are projected back to the shell after the position update, with forces evaluated at the pre-projection positions, so the projection resolves only deep transients that the force law did not stop and does not alter the grasp force.

Friction and release. The tangential term is anchored Coulomb friction with a true static regime, following the classical elastoplastic formulation (Laursen & Simo, 1993; Wriggers, 2006). When a contact forms, its closest point is stored as an anchor $\mathbf{p}_c$ in the rigid body's local frame and $\mathbf{P}_c$ projects onto the tangent plane; the tangential force $\mathbf{f}_t = -k_t \mathbf{P}_c(\mathbf{x}_c - \mathbf{p}_c)$ is limited to the Coulomb cone $\|\mathbf{f}_t\| \le \mu f_n$, and when the limit is exceeded the force is clamped to the cone and $\mathbf{p}_c$ is projected back onto its boundary, which is sliding; $k_t = \kappa k_c$ places the pre-sliding displacement at the scale of fabric. What the static regime buys is visible by contrast with released stacks, which replace the stick regime by a velocity-regularised ramp (Li et al., 2020): a contact loaded to a fraction $\rho = \|\mathbf{f}_t\|/(\mu f_n)$ of the cone then creeps at the steady velocity

$$v = \varepsilon \left(1 - \sqrt{1 - \rho}\right), \tag{3}$$

with ε the regularisation velocity, non-zero for any $\rho > 0$; holding fabric against gravity then requires an inflated μ, and an inflated μ locks fabric wrapped around a finger tip on release through the capstan factor $e^{\mu\theta}$. With the anchor, $v = 0$ for $\rho < 1$, holding depends on μf_n alone without inflating μ (one cloth-side coefficient serves all rigid surfaces, Appendix B.3), and when the gripper opens f_n and hence $\mathbf{f}_t$ vanish and the fabric leaves the finger under its own weight.

Grasp, hold, and release thus arise from the forces of Eq. (2) alone: no attachment constraints, no collision disabling on release, no adhesion term, and no filtering of failed trajectories.

3.3 Co-Simulation with the Rigid-Body Solver

Bidirectional force coupling. Rigid motion drives the cloth through Eq. (2); the same forces act on the rigid bodies with equal magnitude and opposite direction. After the cloth solve of a substep converges, the normal and tangential force of every contact is evaluated at the converged positions and accumulated per rigid body b into a force $\mathbf{F}_b = -\sum_{c\in b} \mathbf{f}_c$ and a torque about its centre of mass $\boldsymbol{\tau}_b = -\sum_{c\in b}(\mathbf{x}_c - \mathbf{c}_b) \times \mathbf{f}_c$, applied by the rigid-body solver (Todorov et al., 2012) as an external load in the next substep. Force transmission is therefore bidirectional: a finger feels the load of the fabric it holds, a packed item is supported by the bag bottom, and a lifted bag carries the weight of its contents to the gripper. The coupling is a staggered scheme at substep granularity.

Gripper actuation. The gripper is a PD position actuator with a force limit. Closing on fabric saturates the actuator, so the grip force equals the actuator's stall force and is independent of the residual jaw opening: a settable physical quantity rather than a stiffness times a leftover gap.

Stability of the staggered scheme. The contact force computed at the start of a substep is held over that substep, so the rigid-body integrator sees it as an explicit spring. For one finger degree of freedom the total contact stiffness it can be exposed to is bounded by

$$k_{\text{crit}} = \frac{4M}{\Delta t^2} + \frac{2c}{\Delta t} - k_p, \tag{4}$$

with M the finger mass including armature, c its damping including the velocity gain, and k_p the position gain, the same form as the passivity bound of haptic virtual walls (Colgate & Schenkel, 1997). For our gripper the summed stiffness of Eq. (2) over a grasp lies a factor of 3–6 below this bound (Appendix B.4), whereas the much stiffer response of fully compacted fabric would exceed it; the normal law is therefore linear, and a nonlinear compression law is left to fully implicit coupling (Sec. 6).

4 Benchmark Design

Platform. We use a dual-arm Piper platform with a 14-dimensional state and action interface: six joints and one gripper value per arm. Policies receive RGB observations from one overhead and two wrist cameras at 25 Hz. Simulation couples our cloth solver with MuJoCo rigid-body dynamics and uses Omniverse Kit for rendering. Simulated camera poses and intrinsics follow the real setup; detailed camera and data specifications are provided in Appendix E.3.

4.1 Task Design

We first summarize common deformable-manipulation situations with five physical regimes: (i) quasi-static, long-horizon folding with self-contact; (ii) configuration recovery under severe self-occlusion; (iii) dynamic momentum transfer during flinging; (iv) friction-limited grasping and hold maintenance without attachment; and (v) manipulation of a volumetric bag whose handles must carry its load, including rigid contents when packed. These regimes are not mutually exclusive task categories: a single task can exercise several of them. We then instantiate the coverage with six representative tasks, shown in Figure 2. Fold T-shirt primarily tests (i); Flatten T-shirt and Flatten silk combine (ii) and (iii) on garments with different material response; Lift silk isolates (iv); Carry bag tests (v) through bimanual handle lifting; and Pack snack combines (v) with rigid pick-and-place, containment, and loaded-bag transport. Success criteria are outcome-based rather than tied to a prescribed grasp point or trajectory, so a policy receives credit for any behavior that reaches the target state.

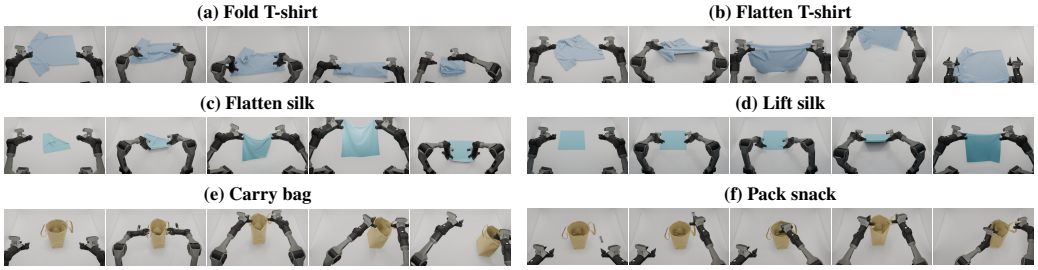

Figure 2: **The FabricBench task suite.** Each panel shows five stages of one successful episode, progressing from left to right. The suite covers (a) quasi-static, long-horizon folding with self-contact; (b–c) flattening under occlusion for garments with distinct material properties; (d) slip-sensitive, attachment-free bimanual grasping; (e) lifting a volumetric bag by its handles; and (f) rigid-object placement into a deformable container followed by loaded-bag transport. Exact success criteria are given in Appendix E.

4.2 Assets

Asset diversity is essential for a deformable benchmark, but meshes produced by generative 3D models (Wang et al., 2025a; Tencent Hunyuan3D Team, 2025) cannot be used directly as simulated cloth. They are typically double-layered closed surfaces rather than single-layer fabric, are split into hundreds of disconnected pieces, and carry UV layouts packed as texture atlases whose per-chart scale does not match the material. A cloth solver that reads the UV layout as the rest pattern then starts from a distorted rest shape, and the garment becomes unstable or diverges within the first frames (Appendix A). We therefore start from high-quality template meshes and expand them through a framework that separates geometry, appearance, and physics: morphology variants edit semantic regions and reconstruct a consistent rest pattern, appearance variants change only render materials and printed patterns, and physical variants assign explicit material-parameter profiles, each leaving the other factors unchanged. Every variant records its seed and recipe and can be rebuilt from its source. We will release 10,000 expanded assets, each passing the same simulation-stability check; the held-out appearance pools of Section 5.3 are drawn from them.

4.3 Training Data

We generate 200 demonstration trajectories per task, recorded from three cameras (two wrist cameras and one top camera) at 25 Hz (Appendix E.3). Each task is trained and evaluated on its own standard task distribution (Appendix E): all tasks randomize object placement, and Carry bag additionally randomizes the handle stiffness and the bag appearance. We train ACT, Diffusion Policy, the VLAs $\pi_{0.5}$ and GR00T, and the WAMs LingBot-VA and Fast-WAM, each on the same data with its official configuration (Appendix F).

4.4 Evaluation Protocol

Evaluation axes. Replacing a whole object changes its appearance, physical properties, and initial state at once. For the two flattening tasks, we instead vary one factor at a time against the same training condition, the C0 configuration with the canonical appearance. The configuration ladder (C0–C2) changes only how the cloth is initially folded or dropped, through seeded programs with matched asset, camera, and placement distribution (Figure 3; generation details in Appendix C). The appearance axis assigns one of 50 held-out appearance variants (Section 4.2) to each episode under C0, leaving geometry and physics unchanged.

Metrics. Each task reports an outcome-based binary success rate (SR), credited for any trajectory that reaches the goal, and a continuous 0–100 score of partial progress (Appendix E). For a shifted condition x, we report the absolute change $\Delta_x = S_x - S_{\mathrm{C0}}$ and, when $S_{\mathrm{C0}} > 0$, the normalized degradation $D_x = (S_{\mathrm{C0}} - S_x)/S_{\mathrm{C0}}$.

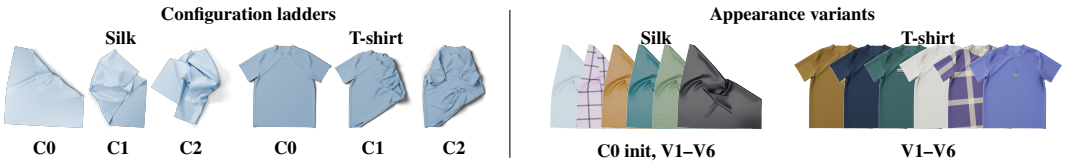

Figure 3: **The two evaluation axes.** Left: configuration ladders. Silk: C0 folds one far corner, C1 randomly folds one or two corners inward, C2 drops the cloth from a two-pin hold. T-shirt: C0 folds the right sleeve corner, C1 folds the right half, C2 applies random deep folds at the four corners. Right: appearance variants of the same assets (silk shown after C0 initialization).

Protocol. Each cell uses 50 episodes from a fixed evaluation seed schedule, disjoint from collection and shared by all policies together with the same episode-to-condition manifest. Evaluation uses no augmentation, and no held-out episode is used for training, checkpoint selection, or tuning.

5 Experiments

5.1 Simulator Validation

We examine whether grasp, hold, and release arise from contact forces alone (Table 1) and whether each contact component is necessary (Table 2).

Cross-simulator comparison. A recorded dual-arm grasp on a flat T-shirt is replayed in each simulator's released configuration at 15 placements, three times each (Appendix B.7). Isaac Sim's PBD cloth forms no bimanual grasp. Released Style3D and Newton VBD hold but never release, as the fingers stay in the fabric; VBD also diverges in 12 of 45 trials. Only our contact grasps, holds, and releases in every trial, still at a lower step cost than Isaac Sim and VBD.

Table 1: **Cross-simulator grasp validation.** The same recorded dual-arm grasp, replayed at 15 placements three times each (45 trials per simulator); success requires grasp, hold, and release. Step time: wall-clock time per physical step on one RTX 4090 with a 9k-vertex T-shirt (Appendix B.7).

Simulator	Success ↑	Grasp ↑	Hold ↑	Release ↑	Diverged ↓	ms / step ↓
Isaac Sim, PBD cloth	0/45	3/45	0/45	–	0/45	5.3
Newton VBD	0/45	39/45	39/45	0/45	12/45	5.6
Newton Style3D	0/45	45/45	45/45	0/45	0/45	3.7
Newton Style3D, our contact	45/45	45/45	45/45	45/45	0/45	4.8

Attachment under closed-loop control. Under the threshold-triggered attach–detach of existing cloth benchmarks, the same checkpoints succeed less often: Diffusion Policy on Lift silk falls from 19/50 to 9/50 (Table 2b), as spurious detachments drop the fabric mid-grasp. These failures stem from the grasp model rather than the policy, so attach–detach underestimates policy performance.

Component ablation. Each component is removed in turn (Table 2a; 20 seeds per cell; edge seeding in Appendix B.6). With vertex spheres of the same material stiffness in place of triangle-level SDF detection, the T-shirt stays on the fingers after the jaws open, the silk intersects the opened fingers, and the bag task diverges in 9 of 20 runs. Without bidirectional coupling, the bag is lifted

Table 2: **Contact ablations.** (a) One component removed at a time (20 seeds). (b) Diffusion Policy on Lift silk with attach–detach grasping vs. our contact (50 episodes).

(a) Variant	T-shirt release ↑	Silk penetration ↓	Pack lift ↑
Our contact	20/20	0/20	20/20
w/o triangle-level detection	7/20	20/20	6/20
w/o bidirectional coupling	20/20	1/20	0/20

(b) Grasp mode	Silk SR ↑
Our contact	19/50
Attach–detach	9/50

but the packed item stays on the table in every run; a check without the load-following condition of Pack snack (Appendix E) would still pass 18 of these runs. Triangle-level detection thus governs release and non-penetration; bidirectional coupling governs load transfer.

5.2 Benchmark Results

Table 3 reports the success rate and mean score of the six policies on the six tasks, each evaluated on its standard distribution. The mean score awards stage-wise partial credit and therefore distinguishes policies with the same success rate.

Table 3: **Benchmark results in simulation.** Success rate (%) and mean score (in parentheses) on each task's standard distribution (C0 for the flattening tasks); $N{=}50$ per cell.

Policy	Lift silk	Flatten silk	Flatten T-shirt	Carry bag	Pack snack	Fold T-shirt	Avg
$\pi_{0.5}$	**98** (98.0)	**86** (94.5)	82 (82.4)	66 (82.5)	92 (97.0)	**36** (63.6)	**77**
LingBot-VA	66 (78.0)	**86** (97.0)	86 (83.0)	50 (70.0)	**98** (98.8)	**36** (57.6)	70
Fast-WAM	58 (78.0)	64 (87.2)	**92** (81.6)	**80** (85.5)	8 (12.4)	0 (22.8)	50
GR00T	70 (79.0)	46 (80.4)	72 (76.9)	40 (58.5)	54 (81.4)	2 (24.2)	47
ACT	28 (62.0)	10 (70.3)	70 (79.6)	38 (62.5)	52 (80.0)	30 (54.4)	38
DP	38 (44.0)	0 (57.3)	78 (78.8)	8 (27.0)	18 (53.2)	0 (16.2)	24

5.3 Generalization Beyond the Training Condition

Table 4 evaluates the two flattening tasks along the configuration ladder and on the 50 held-out appearance variants, with training fixed to C0 and the canonical appearance. Performance generally degrades under both the configuration and the appearance shifts; Section 5.5 analyzes the two axes.

Table 4: **Generalization under configuration and appearance shift.** Success rate (%) and mean score (in parentheses); training on C0 with canonical appearance; $N{=}50$ per cell. C1/C2 follow Figure 3; C0 + app. keeps C0 and applies 50 held-out appearance variants.

	Flatten T-shirt				Flatten silk			
Policy	C0	C1	C2	C0 + app.	C0	C1	C2	C0 + app.
$\pi_{0.5}$	82 (82.4)	44 (73.4)	12 (35.7)	76 (80.3)	**86** (94.5)	20 (60.4)	**2** (42.2)	64 (85.3)
LingBot-VA	86 (83.0)	**62** (73.5)	10 (28.2)	**84** (77.1)	**86** (97.0)	**22** (61.6)	0 (40.4)	**86** (97.0)
Fast-WAM	**92** (81.6)	34 (71.8)	**14** (27.9)	42 (64.1)	64 (87.2)	16 (70.7)	0 (43.1)	0 (74.8)
GR00T	72 (76.9)	34 (62.9)	**14** (31.1)	58 (69.2)	46 (80.4)	18 (71.0)	0 (44.6)	38 (77.8)
ACT	70 (79.6)	16 (36.1)	6 (25.8)	28 (59.1)	10 (70.3)	6 (60.5)	0 (43.4)	4 (65.5)
DP	78 (78.8)	32 (62.9)	10 (26.5)	50 (70.4)	0 (57.3)	0 (55.8)	0 (41.7)	0 (53.1)

5.4 Sim-to-Real Consistency

Zero-shot sim-to-real transfer has been demonstrated for deformable manipulation (Hu et al., 2026b); here we ask a different question, namely whether the difficulty ordering of tasks in simulation is preserved on the real robot. We deploy the $\pi_{0.5}$ checkpoints of Table 3, trained only on simulated demonstrations, zero-shot on a real dual-arm robot and evaluate each task with 20 trials under the same success criteria (Figure 4). Across the six tasks, simulated and real-robot success rates yield Spearman $\rho = 0.94$ and Kendall $\tau_b = 0.87$. These results show that task performance in simulation is strongly correlated with that on the real robot.

5.5 Findings

Finding 1: Current policies remain short of reliable deformable manipulation, especially on long-horizon tasks. With the same 200 demonstrations per task, $\pi_{0.5}$ achieves the highest average success rate (77%), followed by LingBot-VA (70%), while the remaining policies average 24–50% (Table 3). The gap is largest on Fold T-shirt, where no policy exceeds 36% success; in rollouts, once the folded state deviates or a fold fails, policies rarely recover.



Figure 4: **Sim-to-real evaluation of $\pi_{0.5}$.** (a) Simulation (N=50) vs. real-robot (N=20) success rate per task; whiskers are 95% Wilson intervals, dashed line is y=x. ρ and τ_b measure agreement in task ordering, not in absolute rate (bootstrap CIs in Appendix D). (b) Real-robot frames; colors and markers match (a).

Finding 2: Appearance shift degrades performance, but some policies already show a degree of robustness to it. Changing only the garment's color and texture lowers Flatten T-shirt success by only 2–14 percentage points for $\pi_{0.5}$, LingBot-VA, and GR00T, but by 28–50 points for ACT, DP, and Fast-WAM (Table 4). Fast-WAM likely overfits to the training appearance: it has the highest canonical T-shirt success (92%) but the largest drop, and on Flatten silk it falls from 64% to 0%.

Finding 3: Policies degrade systematically as the initial deformation increases, with drops far exceeding those under appearance shift. The configuration ladder holds the asset, its appearance, and the placement distribution fixed and varies only the initial deformation of the cloth. On Flatten T-shirt, success decreases monotonically from C0 to C2 for every policy and reaches 6–14% at C2, a drop of 58–78 percentage points, compared with 2–50 points under appearance shift; on Flatten silk, no policy exceeds 2% at C2 (Table 4). Among policies with nonzero C0 success, the only exception to the larger drop is Fast-WAM on Flatten silk, which already falls to 0% under appearance shift. The deformation state of the object therefore remains a principal challenge for current policies.

6 Limitations and Future Work

FabricBench covers fabric manipulation, namely garments, silk cloth, and fabric bags; ropes, elastoplastic materials, and volumetric soft bodies are left to future versions. The configuration ladder consists of three discrete, script-generated, task-specific levels (C0–C2) rather than a dense sampling of the configuration space; it therefore characterizes how performance degrades as the initial deformation increases, not performance over the full space of deformable configurations. We will also continue to scale up the task suite to test generalist policies: training a single policy on the data of all tasks and evaluating it on each task will further examine whether policies forget individual skills under multi-task training.

7 Conclusion

We presented FabricBench, a standardized benchmark for bimanual fabric manipulation with contact-driven grasping, release, and bidirectional load transfer. Six tasks and controlled configuration and appearance shifts expose limitations in current policies, particularly on long-horizon folding and increasingly deformed initial states. Real-robot evaluation of $\pi_{0.5}$ supports consistency in task difficulty ordering across simulation and reality. We will release the simulator, assets, demonstrations, and evaluation code to support reproducible evaluation.

AI USE STATEMENT

In this work, we used generative AI tools for two tasks with required disclosure: to assist in writing parts of the code, while the methods were designed by the authors; and to assist with translating parts of the paper from drafts written by the authors. We have not used generative AI tools to generate synthetic data sets, develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, design research methodology or experiments, clean or reformat datasets, or analyze and interpret results; proofs and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to edit the paper to improve readability. We have reviewed all AI-assisted work: AI-assisted code was read, tested, and validated by the authors against the simulation checks and experiments reported in this paper, and all AI-edited text was checked by the authors for accuracy against our results and the cited sources. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects or personal data: all demonstrations are generated in simulation by scripted experts (Appendix E.3), and the real-robot experiments only deploy trained policies on fabric objects in a controlled laboratory setting. The assets to be released are expanded from template meshes created by the authors, and all appearance variants are procedurally generated, so they raise no licensing concerns and contain no identifiable people or private content. We use existing policies and models under their official licenses and configurations. We are not aware of direct harmful applications beyond the general dual-use considerations of robot manipulation research.

REPRODUCIBILITY STATEMENT

The cloth model, contact formulation, and co-simulation scheme are described in Section 3, with parameters, solver settings, contact implementation, and GPU execution details in Appendix B. Task definitions, success criteria, and seeded configuration levels are given in Section 4 and Appendices E and C; the data format and the training and evaluation protocol for each policy are given in Appendices E.3 and F, and the statistical procedures in Appendix D. Evaluation seeds are fixed and disjoint from collection seeds. Anonymized source code, including the modified cloth solver, task and contact configurations, data collection, and policy evaluation pipelines, is available at `https://anonymous.4open.science/r/Fabric-Anonymous-23A0/`. Assets and demonstration datasets will be released upon acceptance.

REFERENCES

David Baraff and Andrew Witkin. Large steps in cloth simulation. In *Proceedings of SIGGRAPH*, 1998.

Miklós Bergou, Max Wardetzky, David Harmon, Denis Zorin, and Eitan Grinspun. A quadratic bending model for inextensible surfaces. In *Eurographics Symposium on Geometry Processing (SGP)*, 2006.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 2025a. URL `https://proceedings.mlr.press/v305/black25a.html`.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming

Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems (RSS)*, 2025b. doi: 10.15607/RSS.2025.XXI.010.

David Blanco-Mulero, Oriol Barbany, Gokhan Alcan, Adrià Colomé, Carme Torras, and Ville Kyrki. Benchmarking the sim-to-real gap in cloth manipulation. *IEEE Robotics and Automation Letters*, 9(3):2981–2988, 2024. doi: 10.1109/LRA.2024.3360814.

Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. Projective dynamics: Fusing constraint projections for fast simulation. *ACM Transactions on Graphics*, 33(4), 2014.

Siwei Chen, Yiqing Xu, Cunjun Yu, Linfeng Li, Xiao Ma, Zhongwen Xu, and David Hsu. DaXBench: Benchmarking deformable object manipulation with differentiable physics. In *International Conference on Learning Representations (ICLR)*, 2023.

Tianxing Chen, Yue Chen, Zixuan Li, Junyuan Tang, Kailun Su, Haoran Lu, Weijie Wan, Baijun Chen, Songling Liu, Haowen Yan, Honghao Su, Zhiyang Dou, Kaixuan Wang, Dandan Zhang, Yunze Liu, Yan Qin, Qiwei Liang, Qiwei Wu, Zijian Lin, Wenwei Lin, Yuran Wang, Minghua He, Tianshu Wu, Ruihai Wu, Jingquan Zhou, Kai-Chong Lei, Haibao Yu, Yuanfeng Ji, Weiyang Jin, Guanyu Lin, Xiaofan Li, Qi Xiong, Renjing Xu, Zhongyu Li, Wenhao Chai, Enze Xie, Ziwei Wang, Yao Mu, Hao Dong, Wojciech Matusik, Mingyu Ding, Wenbo Ding, Ping Luo, and Masayoshi Tomizuka. RoboDojo: A unified sim-and-real benchmark for comprehensive evaluation of generalist robot manipulation policies. *arXiv preprint arXiv:2607.04434*, 2026a.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan-ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. RoboTwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026b.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

J. Edward Colgate and Gerd G. Schenkel. Passivity of a class of sampled-data systems: Application to haptic interfaces. *Journal of Robotic Systems*, 14(1):37–47, 1997.

François Faure, Christian Duriez, Hervé Delingette, Jérémie Allard, Benjamin Gilles, Stéphanie Marchesseau, Hugo Talbot, Hadrien Courtecuisse, Guillaume Bousquet, Igor Peterlik, and Stéphane Cotin. SOFA: A multi-model framework for interactive physical simulation. In Yohan Payan (ed.), *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, pp. 283–321. Springer, 2012. doi: 10.1007/8415_2012_125.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: A progressive robustness benchmark for visual-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 38574–38583, 2026.

Genesis Authors. Genesis: A generative and universal physics engine for robotics and beyond. `https://github.com/Genesis-Embodied-AI/genesis-world`, 2024.

Wenkang Hu, Xincheng Tang, Yanzhi E, Yitong Li, Zhengjie Shu, Wei Li, Huamin Wang, and Ruigang Yang. Real garment benchmark (RGBench): A comprehensive benchmark for robotic garment manipulation featuring a high-fidelity scalable simulator. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18306–18314, 2026a. doi: 10.1609/aaai.v40i22.38894.

Wenkang Hu, Haoran Wang, Yitong Li, Liu Liu, Mengao Zhao, Lai Jiang, Xincheng Tang, Junhang Wei, Zhengjie Shu, Zhendong Wang, Zhizhong Su, Huamin Wang, and Ruigang Yang. SimWeaver: Zero-shot RGB sim-to-real for deformable manipulation. *arXiv preprint arXiv:2606.15338*, 2026b.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2025. URL https://proceedings.mlr.press/v270/kim25c.html.

Tod A. Laursen and Juan C. Simo. A continuum-based finite element formulation for the implicit solution of multibody, large deformation-frictional contact problems. *International Journal for Numerical Methods in Engineering*, 36(20):3451–3485, 1993. doi: 10.1002/nme.1620362005.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Luyao Zhang, Mingrui Yu, Zelin Gao, Nan Xue, Boyu Zhou, Xing Zhu, Mingyu Ding, Yujun Shen, and Yinghao Xu. Causal world modeling for robot control. In *Proceedings of Robotics: Science and Systems (RSS)*, 2026a. doi: 10.15607/RSS.2026.XXII.016. LingBot-VA.

Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M. Kaufman. Incremental potential contact: Intersection- and inversion-free, large-deformation dynamics. *ACM Transactions on Graphics*, 39(4), 2020.

Minchen Li, Danny M. Kaufman, and Chenfanfu Jiang. Codimensional incremental potential contact. *ACM Transactions on Graphics*, 40(4), 2021.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 3705–3728. PMLR, 2025. URL https://proceedings.mlr.press/v270/li25c.html. arXiv:2405.05941.

Zeyi Li, Yushi Yang, Shawn Xie, Kyle Xu, Tianxing Chen, Yuran Wang, Zhenhao Shen, Yan Shen, Yue Chen, Wenjun Li, Yukun Zheng, Chaorui Zhang, Siyi Lin, Fei Teng, Hongjun Yang, Ming Chen, Steve Xie, and Ruihai Wu. LeHome: A simulation environment for deformable object manipulation in household scenarios. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2026b.

Xingyu Lin, Yufei Wang, Jake Olkin, and David Held. SoftGym: Benchmarking deep reinforcement learning for deformable object manipulation. In *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pp. 432–448. PMLR, 2021. URL https://proceedings.mlr.press/v155/lin21a.html.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Haoran Lu, Ruihai Wu, Yitong Li, Sijie Li, Ziyu Zhu, Chuanruo Ning, Yan Shen, Longzan Luo, Yuanpei Chen, and Hao Dong. GarmentLab: A unified simulation and benchmark for garment manipulation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2411.01200.

Siyuan Luo, Bingyang Zhou, Chong Zhang, Xin Liu, Zhenhao Huang, Gang Yang, Zhengtao Han, Xiaotian Hu, Eric Yang, Rymon Yu, Ziqiu Zeng, and Fan Shi. FLASH: Fast learning via GPU-accelerated simulation for high-fidelity deformable manipulation in minutes. *arXiv preprint arXiv:2604.17513*, 2026.

Miles Macklin, Matthias Müller, Nuttapong Chentanez, and Tae-Yong Kim. Unified particle physics for real-time applications. *ACM Transactions on Graphics*, 33(4):1–12, 2014.

Miles Macklin, Matthias Müller, and Nuttapong Chentanez. XPBD: Position-based simulation of compliant constrained dynamics. In *Proceedings of the 9th International Conference on Motion in Games*, 2016.

Mohammad Mohajerani, Aravindh Shanmuganathan, Eric Heiden, Anka Chen, Gilles Daviet, Milad Rakhsha, and Antoine Richard. Train a quadruped locomotion policy and simulate cloth manipulation with NVIDIA Isaac Lab and Newton. NVIDIA Technical Blog, `https://developer.nvidia.com/blog/train-a-quadruped-locomotion-policy-and-simulate-cloth-manipulation-with-nvidia-isaac-lab-and-newton/`, 2025. Published September 29, 2025; accessed September 24, 2026.

Yao Mu, Tianxing Chen, Zanxin Chen, Shijia Peng, Zhiqian Lan, Zeyu Gao, Zhixuan Liang, Qiaojun Yu, Yude Zou, Mingkun Xu, Lunkai Lin, Zhiqiang Xie, Mingyu Ding, and Ping Luo. RoboTwin: Dual-arm robot benchmark with generative digital twins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. Position based dynamics. *Journal of Visual Communication and Image Representation*, 18(2):109–118, 2007.

NVIDIA. NVIDIA Isaac Sim. `https://developer.nvidia.com/isaac/sim`, n.d. Undated product page; accessed September 26, 2026.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Daniel Seita, Pete Florence, Jonathan Tompson, Erwin Coumans, Vikas Sindhwani, Ken Goldberg, and Andy Zeng. Learning to rearrange deformable cables, fabrics, and bags with goal-conditioned transporter networks. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

Tencent Hunyuan3D Team. Hunyuan3D 2.0: Scaling diffusion models for high resolution textured 3D assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

The Newton Contributors. Newton: GPU-accelerated physics simulation for robotics and simulation research. `https://github.com/newton-physics/newton`, 2025. Linux Foundation project initiated by Disney Research, Google DeepMind, and NVIDIA; Newton 1.0 released 2026. Deformable solvers include XPBD, VBD, and the Style3D cloth solver.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5026–5033, 2012.

Pascal Volino and Nadia Magnenat-Thalmann. Resolving surface collisions through intersection contour minimization. *ACM Transactions on Graphics*, 25(3):1154–1159, 2006.

Xinjie Wang, Liu Liu, Yu Cao, Ruiqi Wu, Wenkang Qin, Dehui Wang, Wei Sui, and Zhizhong Su. EmbodiedGen: Towards a generative 3D world engine for embodied intelligence, 2025a.

Yuran Wang, Ruihai Wu, Yue Chen, Jiarui Wang, Jiaqi Liang, Ziyu Zhu, Haoran Geng, Jitendra Malik, Pieter Abbeel, and Hao Dong. DexGarmentLab: Dexterous garment manipulation environment with generalizable policy. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b. arXiv:2505.11032, Spotlight.

Peter Wriggers. *Computational Contact Mechanics*. Springer, 2 edition, 2006.

Xuning Yang, Rishit Dagli, Alex Zook, Hugo Hadfield, Ankit Goyal, Stan Birchfield, Fabio Ramos, and Jonathan Tremblay. RoboLab: A high-fidelity simulation benchmark for analysis of task generalist policies. In *Proceedings of Robotics: Science and Systems (RSS)*, 2026. arXiv:2604.09860.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards robust and fair evaluation of vision-language-action models beyond memorization. *arXiv preprint arXiv:2510.03827*, 2025.

Xingyu Zhu, Wenshuo Han, Zhouyu Wang, Yuran Wang, Ruihai Wu, Hao Dong, Fan Tang, Hechang Chen, Hyung Jin Chang, and Yixing Gao. FlatLab: A unified methodology framework and simulation-based benchmark for robotic manipulation of flat objects. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

A Asset Review and Expansion Details

This appendix complements Section 4.2. Each template asset first passes an automated review: the mesh loads and renders, its scale and material dependencies are complete, its deformable simulation remains stable under a standard dynamic check, and its UV layout agrees with the physical rest panel. The last check is needed because a render atlas may pack UV charts at arbitrary image-space scales, whereas the rest panel must preserve the metric lengths and areas of the material.

Expansion varies three factors independently (Table 5). *Appearance* variants apply a deterministic recipe of procedural patterns (e.g., stripes, checks, dots) and screened printed motifs; a single coverage mask drives consistent base-color, roughness, normal, and height maps, which affect rendering only. *Morphology* variants deform semantic regions such as the body, sleeves, collar, and hem with smooth weights anchored at the seams; the rest pattern is then reconstructed from the deformed geometry under seam-length constraints rather than by rescaling the UVs, and the textures are reprojected to the new layout. *Material* variants assign an explicit profile of density, thickness, anisotropic stretch, shear, and bending stiffness, damping, and friction. Every build records its recipe, random seed, and source dependencies, so each released asset can be regenerated exactly, and each passes the same stability check before release.

Table 5: Independent asset-expansion factors.

Factor	Changed	Held fixed
Appearance	render materials and printed patterns	geometry, rest pattern, material
Morphology	local geometry and reconstructed rest pattern	appearance recipe, material
Material	physical-parameter profile	geometry, rest pattern, appearance

Table 6: Template deformable assets used by the six tasks. Extents are the axis-aligned dimensions of the template mesh before task placement.

Asset	Tasks	Extents (m)
Silk	Lift silk, Flatten silk	0.350×0.350 (flat)
T-shirt	Flatten T-shirt, Fold T-shirt	$0.744 \times 0.683 \times 0.045$
Nonwoven bag	Carry bag, Pack snack	$0.222 \times 0.383 \times 0.302$

Pack snack additionally uses a rigid snack bar with a box collision shape, a mass of 0.05 kg, and a friction coefficient of 0.8.

B Simulation Details

B.1 Cloth Model and Parameters

The in-plane energy per triangle is the anisotropic membrane of Baraff & Witkin (1998): with deformation gradient columns $\mathbf{w}_u, \mathbf{w}_v$ along the warp and weft directions of the flat panel, the

stretch conditions are $C_u = A(\|\mathbf{w}_u\| - 1)$, $C_v = A(\|\mathbf{w}_v\| - 1)$ and the shear condition $C_s = A\,\mathbf{w}_u^\top \mathbf{w}_v$, each with its own stiffness k_u, k_v, k_s and quadratic energy $\frac{1}{2}kC^2$. Bending is the discrete quadratic energy of Bergou et al. (2006) on each interior edge with an anisotropic edge stiffness. The three stiffnesses, the bending stiffness, and the areal density are per-asset material parameters stored with the mesh; the fabric thickness t_0 used by the contact formulation is stored alongside them.

B.2 Solver

Each substep solves $(\mathbf{M}/\Delta t^2 + \mathbf{H})\,\delta\mathbf{x} = \mathbf{r}$ with a constant $\mathbf{H}$ assembled from the rest-shape elastic Hessians once at load time, following projective dynamics (Bouaziz et al., 2014). Contact terms (Appendix B.3) are added to the diagonal of the system at every iteration. The linear system is solved by conjugate gradients with a Jacobi preconditioner; the number of linear iterations ramps with the nonlinear iteration index, and the number of nonlinear iterations per 2 ms substep is set per task; velocities are updated from the converged positions. Self-contact uses a BVH broad phase rebuilt every substep, vertex–triangle and edge–edge repulsion within a thickness band, and an edge–face untangling force (Volino & Magnenat-Thalmann, 2006) that acts only on already-intersecting pairs.

B.3 Contact Implementation

Distance queries. Each rigid collision shape (gripper fingers, table, free objects) is voxelised once at load time into a signed distance field with a per-voxel nearest-triangle table, so the closest point on the shape to any query point is obtained exactly from the stored triangle without a tree traversal. For a cloth triangle the minimisation $\min_{\mathbf{x}\in\triangle} \phi(\mathbf{x})$ is seeded at its three vertices, its centroid, and at up to n_e intersection points of the rigid mesh's edges with the triangle plane; each seed is refined by a few projected descent steps inside the triangle and the best point is kept. The edge seeds are needed because a finger edge of thickness comparable to the mesh edge length can pass through the interior of a triangle while all four default seeds sit farther than h from the body.

Normal law. $f_n = k\,(h - d)$ with $k = E_t A/t_0$. We use $E_t = 20$ kPa, the low-strain transverse compression modulus reported for woven and knitted fabrics; t_0 is read from the asset (1 mm for the T-shirt, 2 mm for the silk) and sets both the stiffness and the shell half-thickness, $h = t_0/2$, so that the contact surface coincides with the physical fabric surface. The force acts at the closest point along the field gradient and is distributed to the three vertices with the barycentric weights w_i of that point; the stiffness is added to the Hessian diagonal with the same weights.

Friction coefficient. A single cloth-side coefficient is combined geometrically with the shape coefficient of each rigid surface, and the table and free objects use the same cloth-side coefficient as the fingers. The coefficients are set with the asset and task configurations and released with them.

Anchored friction. On contact formation the closest point is stored in the rigid body's local frame as the anchor $\mathbf{p}$; on every subsequent iteration the tangential offset $\mathbf{s} = (\mathbf{I} - \mathbf{n}\mathbf{n}^\top)(\mathbf{x}_c - \mathbf{p})$ of the current closest point $\mathbf{x}_c$ from the anchor gives $\mathbf{f}_t = -k_t\mathbf{s}$ with $k_t = \kappa k$, $\kappa = 3$. If $\|\mathbf{f}_t\| > \mu f_n$, the force is scaled onto the cone and the anchor is moved so that the stored offset equals the cone limit (return mapping). The anchor is released when the contact deactivates. The pre-sliding displacement is $\mu(h - d)/\kappa$, about 0.1 mm at a typical grasp compression.

Consistent tangential Hessian. The anchor stiffness is added to the Hessian diagonal as the row sum of the full 3×3 block, $\sum_j k_t w_i w_j = k_t w_i$, rather than $k_t w_i^2$, which would under-estimate the stiffness of the rigid translation mode threefold and make the contact oscillate at the cone limit. With this lumping the tangential force on a cloth strip resting on a static plate falls from 7.7 N to below 0.1 N.

Deep-penetration guard. When a triangle's closest point lies deeper than 0.5 mm beyond the shell boundary, its vertices are projected back to the shell after the position update; forces and reaction wrenches are evaluated on the pre-projection positions so that the projection does not starve the force law. This is a safeguard for transients, not the mechanism that resolves contact.

B.4 CO-SIMULATION

After the cloth solve of substep n converges, a kernel re-evaluates every active contact's normal and tangential force at the converged positions and accumulates $-\mathbf{f}$ and $-(\mathbf{x}_c - \mathbf{c}_b) \times \mathbf{f}$ per rigid body b with centre of mass $\mathbf{c}_b$. The wrench is written to the body force buffer consumed by the MuJoCo solver as an applied external wrench in substep $n + 1$. Wrenches act on the gripper finger links and on free rigid objects; the arm links are driven by their own actuators. The gripper joints are position actuators with $k_p = 2 \times 10^4$ N/m, $k_d = 100$ N s/m, and a force limit of 6 N per finger, so a closed grasp exerts 6 N regardless of the residual opening. The stability bound of Sec. 3.3 evaluates, for the finger of our gripper ($M = 0.030$ kg including armature, $c = 105$ N s/m including the velocity gain, $k_p = 2 \times 10^4$ N/m, $\Delta t = 2$ ms), to $k_{\text{crit}} = 1.15 \times 10^5$ N/m; the closed form, a numerical reproduction of the integrator update, and the spectral radius of the true one-step map of the rigid-body solver agree to 0.002%. The summed contact stiffness of the linear law during a grasp is $2\text{–}4 \times 10^4$ N/m, below this bound.

B.5 GPU EXECUTION

A substep consists of about five hundred small kernels (cloth iterations, collision queries, rigid-body step, force exchange). In a direct implementation two costs dominate: host-side kernel-launch latency and the closest-point queries between cloth triangles and rigid meshes. Three measures address them. (i) The entire substep loop is captured once as a CUDA graph and replayed, which removes the launch latency; two graphs are kept for odd substep counts so that the double-buffered state is read consistently, and kinematic targets are interpolated on the device so that no host read-back breaks the graph. (ii) Closest-point queries use the per-shape nearest-triangle table of Appendix B.3 instead of a tree traversal. (iii) The conjugate-gradient inner loop is fused into a few kernels. Table 7 disables each measure in turn on T-shirt flattening with the full contact stack (10 nonlinear iterations, 2 ms substep, one H20 GPU with exclusive access). Removing the CUDA graph or the lookup table each roughly triples the substep time, whereas fusing the conjugate gradients saves 15%. With all three measures in place, the contact formulation takes about 16% of the GPU time of a substep (measured at 20 nonlinear iterations); most of the remainder is the cloth solver's linear solves and the self-collision broad phase.

Table 7: GPU execution ablation on T-shirt flattening with the full contact stack (9,371 vertices, one H20, exclusive access, median of three runs, per-frame synchronised timing).

Configuration	ms / substep	Relative	Kernel launches / substep
Default (graph, fused CG, lookup table)	6.8	1.00	0
Without CUDA graph	19.8	2.92	519
Without fused conjugate gradients	7.8	1.15	0
Without nearest-triangle lookup table	20.4	3.00	0

B.6 ABLATION DETAILS

Criteria of Table 2a. Each cell uses 20 seeds of scripted trajectories. A finger–fabric *intersection event* is a run of at least 10 consecutive frames, with the jaws at least 90% open, in which a triangle of the finger mesh intersects a cloth triangle. *Release* (T-shirt): after the jaws open, the fabric falls back as defined in Appendix B.7, so fabric carried up on a finger counts as a failure whether or not it intersects the finger. *Penetration* (silk): an intersection event occurs after the jaws open. *Lift* (pack): when the bag is raised by more than 5 cm, the packed item rises by at least half the bag's rise. Runs that diverge or fail inverse kinematics count as failures.

Edge seeds. Removing the rigid-edge seeds of the closest-point search (20 seeds per cell) leaves grasp, hold, and the Pack snack task unchanged; on the T-shirt an intersection event follows opening in 2 of 20 runs, and a finger that pierces the fabric on opening stays inside it until separation in 3 of 20, against none with the full model.

B.7 Behavioural Criteria

All grasp-related criteria are binary per hand and computed from cloth vertex positions and gripper state only, so they apply to any simulator. *First grasp*: after the first close command the grasp point rises with the hand by at least half the commanded lift, without retry. *Hold*: the grasp point stays within 30 mm of the tool centre point from the first grasp until the open command, throughout carrying or flinging. *Release*: once the jaws open, the fabric falls back instead of being carried by the fingers, i.e., the highest cloth vertex stays within 5 mm of its height at the moment of opening. Table 1, whose trials end with the arms retracted, additionally requires that no cloth vertex lies within 3 cm of either jaw centre at the end of the trial. Both tests use cloth vertices only and therefore apply to every simulator; the same definition is used in Table 2a. *Success*: grasp, hold, and release all pass and the trial does not diverge. *Diverged*: the cloth velocity exceeds a fixed threshold or a non-finite state appears. Counts are reported per trajectory and never averaged, because release failures are intermittent rather than noisy.

Cross-simulator details. Table 8 breaks down Table 1. The step times in Table 1 are measured on one RTX 4090 with the 9,371-vertex T-shirt as wall-clock time per physical step, each simulator in the configuration of Table 1 (Newton VBD at twice its released substep rate, see below), as the median of three runs excluding rendering and initialization; they are not comparable with the H20 timings of Table 7. *Genesis.* Under the same protocol, with both Piper arms coupled to the cloth, Genesis (Genesis Authors, 2024) (v1.4.2, IPC cloth, parameters of its robot–cloth example) takes 35.9 ms per step, $7.5\times$ our contact (4.8 ms). The grasp does not form in Genesis, so this is a lower bound. *Isaac Sim.* The fingers pass through the fabric while closing, so it is not held between the jaws; one position forms a single-hand grasp. In a separate sweep, commanding the fingertips 15 mm below the table, which is unsafe on the real robot, forms a grasp at only 3 of 19 positions, and the outcome flips under 2 cm shifts. *Newton VBD.* At the released 1440 Hz step the simulation diverges in 39 of 45 trials, during closing or the fling. Doubling the iteration count does not remove it; halving the step reduces it to 12 of 45 (the setting in Table 1). Trials that hold never release: the fingers remain inside the fabric when the jaws open. *Newton Style3D.* The released contact holds in every trial, but the fingers remain inside the fabric and it never releases; it releases only if collisions are disabled at opening.

Table 8: Per-simulator breakdown (45 trials each; same replay as Table 1).

	Grasp	Hold	Release	Diverged
Isaac Sim, PBD cloth	3/45	0/45	–	0/45
Newton VBD, 1440 Hz	30/45	27/45	0/45	39/45
Newton VBD, 1440 Hz, 24 / 48 iter.[a]	–	–	–	3/3, 3/3
Newton VBD, 2880 Hz	39/45	39/45	0/45	12/45
Newton VBD, 5760 Hz[a]	–	–	–	0/3
Newton Style3D	45/45	45/45	0/45	0/45
Newton Style3D, our contact	45/45	45/45	45/45	0/45

[a]Three positions only.

C Configuration Ladder: Generation Details

Each level is generated inside the task environment by a deterministic program whose random choices are driven by the episode seed. For silk, C0 selects one of the two far corners and folds it inward; its sampled displacement is 0.15–0.20 m in the long direction and 0.05–0.15 m laterally. C1 samples a fold count uniformly from one or two, samples that many distinct corners uniformly without replacement from all four corners, and applies the same inward fold sequentially in the sampled order. C2 samples two mesh pins with a separation of 0.10–0.45 m, raises the cloth centroid to 0.32–0.48 m with a 0.15–0.25 m horizontal offset, holds for 1–3 s, releases the cloth, settles it for 2 s, recenters it, and settles it for another 0.3 s. The silk C0/C1 fold primitive executes three segments of 0.4, 0.6, and 0.4 s, places the selected vertex at 0.005 m, pins it for 0.3 s, then releases and settles for 1.5 s.

For the T-shirt, C0 is a right-sleeve-corner inward fold, C1 is a right-half inward fold, and C2 is a randomly selected deep fold over the four corner regions. Figure 3 shows representative states. Within each ladder, the asset identity, physics, task parameters, and planar placement distribution are held fixed.

D SIM-TO-REAL CELLS

Table 9: Per-cell numbers behind Figure 4: sim-to-real difficulty ordering with $\pi_{0.5}$. Simulation cells use $N{=}50$ and completed real-robot cells use $N{=}20$. Simulation rates match Table 3.

Task	Sim SR (%)	Real SR (%)	Real trials
Lift silk	98	70	20
Flatten T-shirt	82	55	20
Flatten silk	86	65	20
Carry bag	66	50	20
Pack snack	92	85	20
Fold T-shirt	36	35	20
Spearman $\rho = 0.94$; bootstrap 95% CI: $[0.26,\ 0.99]$			
Kendall $\tau_b = 0.87$; bootstrap 95% CI: $[0.20,\ 0.97]$			

The correlation intervals use 20,000 within-cell episode bootstrap replicates (random seed 20260924), resampling binary outcomes independently in simulation and on the real robot while holding the six tasks fixed. We report percentile 95% intervals; these quantify episode-sampling uncertainty for the evaluated tasks rather than uncertainty over a larger population of tasks.

E TASK DEFINITIONS AND SUCCESS CRITERIA

E.1 COVERAGE OF PHYSICAL REGIMES

The five regimes in Section 4.1 are overlapping physical requirements rather than a partition of the task set. Table 10 makes the intended mapping explicit. A mark denotes a primary capability exercised by the task, not a claim that all other physical effects are absent.

Table 10: Mapping from the six tasks to the five physical regimes. SC: self-contact; CR: recovery under self-occlusion; MT: dynamic momentum transfer; FH: friction-limited hold without attachment; VB: volumetric-bag load transmission.

Task	SC	CR	MT	FH	VB
Fold T-shirt	✓				
Flatten T-shirt		✓	✓		
Flatten silk		✓	✓		
Lift silk				✓	
Carry bag				✓	✓
Pack snack				✓	✓

E.2 EVALUATOR SEMANTICS

The evaluator observes the simulation at a fixed polling interval (1 s). Unless a task states otherwise, a *sustained* predicate means that it is true at two consecutive evaluation polls. "Ever" means true at any poll; "end-held" means true at the final two polls. Threshold comparisons are strict when the implementation uses $<$ or $>$ and inclusive when it uses $\leq$ or $\geq$. Continuous scores are independent of the binary success flag and remain defined for failed episodes.

Lift silk. At a poll, one hand holds the silk if the nearest cloth particle is less than 0.05 m from its end effector and that particle is above 0.10 m. The whole cloth is lifted if its mean particle height

is above 0.05 m. Success requires both hand-hold predicates and the whole-cloth predicate for two consecutive polls. Each hand contributes 50 points if it is end-held, 25 if it was held at least once but is not end-held, and 0 otherwise. The two hand scores are summed.

Flatten silk. After the policy horizon, both grippers open and the cloth settles for 2 s. Coverage is the union area of table-plane particle footprints with radius 0.005 m, divided by the corresponding area of the canonical flat mesh and clipped to one for scoring. Success requires final coverage at least 0.95 and maximum particle height below 0.05 m. The score is 100 times final settled coverage. The full unclipped coverage and height series are retained.

Flatten T-shirt. The evaluator combines projected coverage c with a planar, rigidly aligned distance d_w to the canonical flat mesh. The distance is a weighted bidirectional correspondence measure after iterative rigid alignment; lower is better. The current success rule is

$$c \geq 0.60, \qquad d_w \leq 0.25, \qquad \bar{z} < 0.03\,\mathrm{m}.$$

The score is

$$100 \left[0.4\ \mathrm{clip}\left(\frac{c - 0.25}{0.78 - 0.25}, 0, 1 \right) + 0.6\ \mathrm{clip}\left(\frac{0.36 - d_w}{0.36 - 0.196}, 0, 1 \right) \right].$$

The episode ends at the first successful poll. Initial, final, and canonical particle arrays are stored to permit offline rescoring.

Fold T-shirt. The evaluator freezes a garment-local frame at the first poll and awards 30 points for the left side fold, 30 for the right side fold, and 40 for the forward fold. A side is latched when its sleeve-center and hem keypoints are both within 0.35 of their initial lateral distance from the torso centerline and rest below 0.10 m. The forward stage is latched when the mean hem keypoint has passed the torso center by at least 0.10 of the initial garment length and rests below 0.14 m. The final 40 points and binary success additionally require a settled envelope: mean height below 0.09 m, maximum height below 0.30 m, footprint width no greater than 0.60 of its initial value, and footprint length no greater than 0.65 of its initial value. Success requires all three stages and the final envelope. Both grippers are released before a 2 s terminal settle.

Carry bag. The handle position, handle stiffness, and bag appearance are randomized in both training and evaluation. Evaluation uses the ten training-pool appearances, with five episodes per appearance and held-out initialization seeds. All vertical displacements are measured relative to the first evaluation poll. A side is valid when the mean height of its strap tip and mid vertices rises by more than 0.15 m and the nearest strap vertex is less than 0.02 m from the corresponding TCP pinch center. The bag predicate requires its mean particle height to rise by more than 0.03 m. Success requires both side predicates and the bag predicate at two consecutive polls. Each side contributes 50 points if sustained, 25 if true at only a nonsustained poll, and 0 otherwise; bag lift gates success but does not add score. The episode ends at the first sustained lift window: Carry bag success is defined by friction-limited handle acquisition and sustained lifting, not by arrival at a separate transport target.

Pack snack. Relative to the first poll, the rigid bar is "up" after rising by more than 0.10 m. It is "in bag" when its table-plane distance from the dynamically estimated bag-mouth center is less than 0.12 m and its height is at least 0.02 m below the estimated mouth rim. Each strap is valid after rising by more than 0.10 m while remaining within 0.02 m of its TCP pinch center. Loaded carry requires the bag mean height to rise by more than 0.05 m while the bar remains in the bag and rises by at least half the bag's rise, so a bar left on the table under the lifted bag does not count. The score assigns 20 points to bar lift, 30 to insertion, 20 to each valid strap, and 10 to loaded carry, with every stage requiring two consecutive polls. Success requires sustained insertion and a sustained joint window in which both straps and loaded carry are valid.

Process traces and failure labels. Besides the binary outcome and the score, each episode logs process traces (coverage over time, semantic-keypoint errors, grasp duration, and stage-completion indicators), which distinguish policies that fail at different points even when their success rates are identical. Failures are labeled by observed behavior: missed acquisition, acquisition followed by

early slip, incomplete fold or flatten stage, failure to settle, failed insertion/containment, one-sided handle acquisition, empty bag lift, and loss during carry. These labels describe observed behavior only and do not infer an unmeasured physical cause. The raw per-poll stage series is retained so that labels can be reproduced or refined offline.

E.3 Demonstration Generation and Data Format

Each task provides a scripted expert expressed through semantic asset keypoints, object-relative targets, and reusable bimanual motion primitives. At the start of an episode, the task randomizer samples the placement and initial state from a seeded distribution, and the planner instantiates the expert trajectory for that scene; collection and evaluation therefore share the same physics, control frequency, randomization, and success checker. Collection seeds are disjoint from evaluation seeds.

Demonstrations are stored in LeRobot format at 25 Hz. Each frame contains three RGB streams, one top camera and two wrist cameras, rendered at 1280×720. State and action are both 14-dimensional: six joints and one gripper value per arm, with the gripper normalized to 0 closed and 1 open.

Cameras. The three simulated cameras replicate the Intel RealSense D435i cameras of the real dual-arm robot and are identical across tasks. The top camera is fixed in the world about 0.77 m above the table, looking forward and 60° below the horizontal. Each wrist camera is rigidly attached to the last link of its arm at the pose obtained from the real robot's calibration, with its optical axis 20° from the link axis. All three are ideal pinhole cameras with a $70.3^\circ \times 43.2^\circ$ field of view (focal length $\approx$ 908 px at 1280×720, principal point at the image centre) and are rendered with path tracing. The real cameras stream at the same resolution, and policies receive images at 25 Hz in both settings.

F Policy Training and Unified Evaluation Protocol

The benchmark covers ACT, Diffusion Policy (DP), $\pi_{0.5}$, GR00T, LingBot-VA, and Fast-WAM. Table 11 records the current reference recipes. The scheduled terminal checkpoint is evaluated; any result from an earlier checkpoint is marked in the corresponding table. When the number of devices changes, steps, per-device batch size, or gradient accumulation are adjusted to preserve the total number of training samples; model revisions and algorithm-specific augmentation remain fixed.

Table 11: Reference training recipes. "Effective batch" includes all devices and gradient accumulation.

Policy	Schedule	Effective batch / devices	Notes
ACT	30k steps	16 on one device	save every 5k
DP	12.5k one-device equivalent	64	6.25k at two devices
$\pi_{0.5}$	30k steps	8	bf16, grad. checkpointing
GR00T	20k steps	global batch 32	16-step prediction horizon
LingBot-VA	30k steps	8 devices, effective 8	task-specific action quantiles
Fast-WAM	5 epochs	8 devices, 16/device	ZeRO-1; no CPU offload

The $\pi_{0.5}$ base checkpoint is pinned to revision `9e55186ad36e66b95cda57bc47818d9e6237ae30`. Fast-WAM inference uses action horizon 32, replans every 32 actions, and uses 20 denoising steps unless a run manifest explicitly overrides them. GR00T predicts a 16-step horizon by default.

Each principal task–condition–policy cell targets $N = 50$ completed episodes from one fixed evaluation-seed schedule shared across policies. This is 50 total episodes, not 50 episodes for each of three seeds. The runner resumes incomplete cells and retries failed workers; an episode JSON that is absent is not counted in N. Success rates are reported with two-sided Wilson 95% confidence intervals (Tables 12 and 13); the mean score is reported as a sample mean. With N=50, an interval spans roughly ±14 points at a 50% success rate and narrows toward the extremes.

Table 12: Wilson 95% confidence intervals (%) for the success rates of Table 3 (N=50 per cell).

Policy	Lift silk	Flatten silk	Flatten T-shirt	Carry bag	Pack snack	Fold T-shirt
$\pi_{0.5}$	90–100	74–93	69–90	52–78	81–97	24–50
LingBot-VA	52–78	74–93	74–93	37–63	90–100	24–50
Fast-WAM	44–71	50–76	81–97	67–89	3–19	0–7
GR00T	56–81	33–60	58–83	28–54	40–67	0–10
ACT	17–42	4–21	56–81	26–52	39–65	19–44
DP	26–52	0–7	65–87	3–19	10–31	0–7

Table 13: Wilson 95% confidence intervals (%) for the success rates of Table 4 (N=50 per cell).

	Flatten T-shirt				Flatten silk			
Policy	C0	C1	C2	C0 + app.	C0	C1	C2	C0 + app.
$\pi_{0.5}$	69–90	31–58	6–24	63–86	74–93	11–33	0–10	50–76
LingBot-VA	74–93	48–74	4–21	71–92	74–93	13–35	0–7	74–93
Fast-WAM	81–97	22–48	7–26	29–56	50–76	8–29	0–7	0–7
GR00T	58–83	22–48	7–26	44–71	33–60	10–31	0–7	26–52
ACT	56–81	8–29	2–16	17–42	4–21	2–16	0–7	1–13
DP	65–87	21–46	4–21	37–63	0–7	0–7	0–7	0–7