

Graph-Operator World Models for Morphology-Parameter Generalization in Continuous Control

Xu Yang¹, Yiqin Yang², Qianchuan Zhao¹

¹Tsinghua University

²The Key Laboratory of Cognition and Decision Intelligence for Complex Systems,
Institute of Automation, Chinese Academy of Sciences

Abstract

World models for continuous control are commonly trained for a fixed physical system and can degrade when known morphology parameters such as link lengths, masses, damping, and actuation change. Existing approaches often provide these parameters as conditioning information, but leave unspecified which part of the learned transition should remain reusable and which part should change with morphology. We propose *Graph-Operator World Models* (GraphOp-WM), a structured world model for generalization across unseen morphology parameters within related articulated robot families. GraphOp-WM represents bodies and their kinematic relations as an attributed graph and factorizes each transition into a morphology-independent local dynamics basis and a morphology-conditioned structured operator. The operator combines node-local modulation, kinematic-tree coupling, and a low-rank global correction, while architectural information separation, basis normalization, and paired-morphology supervision encourage static morphology dependence to be carried by the operator pathway. Graph-level readout and edge-wise action representations provide a compatible interface for reward, value, and TD-MPC-style planning. We further define controlled MuJoCo parameter splits covering interpolation, extrapolation, and held-out compositions of link geometry, mass, damping, and actuation parameters in Hopper, Walker2d, and HalfCheetah.

1 Introduction

World models support sample-efficient reinforcement learning by predicting latent transitions, rewards, and values for planning or imagined policy optimization. DreamerV3 (Hafner et al., 2025), TD-MPC2 (Hansen et al., 2024), and PWM (Georgiev et al., 2025) demonstrate that learned dynamics models can support strong control across broad task suites. Their generality is primarily obtained through model capacity, shared task embeddings, and multitask data. Recent morphology-conditioned world models additionally encode robot specifications or structural embeddings, showing that known embodiment information can improve dynamics modeling under embodiment variation (Danesh et al., 2026; Wang et al., 2026; He et al., 2025). However, conditioning morphology as an input does not by itself specify which components of the transition should be invariant across parameter changes and which components should account for morphology-induced changes in dynamical coupling.

Robotic dynamics additionally expose explicit compositional structure. Even within a fixed articulated topology, changing link geometry, body mass, inertia, joint damping, or actuator gear changes how local motion propagates through the robot. These changes alter the transition field while preserving substantial reusable local structure. A generic context vector can inform a dynamics model that the morphology has changed, but it does not prescribe how morphology parameters should transform the coupling between local dynamic responses.

We study the following question:

Can a compact world model generalize to unseen physical morphology parameters by explicitly separating reusable local dynamics from morphology-induced coupling?

We introduce GraphOp-WM, a graph-structured world model designed around this separation. Bodies are represented as nodes, while joints and actuators are represented as edges. A shared local kernel predicts a

canonical increment for each body using only local dynamic state, local action information, and a discrete body type. It is deliberately prevented from observing masses, lengths, inertias, morphology identifiers, or pooled morphology embeddings. A separate graph encoder converts the MuJoCo morphology graph into a structured transition operator that acts on the shared local increments.

For body-token matrix $Z_t \in \mathbb{R}^{N_m \times d}$, the transition is

$$\hat{Z}_{t+1} = Z_t + \mathcal{A}_{m,t} \bar{B}_t, \quad (1)$$

where N_m is fixed within a robot family, $\bar{B}_t$ is a normalized shared local dynamics basis, and $\mathcal{A}_{m,t}$ is a morphology-parameter-conditioned graph operator. The operator is structured as

$$\mathcal{A}_{m,t} = I + \text{Diag}(d_{m,t}) + E_m \text{Diag}(w_{m,t}) E_m^\top + U_{m,t} \text{Diag}(s_{m,t}) U_{m,t}^\top. \quad (2)$$

The diagonal component captures body-local response, the incidence-matrix component follows the kinematic tree, and the low-rank term captures residual global modes such as trunk-limb and contact-mediated coupling. We do not claim that this operator is an exact inverse mass matrix. It is a constrained transition parameterization motivated by the locality and repeated structure of articulated dynamics.

GraphOp-WM uses graph-level readout for reward and value prediction and attaches actions to actuator edges. This preserves the physical organization of the robot in the predictive and control interfaces without requiring morphology parameters to be collapsed into a single unconstrained context vector.

Our contributions are:

- We formulate morphology-parameter generalization for world models: predicting and planning under unseen changes in physical parameters while keeping the articulated topology fixed within each robot family.
- We introduce a transition factorization that separates a morphology-independent local dynamics basis from a structured operator conditioned on static morphology parameters, combining local, graph-supported, and low-rank global coupling.
- We impose architectural information separation, basis normalization, and paired-morphology supervision so that morphology-dependent transition changes are assigned to the operator rather than absorbed by the shared basis.
- We specify controlled MuJoCo splits for interpolation, extrapolation, and held-out compositions of geometry, mass, damping, and actuation parameters, enabling direct measurement of parameter-efficient out-of-distribution dynamics and planning generalization.

2 Related Work

World models for continuous control. PETS (Chua et al., 2018) and MBPO (Janner et al., 2019) learn probabilistic dynamics for planning or policy optimization. DreamerV3 learns recurrent latent dynamics and trains behavior through imagined trajectories (Hafner et al., 2025). TD-MPC combines a task-oriented latent dynamics model with short-horizon planning and terminal value estimation (Hansen et al., 2022). TD-MPC2 uses a decoder-free latent model and short-horizon planning across a large continuous-control suite (Hansen et al., 2024). PWM pretrains multitask world models and extracts policies through first-order gradients (Georgiev et al., 2025). GraphOp-WM retains the control-centric modeling and planning interface of TD-MPC, but introduces an explicit factorization of reusable local dynamics and morphology-parameter-dependent coupling.

Contextual and morphology-conditioned world models. CaDM infers a dynamics context from recent transitions before conditioning a forward model on that context (Lee et al., 2020), while MAMBA uses recurrent world-model states to represent task information over a meta-episode (Rimon et al., 2024). These approaches are suited to latent properties that must be inferred from interaction. Robot geometry, inertial parameters, and joint connectivity are instead available before interaction in MJCF, URDF, or USD specifications. QWM conditions a Dreamer-style model on normalized quadruped specifications (Danesh et al.,

2026), and WestWorld combines structural embeddings with system-aware mixture-of-experts routing across heterogeneous robotic systems (Wang et al., 2026). A particle-based graph world model provides a shared state and action interface across dexterous embodiments (He et al., 2025). GraphOp-WM targets a narrower but more explicit question: for known changes in physical morphology parameters, it treats morphology as a structured operator over local dynamic responses rather than only as a pooled vector, token embedding, or expert-routing signal. This yields a concrete inductive bias for extrapolating to unseen parameter values and combinations.

Morphology-conditioned control. NerveNet represents an articulated agent as a graph and shares policy computation across body parts (Wang et al., 2018). Shared Modular Policies instantiate one reusable module per actuator and coordinate modules through messages along the morphology (Huang et al., 2020). MetaMorph treats morphology as a conditioning modality for a Transformer controller (Gupta et al., 2022). Body Transformer restricts attention using a sensor-actuator graph (Sferrazza et al., 2025), and URMA uses morphology-agnostic encoders and decoders for multi-embodiment locomotion (Bohlinger et al., 2025). These works focus primarily on transferring control computation across bodies or embodiments, whereas GraphOp-WM applies graph structure to factorize predictive dynamics under morphology-parameter variation.

Graph dynamics and learned operators. Interaction Networks and graph-network simulators establish object- and relation-centric inductive biases for physical prediction and control (Battaglia et al., 2016; Sanchez-Gonzalez et al., 2018). Contrastively learned structured world models use graph transitions over object-centric latent states (Kipf et al., 2020). Compositional Koopman operators combine graph encoders with block-structured linear dynamics for systems with variable object counts (Li et al., 2020). MS-HGNN further incorporates kinematic structure and morphological symmetry into robotic dynamics learning (Xie et al., 2025). GraphOp-WM factorizes prediction into a morphology-independent local basis and a graph-generated operator. The graph-supported term follows the kinematic tree explicitly, while a low-rank component represents residual nonlocal coupling.

3 Problem Setting

We consider a family of control problems indexed by morphology m :

$$\mathcal{M}_m = (\mathcal{S}_m, \mathcal{A}_m, T_m, r_k, \Omega_m, \rho_{0,m}, \gamma), \quad (3)$$

where state and action dimensions may vary with the robot, while k indexes the control objective. Each morphology is represented by an attributed graph

$$\mathcal{G}_m = (\mathcal{V}_m, \mathcal{E}_m^{\text{joint}}, \mathcal{E}_m^{\text{act}}). \quad (4)$$

Body nodes $i \in \mathcal{V}_m$ contain geometry and inertial metadata. Joint edges contain kinematic relations, and actuator edges associate commands with controlled joints.

The training set contains trajectories from morphologies $m \in \mathcal{M}_{\text{train}}$. The primary objective is zero-shot dynamics prediction and planning on unseen variants in $\mathcal{M}_{\text{test}}$ from related articulated families. The benchmark separates interpolation, parameter extrapolation, held-out parameter combinations, and within-family topology extension. Universal transfer to arbitrary robot families remains outside the scope of the formulation.

4 Method

4.1 Morphology Graph

We parse MuJoCo XML or DMControl specifications into a body-joint graph. For body i , static node attributes are

$$p_i^v = [m_i, I_i, \text{geom}_i, \text{type}_i, \text{contact}_i, \text{frame}_i]. \quad (5)$$

For joint edge (i, j) , static attributes are

$$p_{ij}^e = [\text{jointtype}_{ij}, \text{axis}_{ij}, \text{origin}_{ij}, \text{range}_{ij}, \text{damping}_{ij}, \text{gear}_{ij}]. \quad (6)$$

Dynamic body features include local position, orientation, linear and angular velocity, and contact signals. Joint position, joint velocity, and actuator command are attached to the corresponding edge.

Graphs are batched by concatenating nodes and edges and storing a graph-membership index, as in standard packed graph processing. Within each robot family, the graph topology and the correspondence between bodies, joints, and actuators are fixed. Every body token has fixed channel dimension d ; morphology variation is represented through the static node and edge attributes rather than changes in graph size.

4.2 State Tokens and Global Readout

A shared body encoder produces dynamic node tokens

$$z_{t,i} = E_\phi(x_{t-L:t,i}^v, x_{t-L:t,i}^e, \text{type}_i) \in \mathbb{R}^d. \quad (7)$$

The encoder may use a short recurrent or attention history for partial observations. It does not receive masses, lengths, inertias, morphology identifiers, or pooled morphology embeddings. This blocks direct access to static morphology through the state pathway.

For scalar graph-level predictions, a learned global query attends to all body tokens:

$$\alpha_{t,i} = \text{softmax}_i \left(\frac{q_g^\top W_k z_{t,i}}{\sqrt{d}} \right), \quad (8)$$

$$h_t^g = \sum_{i=1}^{N_m} \alpha_{t,i} W_v z_{t,i}. \quad (9)$$

We concatenate this readout with the root-body token:

$$g_t = [z_{t,\text{root}}; h_t^g] \in \mathbb{R}^{2d}. \quad (10)$$

Thus reward and value heads have fixed input dimension without fixing the number of bodies.

4.3 Morphology-Independent Local Dynamics Basis

For each body, the shared local kernel predicts an unconstrained candidate increment using local state, incident dynamic joint features, and local actuator commands:

$$B_{t,i} = b_\theta \left(z_{t,i}, \text{Agg}_{j \in \mathcal{N}(i)} [q_{t,ij}, \dot{q}_{t,ij}, a_{t,ij}], \text{type}_i \right). \quad (11)$$

The kernel is shared across all bodies and morphologies and has no access to static morphology parameters. To remove the scale ambiguity between the basis and operator, we normalize each graph's basis by its root-mean-square magnitude:

$$\bar{B}_t = \frac{B_t}{\sqrt{\frac{1}{N_m d} \|B_t\|_F^2 + \epsilon}}. \quad (12)$$

4.4 Structured Morphology Operator

A morphology encoder processes only static graph attributes and returns node coefficients, edge coefficients, and r node-aligned global modes:

$$(d_m, w_m, \tilde{U}_m, s_m) = G_\psi(\mathcal{G}_m). \quad (13)$$

Here $d_m \in \mathbb{R}^{N_m}$, $w_m \in \mathbb{R}^{|\mathcal{E}_m^{\text{joint}}|}$, $\tilde{U}_m \in \mathbb{R}^{N_m \times r}$, and $s_m \in \mathbb{R}^r$. The same node and edge encoders are reused for every graph, so their parameter dimensions do not depend on N_m . We orthogonalize the global modes within each graph:

$$U_m = \text{qr}(\tilde{U}_m), \quad U_m^\top U_m = I_{r_m}, \quad (14)$$

where $r_m = \min(r, N_m)$ is the effective rank for morphology m .

The static coefficients are modulated by shared state-dependent gates:

$$d_{m,t,i} = d_{m,i} \tanh(g_d(z_{t,i})), \quad (15)$$

$$w_{m,t,ij} = w_{m,ij} \tanh(g_e(z_{t,i}, z_{t,j}, x_{t,ij}^e)), \quad (16)$$

$$s_{m,t} = s_m \odot \tanh(g_s(h_t^g)). \quad (17)$$

The gates contain no trainable morphology identifiers; they only activate the static structure according to the current pose, velocity, and contact state.

Let $E_m \in \mathbb{R}^{N_m \times |\mathcal{E}_m^{\text{joint}}|}$ be the oriented incidence matrix of the kinematic graph. The transition operator is

$$\mathcal{A}_{m,t} = I + \text{Diag}(d_{m,t}) + E_m \text{Diag}(w_{m,t}) E_m^\top + U_m \text{Diag}(s_{m,t}) U_m^\top. \quad (18)$$

The predicted transition is

$$\hat{Z}_{t+1} = Z_t + \mathcal{A}_{m,t} \bar{B}_t. \quad (19)$$

Equation (18) is not presented as an exact rigid-body solver. Its components encode three inductive biases: local response, coupling supported by the kinematic tree, and a small number of nonlocal residual modes. The low-rank term is retained only if numerical transition-Jacobian diagnostics show a rapidly decaying residual spectrum after removing local and graph-supported components.

4.5 Information-Separated Factorization

The architecture enforces information separation: static morphology attributes enter only $\mathcal{A}_{m,t}$, while the local basis observes only dynamic local state and action. Basis normalization removes arbitrary reciprocal scaling between $\bar{B}_t$ and $\mathcal{A}_{m,t}$. We additionally use paired-morphology transitions within each robot family.

For two variants initialized at an aligned normalized pose and driven by the same normalized action sequence, a single local basis is used with two operators:

$$\widehat{\Delta Z}_t^{m_1} = \mathcal{A}_{m_1,t} \bar{B}_t, \quad (20)$$

$$\widehat{\Delta Z}_t^{m_2} = \mathcal{A}_{m_2,t} \bar{B}_t. \quad (21)$$

The paired loss is

$$\mathcal{L}_{\text{pair}} = \left\| \widehat{\Delta Z}_t^{m_1} - \Delta Z_t^{m_1} \right\|_2^2 + \left\| \widehat{\Delta Z}_t^{m_2} - \Delta Z_t^{m_2} \right\|_2^2. \quad (22)$$

This supervision assigns morphology-dependent transition differences to the operator while retaining a shared local basis. Operator-swap analysis uses the same dynamic state and basis with operators from controlled morphology variants and compares the induced transition changes with simulator rollouts.

4.6 Reward, Value, and Variable-Action Heads

Robot dynamics are shared across control objectives, but reward semantics may be task dependent. We therefore allow a fixed-dimensional goal or task embedding e_k to enter reward, value, and policy heads, but never the transition basis or morphology operator.

The reward is decomposed into global and local components:

$$\hat{r}_t = r_\eta^g(g_t, e_k) + \sum_{i=1}^{N_m} r_\eta^v(z_{t,i}) + \sum_{e \in \mathcal{E}_m^{\text{act}}} r_\eta^a(z_{t,e}, a_{t,e}). \quad (23)$$

Here $z_{t,e}$ is an actuator-edge representation computed from its incident body tokens and static joint attributes. The global term captures progress, height, and uprightness; the local sums capture body contact and actuator effort. For diagnostic experiments, known MuJoCo reward functions can be evaluated directly from predicted physical quantities, isolating transition quality from reward-model error.

The value head uses the fixed-dimensional readout:

$$V_\xi(Z_t, k) = V_\xi(g_t, e_k). \quad (24)$$

Algorithm 1 Training GraphOp-WM

- 1: Sample morphology-parameter variants and trajectory segments from $\mathcal{D}$.
 - 2: Pack body nodes, joint edges, actuator edges, and graph-membership indices.
 - 3: Encode dynamic body tokens Z_t without static morphology attributes.
 - 4: Predict and normalize the shared local basis $\bar{B}_t$.
 - 5: Encode static morphology coefficients from $\mathcal{G}_m$ and apply state-dependent gates.
 - 6: Construct $\mathcal{A}_{m,t}$ using Eq. (18).
 - 7: Predict $\hat{Z}_{t+1}$ using Eq. (19) and roll forward for H_{train} steps.
 - 8: Compute dynamics, reward, TD, paired-morphology, and operator losses.
 - 9: Update model parameters and target value parameters.
-

For action values, actions are embedded on actuator edges, followed by an action-conditioned graph readout:

$$Q_\omega(Z_t, A_t, k) = Q_\omega(\text{Readout}_Q(Z_t, \mathcal{G}_m, A_t), e_k). \quad (25)$$

Both heads share weight dimensions across arbitrary body and actuator counts.

When a policy prior is used, each actuated edge independently predicts a Gaussian action conditioned on its incident body tokens and the global readout:

$$(\mu_{t,e}, \sigma_{t,e}) = \pi_\rho(z_{t,i}, z_{t,j}, p_{ij}^e, g_t, e_k), \quad e = (i, j) \in \mathcal{E}_m^{\text{act}}. \quad (26)$$

The number of output actions therefore equals the number of actuators in the current morphology.

4.7 Training Objective

The primary dynamics loss predicts token-aligned physical quantities or anchored latent targets:

$$\mathcal{L}_{\text{dyn}} = \sum_{h=1}^{H_{\text{train}}} \rho^{h-1} \frac{1}{N_m} \left\| \hat{Z}_{t+h} - Z_{t+h}^{\text{tar}} \right\|_F^2, \quad (27)$$

where the loss is evaluated only over nodes present in each packed graph. For proprioceptive experiments, an observation head anchors position, velocity, root motion, and contact channels.

The value target is a standard bootstrapped TD target rather than self-consistency between two unconstrained value predictions:

$$y_t = r_t + \gamma(1 - d_t) \text{sg } V_{\bar{\xi}}(Z_{t+1}, k), \quad (28)$$

$$\mathcal{L}_V = \|V_{\bar{\xi}}(Z_t, k) - y_t\|_2^2. \quad (29)$$

The operator is regularized around the identity and by the magnitude of its structured corrections:

$$\mathcal{L}_{\text{op}} = \lambda_d \|d_{m,t}\|_2^2 + \lambda_w \|w_{m,t}\|_2^2 + \lambda_s \|s_{m,t}\|_1. \quad (30)$$

The complete objective is

$$\mathcal{L} = \mathcal{L}_{\text{dyn}} + \lambda_r \mathcal{L}_r + \lambda_Q \mathcal{L}_Q + \lambda_V \mathcal{L}_V + \lambda_{\text{pair}} \mathcal{L}_{\text{pair}} + \lambda_{\text{op}} \mathcal{L}_{\text{op}}. \quad (31)$$

5 Model-Based Control with Variable Graphs

TD-MPC-style planning does not conceptually require a fixed latent or action dimension. At a decision step, all candidate trajectories correspond to the current morphology, whose node and actuator counts are known. For morphology m , MPC samples

$$A_{t:t+K-1} \in \mathbb{R}^{K \times N_{\text{sample}} \times |\mathcal{E}_m^{\text{act}}|}, \quad (32)$$

Table 1: MuJoCo morphology parameterization. Values are multiplicative factors relative to each canonical Hopper, Walker2d, or HalfCheetah model. Inertia is deterministically recomputed after geometry and mass changes.

Morphology quantity	Training envelope	Interpolation set	Extrapolation set	Held-out composition
Link length	[0.80, 1.20]	{0.85, 0.95, 1.05, 1.15}	[0.60, 0.75] $\cup$ [1.25, 1.40]	distal link 1.25 with actuator gear 0.70
Link mass	[0.75, 1.25]	{0.85, 0.95, 1.05, 1.15}	[0.50, 0.70] $\cup$ [1.30, 1.50]	distal mass 1.35 with torso mass 0.75
Torso mass	[0.80, 1.20]	{0.90, 1.00, 1.10}	[0.60, 0.75] $\cup$ [1.25, 1.40]	torso mass 1.30 with limb mass 0.70
Joint damping	[0.70, 1.30]	{0.80, 0.90, 1.10, 1.20}	[0.40, 0.60] $\cup$ [1.40, 1.60]	damping 1.40 with actuator gear 0.70
Actuator gear	[0.80, 1.20]	{0.90, 0.95, 1.05, 1.10}	[0.60, 0.75] $\cup$ [1.25, 1.40]	gear 0.70 with distal link length 1.25
Body inertia	recomputed	recomputed	recomputed	recomputed

replicates the current graph state across candidates, and rolls out GraphOp-WM. The planning objective is

$$A_{t:t+K-1}^* = \arg \max_A \sum_{k=0}^{K-1} \gamma^k \hat{r}_{t+k} + \gamma^K V_{\hat{\xi}}(\hat{\mathcal{Z}}_{t+K}, k). \quad (33)$$

Different morphologies instantiate different action tensors at separate planning calls; no global maximum action dimension is required.

To adapt TD-MPC2, we replace its vector encoder and MLP transition with packed graph modules, replace reward and value inputs with Eq. (10), and replace the fixed policy output with Eq. (26). The MPPI/CEM optimization logic and temporal-difference targets remain unchanged.

6 MuJoCo Benchmark Definition

6.1 Environment Families and Morphology Construction

The benchmark uses the MuJoCo physics engine (Todorov et al., 2012) and three locomotion families: Hopper, Walker2d, and HalfCheetah. Each canonical MJCF model is converted into the attributed graph in Section 4.1 and then transformed by deterministic MJCF edits. Parameter variants scale link geometry, body mass, torso mass, joint damping, and actuator gear relative to the canonical model. Body inertias are recomputed from the scaled geometry and mass rather than varied independently. Ground friction, gravity, integration time step, termination rules, and reward definitions remain fixed within each family.

6.2 Parameter Splits

All values in Table 1 are multipliers of the canonical MJCF quantity. Training morphologies are generated by maximin Latin-hypercube sampling over the training ranges. Interpolation morphologies use unseen values and combinations inside the training envelope. Extrapolation morphologies lie outside that envelope. The compositional split combines individually observed directions into jointly unseen morphologies.

7 Conclusion

We introduced GraphOp-WM, a structured world model for morphology-parameter generalization in continuous control. The method represents an articulated robot as a graph and separates a morphology-independent local dynamics basis from a structured operator conditioned on physical morphology parameters. Architectural information separation, basis normalization, and paired-morphology supervision assign reusable and morphology-dependent components distinct roles in the transition. The accompanying MuJoCo benchmark

defines interpolation, extrapolation, and held-out composition splits for studying out-of-distribution prediction and planning under changes in geometry, mass, damping, and actuation. GraphOp-WM is intended as a compact structural prior for parameter generalization within related articulated robot families rather than a universal cross-topology controller.

References

- Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL <https://proceedings.neurips.cc/paper/2016/hash/3147da8ab4a0437c15ef51a5cc7f2dc4-Abstract.html>.
- Nico Bohlinger, Grzegorz Czechmanowski, Maciej Piotr Krupka, Piotr Kicki, Krzysztof Walas, Jan Peters, and Davide Tateo. One policy to run them all: An end-to-end learning approach to multi-embodiment locomotion. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 3356–3378. PMLR, 2025. URL <https://proceedings.mlr.press/v270/bohlinger25a.html>.
- Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL <https://proceedings.neurips.cc/paper/2018/hash/3de568f8597b94bda53149c7d7f5958c-Abstract.html>.
- Mohamad H. Danesh, Chenhao Li, Amin Abyaneh, Anas Houssaini, Kirsty Ellis, Glen Berseth, Marco Hutter, and Hsiu-Chin Lin. Toward hardware-agnostic quadrupedal world models via morphology conditioning, 2026. URL <https://arxiv.org/abs/2604.08780>.
- Ignat Georgiev, Varun Giridhar, Nicklas Hansen, and Animesh Garg. PWM: Policy learning with multi-task world models. In *International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=hOELrZfg0J>.
- Agrim Gupta, Linxi Fan, Surya Ganguli, and Li Fei-Fei. MetaMorph: Learning universal controllers with transformers. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=0pmqtk_GvYL.
- Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2. URL <https://doi.org/10.1038/s41586-025-08744-2>.
- Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=FzpfPa6unv>.
- Nicklas A. Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 8387–8406. PMLR, 2022. URL <https://proceedings.mlr.press/v162/hansen22a.html>.
- Zihao He, Bo Ai, Tongzhou Mu, Yulin Liu, Weikang Wan, Jiawei Fu, Yilun Du, Henrik I. Christensen, and Hao Su. Scaling cross-embodiment world models for dexterous manipulation, 2025. URL <https://arxiv.org/abs/2511.01177>.
- Wenlong Huang, Igor Mordatch, and Deepak Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4455–4464. PMLR, 2020. URL <https://proceedings.mlr.press/v119/huang20d.html>.

- Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL <https://proceedings.neurips.cc/paper/2019/hash/5faf461eff3099671ad63c6f3f094f7f-Abstract.html>.
- Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=H1gax6VtDB>.
- Kimin Lee, Younggyo Seo, Seunghyun Lee, Honglak Lee, and Jinwoo Shin. Context-aware dynamics model for generalization in model-based reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5757–5766. PMLR, 2020. URL <https://proceedings.mlr.press/v119/lee20g.html>.
- Yunzhu Li, Hao He, Jiajun Wu, Dina Katabi, and Antonio Torralba. Learning compositional koopman operators for model-based control. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=H1ldzA4tPr>.
- Zohar Rimón, Tom Jurgenson, Orr Krupnik, Gilad Adler, and Aviv Tamar. MAMBA: An effective world model approach for meta-reinforcement learning. In *International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=1REOH6mU7M>.
- Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin Riedmiller, Raia Hadsell, and Peter Battaglia. Graph networks as learnable physics engines for inference and control. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4470–4479. PMLR, 2018. URL <https://proceedings.mlr.press/v80/sanchez-gonzalez18a.html>.
- Carmelo Sferrazza, Dun-Ming Huang, Fangchen Liu, Jongmin Lee, and Pieter Abbeel. Body transformer: Leveraging robot embodiment for policy learning. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 3407–3424. PMLR, 2025. URL <https://proceedings.mlr.press/v270/sferrazza25a.html>.
- Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109. URL <https://doi.org/10.1109/IROS.2012.6386109>.
- Tingwu Wang, Renjie Liao, Jimmy Ba, and Sanja Fidler. NerveNet: Learning structured policy with graph neural networks. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=S1sqHMZCb>.
- Yuchen Wang, Jiangtao Kong, Sizhe Wei, Xiaochang Li, Haohong Lin, Hongjue Zhao, Tianyi Zhou, Lu Gan, and Huajie Shao. WestWorld: A knowledge-encoded scalable trajectory world model for diverse robotic systems, 2026. URL <https://arxiv.org/abs/2603.14392>.
- Fengze Xie, Sizhe Wei, Yue Song, Yisong Yue, and Lu Gan. Morphological-symmetry-equivariant heterogeneous graph neural network for robotic dynamics learning. In *Proceedings of the 7th Annual Learning for Dynamics & Control Conference*, volume 283 of *Proceedings of Machine Learning Research*, pages 1392–1405. PMLR, 2025. URL <https://proceedings.mlr.press/v283/xie25a.html>.