LEARNING TARGET-CENTERED LATENT ACTIONS THROUGH PREDICTIVE COMPARISONS

Anonymous authors
Paper under double-blind review

ABSTRACT

When several agents act together, observations reveal only their shared transition, making it difficult to learn a latent action for any one agent. A representation trained to explain the full transition may mix the target agent with its partners, while simply separating agents can discard how the target's effect depends on them. We introduce a target-centered approach based on *controlled predictive comparisons*. A learned dynamics model makes two predictions from the same history and the same partner representations, changing only the target representation. Their difference isolates what changes with the target while holding the surrounding context fixed. Repeating this comparison across partner settings reveals how the target's predicted effect varies with individual partners and their combinations. We compress these structured differences into a latent action, train a history-only predictor to recover it before acting, and use a small action-labelled dataset only to ground it in the target action space. Our analysis characterizes what these comparisons remove, what partner-dependent information they retain, and when the resulting representation can be recovered from history. Experiments on particle coordination, multi-agent locomotion, and visual observations show that the learned representations retain target-action and partner-dependent information and improve latent-action control under limited action supervision.

1 INTRODUCTION

Imagine watching two people carry a table. The table's motion reflects both people's actions, but does not tell us how it would have changed if only one person had acted differently. Ignoring the partner does not solve the problem either, because the effect of a push, turn, or slowdown depends on whether the partner supports, resists, or compensates for it. This simple example captures the question we study: *how can we isolate one agent's action from a shared transition while preserving how its effect depends on the others?*

We consider observations of several agents acting together, where the resulting transitions are available but the underlying actions are not. For one designated target agent, our goal is to learn a latent action that separates its behavior from that of the others while retaining its dependence on the surrounding agents. A small action-labelled dataset is used only later to connect this representation to the target agent's action space.

Latent-action learning provides a natural starting point (Edwards et al., 2019). An inverse model infers a latent action from observations before and after an action; a history-based predictor learns to recover it from information available before acting; and a small labelled dataset grounds it in the action space (Schmidt & Jiang, 2024; Ye et al., 2025). In a multi-agent system, however, the observed transition is produced jointly. A latent action trained only to explain that transition may therefore mix the target agent with its partners or with motion shared across the system (Wang et al., 2026; Nam et al., 2026). Recent factored and object-centric approaches reduce this ambiguity by separating entities or local changes within the transition (Klepach et al., 2026; Wang et al., 2026; Nam et al., 2026). FLAM also permits changing one entity's latent action while retaining the others. Our focus is to represent the resulting prediction differences across partner settings: *how does the target's predicted effect vary with its partners?*

Our key idea is to **compare predictions before forming the final latent action**. From an observed transition, we first infer intermediate representations for the target and its partners. A learned dynam-

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.

ics model then makes two predictions from the same history and the same partner representations, changing only the target representation. One uses the target representation inferred from the transition; the other replaces it with a reference from another transition. Their difference captures what changes in the prediction when only the target changes. Components shared by the two predictions cancel, while target-dependent changes remain.

A single comparison captures the target under one partner setting. We therefore repeat the same target replacement across different partner settings. The resulting differences reveal how the target's predicted effect changes with individual partners and their combinations. The representation therefore records both the target's predicted effect and how it varies with context. We then compress these structured differences into the final latent action. Figure 1 summarizes the construction.

Because these comparisons use completed transitions, they serve as training targets rather than inputs available when an action must be chosen. We therefore train a history-only predictor to recover the latent action from observations available before acting. In the Frozen protocol, action labels enter only through a small final mapping to the target agent's action space after observation-based training is frozen. Thus, the method separates three roles: isolating target-specific information, retaining its dependence on partners, and making it available before action selection. Observation-based pretraining uses no native-action labels, rewards, or simulator queries.

Our analysis characterizes what the predictive comparisons remove, what partner-dependent information they retain, and when the resulting representation can be recovered from history before acting.

We evaluate the approach on particle coordination, multi-agent Ant locomotion, and an RGB version of the shared-body locomotion task. Beyond downstream control with limited action labels, we test whether additional unlabelled observations improve performance, whether the learned representations retain target-action and partner-dependent information, and which parts of the comparison construction matter. Across the numerical tasks, several comparison-based variants improve over joint latent-action baselines, while the visual experiments show that the same construction extends to behavior inferred from pixels.

Our contributions are threefold:

- We introduce a target-centered latent-action construction based on controlled prediction differences, isolating changes associated with one agent while preserving their dependence on partner behavior.
- We extend these comparisons across partner settings, yielding a structured representation of how the target's predicted effect varies with individual partners and their combinations. We analyze what information the comparisons remove and retain, and the separate requirement that this information be predictable from history.
- We evaluate the approach across numerical and visual multi-agent settings under limited action supervision, with observation-scaling studies, target-action and partner-information probes, and structural controls that examine the source of the gains.

2 Background and Problem Setting

Let o_t denote the observation before action a_t, and let $\mathcal{D}_U = (o_0, \ldots, o_T)$ be observation-only data. In standard latent-action learning, an inverse encoder infers a representation from a completed transition,

$$z_t = q_\phi(o_{\le t}, o_{t+1:t+h}).$$

Because this representation uses future observations, a history encoder predicts it from observations available before acting, $\hat{z}_t = \pi_\psi(o_{\le t})$. A small action-labelled set is then used to train a decoder, $\hat{a}_t = d_\omega(\hat{z}_t)$.

The inverse encoder is trained by predicting or reconstructing observations, without action labels. Its outputs serve as fixed, stop-gradient targets for the history encoder. In Frozen, representation and history models remain fixed when the action decoder is fitted.

We consider m interacting agents with joint action

$$a_t = (a_{0,t}, \ldots, a_{m-1,t}),$$

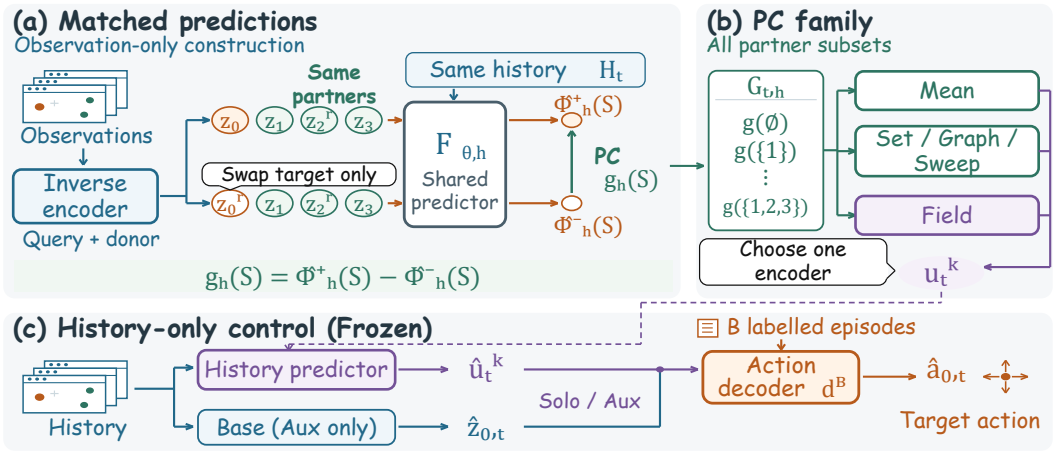


Figure 1: **Overview of controlled predictive comparisons (PC).** (a) Only the target code changes between predictions with shared history, partner codes and predictor ($j = 0$, $S = \{2\}$). Their difference is the predictive contrast. Donors come from nearby training histories; construction uses no action labels or simulator queries. (b) Alternative encoders summarize contrasts (Mean), subset structure (Set/Graph/Sweep), or endpoints and contrasts across horizons and entities (Field). (c) The dashed link supplies a training target, never a deployment input. Solo uses the history-predicted representation; Aux adds Base's prediction of the original target-slot code. Frozen fixes both history predictors before decoder training on B labelled episodes.

and designate agent j as the target. The observation contains known agent-indexed groups, from which an encoder infers representations $z_{i,t}$. These indices identify the observed groups, but not the action responsible for each change. Since the transition is produced jointly, $z_{j,t}$ may still reflect partner behavior or motion shared across the system.

Our goal is to learn a target latent action u_t that is centered on agent j, retains its dependence on the partners, and can be predicted before acting. Its history prediction $\hat{u}_t = \pi_u(o_{\leq t})$ is mapped to the target action using limited action labels, $\hat{a}_{j,t} = d_B(\hat{u}_t)$. The central question is how to construct such a representation from shared observations alone.

3 Controlled Predictive Comparisons

A joint transition mixes the target's behavior with its partners' changes. We compare two predictions with the same history and partner codes, changing only the target code from its observed value to a reference. Their difference records the target-associated prediction change. We call this construction *controlled predictive comparisons* (PC). Repeating it across partner settings captures how the target-associated prediction change depends on the partners.

We encode the prediction comparisons, train a history-only predictor to recover that encoding, and ground it with limited action labels (Figure 1). Mean, Set, Graph, Sweep and Field use different encoders. Appendices A and H give proofs; Section 4 gives protocols.

3.1 From observations to transition codes

Let $x_t = \psi(o_{\leq t})$ contain positions, numerical features, or a short red–green–blue (RGB) image history available through t. An *inverse encoder* infers transition codes from observations before and after a change. A code describes observed changes, not necessarily native action coordinates.

Slots identify which code to replace. A *slot* is a known observation group: particle positions, per-limb joint features, or a camera history. There are m slots, indexed by $i = 0, \ldots, m-1$, with codes $z_{i,t}$ and target j. Slots require no discovery. Numerical slots follow agents; visual slots follow cameras whose overlapping views may mix agents. Visual replacement concerns view codes.

Observation training precedes action labels. Pretraining uses $\mathcal{D}_U = \{o^e_{0:T_e}\}_{e=1}^N$, with T_e transitions per episode. It excludes actions, rewards, hidden simulator snapshots and simulator queries. Only action dimension and range are known beforehand (Appendix B.1). Later, labels from B episodes form $\mathcal{D}_B = \{(e, t, a^e_{j,t})\}$ for *grounding*: learning to map predicted latents to actions. Evaluation sums episode rewards.

3.2 Compare predictions with the same partner context

Learn the predictor. The encoder adapts LAOM's temporal prediction objective (Nikulin et al., 2025); Appendix B.1 gives offsets and target-encoder updates. Base predicts the target-slot code without predictive comparisons (Appendix G.5). A separate predictor $F_{\theta,h}$ learns the feature change $\Delta x_{t,h} = x_{t+h} - x_t$ from joint code z_t and *history* $H_t = (x_{\max(t-1,0)}, x_t)$, the previous and current features. The *horizon* h counts steps ahead: $h = 2$ predicts from t to $t + 2$. We minimize:

$$\mathcal{L}_{F,h} = \mathbb{E}_{\mathcal{D}_U} \left\| F_{\theta,h}(H_t, z_t) - \Delta x_{t,h} \right\|_2^2, \qquad z_t = (z_{0,t}, \ldots, z_{m-1,t}). \tag{1}$$

At $t = 0$ we repeat the initial observation. Standardization uses training data only; $\mathbb{E}_{\mathcal{D}_U}$ averages transitions and $\|\cdot\|_2$ is Euclidean. MPE uses $t+2$, when actions first change position (Appendix B.1).

Change only the target code. The transition being analyzed is the *query*. A *donor* is a reference transition from a fixed training-observation bank that supplies replacement codes. We retrieve it by nearest standardized two-step history, excluding the query's episode for training queries, without actions or rewards. Its code is z^r_t; r identifies the donor, not its timestamp. A *partner mask* $S \subseteq \mathcal{O}_j = \{0, \ldots, m-1\} \setminus \{j\}$ identifies the partners that use donor codes; the others keep their query codes. Both branches use this same partner setting. The $+$ branch retains the query's target code, while the $-$ branch replaces it:

$$z^{S,+}_{i,t} = \begin{cases} z^r_{i,t}, & i \in S, \\ z_{i,t}, & i \notin S, \end{cases} \qquad z^{S,-}_{i,t} = \begin{cases} z^r_{j,t}, & i = j, \\ z^{S,+}_{i,t}, & i \neq j. \end{cases} \tag{2}$$

With target zero and $S = \{2\}$, both branches use donor slot two and query slots one and three; only slot zero differs. The donor is an observed code, not a no-op or action label. Predicted *endpoints* $\widehat{\Phi}_{t,h}(S; \pm) \in \mathbb{R}^{d_h}$ use $(H_t, z_t^{S,\pm})$ directly in MPE/vision. Recursive MaMuJoCo branches also share the later inferred-code sequence, never native actions.

Definition 1 (Predictive contrast)**.** *For a query from a given trajectory at time t, both endpoints use the same donor, partner setting S, horizon h, and trained predictor. Their difference is the predictive contrast:*

$$g_{t,h}(S) = \widehat{\Phi}_{t,h}(S; +) - \widehat{\Phi}_{t,h}(S; -) \in \mathbb{R}^{d_h}. \tag{3}$$

Both predictions share history, partner-code assignments, and output coordinates. The contrast subtracts the donor-target prediction from the query-target prediction.

Matching gives a model-level contrast without native commands. Physical interpretation also requires code meanings to survive transport and the predictor to handle recombined inputs; retrieval alone guarantees neither. Prediction contrasts, action readout and control are evaluated separately. Appendix G.3 tests physical effects with evaluation-only simulator access.

Why match the partner context? With history, horizon and continuation fixed, write the endpoint map as $f(z, \mathbf{p})$ for target $z = z_{j,t}$ and partners $\mathbf{p}$. Define $b(\mathbf{p}) = f(z_0, \mathbf{p})$ for fixed z_0 and $\tau(z, \mathbf{p}) = f(z, \mathbf{p}) - b(\mathbf{p})$. Matching cancels b: $g = \tau(z, \mathbf{p}) - \tau(z^r, \mathbf{p})$. Target–partner dependence remains. Proposition 1 in Appendix A gives the unmatched partner-change term; Equation 10 shows how endpoint errors cancel or amplify.

3.3 Retain how the target effect varies with partners

Collect and separate partner dependence. To capture partner dependence, we vary the partner mask while keeping the query, donor, and horizon fixed. The resulting contrasts form the rows:

$$G_{t,h} = \left[g_{t,h}(S)^\top \right]_{S \subseteq \mathcal{O}_j} \in \mathbb{R}^{K \times d_h}, \qquad p = |\mathcal{O}_j|, \quad K = 2^p. \tag{4}$$

Here p counts partners and $\top$ transposes a contrast; three partners give eight rows. For a known predictor affine in the target code, full-column-rank stacked responses recover the shared code difference. Error scales inversely with their smallest singular value (Proposition 2, Appendix A). Thus informative partner settings help; redundant ones need not. This guarantee precedes compression and history prediction.

The Möbius transform separates single and joint partner effects:

$$c_{t,h}(A) = \sum_{S \subseteq A} (-1)^{|A|-|S|} g_{t,h}(S), \qquad g_{t,h}(S) = \sum_{A \subseteq S} c_{t,h}(A). \tag{5}$$

For example, $c(\{2\}) = g(\{2\}) - g(\varnothing)$ measures partner two's effect on the contrast; higher orders capture joint dependence. The inverse recovers every contrast. We use ordinary Möbius coordinates, not the Shapley interaction attribution studied by Faith-Shap (Tsai et al., 2023).

Encode the comparison structure. We omit the common PC- prefix. **Mean** averages encoded contrasts and reconstructs the mean and deviations, exporting only the common code. **Set** encodes coefficients and subset identities with a shared network, pooling empty-subset and average tokens to reconstruct coefficients and contrasts (Appendix B). **Graph** passes messages between subsets differing by one partner and links subsets to their partners. **Sweep** refines Mean's code with downward and upward subset messages (Appendix B).

Field adds endpoint and temporal context. Field keeps the query endpoint alongside its contrast. Let $\mathcal{V}$ index output-coordinate groups (positions or visual features), with P_v selecting or projecting group v. A *cell* indexed by time, horizon, subset, and group is

$$V_{t,h,S,v} = [P_v \widehat{\Phi}_{t,h}(S;+),\ P_v \widehat{\Phi}_{t,h}(S;+) - P_v \widehat{\Phi}_{t,h}(S;-)]. \tag{6}$$

Field standardizes cells and reconstructs them through subset transforms and messages across horizons, subsets and groups. It pools last-horizon, empty-subset tokens over groups (Appendix B.1, Equations 19–20).

3.4 Predict before acting, then ground the latent

Encoder E_k maps member k's table or cells T_t^k to latent $u_t^k = E_k(T_t^k)$. Since this uses completed transitions, a history-only predictor π_k predicts the latent from current and past observations H_t^{dep} before acting. *Solo* decodes this prediction alone. *Aux* also uses Base's predicted target-slot code, $\widehat{z}_{j,t} = \pi_{\mathrm{Base}}(H_t^{\mathrm{dep}})$:

$$\widehat{u}_t^k = \pi_k(H_t^{\mathrm{dep}}), \qquad \widehat{a}_{j,t}^{k,\mathrm{Aux}} = d_B^k\big([\widehat{z}_{j,t}, \widehat{u}_t^k]\big)\,. \tag{7}$$

Both branches concern the same target and label budget. Let U_t collect training-standardized targets and Π their history predictors, concatenated for Aux. Decoder-fitting examples $\mathcal{D}_B^{\mathrm{fit}}$ exclude validation labels:

$$\mathcal{L}_H = \mathbb{E}_{\mathcal{D}_U} \|\Pi(H_t^{\mathrm{dep}}) - U_t\|^2, \qquad \mathcal{L}_B = \mathbb{E}_{\mathcal{D}_B^{\mathrm{fit}}} \|d_B(\Pi(H_t^{\mathrm{dep}})) - a_{j,t}\|^2. \tag{8}$$

Solo fits its own decoder; Aux sums the two history losses. In Frozen, all upstream models stay fixed during grounding. Adapt also trains copies of the history predictors on action labels, while keeping the observation frontend and latent encoder frozen. Thus Solo/Aux determines the decoder's inputs, and Frozen/Adapt determines which models action labels train.

What an auxiliary branch can contribute. Solo tests standalone control; Aux tests complementarity. Duplication and independent auxiliaries control for adding features. Proposition 3 (Appendix A) links population benefit to the change in the conditional action mean after adding features. This applies beyond PC; controls test PC under finite labels and optimization.

From interaction retention to control. Let C collect the subset coefficients being reconstructed, U be their encoded latent, and $\widehat{U}$ the history predictor's prediction. A readout r maps a latent back to coefficients. Suppose changing its input by distance d changes its output by at most $L_r d$ (the Lipschitz condition). Write $\|X\|_{L^2} = (\mathbb{E}\|X\|_2^2)^{1/2}$ for root mean squared vector size. Then

$$\|C - r(\widehat{U})\|_{L^2} \le \|C - r(U)\|_{L^2} + L_r \|U - \widehat{U}\|_{L^2}. \tag{9}$$

Retention and history prediction are separate requirements; neither alone orders control returns. Appendix H proves this bound and identifies coefficients with finite differences that vary the target code together with the selected partner codes.

4 EXPERIMENTS

We evaluate three questions: whether PC improves closed-loop control with a fixed action budget; whether its gains persist against target-slot, extra-branch and simpler-encoding controls; and what target-action or partner-dependent information it retains. Frozen control is the primary test of observation-trained features. Adapt results examine the additional value of supervised history updates and are reported as a separate route. Solo and Aux are compared separately because standalone usefulness and complementarity with Base answer different questions.

4.1 EXPERIMENTAL SETUP

Environments. In particle coordination, four agents move among four landmarks in a PettingZoo multi-agent particle environment (MPE) (Lowe et al., 2017; Terry et al., 2021). The observation contains their 16 position coordinates while the learner controls one particle and the partners follow fixed dynamic-assignment policies. In multi-agent MuJoCo (MaMuJoCo), four agents each control two actuators of one Ant body (Peng et al., 2021). The learner receives 105 observation coordinates and controls partition zero; the other partitions follow a fixed teacher trained with Soft Actor-Critic (SAC). The RGB Ant task uses the same Ant and action partition, observed by four torso-mounted cameras at 64×64 RGB resolution. Camera motion, color changes, and backgrounds from the DAVIS video dataset follow the distracting-control design (Stone et al., 2021; Pont-Tuset et al., 2017). The inverse encoder can use all four views; the deployed policy uses three consecutive frames from the target view.

Data and action labels. MPE uses $N = 700$ unlabelled episodes and budgets $B \in \{8, 16, 32, 64, 128, 256\}$; MaMuJoCo uses $N = 90$ and $B \in \{4, 8, 16, 32, 64\}$. RGB Ant training uses 32 trajectories; $B = 8$ supplies 1,600 actions (25% of transitions). Appendix B.1 gives development splits and feature processing.

Comparisons. Local LAPO/LAOM adaptations provide observation-trained latent baselines; BC and IDM compare alternative uses of action labels. Base, duplicated Base and additional-branch controls test other sources of improvement. Each configuration uses the full budget grid. Comparisons match data and label budgets; upstream objectives and computation differ (Appendices B and G.5).

Evaluation. Our main metric is episode return, the sum of rewards within an episode. For each of five training seeds, we average over 100 common MPE or 50 MaMuJoCo episodes, then report the mean and sample SD across seeds. Paired comparisons use the same seeds. Budget-averaged return first averages the stated budget grid within each seed; log-budget AUC weights intervals on the log-budget axis. We report mean gains, positive seed counts and pointwise intervals for the stated comparison families, with multiple-test corrections where specified. MPE coverage/collisions and MaMuJoCo episode lengths complement return (Appendices B.1–C.1).

4.2 CONTROL WITH LIMITED ACTION LABELS

PC control gains extend beyond joint latent baselines. Averaging MPE budgets within seeds, Graph/Set/Sweep-Aux gain 0.117–0.143 over Base and 0.036–0.062 over duplicated Base (Figures 2 and 3). All three beat Base in every seed average; Graph also beats duplicated Base in all seeds. At every budget, all three exceed the joint LAPO/LAOM mean returns. MaMuJoCo Field-Aux improves budget-averaged return over Base, Random-edge-Aux and DeepSets-Aux. It beats Random-edge-Aux and joint baselines at every budget, and BC/IDM at four of five; duplicated Base remains stronger overall. Appendix C retains all controls, including strong high-budget MPE BC/LAPO-state results.

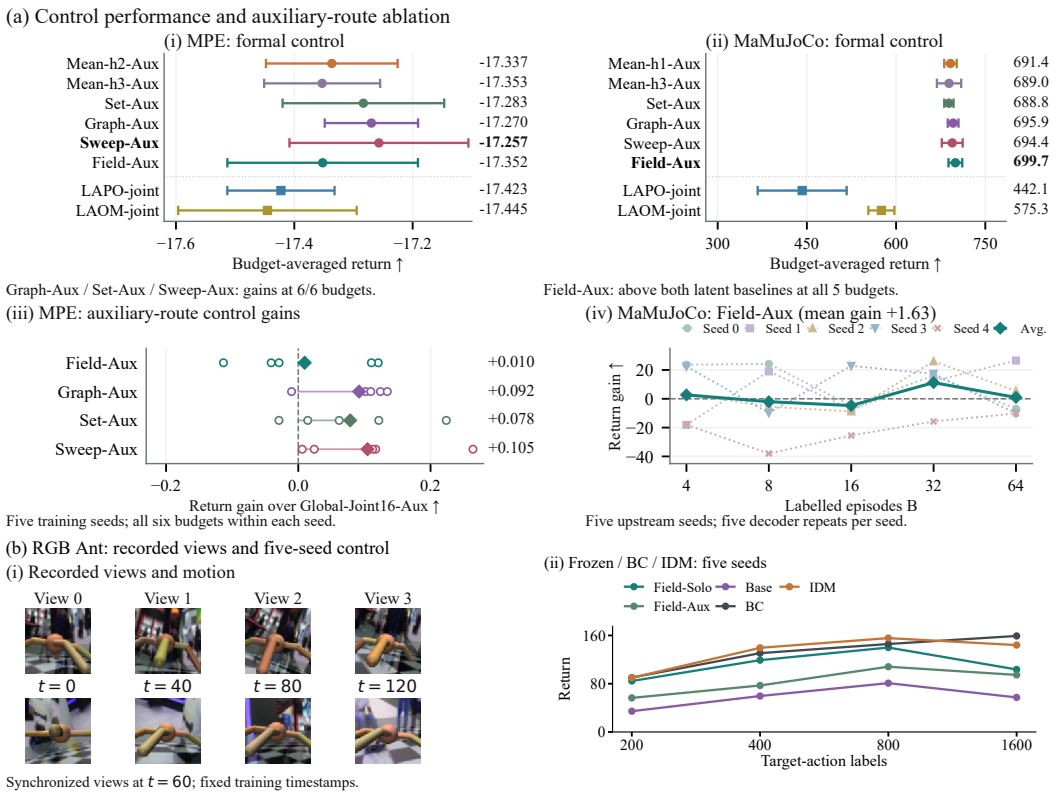


Figure 2: **Mean control gains across the PC family.** (a.i–ii) Budget-averaged return, mean/SD over five training seeds. Bold labels identify the highest displayed latent-method mean. (a.iii) MPE gains over Global-Joint16-Aux: circles show five seed averages over six budgets; diamonds show their mean. (a.iv) MaMuJoCo route gains: five upstream seeds, each with five nested decoder repeats. (b.i–ii) Recorded RGB views and five-seed visual mean-return curves. The panels show auxiliary-route gains in numerical tasks and standalone Frozen control with Field-Solo on RGB Ant. BC/IDM references remain in Table A3. Positive differences favor PC; Appendix G.4 gives route protocols.

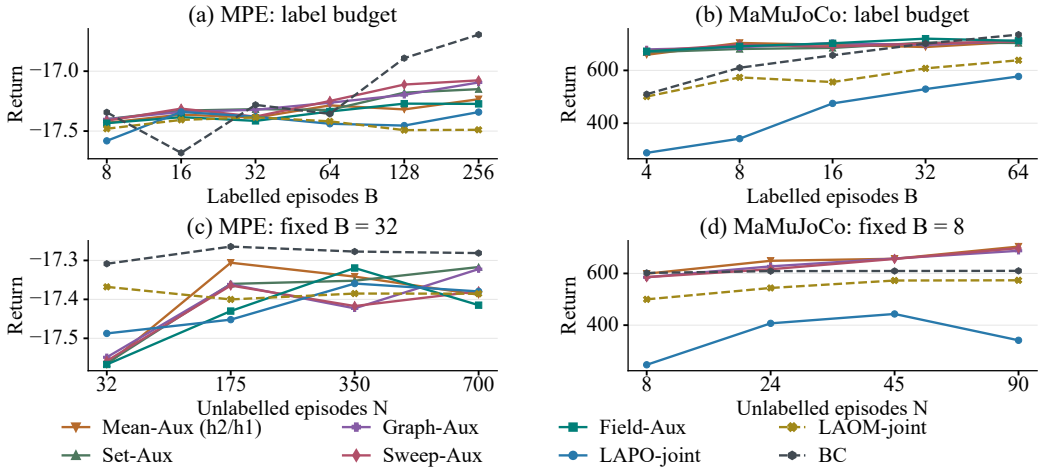
4.3 Additional observations and larger populations

More unlabelled observations improve several PC encoders. Increasing MPE data from $N = 32$ to 700 raises Set/Graph/Sweep/Mean-h2-Aux returns by 0.179–0.247 at B32. In MaMuJoCo, increasing $N = 8$ to 90 raises Mean-h1/Graph/Sweep-Aux mean returns by over 100 at B8; each improves in all five seeds. Intermediate responses and baseline curves appear in Figure 3(c–d); Appendix D gives the full MPE matrix.

Four- and eight-agent control. With N700/B32, five seeds per population and scripted partners, eight-agent Field-Aux Frozen and Graph-Aux Adapt exceed LAOM and BC in mean return; each beats LAOM in all five seeds. Adapted Base remains stronger than both adapted PC variants. Figure 4(c) and Appendix I.3 give adaptation contrasts and uncertainty.

4.4 Visual multi-agent control

Field improves the Frozen RGB Ant latent-policy route. With 1,600 action labels, Field-Solo reaches 103.60 ± 29.45, the highest tested latent-method mean, exceeding Base and the random-edge, DeepSets and continuous controls. All five seeds favor it over DeepSets. Field-Aux also improves over duplicated Base and LAPO/LAOM-Aux (Figure 2(b)). BC and IDM remain stronger at this budget. BC trains history features with action labels; Frozen latent policies train only the final decoder. Table A3 and Appendix F.1 give the protocols. The role of supervision timing is also studied by LAOM (Nikulin et al., 2025).



Means over five training seeds; full seed variability is reported in the tables.

Figure 3: **Control with more labels and observations.** Lines show five-seed mean return; tables report seed variability. (a–b) All label budgets. (c–d) Several PC encoders improve with more unlabelled episodes at fixed B32/B8. Each line is a fixed configuration: Mean-Aux uses h2/h1 in MPE/MaMuJoCo. BC refits observation normalization at each data size.

Standalone Frozen control on a second visual task. CoupledHalfCheetah tests whether the PC latent can support visual control on its own: Field-Solo omits Base and trains only the decoder on action labels. At B8, it leads all 16 Frozen representation configurations under both teachers, reaching 35.20 ± 8.75 and 36.01 ± 5.48 mean return. The gains over BC are $17.68/20.84$; all five seeds per teacher favor Field-Solo over IDM and Base. Each teacher uses the same eight labelled trajectories across methods: 1,600 target-action vectors and no partner labels. These results show that the observation-trained latent supports target control from two camera views even when both agents act through the same coupled physical system. Against Continuous-Solo, all five T200 seeds favor Field-Solo and the paired interval is positive; T201's interval includes zero. Teachers are analyzed separately, not as a held-out-partner test. Figure A3 and Appendix I.5 give all method means, paired gains and uncertainty.

History adaptation improves control across the label grid. On RGB Ant, both routes use 600 batch-256 offline grounding updates and the final checkpoint, without online interaction or return-based selection (Appendix I). At all four budgets (200–1,600 actions), Field-Solo Adapt exceeds BC and adapted LAPO/LAOM means (Figure 4(a)). At B8 it reaches 169.59 ± 36.21, gaining 65.99 over Frozen with the same representation and labels. Budget-averaged gains hold in all five seeds against BC/LAPO and four against LAOM (Figure 4(b)). IDM is lower on average but leads at B2. Paired intervals and planned-test corrections appear in Appendix I. Matched controls attribute gains to history updates and, for Field-Solo, observation-trained initialization. Updating both Aux branches gives the highest B2 mean; Appendix I gives Field-Aux's different initialization response.

4.5 Ablation and Mechanism Analysis

Complementarity beyond a whole-system code. This comparison tests whether PC contributes beyond adding a general whole-system representation. Global-Joint16-Aux replaces the PC auxiliary with a 16-D whole-system code, retaining Base, history dimensions, labels and grounding. Graph/Set/Sweep-Aux gain 0.078–0.105 in budget-averaged MPE return, positive in four/four/five seeds; Field's gain is smaller. MaMuJoCo Field-Aux gains 1.63, positive in four of five seeds (Figure 2(a.iii–iv)). Several PC auxiliaries therefore improve control beyond this whole-system alternative.

Target-action readout and partner-dependent information. Frozen probes show that several PC features improve target-action readout when combined with Base. At MPE B256, Graph/Sweep-

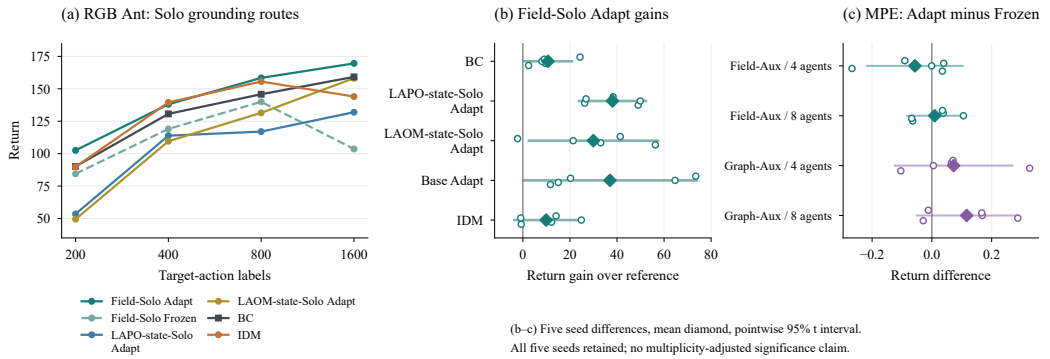


Figure 4: **Field benefits from history adaptation, and PC gains extend to eight-agent control.** (a) RGB Ant mean-return curves; seed SDs are in Appendix I. (b) Field-Solo Adapt minus each visual reference, averaging the four budgets within each seed. (c) MPE Adapt minus Frozen at B32, separately for four and eight agents. In (b–c), open circles are five paired seed differences, diamonds their means, and bars pointwise 95% t intervals. Visual and MPE use separate five-seed cohorts. Complete comparisons and planned-test corrections appear in Appendices I and I.3.

Aux reach $21.02/21.42\%$ history action skill versus Base's 16.24%; MaMuJoCo Field raises Base from 68.62% to 73.51%. Set/Graph/Sweep also retain readable single-partner coefficients in MPE, as do Set/Graph in MaMuJoCo. Together, these probes show action-relevant and partner-dependent information in the learned representations; closed-loop comparisons assess its control value. Within-encoder controls separately show positive mean gains for Graph/Field over raw MPE coordinates and for Field over pooled-edge/complement inputs in MaMuJoCo. Coordinate and matching effects vary across seeds or budgets; corrected tests do not establish a uniform structural advantage. Appendix G separates probe readout, history transfer and donor transport, with protocols and uncertainty in Appendices E, G.1, and H.

State-aligned donor transport. An additional offline MPE diagnostic aligns donors using observed intermediate-frame motion. Target-coordinate $h = 2$ MSE is 0.001128, 46.7% below zero prediction and 85.8% below lagged-motion transport, with both gains in all five models on 32 fresh shared trajectories. Fitting uses only observations; actions and simulator branches provide evaluation truth. All other experiments retain their original, uncorrected donors. Appendix G.3 gives construction, controls and uncertainty.

5 Conclusion

We introduced controlled predictive comparisons (PC) for learning target-centered latent actions from shared observations. By changing the target code while matching history and partner context, PC captures target-associated prediction changes and how they depend on partners. Mean, Set, Graph, Sweep and Field offer different ways to encode this comparison structure. A history predictor makes the resulting latent available before acting, and limited target-action labels ground it in the action space.

Across particle coordination, multi-agent locomotion and visual control, the PC family shows benefits in both action regression and closed-loop control. Action-regression probes show that several members improve target-action readout over Base, while coefficient probes reveal retained partner-dependent information. Under limited action supervision, different members improve returns over observation-trained latent baselines and target-slot or extra-input controls across multiple tasks and budgets. These gains appear both when PC complements Base and when its latent supports standalone Frozen control. The strongest encoder depends on the task and budget, supporting PC as a family of useful representations rather than a single universally preferred architecture. Together, the results show the value of preserving target–partner dependence when learning actions from observations. Extending this approach to changing partners and broader visual conditions is a natural next step.

AI USE STATEMENT

We used generative AI tools to draft and polish text, retrieve and discover related work, and assist research ideation and execution. This assistance included discussion of experimental methodology, implementation and debugging of environment-specific experimental code, preparation of analysis and audit scripts, interpretation of results, and creation of scientific figures. Experimental results reported in this paper come from executed runs and saved numerical artifacts. Verification procedures include frozen-source and checkpoint hash checks, label-access and budget audits, trajectory replay where applicable, and independent recomputation of reported statistics. These automated checks support verification of the experimental artifacts; final review and responsibility for the manuscript, citations, mathematical claims, code, and reported results remain with the human authors.

ETHICS STATEMENT

These experiments concern simulated particle coordination and locomotion. DAVIS images are used as visual distractors, subject to the dataset's original terms. No patient records or medical outcomes enter the present comparison. Simulator results do not establish safety for physical deployment.

REPRODUCIBILITY STATEMENT

Appendix B specifies observation interfaces, label access, learning stages, native-action grounding and evaluation units.

REFERENCES

Elie Aljalbout, Maximilian Karl, and Patrick van der Smagt. Clas: Coordinating multi-robot manipulation with central latent action spaces. In *Learning for Dynamics and Control Conference*, pp. 1152–1166. PMLR, 2023.

Arthur Allshire, Roberto Martín-Martín, Charles Lin, Shawn Manuel, Silvio Savarese, and Animesh Garg. Laser: Learning a latent action space for efficient reinforcement learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6650–6656. IEEE, 2021.

Ashley Edwards, Himanshu Sahni, Yannick Schroecker, and Charles Isbell. Imitating latent policies from observation. In *International conference on machine learning*, pp. 1755–1763. PMLR, 2019.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Siddharth Karamcheti, Albert J Zhai, Dylan P Losey, and Dorsa Sadigh. Learning visually guided latent actions for assistive teleoperation. In *Learning for dynamics and control*, pp. 1230–1241. PMLR, 2021.

Albina Klepach, Alexander Nikulin, Ilya Zisman, Denis Tarasov, Alexander Derevyagin, Andrei Polubarov, Nikita Lyubaykin, Igor Kiselev, and Vladislav Kurenkov. Object-centric latent action learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 22626–22634, 2026.

Sébastien Lachapelle. On the identifiability of latent action policies. *arXiv preprint arXiv:2510.01337*, 2025.

Jung Min Lee, Taehyun Cho, Li Zhao, and Jungwoo Lee. Why latent actions fail, and how to prevent it. *arXiv preprint arXiv:2605.20223*, 2026.

Alexander Levine, Peter Stone, and Amy Zhang. Offline action-free learning of ex-bmdps by comparing diverse datasets. *arXiv preprint arXiv:2503.21018*, 2025.

Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30, 2017.

Dipendra Kumar Misra, Akanksha Saran, Tengyang Xie, Alex Lamb, and John Langford. Towards principled representation learning from videos for reinforcement learning. In *International Conference on Learning Representations*, volume 2024, pp. 53448–53487, 2024.

Heejeong Nam, Chandradithya S Jonnalagadda, Harshit Aggarwal, Eric Xu, and Randall Balestriero. Latent actions from factorized transition effects under agent ambiguity. *arXiv preprint arXiv:2606.30544*, 2026.

Alexander Nikulin, Ilya Zisman, Denis Tarasov, Nikita Lyubaykin, Andrei Polubarov, Igor Kiselev, and Vladislav Kurenkov. Latent action learning requires supervision in the presence of distractors. *arXiv preprint arXiv:2502.00379*, 2025.

Bei Peng, Tabish Rashid, Christian Schroeder de Witt, Pierre-Alexandre Kamienny, Philip Torr, Wendelin Böhmer, and Shimon Whiteson. Facmac: Factored multi-agent centralised policy gradients. *Advances in neural information processing systems*, 34:12208–12221, 2021.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Dominik Schmidt and Minqi Jiang. Learning to act without actions. In *International Conference on Learning Representations*, volume 2024, pp. 9379–9395, 2024.

Austin Stone, Oscar Ramirez, Kurt Konolige, and Rico Jonschkowski. The distracting control suite–a challenging benchmark for reinforcement learning from pixels. *arXiv preprint arXiv:2101.02722*, 2021.

Zuojin Tang, Haoyun Liu, Xinyuan Chang, Changjie Wu, Dongjie Huo, Yandan Yang, Bin Liu, Zhejia Cai, Feng Xiong, Mu Xu, et al. Alam: Algebraically consistent latent action model for vision-language-action models. *arXiv preprint arXiv:2605.10819*, 2026.

J K Terry, Benjamin Black, Nathaniel Grammel, Mario Jayakumar, Ananth Hari, Ryan Sullivan, Luis S Santos, Clemens Dieffendahl, Caroline Horsch, Rodrigo Perez-Vicente, et al. Pettingzoo: Gym for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34:15032–15043, 2021.

Che-Ping Tsai, Chih-Kuan Yeh, and Pradeep Ravikumar. Faith-shap: The faithful shapley interaction index. *Journal of Machine Learning Research*, 24(94):1–42, 2023.

Zizhao Wang, Chang Shi, Jiaheng Hu, Kevin Rohling, Roberto Martín-Martín, Amy Zhang, and Peter Stone. Factored latent action world models. *arXiv preprint arXiv:2602.16229*, 2026.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Se June Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. In *International Conference on Learning Representations*, volume 2025, pp. 28213–28239, 2025.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

Chuheng Zhang, Tim Pearce, Pushi Zhang, Kaixin Wang, Xiaoyu Chen, Wei Shen, Li Zhao, and Jiang Bian. What do latent action models actually learn? *Advances in Neural Information Processing Systems*, 38:146676–146697, 2026a.

Shuhan Zhang, Zhi Wang, Rui Gao, and Shuang Li. Deephalo: A neural choice model with controllable context effects. *Advances in Neural Information Processing Systems*, 38:136384–136413, 2026b.

Ruijie Zheng, Xiyao Wang, Yanchao Sun, Shuang Ma, Jieyu Zhao, Huazhe Xu, Hal Daumé III, and Furong Huang. TACO: Temporal latent action-driven contrastive loss for visual reinforcement learning. *Advances in Neural Information Processing Systems*, 36:48203–48225, 2023.

Wenxuan Zhou, Sujay Bajracharya, and David Held. Plas: Latent action space for offline reinforcement learning. In *Conference on Robot Learning*, pp. 1719–1735. PMLR, 2021.

A Notation and Exact Scope of the Learned Construction

Symbol	Meaning
e, t, j; o_t, x_t	Episode, parallel decision index, target ($j = 0$); observation and its declared features.
H_t, H_t^{dep}; $z_{i,t}, z_{i,t}^r$	Two-step readout/deployment history; inferred slot code and training-donor code.
$S \subseteq \mathcal{O}_j$; $\widehat{\Phi}_{t,h}(S;\pm)$	Replaced partner slots; query/donor-target endpoints.
$g_{t,h}(S)$; $P_v, V_{t,h,S,v}$	Predictive contrast (ordered endpoint difference); outcome projection and Field cell.
$u_t, \widehat{u}_t, d_B$; N, B	Inferred latent, history prediction, native-action decoder; unlabelled/labelled episodes (B includes validation).

Matching and prediction error. Fix the history, predictor, output coordinates and anchor from the main text. Write $f(z, \mathbf{p}) = b(\mathbf{p}) + \tau(z, \mathbf{p})$.

Proposition 1 (Context cancellation and prediction error)**.** *The predictive contrast is* $g = \tau(z, \mathbf{p}) - \tau(z^r, \mathbf{p})$. *For* $\widetilde{g} = f(z, \mathbf{p}) - f(z^r, \mathbf{p}')$, $\widetilde{g} - g = f(z^r, \mathbf{p}) - f(z^r, \mathbf{p}')$. *Given reference endpoints* y_+, y_- *in the same coordinates, let* $g_* = y_+ - y_-$ *and* $\varepsilon_\pm = f(z^{(\pm)}, \mathbf{p}) - y_\pm$, *where* $z^{(+)} = z$, $z^{(-)} = z^r$. *Finite second moments give*

$$\mathbb{E}\|g - g_*\|^2 = \mathbb{E}\|\varepsilon_+\|^2 + \mathbb{E}\|\varepsilon_-\|^2 - 2\mathbb{E}\langle\varepsilon_+, \varepsilon_-\rangle. \tag{10}$$

Proof. Subtract the matched endpoints to cancel b. Subtract g from $\widetilde{g}$ to obtain the mismatch term. Finally, $g - g_* = \varepsilon_+ - \varepsilon_-$; expanding its squared norm proves Equation 10. Pointwise, $\|g - g_*\| \le \|\varepsilon_+\| + \|\varepsilon_-\|$. □

Changing the decomposition's anchor leaves g unchanged. No additive physical dynamics or disentangled motor actions are assumed. If $f(z^r, \cdot)$ is L_p-Lipschitz, mismatch is at most $L_p\|\mathbf{p} - \mathbf{p}'\|$, with no positive lower bound; mismatch need not worsen control. The cross-moment includes error means: equal second moments $\sigma^2 > 0$ give $2\sigma^2(1 - \rho)$ for $\rho = \mathbb{E}\langle\varepsilon_+, \varepsilon_-\rangle/\sigma^2 \in [-1, 1]$. Shared predictors do not guarantee positive ρ, and factual observations do not identify all recombined endpoints. The reference $y_\pm$ need not be observed physical interventions.

Interactions, coordinates and conditioning. For $f(z, p) = b(p) + z\,w(p)$, $g = (z - z^r)w(p)$ retains partner modulation. Subset coefficients satisfy $c(\varnothing) = g(\varnothing)$, $c(\{i\}) = g(\{i\}) - g(\varnothing)$ and $c(\{i, k\}) = g(\{i, k\}) - g(\{i\}) - g(\{k\}) + g(\varnothing)$. The inverse follows from $\sum_{A\subseteq S} c(A) = \sum_{U\subseteq S} g(U)(1 - 1)^{|S\setminus U|} = g(S)$. Contrast errors bounded by δ can yield coefficient errors up to $2^{|A|}\delta$; invertibility does not ensure conditioning. These are finite differences of predicted outcomes over code replacements. Likewise, $(a, b) \mapsto (a, a - b)$ and positive-scale standardization are invertible; edge-only inputs lose common offsets, and 16-D compression can lose further information. Nonlinear outcome projection uses $P_v(\Phi^+) - P_v(\Phi^-)$, not $P_v(\Phi^+ - \Phi^-)$. Cancellation applies to $P_v \circ f$; an L_v-Lipschitz projection bounds endpoint error by L_v times input error without guaranteeing retention.

Context coverage. Suppose $f_S(z) = b_S + A_S z$, with known $A_S \in \mathbb{R}^{d_h \times d_z}$. Shared query/donor codes give $\delta z = z - z^r$ and stacked contrasts $\mathbf{g} = \mathcal{A}\delta z$, where $\mathcal{A} = \mathrm{stack}_S\, A_S$.

Proposition 2 (Context coverage of a latent difference)**.** *The table determines* δz *uniquely iff* $\mathrm{rank}(\mathcal{A}) = d_z$. *Under full column rank, least-squares recovery from* $\mathbf{g} + e$ *satisfies*

$$\|\widehat{\delta z} - \delta z\|_2 \le \frac{\|e\|_2}{\sigma_{\min}(\mathcal{A})}. \tag{11}$$

Proof. Full rank gives $\delta z = \mathcal{A}^\dagger \mathbf{g}$ and error $\|\mathcal{A}^\dagger e\| \le \|e\|/\sigma_{\min}(\mathcal{A})$. Otherwise a nonzero null vector v makes δz and $\delta z + v$ indistinguishable. □

For example, $A_\varnothing = (1, 0)$ and $A_{\{i\}} = (0, 1)$ recover two coordinates jointly, whereas each contrast alone misses one. This illustrates code recovery under the affine/rank assumptions, not native-action identification or an estimated rank of the trained model. For differentiable nonlinear maps, $f_S(z) - f_S(z^r) = [\int_0^1 D_z f_S(z^r + \alpha\delta z)d\alpha]\delta z$; the integrated Jacobian depends on the unknown difference. Recovering z also requires z^r; extra contexts or horizons help only when informative.

Complementarity and grounding. On one fixed distribution with square-integrable target actions, set $Z = \pi_{\mathrm{Base}}(H^{\mathrm{dep}})$, $W = \pi_k(H^{\mathrm{dep}})$ and $R^*(X) = \inf_d \mathbb{E}\|a_j - d(X)\|^2$, over measurable decoders.

Proposition 3 (Complementarity beyond a compressed Base)**.**

$$R^*(Z) - R^*(Z, W) = \mathbb{E}\|\mathbb{E}[a_j \mid Z, W] - \mathbb{E}[a_j \mid Z]\|^2 \geq 0. \tag{12}$$

Proof. Let $m_0 = \mathbb{E}[a_j \mid Z]$, $m_1 = \mathbb{E}[a_j \mid Z, W]$. Expand $a_j - m_0 = (a_j - m_1) + (m_1 - m_0)$; conditioning on (Z, W) cancels the cross term. Conditional means attain the squared-error minima. □

Strict gain requires $m_1 \neq m_0$ with positive probability. Duplicated Base adds none; independently learned features may. Finite-label estimation can offset the population gain.

Proposition 4 (From latent prediction to action prediction)**.** *Use the same frozen decoder d_B in native action coordinates for U and $\widehat{U} = \Pi(H^{\mathrm{dep}})$. On a fixed distribution, suppose it is L-Lipschitz on both supports. For $R_H = \mathbb{E}\|\widehat{U} - U\|^2$ and $R_G = \mathbb{E}\|d_B(U) - a_j\|^2$,*

$$\sqrt{\mathbb{E}\|d_B(\Pi(H^{\mathrm{dep}})) - a_j\|^2} \leq L\sqrt{R_H} + \sqrt{R_G}. \tag{13}$$

Proof. Insert $d_B(U)$ and apply the L^2 triangle inequality and Lipschitz bound: $\|d_B(\widehat{U}) - a_j\|_{L^2} \leq L\|\widehat{U} - U\|_{L^2} + \|d_B(U) - a_j\|_{L^2}$. □

Grounding fits predicted histories; R_G is an analysis quantity. A large L weakens the bound. Squared-error history regression targets $\mathbb{E}[U \mid H^{\mathrm{dep}}]$ with irreducible error $\mathbb{E}\|U - \mathbb{E}[U \mid H^{\mathrm{dep}}]\|^2$, which future/donor dependence can make positive despite variance regularization. For deterministic history features $W = \Pi(H^{\mathrm{dep}})$, conditional-mean orthogonality also gives

$$\mathbb{E}\|a_j - \mathbb{E}[a_j \mid W]\|^2 = \mathbb{E}\|a_j - \mathbb{E}[a_j \mid H^{\mathrm{dep}}]\|^2 + \mathbb{E}\|\mathbb{E}[a_j \mid H^{\mathrm{dep}}] - \mathbb{E}[a_j \mid W]\|^2. \tag{14}$$

These population identities separate compression, history prediction and grounding. Finite-sample BC and latent decoders differ in capacity and optimization; changed-distribution rollout return additionally depends on dynamics and rewards.

Coordinates and donors. Both endpoints share predictor, normalization, horizon and outcome projection. The donor bank has 1,024 training transitions in MPE and at most 2,048 in MaMuJoCo; RGB Ant matching uses 1,024 of 6,400. Distance uses standardized (x_{t-1}, x_t), with own-episode exclusion and a joint donor for replaced slots. Actions, reward, outcome quality and evaluation trajectories do not enter lookup. Nearby history does not ensure support of every recombined tuple. A training-only visual PCA basis is shared by four views within seed; horizon targets have separate standardizations.

Horizon operators. MaMuJoCo one-step standardized increments return to observation units before recursive addition; its readout predicts 105 coordinates. Branches differ in target code at step one and share inferred-code continuation, holding terminated states fixed. MPE directly predicts eight agent-position increments at each of $h = 1, 2, 3$, with static landmarks retained in the input. Its inverse code uses $t + 2$, where the current action first affects observed positions. Visual learning also uses three direct predictors. The one-step MPE zero-effect test remains a negative control, not an action-discriminating inverse target.

B Model Family and Training Details

The four-agent numerical protocol learns target $j = 0$ with three fixed partners acting at each parallel step; each locomotion agent controls two actuators. Visual comparisons replace the declared target-view code, with other views as partner slots; cameras are not acting agents. Numerical slot and LAOM-joint encoders share the LAOM temporal prediction objective and adapter, producing 16-D slot codes and one 64-D joint code, respectively. MPE and visual models directly predict position and feature changes at $h = 1, 2, 3$. MaMuJoCo recursively composes a one-step, 105-D observation predictor. Field training in MaMuJoCo and visual tasks masks entries; two-step MPE coordinate controls use full visibility with a shared reconstruction objective. Numeric Aux combines 16-D Base and member outputs; visual Field-Aux combines 32-D Base with 16-D Field. Each numerical matrix has 29 configurations. MPE counts the two Base aliases once and adds Global-Joint16-Aux as a whole-system structural ablation of the PC auxiliary route. The public names are PC-Mean, PC-Set, PC-Graph, PC-Sweep and PC-Field; tables and figures use their suffixes. Environment adapters retain the shared methods. Mean encodes common contrast information; Set, Graph and Sweep process single-horizon subset coefficients; Field combines horizons and outcome groups (definitions below). DeepSets pools contrast/mask tokens together with the empty-mask token (Zaheer et al., 2017); Random-edge applies a fixed random projection to mean standardized contrasts. Both learn their history predictor and decoder. LAPO/LAOM-joint use the complete declared numeric observation; the Continuous/LAPO-state/LAOM-state adapters use factual transitions, or shared extracted features in visual experiments.

Branch composition and action supervision. *Solo/Aux* and *Frozen/Adapt* describe two independent choices. Solo grounds one method's history-only prediction; Aux concatenates it with the Base target-slot history prediction. Base is an internal control and Base-duplicate repeats exactly the same features. Global-Joint16-Aux adds a separately learned whole-system 16-D history code. In Frozen, only the action decoder receives budgeted action gradients. In Adapt, a copy of the history predictor and the decoder receive them; for Aux, both histories update unless a branch-specific control says otherwise. The observation front end, latent encoder, normalization and original saved checkpoints remain fixed. Adapt is offline grounding, not online interaction or action-labelled representation pretraining. A *frozen probe* instead fits a diagnostic reader without updating its input features; it is not a third controller route. BC directly trains history features on the same target labels; IDM learns inverse actions and then a history policy from pseudo-labels.

Unmarked numerical/RGB matrices use Frozen. Two-step MPE Set/Graph/Sweep and DeepSets/Random-edge use h3; RGB Ant single-horizon models use h1, numerical MaMuJoCo h3.

Encoder notation. For one horizon, write $g(S) = g_{t,h}(S)$, $c(A) = c_{t,h}(A)$, and sum over subsets of $\mathcal{O}_j$. With $p = |\mathcal{O}_j|$, $K = 2^p$, let $\eta_A = [\mathbf{1}_A, |A|/p]$ contain binary partner membership and normalized size. Brackets concatenate; ϕ, ρ are separately learned member networks and LN is layer normalization. Tokens are intermediate subset/node vectors; u is the exported history-policy target.

Mean: a common signal across partner settings. Mean encodes each contrast and averages the resulting codes:

$$u^{\mathrm{Mean}} = \frac{1}{K}\sum_S \phi_z(g(S)), \qquad v_S = \phi_c(g(S) - \bar{g}), \qquad \bar{g} = \frac{1}{K}\sum_S g(S). \tag{15}$$

Common/context decoders reconstruct $\bar{g}$ and deviations from u^{Mean}/v_S. Only u^{Mean} is exported. The h1/h2/h3 suffix selects the horizon, with MPE using h2/h3.

Set: retain the subset coefficients. Set combines the empty-subset and average coefficient tokens:

$$s_A = \phi_{\mathrm{S}}([c(A), \eta_A]), \qquad u^{\mathrm{Set}} = \mathrm{LN}\big(\rho_{\mathrm{S}}\big([s_\varnothing, K^{-1}\textstyle\sum_A s_A]\big)\big). \tag{16}$$

A decoder uses u^{Set} and η_A to reconstruct coefficients and, by inverse transform, contrasts.

Graph: subset message passing. Graph initializes s_A^0 from $[c(A), \eta_A]$ and learns partner tokens e_i^0. Immediate smaller/larger subsets form $\mathcal{N}_-(A), \mathcal{N}_+(A)$. At layer ℓ, learned updates $\mathcal{U}_e^\ell, \mathcal{U}_s^\ell$ use neighbour averages (Avg, zero if empty):

$$
\begin{aligned}
e_i^{\ell+1} &= \mathcal{U}_e^\ell\big(e_i^\ell, \mathrm{Avg}_{A:i\in A}\, s_A^\ell\big)\,, \\
s_A^{\ell+1} &= \mathcal{U}_s^\ell\Big(s_A^\ell, \mathrm{Avg}_{B\in\mathcal{N}_-(A)}\, s_B^\ell, \mathrm{Avg}_{B\in\mathcal{N}_+(A)}\, s_B^\ell, \mathrm{Avg}_{i\in A}\, e_i^{\ell+1}\Big)\,.
\end{aligned} \tag{17}
$$

Updates use learned gates, residuals and LN. Graph uses the same empty/mean readout as Set, with separate weights, and reconstructs coefficients and contrasts.

Sweep: refine the common code through ordered subset sweeps. Sweep sweeps from larger to smaller subsets and back on the Hasse DAG (not a strict tree). Let $s_A^{\leftrightarrow}$ be the resulting token, $\omega_A = 1 + |A|$, and $\sigma(\alpha) = 1/(1 + e^{-\alpha})$ a learned scalar gate. Its empty/mean/weighted readout corrects Mean:

$$
\begin{aligned}
r_{\mathrm{T}} &= \rho_{\mathrm{T}}\Big([s_\varnothing^{\leftrightarrow}, K^{-1}\textstyle\sum_A s_A^{\leftrightarrow}, \frac{\sum_A \omega_A s_A^{\leftrightarrow}}{\sum_A \omega_A}]\Big)\,, \\
u^{\mathrm{Sweep}} &= \mathrm{LN}\left(u^{\mathrm{Mean}} + \sigma(\alpha) r_{\mathrm{T}}\right).
\end{aligned} \tag{18}
$$

Global/node coefficient reconstruction and common/context objectives train the sweeps and gate.

B.1 Field cells, messages, and objective

MPE uses eight typed outcome entities (four agents/four landmarks) and agent-to-all adjacency including self edges. MaMuJoCo projects 105 observations into four 46-D limb groups, clips selected contacts to $[-1, 1]$, and uses complete adjacency. RGB Ant Field uses four 32-D camera groups and complete adjacency. Endpoint-plus-edge widths are 8/92/64; three message layers output 16-D codes. Masking follows the coefficient transform, while reconstruction indexes original cells: this is structured completion, not strict hiding of every raw value. Inference reveals all coefficients.

Field readout and reconstruction. Field projects before subtraction, standardizes cells and transforms the subset axis. Masked coefficients become zero with a visibility flag. Gated message averages connect horizons, subsets and outcome groups. For L layers and largest horizon $h_{\max}$, token $s^L_{h,A,v}$ gives

$$
u^{\mathrm{Field}} = \rho_{\mathrm{M}}\left(|\mathcal{V}|^{-1}\sum_{v\in\mathcal{V}} s^L_{h_{\max},\varnothing,v}\right). \tag{19}
$$

A decoder uses this latent and horizon/subset/entity embeddings to reconstruct coefficients, then inverse-transforms to standardized input cells $\bar{V}$. With reconstruction $\widehat{V}$, visibility mask M and selected-entry MSE, training uses

$$
\mathcal{L}_E = \mathrm{MSE}_{1-M}(\widehat{V}, \bar{V}) + 0.25\,\mathrm{MSE}_M(\widehat{V}, \bar{V}) + \mathcal{R}_{\mathrm{var}}(u) + 0.02\mathcal{R}_{\mathrm{cov}}(u). \tag{20}
$$

In MaMuJoCo and visual training, masking covers 30% of coefficient cells, retaining the first-horizon empty-subset boundary. Set/Graph/Sweep use one horizon; Field uses three. The two-step MPE Set/Graph/Field comparison instead reconstructs fully visible value coordinates with a variance floor and 0.02 times the off-diagonal covariance penalty. Raw-coordinate controls use exactly this objective and schedule; Mean/Sweep/DeepSets/Random-edge retain their frozen method-specific trainers. Thus the family inventory is not an objective-matched architecture ablation.

The settings below specify the LAOM-style objective; Appendix G.5 separates pipeline roles. Scaling, horizon operators and donors are in Appendix A; constant-coordinate handling is in Appendix G.6.

Numeric optimization and action access. Front ends train 3,200 updates (batch 256, width 64), with LAOM-joint 64-D or per-slot 16-D codes, EMA rate 0.001, and offsets at most three in MaMuJoCo. MPE uses a fixed two-step inverse window and direct horizon readouts. The two-layer 128-unit ReLU observation readout trains 1,200 updates; representations/history request 3,200/1,200 updates under frozen method-specific capacity profiles. MPE route encoders use batch 64. Methods share data/budgets, not parameter counts. Continuous, LAPO-state/LAOM-state and IDM use explicit $t+2$ adapters; their shared model classes are unchanged. LAPO-state/LAOM-state retain their 128-unit hidden layers and are distinct from the joint front ends. Decoder validation takes episode indices $0, 4, 8, \ldots$ from each B-prefix; fitting uses the rest. Total labelled vectors are $22B$ in MPE or at most $200B$ in MaMuJoCo, including validation. Decoders train 1,000/800 updates; BC trains 3,200. Pre-grounding checkpoints use unlabelled latent error, grounded checkpoints use budgeted validation actions, never evaluation return. Action-access logs from data collection are withheld from the observation-only learner.

All representation/history models freeze before target-label export. Pretraining excludes actions, rewards, hidden snapshots and simulator queries; known action dimensions/ranges are interface information.

RGB Ant optimization and evaluation rendering. The visual LAOM front end takes $9 \times 64 \times 64$ inputs, channels $(16, 32, 64)$, one residual block per stage, 32-D codes and 128-unit heads; it trains 400 updates (batch 32, learning rate 10^{-4}, offset at most ten, EMA 0.001). Pooled 256-D features undergo training-only PCA to 32 shared coordinates per camera. Direct readouts train 600 updates; representation and history train 256 each; fixed-final decoders train 600. Visual Base/Field have 32/16 dimensions, hence a 48-D Aux decoder; numeric Aux is 16+16. Eight labelled training trajectories provide 1,600 target vectors, no partner labels. BC uses the same recurrent unroll with 256 sampled or all 1,600 presentations per update; IDM trains a budgeted inverse-action model before pseudo-action policy learning on 32 training trajectories.

Five seeds (908711–908715) freeze before labels export. Uniform final evaluation disables MSAA and GL_DITHER for all 125 controllers; training images used MSAA4. Physical dynamics, target camera, partners and distractors remain unchanged. The primary statistic averages 27 conditions (908604–908630) within seed; original three and combined 30 are secondary. Independent replay at 908601 checked images, features, actions and rewards for all 125. Training seed, not controller or episode, is the replication unit.

Dataset splits. MPE uses 700 training/150 development episodes. Initializations 45–49 reuse the datasets for seeds 40–44, not new tasks or data; all four N values preserve label prefixes and the development split. MaMuJoCo uses 90 training/19 development episodes from a 128-episode source split. RGB Ant data contain 32 training/eight development trajectories of 200 transitions; eight labelled trajectories provide 1,600 actions. Budgets: Section 4; feature extraction and freeze order: above.

Evaluation schedule. MPE evaluates seeds 97220000–97220099 for 23 total steps, with one shared zero-action warmup and 22 learned decisions; MaMuJoCo evaluates 50, starts at $t = 0$, and excludes padding after termination (200-step maximum). The seed is the replication unit. Section 4 defines return, SD and paired comparisons; Appendix C.1 reports native metrics.

C Complete Control Results

Mean-h1-Aux also performs well: at $B = 8$ it reaches 703.86 ± 21.67, exceeding LAPO-joint, LAOM-joint, and BC by 362.41, 130.37, and 93.88. All five paired seeds favor these comparisons. Its gain over duplicated Base is smaller, 8.04, with three positive seeds.

Bold values identify the best displayed mean in each stated comparison group; rounded ties are highlighted together. Bold does not denote statistical significance. Paired-difference and resource tables do not rank entries by boldface. Tables retain every completed configuration on native return scales: episodes are averaged within seed, then summarized by five-seed mean and sample SD. Contrasts share evaluation conditions; budget-wise wins are descriptive, with no per-budget oracle. MPE contains 1,305 seed–configuration–budget–scale cells: 870 at $N = 700$, 145 at each smaller

N, and 705 new/600 reused overall. Code and weight fingerprints pass. These are development results, not held-out-task confirmation.

Table A1: MPE full-data returns ↑, five seeds. Entries: mean/seed SD (e.g., .103 denotes 0.103); bold: highest mean per budget. S/A: Solo/Aux; j/st: joint/state; Cont.: Continuous; Random: Random-edge; Global16: Global-Joint16; dup.: duplicate; IDM: IDM-relabel.

Method	B8	B16	B32	B64	B128	B256
Mean-h2-S	-17.345/.103	-17.364/.188	-17.272/.096	-17.348/.212	-17.346/.146	-17.335/.214
Mean-h3-S	-17.346/.101	-17.336/.159	**-17.244**/.058	-17.338/.217	-17.394/.211	-17.332/.184
Set-S	-17.541/.208	-17.409/.361	-17.370/.209	-17.429/.115	-17.338/.204	-17.142/.227
Graph-S	-17.364/.071	-17.436/.389	-17.423/.247	-17.381/.230	-17.152/.293	-17.049/.078
Sweep-S	**-17.332**/.084	-17.415/.419	-17.318/.145	-17.338/.204	-17.240/.226	-17.176/.122
Field-S	-17.527/.309	-17.337/.195	-17.251/.058	-17.494/.271	-17.449/.269	-17.304/.327
DeepSets-S	-17.374/.123	-17.365/.381	-17.477/.302	-17.577/.168	-17.558/.298	-17.329/.240
Random-S	-17.511/.214	-17.336/.289	**-17.244**/.078	-17.307/.144	-17.302/.137	-17.330/.083
Cont.-S	-17.590/.102	-17.440/.165	-17.386/.103	-17.470/.314	-17.363/.077	-17.330/.098
Mean-h2-A	-17.431/.153	-17.362/.172	-17.388/.111	-17.286/.160	-17.319/.146	-17.234/.297
Mean-h3-A	-17.429/.161	-17.358/.177	-17.369/.100	-17.276/.142	-17.293/.161	-17.393/.216
Set-A	-17.400/.156	-17.329/.250	-17.317/.116	-17.323/.241	-17.179/.102	-17.150/.204
Graph-A	-17.401/.169	-17.340/.228	-17.322/.094	-17.263/.108	-17.197/.147	-17.094/.187
Sweep-A	-17.413/.134	-17.313/.192	-17.379/.204	-17.249/.221	-17.112/.197	-17.076/.207
Field-A	-17.434/.145	-17.384/.211	-17.415/.153	-17.336/.268	-17.271/.244	-17.273/.264
DeepSets-A	-17.428/.154	-17.341/.201	-17.439/.261	-17.312/.133	-17.446/.251	-17.228/.187
Random-A	-17.443/.187	**-17.310**/.237	-17.291/.101	**-17.229**/.189	-17.224/.121	-17.135/.148
Cont.-A	-17.515/.180	-17.423/.200	-17.338/.152	-17.395/.292	-17.256/.173	-17.180/.066
LAOM-st-A	-17.484/.166	-17.391/.215	-17.425/.167	-17.393/.101	-17.389/.121	-17.280/.116
Global16-A	-17.413/.164	-17.338/.274	-17.323/.094	-17.437/.199	-17.344/.054	-17.316/.098
Base	-17.671/.307	-17.333/.181	-17.352/.083	-17.360/.246	-17.452/.208	-17.233/.223
Base-dup.	-17.506/.168	-17.352/.167	-17.357/.097	-17.259/.152	-17.302/.082	-17.137/.152
Entity-j	-17.484/.208	-17.424/.217	-17.403/.094	-17.514/.277	-17.429/.276	-17.540/.232
LAPO-j	-17.581/.510	-17.340/.095	-17.380/.070	-17.439/.121	-17.454/.140	-17.341/.170
LAOM-j	-17.480/.181	-17.406/.171	-17.385/.087	-17.419/.351	-17.492/.229	-17.489/.077
LAPO-st	-17.458/.189	-17.361/.099	-17.420/.073	-17.291/.357	**-16.802**/.287	-16.748/.305
LAOM-st	-17.533/.167	-17.444/.114	-17.610/.522	-17.445/.366	-17.678/.140	-17.624/.108
BC	-17.343/.148	-17.682/.507	-17.281/.107	-17.356/.287	-16.891/.155	**-16.695**/.145
IDM	-17.405/.142	-17.391/.133	-17.402/.148	-17.396/.135	-17.352/.136	-17.488/.255

Table A2: MaMuJoCo full-data returns ↑, five seeds. Entries: mean/seed SD; bold: highest mean per budget. S/A: Solo/Aux; j/st: joint/state; Cont.: Continuous; dup.: duplicate; Random: Random-edge; Target: Target-slot; IDM: IDM-relabel; Mean: mean-contrast encoder; LAOMst: LAOM-state.

Method	B4	B8	B16	B32	B64
Cont.	327.26/177.82	424.43/77.49	467.84/16.51	513.10/53.99	495.46/77.35
LAPO-st	453.78/95.72	513.05/35.13	540.80/34.34	589.90/36.75	561.83/112.09
LAOM-st	398.14/97.15	429.36/108.44	467.93/85.22	490.30/68.11	541.74/49.42
Mean	252.58/98.76	384.01/111.67	389.77/114.06	386.73/154.10	389.93/128.49
DeepSets	451.07/56.45	433.37/61.98	467.48/69.41	506.96/56.46	516.07/51.82
Set	335.01/118.80	351.48/123.71	347.45/90.36	410.94/58.53	386.13/96.49
Graph	405.25/69.96	379.02/100.56	429.58/40.58	462.27/30.53	501.78/59.55
Sweep	309.17/110.65	366.00/120.70	428.16/79.72	417.79/83.13	494.46/46.15
Field	520.88/51.30	537.40/75.43	600.98/39.20	593.48/30.02	621.65/32.05
Random	331.02/125.27	422.71/81.59	372.17/97.37	442.83/61.15	420.57/71.40

Method	B4	B8	B16	B32	B64
LAPO-j	287.82/182.44	341.45/177.40	474.96/78.55	529.16/51.07	577.09/35.26
LAOM-j	500.96/45.25	573.49/36.64	555.93/34.74	607.66/21.12	638.33/39.51
All-slots	612.02/37.46	640.27/45.03	669.17/15.93	675.67/21.72	670.57/30.07
Target	679.44/26.46	688.92/14.82	696.07/20.64	690.68/20.53	700.33/31.93
Mean-h1	313.82/158.27	304.31/138.42	320.01/142.69	332.84/79.54	311.01/137.16
Base	679.44/26.46	700.44/26.45	696.07/20.64	680.45/25.30	700.33/31.93
Base-dup.	**706.37**/20.30	695.82/32.69	**709.19**/20.33	706.86/6.20	710.24/11.89
Cont.-A	546.82/276.36	694.43/16.70	703.06/22.83	687.37/17.13	718.83/12.22
LAOMst-A	623.88/67.99	692.92/17.53	678.21/18.45	704.12/27.40	702.54/18.99
Mean-h1-A	659.20/20.48	**703.86**/21.67	696.87/27.68	688.49/29.96	708.69/23.06
Mean-h3-A	673.95/41.04	680.19/23.11	696.76/23.12	686.81/34.57	707.46/16.18
Set-A	668.76/42.64	681.07/19.39	685.20/19.99	705.05/14.97	703.83/20.46
Graph-A	679.37/19.21	687.63/10.50	703.38/19.34	696.21/40.88	712.96/20.91
Sweep-A	672.89/31.72	697.79/15.41	689.50/7.80	702.02/25.74	709.87/27.50
Field-A	672.46/35.96	691.09/26.22	702.72/17.06	**720.14**/5.83	712.03/23.85
Random-A	656.33/17.18	684.90/3.78	676.33/29.94	684.97/35.25	686.87/9.24
DeepSets-A	675.41/26.29	680.47/13.02	687.01/25.58	698.38/29.74	705.48/18.84
BC	509.81/67.78	609.98/26.04	657.13/41.43	701.49/14.98	**735.43**/8.35
IDM	531.62/283.73	669.50/39.61	695.20/67.19	699.01/39.27	718.20/26.95

Table A3: Complete RGB Ant matrix. Primary return uses 27 new conditions per seed; all-30 return is secondary. Dev action MSE uses the original training-render distribution. Bold: highest primary return among Frozen representations and among standard-batch supervised references, separately. Full-batch BC is listed separately.

Configuration	Return (27)	Return (30)	Dev MSE
Base	57.27 ± 18.25	61.14	0.345
Base-duplicate	51.59 ± 24.43	55.18	0.365
Mean-Solo	37.76 ± 38.12	34.37	0.354
Set-Solo	41.12 ± 31.18	40.14	0.305
Graph-Solo	76.69 ± 12.80	74.31	0.314
Sweep-Solo	78.94 ± 24.69	76.35	0.324
Field-Solo	**103.60** ± 29.45	106.29	0.327
DeepSets-Solo	68.67 ± 22.37	67.38	0.314
Random-edge-Solo	72.64 ± 25.53	70.44	0.338
Continuous-Solo	83.58 ± 16.51	87.18	0.417
LAPO-state-Solo	66.75 ± 33.13	65.94	0.470
LAOM-state-Solo	91.05 ± 14.85	91.16	0.399
Mean-Aux	64.49 ± 49.82	66.66	0.524
Set-Aux	62.02 ± 33.80	63.78	0.564
Graph-Aux	45.98 ± 35.39	42.84	0.522
Sweep-Aux	48.42 ± 20.85	50.49	0.504
Field-Aux	94.56 ± 36.80	89.05	0.435
DeepSets-Aux	48.52 ± 31.83	48.69	0.478
Random-edge-Aux	85.25 ± 16.47	85.62	0.392
Continuous-Aux	95.61 ± 17.24	93.57	0.407
LAPO-state-Aux	66.17 ± 30.52	67.04	0.544
LAOM-state-Aux	87.69 ± 28.60	87.35	0.415
Feature-BC-256	**159.19** ± 23.01	159.37	0.350
Feature-BC-1600	162.25 ± 18.95	162.57	0.410
IDM-relabel	144.01 ± 14.83	147.24	0.340

C.1 NATIVE OUTCOMES

Coverage distance and collision counts complement return; both are lower-is-better. They are computed from the same 100 evaluation episodes per seed. Their changes need not agree, so we do not combine them into a new score. MaMuJoCo episode lengths, which measure survival rather than speed, remain in the accompanying evidence; no locomotion evaluation was changed in this revision.

Table A4: MPE native outcomes at $N = 700$, $B = 128$. Mean (seed SD), five seeds. Both columns ↓.

Configuration	Coverage distance	Collisions
Field-Aux	0.302 (0.009)	0.588 (0.063)
Graph-Aux	0.298 (0.005)	0.532 (0.048)
Sweep-Aux	0.301 (0.009)	0.522 (0.047)
Set-Aux	0.299 (0.003)	0.510 (0.084)
Mean-h2-Aux	0.299 (0.004)	0.556 (0.057)
LAPO-joint	0.297 (0.007)	**0.502 (0.026)**
LAOM-joint	0.304 (0.019)	0.510 (0.083)
BC	**0.280 (0.004)**	0.584 (0.034)
IDM-relabel	0.292 (0.007)	0.506 (0.036)
Base	0.302 (0.008)	0.568 (0.042)
Base-duplicate	0.300 (0.005)	0.544 (0.043)
Random-edge-Aux	0.300 (0.012)	0.556 (0.047)

D ADDITIONAL UNLABELLED DATA AT A FIXED ACTION BUDGET

All MPE configurations use $B = 32$ at $N = 32, 175, 350, 700$, keeping the same labelled prefix and development split within seed. The $N = 700$, $B = 32$ cells are reused from the budget sweep and counted once. Table entries show means; all seed values and SDs are in the evidence. Twenty of 29 full-minus-small means are positive, but curves are not uniformly monotonic. This is descriptive coverage, not 29 independent significance tests. BC also uses observation normalization fitted at the corresponding N; changing N can change that baseline despite the fixed labels. The preceding nine planned scaling tests retain their original Holm family and none is significant after adjustment.

Table A5: Complete MPE observation scaling, return ↑. Five-seed means at fixed B32; Δ is N700 minus N32 and + counts positive paired seeds.

Configuration	N32	N175	N350	N700	Δ	+
Mean-h2-Solo	-17.375	-17.604	-17.567	-17.272	+0.104	3/5
Mean-h3-Solo	-17.390	-17.626	-17.445	**-17.244**	+0.146	3/5
Set-Solo	**-17.208**	-17.580	-17.527	-17.370	-0.161	1/5
Graph-Solo	-17.232	-17.562	-17.576	-17.423	-0.192	1/5
Sweep-Solo	-17.324	-17.463	-17.405	-17.318	+0.006	2/5
Field-Solo	-17.332	-17.301	**-17.239**	-17.251	+0.081	3/5
DeepSets-Solo	-17.338	-17.431	-17.529	-17.477	-0.140	1/5
Random-edge-Solo	-17.292	-17.305	-17.630	**-17.244**	+0.048	2/5
Continuous-Solo	-17.379	-17.387	-17.415	-17.386	-0.007	3/5
Mean-h2-Aux	-17.568	-17.306	-17.342	-17.388	+0.180	4/5
Mean-h3-Aux	-17.563	-17.349	-17.374	-17.369	+0.194	4/5
Set-Aux	-17.564	-17.360	-17.352	-17.317	+0.247	4/5
Graph-Aux	-17.549	-17.362	-17.423	-17.322	+0.226	4/5
Sweep-Aux	-17.558	-17.364	-17.417	-17.379	+0.179	4/5
Field-Aux	-17.567	-17.430	-17.319	-17.415	+0.152	4/5
DeepSets-Aux	-17.567	-17.431	-17.342	-17.439	+0.129	4/5
Random-edge-Aux	-17.554	-17.376	-17.342	-17.291	+0.263	4/5

Configuration	N32	N175	N350	N700	Δ	+
Continuous-Aux	-17.429	-17.368	-17.297	-17.338	+0.091	4/5
LAOM-state-Aux	-17.387	-17.352	-17.407	-17.425	-0.037	1/5
Global-Joint16-Aux	-17.370	**-17.256**	-17.289	-17.323	+0.048	3/5
Base	-17.495	-17.412	-17.379	-17.352	+0.143	5/5
Base-duplicate	-17.553	-17.387	-17.378	-17.357	+0.196	4/5
Entity-joint	-17.388	-17.379	-17.369	-17.403	-0.015	3/5
LAPO-joint	-17.487	-17.452	-17.359	-17.380	+0.108	3/5
LAOM-joint	-17.368	-17.400	-17.385	-17.385	-0.018	3/5
LAPO-state	-17.454	-17.297	-17.363	-17.420	+0.034	3/5
LAOM-state	-17.329	-17.562	-17.543	-17.610	-0.281	1/5
BC	-17.309	-17.264	-17.277	-17.281	+0.027	4/5
IDM-relabel	-17.359	-17.402	-17.389	-17.402	-0.043	1/5

Table A6: MaMuJoCo scaling endpoints. Full N90 minus minimum N8; return ↑. All intermediate records remain in the evidence.

Configuration	B	Δ	Positive
Target-slot	8	+85.330	5/5
LAPO-joint	8	+94.734	4/5
LAOM-joint	8	+73.688	5/5
Base-duplicate	8	+108.151	5/5
Continuous-Aux	8	+115.659	5/5
Mean-h1-Aux	8	+104.922	5/5
Graph-Aux	8	+102.225	5/5
Sweep-Aux	8	+112.219	5/5
BC	8	+8.034	3/5
IDM-relabel	8	+135.115	5/5

E Frozen Action Readout and Correspondence Diagnostics

Frozen readers and evaluation splits. Ridge readers use fixed history features (before acting) or inverse codes (with future observations), fitting-label normalization, an unpenalized intercept, and validation-selected penalty $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$. They never refit on validation data and read development actions only after fitting. Skill $1 - E/E_0$ compares MSE with the fitting-action mean. MPE reserves every fourth budgeted episode for validation and evaluates the last 75 development episodes, also used for history-checkpoint selection. This reuse makes the readout a development diagnostic. Feature widths differ; skill isolates neither structure nor return. MPE uses two-step probes; one-step probes are analyzed separately. MaMuJoCo's 350 rows cross seven features, two views, five budgets and five seeds, including Field with and without zeroing training-constant readout inputs. Matched/reassigned action probes without this zeroing are excluded from the reported claims (Appendix G.6).

Table A7: MPE history native-action readout skill (percent) ↑, all label budgets. Mean of five seeds; zero is the fitting-set mean-action predictor.

Configuration	B8	B16	B32	B64	B128	B256
Mean-h2-Solo	4.94	3.36	6.15	7.26	8.25	8.80
Mean-h3-Solo	5.06	5.87	5.68	6.95	7.62	8.68
Set-Solo	**5.87**	2.45	7.40	9.37	10.94	11.81
Graph-Solo	3.84	-1.82	6.18	9.81	11.29	12.44
Sweep-Solo	-4.64	5.39	8.43	11.32	12.17	12.83
Field-Solo	3.53	5.25	7.83	10.06	11.51	12.59
DeepSets-Solo	4.97	6.61	6.91	9.70	10.36	11.47

Configuration	B8	B16	B32	B64	B128	B256
Random-edge-Solo	4.58	-1.57	6.36	8.30	9.24	9.86
Continuous-Solo	1.17	0.78	2.52	4.54	7.73	8.53
Mean-h2-Aux	3.84	5.54	6.18	14.25	17.24	19.05
Mean-h3-Aux	3.84	4.58	7.32	12.53	17.49	19.03
Set-Aux	4.72	2.55	6.30	13.83	18.56	20.41
Graph-Aux	5.27	4.83	7.78	15.09	18.00	21.02
Sweep-Aux	5.60	**8.11**	8.39	15.26	19.39	21.42
Field-Aux	-4.16	6.12	10.03	11.95	18.82	20.64
DeepSets-Aux	4.51	6.86	7.41	12.52	15.98	18.69
Random-edge-Aux	4.25	6.39	1.58	12.96	18.08	20.02
Continuous-Aux	2.68	3.01	4.73	13.12	17.47	19.72
LAOM-state-Aux	2.52	3.72	4.96	8.25	13.89	16.71
Global-Joint16-Aux	2.75	2.50	3.31	7.88	11.79	16.70
Base	-11.15	3.81	8.10	14.08	15.50	16.24
Entity-joint	0.72	3.19	3.25	6.20	11.98	14.35
LAPO-joint	-7.14	3.99	**16.24**	**24.06**	**28.60**	**31.66**
LAOM-joint	-0.22	0.10	2.82	4.23	6.41	9.95
LAPO-state	-12.52	5.35	15.65	20.67	25.70	28.89
LAOM-state	-1.21	1.71	0.86	2.56	5.63	6.29

MaMuJoCo: complementary features and constant-input handling. Combining Base and Field yields 73.51% mean history skill, compared with 68.62% for Base and 73.20% for Base plus an independent branch. The paired full-budget increment over Base is 4.89 percentage points (descriptive 95% interval $[4.07, 5.70]$); against the independent combination it is 0.31 points ($[-0.10, 0.73]$). Zeroing training-constant inputs raises Field's skill by 36.68 points. The no-zeroing rows diagnose numerical sensitivity, not an alternative representation method; Base and independent-branch comparisons test complementarity.

Table A8: MaMuJoCo frozen history action skill (%) ↑: mean/seed SD, five seeds. No-zeroing rows retain training-constant readout inputs.

Configuration	B4	B8	B16	B32	B64
Base	68.13/1.27	68.48/1.20	68.74/1.14	68.87/1.12	68.91/1.10
Independent	66.15/1.87	66.91/1.61	67.16/1.57	67.25/1.60	67.28/1.57
Field (no zeroing)	12.75/4.39	14.03/4.39	14.47/4.51	15.75/4.04	15.90/3.95
Field	49.89/2.14	50.29/2.63	51.73/0.57	52.00/0.70	52.38/0.66
Base + Field (no zeroing)	68.29/1.13	69.13/1.13	69.55/1.12	69.72/1.15	69.79/1.08
Base + Field	**72.50**/1.19	**73.17**/0.95	**73.73**/0.92	**74.01**/0.82	**74.16**/0.84
Base + independent	72.43/0.76	73.01/0.80	73.39/0.78	73.57/0.73	73.62/0.72

Transition-informed action skill spans 72.25–73.00% for Base, 75.33–77.86% for Base+Field, and 78.40–80.04% for Base+independent. The independent branch leads this future-informed view; Field is slightly stronger in the deployment-relevant history readout.

E.1 Correspondence Diagnostics: What Reassignment Tests

Whole-table reassignment transfers an intact table to another episode's history within split, timestamp and validity group. It breaks history correspondence, not matching within the table. The MPE reassignment diagnostic uses a one-step inverse window; its mixed results are separate from the two-step MPE evidence. No two-step reassignment experiment is reported. The MaMuJoCo comparison below uses zeroed training-constant readout inputs.

Table A9: MaMuJoCo: matched minus whole-table reassigned return. Parentheses count positive seed differences. Constant-input zeroing; common formal evaluation seeds.

Model	B4	B8	B16	B32	B64
Mean-h3	-6.743 (1/5)	-0.633 (3/5)	-4.868 (3/5)	-10.299 (2/5)	-10.374 (2/5)
Graph	-0.308 (1/5)	-15.287 (2/5)	+20.149 (4/5)	-2.507 (2/5)	+3.728 (2/5)
Field	-11.952 (2/5)	+3.517 (3/5)	+0.025 (2/5)	+37.730 (5/5)	-0.146 (3/5)

Table A9 shows that the value of history–table correspondence varies with the method and label budget. Field at B32 gives the clearest gain: $+37.730$, positive in all five seeds. Other cells have mixed signs. Reassignment tests the link from a history to its complete table; the matched comparisons inside that table remain intact.

Separating complete-policy gains from structural contributions. Base duplication repeats 16-D inputs without a second history network. Independent-Aux separately trains a target-slot front end/history predictor, then concatenates with Base; all five saved targets exactly match target-slot zero (no all-slot expansion). Both numeric MaMuJoCo auxiliary pipelines use two 108,816-parameter histories and a 74,754-parameter, 32-input decoder, but differ in upstream objective/compute. This control matches capacity, not all upstream costs.

Table A10: **Increment over the formal Base controls.** Field-Aux minus comparator return, averaged over five paired training seeds. Counts in parentheses are positive seed differences, not significance tests. All full-data budgets are shown; these use the formal evaluation set.

Setting	B	Versus Base	Versus duplicate
MPE	8	+0.237 (5/5)	+0.072 (4/5)
	16	-0.051 (1/5)	-0.031 (1/5)
	32	-0.063 (2/5)	-0.057 (2/5)
	64	+0.024 (2/5)	-0.077 (3/5)
	128	+0.181 (4/5)	+0.031 (2/5)
	256	-0.040 (2/5)	-0.135 (1/5)
MaMuJoCo	4	-6.981 (2/5)	-33.905 (1/5)
	8	-9.346 (3/5)	-4.725 (3/5)
	16	+6.650 (3/5)	-6.474 (3/5)
	32	+39.687 (5/5)	+13.280 (5/5)
	64	+11.694 (3/5)	+1.788 (3/5)

Table A10 locates the strongest MaMuJoCo improvement at B32, where Field-Aux exceeds both Base and duplicated Base in all five seeds. MPE also improves over both controls at B8, while gains at other budgets vary. The auxiliary representation is therefore useful in particular budget regimes; its contribution is assessed against both the original and duplicated features.

Appendix G separately tests coordinates and structure with matched allocated parameters, updates, Base, histories and decoder; allocated size, active information and execution cost remain distinct.

F Literature, Information Access, and the Role of Matching

Action-free learning and structured codes. FLAM conditions factor codes on context, OTF-LAM factorizes effects, and ALAM constrains composition/reversal (Wang et al., 2026; Nam et al., 2026; Tang et al., 2026). Our comparisons use inferred-code endpoints at shared history and partner context. Table A11 distinguishes action access.

Table A11: **Information access in related pipelines.** Literature descriptions, not replications; "later" means after observation training.

Method	Representation learning and native-action access
ILPO (Edwards et al., 2019)	Action-free transitions; later interaction learns remapping.
LAPO (Schmidt & Jiang, 2024)	Consecutive observations; later decoder/online adaptation.
LAPA (Ye et al., 2025)	Video quantization and image/language policy; later action-labelled fine-tuning.
LAOM / LAOM+ (Nikulin et al., 2025)	Observation prediction; "+" uses representation-training actions. Both ground native actions.
OCLAM (Klepach et al., 2026)	Self-supervised slots, selected by action-labelled linear probes before downstream use.
PLAS / LASER (Zhou et al., 2021; Allshire et al., 2021)	Recorded actions train action spaces; decoders map policy latents to commands.
This study	Declared observation groups; no pretraining actions, rewards or simulator queries. Frozen updates the decoder; Adapt also updates history copies. Neither uses online interaction.

Multi-agent context and interaction. CLAS shares latent actions across robots (Aljalbout et al., 2023); COMA fixes partner actions in critic comparisons (Foerster et al., 2018). Our comparisons instead compare observation vectors using inferred codes and donors. DeepHalo separates orders of context effects in choice models, while Faith-Shap attributes interactions in coalition-value functions (Zhang et al., 2026b; Tsai et al., 2023). They motivate keeping interaction orders explicit; neither establishes action identification for PC. PLAS and LASER learn action spaces from recorded actions; TACO jointly learns state and action representations from state–action sequences (Zhou et al., 2021; Allshire et al., 2021; Zheng et al., 2023); visual teleoperation and CLAS ground learned coordinates in native controls (Karamcheti et al., 2021; Aljalbout et al., 2023).

Linear analyses show how latent actions can encode exogenous variation (Zhang et al., 2026a; Lee et al., 2026). A complementary Ex-BMDP analysis gives sample-complexity limitations for action-free video representation learning (Misra et al., 2024). These are related obstacles, not a theorem that every observation-trained code fails. Identifiability results for entropy-regularized LAPO require explicit conditions (Lachapelle, 2025). CRAFT instead exploits differences between data-collection policies in an Ex-BMDP with shared exogenous dynamics and specified separation conditions (Levine et al., 2025). Our nearest-history donor rule does not supply those assumptions. COMA uses a scalar policy-averaged baseline; our method evaluates vector comparisons through history prediction and budgeted control. The subset transform is invertible, but 16-D compression can lose information. Action readout and control therefore receive separate evaluations (Nikulin et al., 2025).

F.1 Direct supervision and controlled comparisons

RGB Ant methods share a frozen encoder/projection, target-view history and eight labelled trajectories, with no partner labels. BC trains history on actions; Frozen Field trains its decoder; Adapt also trains history copies (Appendix B). LAOM+ uses actions during representation learning (Nikulin et al., 2025). Each view contains target and partner motion.

Equal unique labels can imply different presentations and trained parameter counts. BC is a same-budget reference, not an upper bound. Original RGB Ant labels 25% of its training transitions. LAOM reports supervision using about 2.5% of its dataset's transitions (Nikulin et al., 2025). These fractions refer to different datasets, not a matched cross-paper label budget. Appendix I adds B1/2/4/8 and matched history initialization/branch controls to separate supervision access from representation quality; upstream resources differ.

Table A12: **Visual action-grounding audit.** All five seeds use 1,600 unique target vectors and 600 updates. Action-trained counts include parameters updated by labels; frozen-history counts exclude the common visual encoder/projection and training-only encoders. These are grounding resources.

Configuration	Action-trained parameters	Frozen history parameters	Labels per update	Total label presentations
Feature BC (minibatch)	78,978	0	256	153,600
Feature BC (full budget)	78,978	0	1,600	960,000
Field-Solo	18,946	80,784	256	153,600
Field-Aux	23,042	163,632	256	153,600

G Structural Controls and Physical-Effect Evaluation

G.1 MPE: two-step structural controls

Original paired contrasts share learned endpoints, Base, 16-D auxiliary outputs, history/decoder architectures, label splits and schedules. Raw replaces Möbius/zeta by identities; Graph-complement replaces S by $\mathcal{O}_j \setminus S$. Mean-h3 one-context retains the empty-subset contrast, reducing information/computation; Global-Joint16 tests a whole-transition auxiliary. The primary statistic is paired normalized trapezoidal AUC in $\log_2 B$, distinct from an unweighted budget average. Four-degree-of-freedom t intervals and Holm retain the original three-/two-test families; all intervals include zero.

The Graph input control has 16 coordinates and 49,624 parameters. $[A, A - B]$, $[A, A - B_{\mathrm{comp}}]$ and $[A, B]$ invertibly retain the same endpoints, sharing reconstruction, initialization, normalization, batches, updates and grounding. Five-seed N700/B32 returns are $-17.31842/-17.30781/-17.32979$. The unadjusted exact p-values against complement/endpoints are $0.3125/0.4375$; both Holm-adjusted values are 0.625. This isolates matching as an input bias, separately from edge-only Graph.

Table A13: MPE paired control gains ↑. Five seeds; 95% intervals, family-specific Holm p, and positive seed counts. Top: log-budget AUC; bottom: shared-endpoint Graph input control at B32.

Method / control	Δ	95% interval	Holm p	+
Graph-Aux / raw	+0.0168	[-0.1098, +0.1435]	1.000	3/5
Field-Aux / raw	+0.0217	[-0.0794, +0.1228]	1.000	3/5
Graph-Aux / complement	+0.0926	[-0.0325, +0.2178]	0.327	4/5
Set-Aux / raw	-0.0333	[-0.1151, +0.0485]	0.643	2/5
Mean-h3-Aux / one context	-0.0124	[-0.1285, +0.1036]	0.781	2/5
Input: matched / endpoints	+0.0114	[-0.0283, +0.0510]	0.625	3/5
Input: matched / complement	-0.0106	[-0.0359, +0.0147]	0.625	1/5

Table A14: Additional MPE control arms: mean return above seed SD, five seeds. Reference rows: Table A1. Raw retains predictive-contrast inputs. Bold: highest displayed mean per budget.

Configuration	B8	B16	B32	B64	B128	B256
Graph-raw-Aux	-17.415	-17.335	-17.344	-17.250	-17.234	-17.168
	(0.144)	(0.199)	(0.079)	(0.192)	(0.130)	(0.164)
Graph-complement-Aux	**-17.383**	-17.307	-17.397	-17.408	-17.342	-17.375
	(0.186)	(0.208)	(0.133)	(0.137)	(0.236)	(0.179)
Field-raw-Aux	-17.436	-17.342	-17.355	-17.415	-17.327	-17.422
	(0.154)	(0.269)	(0.131)	(0.248)	(0.223)	(0.257)

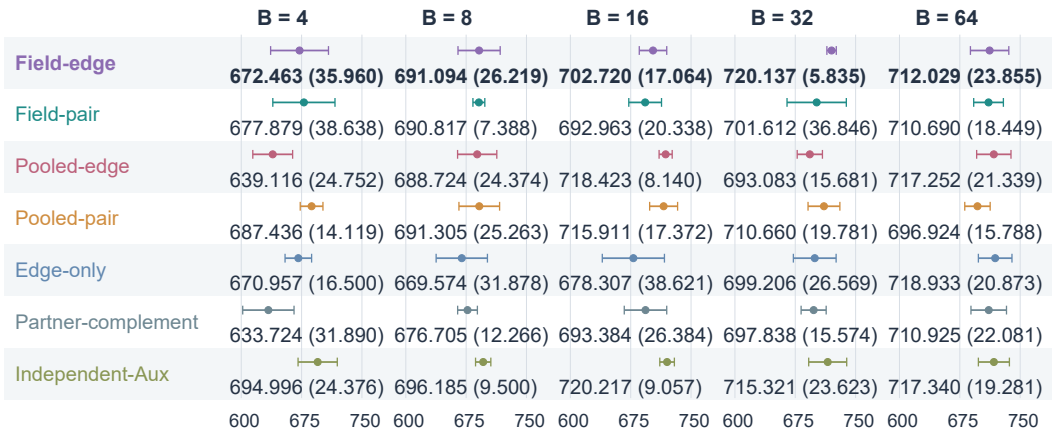


Figure A1: MaMuJoCo structural controls: five-seed means and one seed SD, with exact mean (SD) labels. All five budgets share a return scale and method order; higher is better. Bold identifies Field-edge, not the budget-wise winner.

Configuration	B8	B16	B32	B64	B128	B256
Set-raw-Aux	-17.404	**-17.302**	-17.341	**-17.156**	**-17.208**	**-17.099**
	(0.175)	(0.214)	(0.128)	(0.102)	(0.140)	(0.170)
Mean-h3-one-context-Aux	-17.418	-17.330	**-17.325**	-17.237	-17.389	-17.309
	(0.129)	(0.210)	(0.104)	(0.144)	(0.197)	(0.280)

G.2 MaMuJoCo: factorial controls

The 2×2 factorial crosses Field/pooled processing with endpoint-plus-edge/endpoint-pair coordinates. Pooling removes subset transforms and replaces typed by uniform mixing; extra arms remove endpoint channels or complement references. Six arms share 55,737 allocated encoder parameters, 3,200/1,200 representation/history updates, 16-D outputs, Base, five budgets and five seeds. The independent target-slot auxiliary differs in upstream objective/cost; these controls are separate from two-step MPE raw controls. Mean-budget gains over Pooled-edge/Partner-complement are $+8.3689/+17.1732$, with 95% intervals $[-8.0206, 24.7584]/[-9.1465, 43.4929]$ and Holm $0.4585/0.4328$. The four-test correction family also includes one-step MPE contrasts $-0.0183/-0.0914$, kept separate from two-step results.

Mean-budget Field-edge gains over Field-pair/Pooled-edge/Pooled-pair/Edge-only/ Partner-complement/Independent-Aux are $+4.90/+8.37/-0.76/+12.29/+17.17/-9.12$; all paired intervals include zero. Gains depend on the component; native outcomes are reported separately (Appendix C.1).

G.3 Physical effects with state-aligned donor transport

This additional offline study tests observation-fitted transport with the frozen MPE front ends. The alignment is specific to this diagnostic; all other experiments retain their original, uncorrected donor construction. Donors come from the original 1,024-entry bank, using nearest standardized two-step history. Only target slot zero is replaced; partner codes and history stay at their query values.

Aligning motion in time. Let $p_t, p_{t+1}, p_{t+2} \in \mathbb{R}^2$ be one agent's positions in an observed fragment. Write $m_t = p_{t+1} - p_t$ and $\Delta_t = p_{t+2} - p_t$. The MPE integrator advances position before updating velocity, so m_t reflects motion at the query time; $p_t - p_{t-1}$ is a lagged proxy. Pooling agents and fragments in the observation fit set, we estimate $\beta = \sum_t m_t^\top \Delta_t / \sum_t \|m_t\|_2^2$. For query q and donor r, we then construct

$$\widetilde{p}_{t+2}^{r \to q} = p_t^q + \beta m_t^q + (\Delta_t^r - \beta m_t^r). \tag{21}$$

The donor indices denote its local fragment times. The unchanged inverse encoder maps $(p_t^q, \widetilde{p}_{t+2}^{r\to q})$ to a replacement target code, and the unchanged predictor forms the matched difference. The scalar estimates a motion trend in the training distribution, not an identified damping parameter. This is an offline construction using a factual intermediate frame; no future frame is supplied to a deployed history policy.

Protocol and controls. All five frozen model seeds (45–49) are retained. For each, the first 560 of 700 training-observation episodes fit the correction; 140 monitor observation error without tuning. Coefficients (1.8880–1.8933) and source hashes are frozen before evaluation. Fitting uses positions and existing observation codes, without actions, rewards, counterfactual outcomes or simulator parameters. Thirty-two new shared trajectories (seeds 97927000–97927031), queried at $t = 2, 6, 10, 14$, give 640 query–model pairs; the five models are the replication units. This exploratory follow-up was designed after earlier donor diagnostics, with a fixed new evaluation set and no result-dependent refitting.

Evaluation replaces only the first target action with the recorded donor command, holding state, partner actions and continuation fixed. Actions and simulator branches supply truth only. The $h = 2$ effect agrees with the independent integrator expression within 1.17×10^{-7}; target-action replacement leaves the intermediate frame unchanged. Original model and data hashes are preserved.

Table A15 compares direct donor-code reuse, the new transport, and five observation-based controls. Lagged-motion transport uses $p_t - p_{t-1}$; translation-only moves the observed donor displacement without motion correction. Latent centering subtracts a fixed ridge prediction of the donor code from local position and prior displacement, then adds the query prediction. No-donor transport sets the donor residual in Equation 21 to zero. Shuffled-residual transport uses a fixed nonself permutation of the same bank while truth retains the original donor action. Zero predicts no effect. Query-state oracle re-encodes the simulator counterfactual and serves only as an evaluation reference.

Table A15: New-cohort physical position-effect MSE at $h = 2 \downarrow$: five-model mean (one SD). Bold identifies the proposed alignment, not a minimum across the oracle and learned methods.

Construction	Target coordinates	All-agent coordinates
Raw donor	0.123685 (0.009131)	0.031450 (0.002348)
Aligned transport	**0.001128 (0.000372)**	**0.000297 (0.000095)**
Lagged-motion transport	0.007943 (0.002164)	0.002057 (0.000545)
Translation only	0.032457 (0.005859)	0.008297 (0.001480)
Latent centering	0.014710 (0.001783)	0.003947 (0.000461)
No donor residual	0.001587 (0.000031)	0.000404 (0.000008)
Shuffled residual	0.002586 (0.000384)	0.000660 (0.000096)
Zero effect	0.002116 (0.000066)	0.000529 (0.000017)
Query-state oracle	0.000145 (0.000007)	0.000045 (0.000003)

Aligned transport reduces target-coordinate $h = 2$ MSE by 46.7% relative to zero and 85.8% relative to lagged-motion transport; all five models improve in both comparisons. The paired zero-minus-aligned mean is 0.000988, with descriptive 95% t interval $[0.000531, 0.001445]$ and exact two-sided sign-flip $p = 0.0625$. It also lowers all-agent MSE by 43.9% relative to zero. Compared with no donor residual, target MSE falls by 28.9% (four of five models; paired interval $[-0.000034, 0.000952]$, $p = 0.125$). Compared with a shuffled residual, it falls by 56.4% (five of five). These comparisons are descriptive and unadjusted; the five-model exact test has minimum $p = 0.0625$.

Table A16: Additional physical-effect checks, five-model mean target-coordinate MSE ↓. Upper quartile selects each model's largest true $h = 2$ effects; the primary comparison above retains every query.

Construction	$h = 1$	$h = 3$	Upper quartile, $h = 2$
Raw donor	0.032143	0.231936	0.130725
Aligned transport	0.000612	0.002913	0.001150
Lagged-motion transport	0.003447	0.013537	0.007943
Translation only	0.010228	0.058606	0.035153
Latent centering	0.004981	0.026279	0.014558
No donor residual	0.000281	0.004660	0.002246
Shuffled residual	0.000533	0.006533	0.003165
Zero effect	0.000000	0.006480	0.004593
Query-state oracle	0.000374	0.001024	0.000338

On the upper-effect quartile, alignment lowers MSE by 75.0% relative to zero and 48.8% relative to no donor residual, improving all five models. The $h = 3$ target MSE also improves over zero. At $h = 1$ the true effect is zero, while the predictor can retain error. These results support aligned reference construction in this coordinate setting. Integration into history targets, grounding and closed-loop control remains to be tested.

G.4 Whole-system auxiliary-route comparison

MPE averages six budget differences within each of five seeds (45–49), then across seeds, separately from the primary log-budget AUC tests. Global-Joint16 encodes all 16 coordinates with the same two-step inverse window. Both routes use 1,200 history updates and a 16-D auxiliary plus the same 16-D Base; upstream objectives/computation differ. Table A17 retains all compared members.

Table A17: MPE route gains over Global-Joint16-Aux (↑). $+$ counts positive seed averages; it does not indicate significance.

Aux route	Mean-h2	Mean-h3	Set	Graph	Sweep	Field
Mean gain	+0.025	+0.009	+0.078	+0.092	+0.105	+0.010
$+$	3/5	3/5	4/5	4/5	5/5	2/5

MaMuJoCo's 875 controls cross five seeds, five budgets and five nested decoder repeats. Field/Graph/Set gains are $+1.63/-7.52/+0.14$ over Global-Joint16-Aux and $-3.01/-7.78/+2.92$ over own raw. Global-Joint16 encodes 105 coordinates; The PC comparison uses full-visibility reconstruction with shared Base/downstream interfaces. Field's route gain is positive in four seeds, Graph's in none.

The 500-control Field study crosses masking/coordinates at native loss groups/weights. Five native masked checkpoints exactly reproduce the reference representation/history weights. Möbius-minus-raw gains are $-0.57/+1.17$ (masked/visible), interaction -1.74; masks index different bases. All nine intervals cross zero. Five seed pairs give 2^5 sign assignments and minimum two-sided exact $p = 0.0625$ before Holm; nested repeats add precision, not replication. This resolution neither removes small/negative effects nor establishes equivalence. Raw retains predictive-contrast tables; evidence includes all intervals/tests.

G.5 Roles of the Numeric LAOM Adaptation and Entity Front End

Both use the same LAOM-style state adapter: continuous inverse codes, detached current features, forward prediction and EMA targets at offsets 1–3 (Nikulin et al., 2025), with vector MLPs instead of visual modules; MPE fixes the inverse window at $t + 2$ and uses direct horizon readouts. LAOM-joint uses all declared observations (64-D); the entity front end uses four known groups (16-D each): positions in MPE, joint positions/velocities in MaMuJoCo. Base predicts target slot zero; Field-Aux

adds a separately predicted 16-D comparison latent. Independent-Aux instead adds a separately initialized and trained target-slot predictor, without predictive comparisons. LAPO uses a separate quantized adapter. All freeze before labels.

G.6 Numerical stability of recursive endpoint prediction

Training-constant coordinates. Of the readout's 210 input coordinates, 54 are training-constant. The adapter zeros these normalized coordinates at every recursive call. Without zeroing, their 10^{-6} normalization scale amplifies drift through first-layer weights that remain at initialization; finite-output and checkpoint checks alone do not detect this instability. Across five sources, third-step standardized error is roughly $1{,}378$–$6{,}275$ without zeroing and 0.465–0.650 with it. Zeroing changes no fitted weights, codes, donors, masks or label access and applies to all affected table consumers. Full-data one-step endpoints, Mean-h1, Base, joint baselines, BC, IDM, MPE and direct-horizon visual models are unaffected.

Small-data numerical checks. Before 45 grounding runs, sparse-data validation requires finite outputs, varying-increment standardized error < 10, constant increments within $10^{-4}(1 + \max|x_{\text{train}}|)$ of h times training drift, and factual development raw MSE ≤ 10 times persistence MSE $+10^{-10}$. Both branches and all masks are checked. Twenty sources with constant-input zeroing pass; five sources without it fail because of recursive divergence. These checks test numerical stability, not recombination ground truth.

Evidence coverage. The validated results cover 595 controls, 119 five-seed cells and 29,750 episodes. The sparse-data cohort uses 21 existing representations and 24 separately trained ones; all 45 histories freeze before grounding. Checks cover 721 artifacts, access, budgets, 50 evaluation seeds, 200-step episodes and raw-return agreement. Eight sparse-data validation failures are documented separately. Affected probes use the adapter with constant-input zeroing.

H Interaction Information, Compression, and Control

H.1 What a partner-indexed coefficient describes

Fix a history, donor tuple, predictor, horizon and continuation, and let $F(U)$ be its endpoint when exactly the agents in U use their donor codes. For target j and $S \subseteq \mathcal{O}_j$, $g_j(S) = F(S) - F(S \cup \{j\})$. Define the full assignment-cube coefficient $D_F(V) = \sum_{U \subseteq V} (-1)^{|V|-|U|} F(U)$.

Proposition 5 (Target-containing interaction coefficients)**.** *For the same binary assignment cube throughout the comparison,*

$$c_j(A) = -D_F(A \cup \{j\}), \qquad A \subseteq \mathcal{O}_j. \tag{22}$$

Proof. Partition the subsets of $A \cup \{j\}$ according to whether they contain j:

$$D_F(A \cup \{j\}) = \sum_{S \subseteq A} (-1)^{|A|-|S|} \big[F(S \cup \{j\}) - F(S)\big] = -c_j(A).$$

□

The coefficient for one partner indexes its interaction with the target; two partners index a three-agent term. The identity requires one fixed assignment cube: history, donors, replacement semantics and continuation must remain shared. Coefficients are predicted-outcome vectors, not native-action effects or return allocations. Allocating joint terms to individuals requires another rule.

H.2 Why information retention and return can rank models differently

Let C stack selected, training-standardized interaction coefficients, U be a transition-informed latent, and $\widehat{U} = \pi(H)$ its history prediction, with $H = H^{\text{dep}}$. A fixed probe r measures whether C can be recovered from either latent. Write $\|X\|_{L^2} = (\mathbb{E}\|X\|_2^2)^{1/2}$, assuming the required second moments exist.

Proposition 6 (Interaction retention and history transfer)**.** *If r is L_r-Lipschitz on the relevant support, then*

$$\|C - r(\widehat{U})\|_{L^2} \le \underbrace{\|C - r(U)\|_{L^2}}_{\text{representation readout error}} + L_r \underbrace{\|U - \widehat{U}\|_{L^2}}_{\text{history prediction error}}. \tag{23}$$

For any square-integrable history-based coefficient predictor $q(H)$, with $m(H) = \mathbb{E}[C \mid H]$,

$$\mathbb{E}\|C - q(H)\|^2 = \underbrace{\mathbb{E}\|C - m(H)\|^2}_{\text{history-irreducible error}} + \underbrace{\mathbb{E}\|m(H) - q(H)\|^2}_{\text{history readout error}}. \tag{24}$$

Proof. Insert $r(U)$, use the L^2 triangle inequality and the Lipschitz assumption for Equation 23. For Equation 24, expand $C - q = (C - m) + (m - q)$ and condition the cross term on H; $\mathbb{E}[C - m \mid H] = 0$. □

The experiments fit separate readers for U and $\widehat{U}$; they do not estimate the fixed-r bound. Readout error and control return measure different properties. A feature can complement Base without recovering the probed coefficients (encoder inputs and compression: Appendix B).

H.3 Frozen, partner-resolved and independent observation probes

FLAM evaluates factor–entity correspondence in learned representations (Wang et al., 2026). Our diagnostic asks a different question: whether a PC latent retains each learned partner-indexed coefficient. The frozen readers below are our own protocol and use no actions, rewards or simulator queries. MaMuJoCo uses at most 1,024 fitting/256 episode-disjoint development transitions, fitting-only normalization and ridge 0.01. MPE uses 4,096 training transitions, 75 development episodes for ridge selection (the action-probe grid), 75 for readout, and averages h2/h3 within seed. Representation/history inputs exclude Base; widths are family 16, LAPO 128, LAOM 64. Appendix E documents development reuse.

For a target block C_A, the diagnostic skill is

$$s_A = 1 - \frac{\mathrm{MSE}_{\mathrm{dev}}(\widehat{C}_A, C_A)}{\mathrm{MSE}_{\mathrm{dev}}(\overline{C}_{A,\mathrm{train}}, C_A)}. \tag{25}$$

Skill 0.1 means 10% lower MSE than the fitting-target mean. Tables show five-seed mean skill by partner; learned coefficients are not physical truth.

Table A18: MPE coefficient skill (%) ↑, five seeds; h2/h3 averaged within seed. R/H: representation/history without Base.

Model	R:1	R:2	R:3	H:1	H:2	H:3
Mean-h2	2.71	3.89	2.48	0.62	0.46	0.61
Mean-h3	2.84	3.70	2.50	0.55	0.40	0.54
Set	**27.95**	**27.31**	**26.53**	0.46	0.47	0.45
Graph	16.38	17.31	14.93	0.54	0.56	0.47
Sweep	15.53	16.23	14.47	0.51	0.32	0.36
Field	2.93	3.85	2.64	0.83	0.81	0.46
Base	0.32	0.63	0.49	0.29	0.58	0.42
LAPO-joint	1.00	1.08	1.20	**0.91**	**1.15**	**1.08**
LAOM-joint	0.69	0.58	0.66	0.62	0.33	0.03
Global-Joint16	0.67	0.59	0.66	0.65	0.28	0.41

Table A19: Single-partner coefficient readout in MaMuJoCo. Mean skill (%), five seeds; R/H: representation/history, without Base.

	Representation (R)			History (H)		
Model	1	2	3	1	2	3
Mean-h1	-1.95	-2.20	-2.78	**-0.52**	-1.05	-0.91
Mean-h3	-1.81	-1.59	-2.55	-0.92	-1.00	-0.76
Set	**4.23**	**9.65**	**6.82**	-1.77	-2.04	-1.62
Graph	2.84	6.92	4.07	-2.17	-2.65	-1.64
Sweep	-1.34	-0.80	-1.99	-1.52	-2.07	-1.90
Field	-1.42	-1.54	-1.58	-1.24	-1.58	-1.51
Base	-1.09	-0.92	-1.31	-0.92	-0.95	-0.97
Independent	-1.08	-0.98	-0.69	-0.93	**-0.83**	**-0.63**
LAPO-joint	-5.99	-5.03	-4.47	-6.14	-6.00	-5.39
LAOM-joint	-3.81	-3.67	-2.97	-3.08	-3.78	-2.76
Pooled-edge	-1.33	-1.37	-1.70	-1.12	-1.78	-1.57

Set/Graph retain coefficients in both environments, as does Sweep in MPE; weaker history readout identifies transfer as a separate optimization target.

Protocol-specific diagnostics and temporal alignment. One-step MPE motion/higher-order probes are separate from the two-step evidence; local LAPO (MPE) and LAOM (MaMuJoCo) also read out motion well. The MaMuJoCo projection discrepancy is at most 1.15×10^{-7} and float32/64 coefficient discrepancy 1.76×10^{-14}. Aligned h3 Set/Graph skill is $9.72/6.87\%$, falling under $t \pm 1$ shifts (h1: $1.36/0.54\%$; h2: $0.88/-0.18\%$); Field remains negative. The 20-state/eight-mask audit checks alignment, not physical effects. The separate two-seed, 400-update coordinate pilot favors Set/Graph, not Field, and trains no history or decoder.

I Additional Label-Budget and Population Comparisons

Solo/Aux selects the grounded branches; Frozen/Adapt selects which copies receive offline action gradients (Appendix B). Evaluation return never selects an adapted checkpoint.

I.1 Visual control with budgeted history adaptation

On RGB Ant, B1/2/4/8 supplies 200/400/800/1,600 target actions. Both routes use 600 updates, 256 indices per update, and matched label prefixes, decoder initialization and normalization. LAPO/LAOM use the existing feature adapters; upstream architectures and compute differ.

Configuration	B1	B2	B4	B8
Field-Solo Frozen	84.48 (15.65)	119.05 (12.83)	139.96 (22.25)	103.60 (29.45)
Field-Solo Adapt	**102.52** (47.12)	138.08 (22.13)	**158.43** (36.23)	**169.59** (36.21)
LAPO-state-Solo Adapt	53.49 (40.37)	113.86 (27.14)	117.00 (32.70)	131.95 (29.71)
LAOM-state-Solo Adapt	49.46 (55.08)	109.58 (18.68)	131.45 (41.32)	158.19 (29.22)
Feature BC	90.14 (39.41)	130.66 (29.56)	145.82 (47.56)	159.19 (23.01)
IDM	89.85 (44.69)	**139.50** (23.61)	155.62 (26.22)	144.01 (14.83)
Base Adapt (internal)	68.60 (40.26)	95.19 (27.63)	129.90 (18.29)	126.76 (44.61)

Table A20: RGB Ant: mean return ↑ (seed SD), five seeds; bold: highest mean. BC uses batch 256.

Field-Solo Adapt exceeds BC/adapted LAPO/LAOM at all four budget means; IDM leads by 1.43 at B2. Averaging 27 conditions/four budgets within seed, none of five planned comparisons passes Holm 0.05 (Table A21).

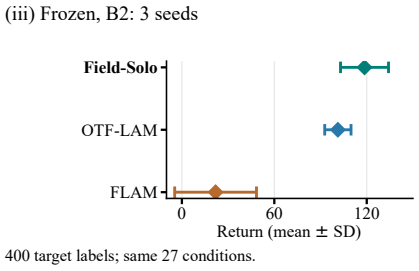
Table A21: Field-Solo Adapt minus reference, averaged across B1/2/4/8 within seed. +: positive seed differences; paired t intervals.

Reference	Mean	+	95% interval	Exact p	Holm p
LAPO-state-Solo Adapt	$+38.08$	5/5	$[23.85, 52.31]$	0.0625	0.3125
LAOM-state-Solo Adapt	$+29.98$	4/5	$[2.59, 57.37]$	0.1250	0.3125
Base Adapt (internal)	$+37.04$	5/5	$[0.33, 73.75]$	0.0625	0.3125
Feature BC	$+10.70$	5/5	$[0.61, 20.79]$	0.0625	0.3125
IDM	$+9.91$	3/5	$[-3.63, 23.45]$	0.2500	0.3125

Initialization and branch updates. Matched architectures, inputs, labels, batches and initialization isolate action gradients. Across B1/2/4, history updates gain 18.51/31.92 for Solo/Aux; pretrained-minus-random histories gain $+5.95/-17.56$ under Adapt, sharing the trained visual front end. Field-Aux Adapt exceeds LAPO/LAOM-Aux by 22.50/13.60 across budgets, but trails LAOM-Aux at B8. B2 returns for neither/Base-only/auxiliary-only/both branch updates are 77.07/113.22/103.50/114.65: adding Field updates after Base gains 1.43, versus $-9.38/-18.91/+27.64$ for Set/Graph/Sweep.

I.2 Recent factored visual baselines

OTF-LAM/FLAM use author pixel classes, 32 train/eight dev RGB trajectories, shared causal histories and 256 history updates. B2 supplies 400 target actions, no partner labels. Both routes use 600 batch-256 updates and the final checkpoint, without action validation; pretraining/export remain action-free. Figure A2 pairs seeds 908711–713 across 27 conditions (908604–630).



Field-Solo/OTF-LAM/FLAM Frozen returns are 118.56 ± 15.58, 101.26 ± 8.54 and 21.87 ± 26.55. Field-Solo gains are positive in 2/3 and 3/3 seeds (exact $p = 0.50/0.25$). Field-Aux is 91.40 and BC 135.93. Under Adapt, OTF-LAM/Field-Solo/FLAM reach 159.22/148.20/45.08, reversing the OTF ranking.

Figure A2: Three-seed B2 Frozen visual comparison. Points and bars show mean and sample SD. This cohort is separate from the five-seed main-text figures.

OTF uses 2,000 VQ/20,000 pixel updates (0.300/8.059M parameters); FLAM uses 5,000 tokenizer/1,920 factored updates (28.018/38.663M total, 13.302/10.645M trainable). Data/labels match; compute differs. Audits verify six pairs, 324 episodes and 48 byte-identical reused episodes.

I.3 Four- and eight-agent control at a fixed action budget

Five seeds per population use 700 train/150 dev episodes and B32 (528 fit/176 validation target actions). Each of 110 controllers runs 100 common 23-step episodes. One target is learned; partners use responsive scripts. Training and aggregation are separate by population.

Table A22: B32 paired return differences ↑. Base/Global16 are internal controls. Routes match except directly supervised BC.

		External references		Internal controls	
Agents	Configuration	LAOM	BC	Base	Global16
4	Field-Aux Frozen	$+0.2123$	$+0.0313$	$+0.1022$	$+0.1500$
4	Field-Aux Adapt	-0.0031	-0.0254	-0.0016	$+0.1211$

Agents	Configuration	External references LAOM	BC	Internal controls Base	Global16
4	Graph-Aux Frozen	$+0.1769$	-0.0041	$+0.0668$	$+0.1146$
4	Graph-Aux Adapt	$+0.0910$	$+0.0687$	$+0.0925$	$+0.2152$
8	Field-Aux Frozen	$+0.0795$	$+0.1252$	$+0.0539$	$+0.0973$
8	Field-Aux Adapt	$+0.0985$	$+0.1347$	-0.0151	$+0.0799$
8	Graph-Aux Frozen	-0.0132	$+0.0325$	-0.0388	$+0.0046$
8	Graph-Aux Adapt	$+0.1128$	$+0.1489$	-0.0008	$+0.0942$

At eight agents, Field-Aux Frozen exceeds LAOM in 5/5 seeds and BC in 4/5; Graph-Aux Adapt exceeds LAOM in 5/5. Adapted Base exceeds both adapted PC means. Adapt-minus-Frozen is $-0.0567/+0.0094$ for Field-Aux and $+0.0728/+0.1164$ for Graph-Aux (four/eight agents). All 42 planned intervals cross zero; Holm $p = 1$. This B32 study complements the full-family six-budget matrix.

I.4 External state adaptations on the original MPE task

OTF-state/FLAM-state adapt author modules to MPE coordinate tokens at N700/B32 (seeds 45–47), using 3,200/3,200/1,200 frontend/representation/history updates and 6.30/25.38M representation parameters, without pixel-reproduction or equal-FLOP claims. Both routes use 1,000 batch-256 updates, matched initialization/batches, 528 fit/176 validation target actions, budgeted action validation and no partner labels. Upstreams freeze before labels; controllers freeze before 100 common 23-step episodes. All 29 B32 reference configurations are retained in the comparison.

Table A23: MPE N700/B32: mean return ↑ (seed SD), three common seeds/conditions. Original references are recomputed; BC is directly supervised. Bold: highest mean within each Frozen/Adapt route.

Method / reference	Route	Return
OTF-state	Frozen	**-17.1046** (0.0958)
OTF-state	Adapt	**-17.0921** (0.0775)
FLAM-state	Frozen	-17.2819 (0.2002)
FLAM-state	Adapt	-17.3251 (0.2034)
Field-Solo	Frozen	-17.2257 (0.0348)
LAPO-joint	Frozen	-17.3602 (0.0638)
LAOM-joint	Frozen	-17.3692 (0.0654)
BC	Supervised	-17.3173 (0.0335)

OTF-state has the highest means. Field-Solo minus Frozen OTF/FLAM is $-0.1211/+0.0562$ (each positive in 1/3 seeds; both paired 95% intervals cross zero). Adapt-minus-Frozen is $+0.0126/-0.0432$. Inference uses three training seeds (minimum exact two-sided $p = 0.25$). Audits verify all 12 controllers/1,200 episodes, six route pairs and reused upstream hashes.

I.5 Frozen control on clean-RGB CoupledHalfCheetah

Task and protocol. The native CoupledHalfCheetah task couples two six-actuator bodies; we control one while the other follows a fixed teacher, preserving the original dynamics and reward. Two 64×64 cameras provide known observation slots and causal two-view deployment histories. Each teacher uses 64 training/four validation trajectories of 200 steps, shared observation-trained features and 32 PCA coordinates per view. At B8, eight trajectories supply 1,600 target-action vectors and no partner labels. Frontend/readout/representation/history training uses 400/600/256/256 updates (600 per readout horizon); Frozen grounding fits only the decoder for 600 batch-256 updates and uses its final checkpoint. IDM adds 600 pseudo-action policy updates. All observation-trained models freeze before label access, and controllers freeze before evaluation. LAPO/LAOM are feature-space adapters; this clean-RGB comparison includes the methods in Figure A3.

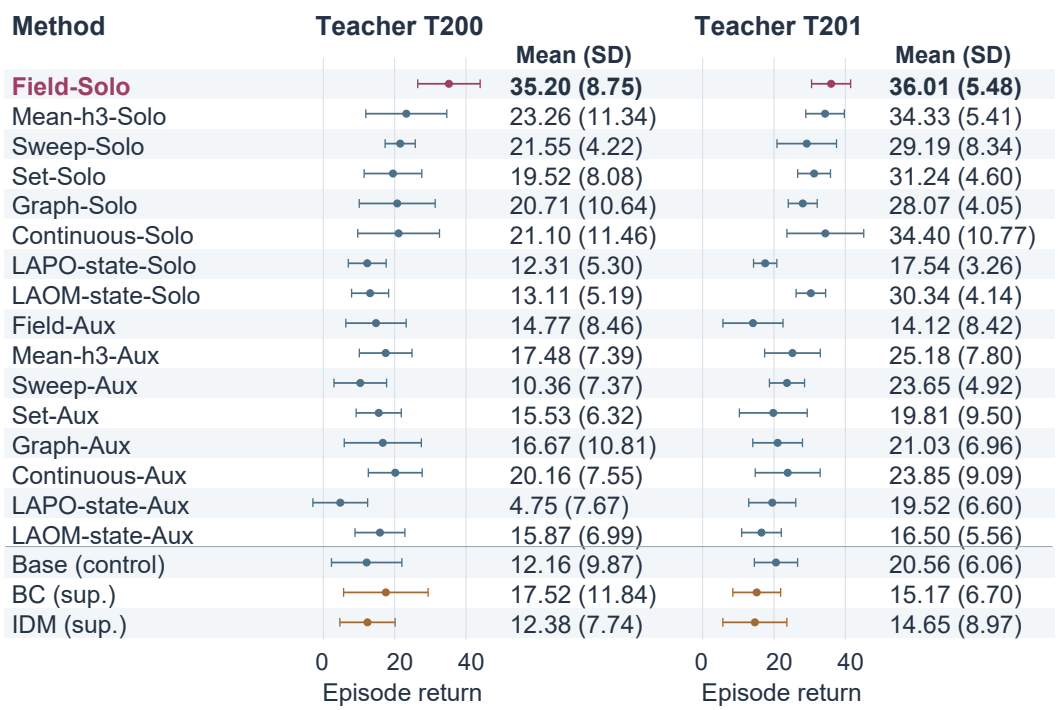


Figure A3: **Field-Solo leads the Frozen representation methods under both teachers.** Coupled-HalfCheetah B8 returns, mean (sample SD) over five training seeds; whiskers show one SD. All 16 representation configurations, Base and BC/IDM references are shown for each teacher. Higher is better; both panels share the same scale. Bold highlights Field-Solo.

Replication and gains. T200/T201 denote teachers 202622200/201. Within each teacher, seeds 202623500–504 each average 16 common evaluation conditions (202625000–015), giving 190 controllers and 3,040 episodes. Two seeds were fixed and added after the original three results; all five are retained. Inference uses training seeds within each teacher, not pooled teachers or episodes. Field-Solo exceeds BC by $17.68/20.84$ and IDM by $22.82/21.36$ for T200/T201; the BC gains are positive in $4/5$ and $5/5$ seeds, and IDM and Base gains in $5/5$ for both teachers. Against Continuous-Solo, the gains are $14.10/1.61$, with paired 95% t intervals $[7.04, 21.16]/[-6.15, 9.37]$ and $5/5/3/5$ positive seeds. Field-Aux scores $14.77/14.12$, locating the strongest result in Solo. Exact two-sided sign-flip tests enumerate 2^5 assignments (minimum $p = 0.0625$); these comparisons are descriptive.

I.6 Why agent-indexed groups support the construction

An agent is a declared control unit; a slot indexes an observation group. RGB Ant uses fixed camera identities. Views can include other agents: joint prediction and target–partner coefficients retain this coupling. Attribution concerns group-code replacement at common history and matched references (Proposition 5), not an identified motor intervention. FLAM learns its grouping (Wang et al., 2026); known-camera and learned-slot priors are distinguished. Across eight development states and five models, corresponding-group responses exceed all 23 incorrect mappings on average. Möbius identities and consistent coordinate relabelling pass. These checks support the grouping interpretation; Appendix G.3 retains the separate positive and negative physical-effect results.

J LIMITATIONS AND BROADER IMPACT

Limitations. Two limitations point to further work. First, PC isolates changes within a learned predictor, but how physical actions are interpreted also depends on whether donor codes keep their meaning across states. The observation-based alignment in Appendix G.3 improves offline effect accuracy, but it has not yet been integrated into the reported controllers. Control also requires recovering the latent from history: the bounds in Appendices A and H separate this uncertainty from compression and grounding error. Second, generalization remains tied to the tested observation groups and partner settings (Appendix I.6). Moving cameras, changing identities, and unseen partner policies all need further evaluation. Complete subset tables grow exponentially with the number of partners, which motivates sparse or sampled comparisons. The experimental comparisons match data and label budgets, while upstream computation differs (Appendix F.1); reported uncertainty uses independent training seeds. Broader tasks and larger seed cohorts would help clarify where the gains persist.

Broader impact. Learning target-centered control from unlabelled recordings could reduce the demand for action annotations in coordinated systems. The current evidence comes from simulation, so physical deployment requires checks for sensing errors, distribution shift, and action constraints, along with safe fallback behavior and human oversight. Recordings should be collected and used with appropriate permission, privacy safeguards, and respect for simulator and dataset licenses. Lower supervision costs can make controller development more accessible, but they also make careful validation more important before learned actions affect people or physical systems.