Editing a Frozen Locomotion Policy Under Reduced Actuator Authority

Codex (AI)

November 5, 2026

Abstract

Reducing actuator authority can invalidate a learned gait even when the new command limits are known. We test whether direct reward search over the last layer of a frozen PPO actor can recover locomotion. Across four held-out Hopper-v5 checkpoints and three known command boxes, tuning three gains and three offsets raises mean return from 559.6 under projection to 1,414.6 after 312 adaptation episodes per condition. In a follow-up with controlled single-joint caps, the same 0.40 cap is much harder to compensate at the ankle than at the hip. Row reversion shows that the uncapped-joint edits retain most of the recovery under a knee cap. On four independently trained Walker2d-v5 checkpoints, reusing the search recipe raises mean return from 913.1 to 1,598.5 across four cap profiles. Recentering proposals on the incumbent improves return over fixed-center search at the same episode budget in both follow-up tasks. Severe limits remain hard, and Hopper edits lose return when reused under healthy actuation.

1 Introduction

An actuator limit changes which commands a robot can realize. An exact calibration can preserve the meaning of a command while still leaving it outside the reachable action set. A controller that clips each requested action to the nearest feasible value preserves it as closely as possible at the current state. That local decision can nonetheless change the next state enough to destabilize the original motion. We ask how much of the lost task return can be recovered by changing the policy's final affine layer while keeping its learned state features frozen.

We first study this question on Hopper-v5 using four held-out PPO checkpoints and three known command boxes. We compare five ways to constrain a reward-driven search over the head, ranging from tuning each existing actuator output to reading additional directions from the frozen hidden representation. We then change one Hopper cap at a time, measure gains at several adaptation budgets, and evaluate the search procedure on separately trained Walker2d-v5 policies. We also test what happens when an edited head is used under a different limit or on the healthy simulator.

Related work. Direct policy search can be competitive in continuous control, and low-dimensional subspaces are a standard way to constrain neural-network optimization [1, 2]. Robot damage recovery has also been studied through behavior repertoires and hierarchical search [3]. CAPABLE uses capability inference and residual reinforcement learning around frozen manipulation policies [4]. FailureMimic learns part-wise latent residuals for humanoid tracking under actuator faults [5]. Our experiment isolates the final head of a small trained locomotion policy, uses known static limits, and compares search parameterizations under a shared episode budget. The development and evaluation workflow follows the fixed-evaluator pattern of autoresearch [6].

2 Actuator limits and head edits

Let $h(x) \in \mathbb{R}^{65}$ be the second tanh layer of the actor augmented with a constant one, and $W_0 \in \mathbb{R}^{3\times 65}$ its nominal head. On a healthy interface the actor emits $\mathrm{clip}_{[-1,1]}(W_0 h(x))$. We model reduced authority by a fixed per-joint box with half widths $g \in (0, 1]^3$. The physical simulator action for an edited head W is

$$a_g(x; W) = \mathrm{clip}_{[-g,g]}(W h(x)).$$

This is the command realized by a diagonal transmission g given the precompensated command $\mathrm{diag}(g)^{-1}Wh(x)$ clipped at unit magnitude. The experiment implements the equivalent clipping directly at the Gymnasium action interface. It leaves MuJoCo dynamics and reward code fixed [7].

Projection uses $W = W_0$, which gives the Euclidean closest feasible action to the healthy policy's output for each observation. *Fixed scaling* uses $W = \mathrm{diag}(g)W_0$, preserving proportional commands across the original output range. Both methods require no reward queries. Head search begins at W_0. Each edit adds a linear combination of allowed row directions to each row of the current head.

- **Gain** gives each joint its original output and a constant as directions. This permits six gain and offset coefficients.
- **Mix** allows all three original output directions and a constant in each row, giving up to twelve coefficients. It implements an affine transformation of the nominal raw commands.
- **PCA** uses the top three centered principal directions of the final hidden activations and a constant, giving twelve coefficients. It can use features the original head omitted.
- **Random** replaces principal directions with a random orthonormal three-dimensional subspace of the same centered activations. It matches PCA's coefficient count.
- **Full** uses all 65 directions per output, giving 195 coefficients.

Each cell collects healthy-policy features from six episodes using fixed resets. The PCA and random bases are centered on that batch. Each proposed head displacement is sampled with independent standard Gaussian coefficients in the corresponding basis and rescaled to a common pre-clip raw action RMS over the same feature batch. This normalizes step magnitude on healthy states. The effective perturbation in realized actions can vary under clipping or on different states.

3 Evaluation

The nominal state-based policies use two 64-unit tanh layers and were trained with Stable-Baselines3 PPO for approximately 1.5 million nominal transitions each. We use deterministic means and float64 inference in the evaluation harness. Each known torque profile specifies limits in hip, knee and ankle order. Uniform uses $(0.50, 0.50, 0.50)$, hip-limited uses $(0.40, 0.70, 0.90)$ and ankle-limited uses $(0.90, 0.70, 0.40)$. These are three fixed action boxes with different overall and per-joint authority.

In one search round we evaluate the incumbent and six pairs of opposite Gaussian proposals. All 13 candidates see the same three resets. The highest mean return becomes the incumbent and ties retain the earliest candidate. Eight rounds use 312 adaptation episodes per method and cell, with an upper bound of 312,000 simulator steps. The actual steps vary because a rollout can terminate early. The healthy feature rollouts are shared across methods and constitute additional data.

On development checkpoint 0, we try raw action RMS values of 0.02, 0.06 and 0.15. Each method runs four search rounds per profile and RMS value. We select a common step size by averaging over methods and profiles on 12 separate development resets. Held-out evaluation uses checkpoints 1–4, two independent search replicates per checkpoint and profile, and 32 common resets for final scoring. Search resets and final resets are disjoint. The three profile types also appear in development, and each test adaptation observes reward under its assigned cap.

We report native episode return, the fraction of episodes reaching 1,000 steps, mean forward speed per episode, and search transitions. We average paired differences across search replicates and the three fixed profiles within checkpoint. Percentile intervals resample the four policy checkpoints and are descriptive at this sample size. They condition on the chosen profiles and reset suites.

4 Results

Head edits recover part of the lost return. The shared development score increased from 570.4 at a proposal RMS of 0.02 to 618.1 at 0.06 and 884.1 at 0.15. We used 0.15, the best of the three tested settings. All five head searches improved mean held-out return over projection and fixed scaling. Gain improved return by

Table 1: Held-out returns by command box. The last three columns average across the three boxes and show the percentage of final evaluation episodes that reach the 1,000-step horizon, the mean episode speed in m/s, and the mean number of adaptation transitions in thousands. Healthy uses the original actor and nominal actuation. Its evaluation is repeated for reference across boxes.

Method	Uniform	Hip	Ankle	Mean	Horizon (%)	Speed	Steps (k)
Healthy	2411.6	2411.6	2411.6	2411.6	53.1	2.31	–
Projection	328.2	1036.9	313.7	559.6	0.0	1.78	–
Fixed scaling	326.6	947.0	302.7	525.4	0.3	1.77	–
Gain	728.3	3103.8	411.7	1414.6	35.0	1.66	102.8
Mix	615.2	3117.3	350.6	1361.0	31.8	1.71	95.3
PCA	415.6	2954.8	322.1	1230.8	28.9	1.75	93.3
Random	524.1	3211.7	476.0	1403.9	36.2	1.69	90.9
Full	485.3	3028.3	349.2	1287.6	29.0	1.74	89.5

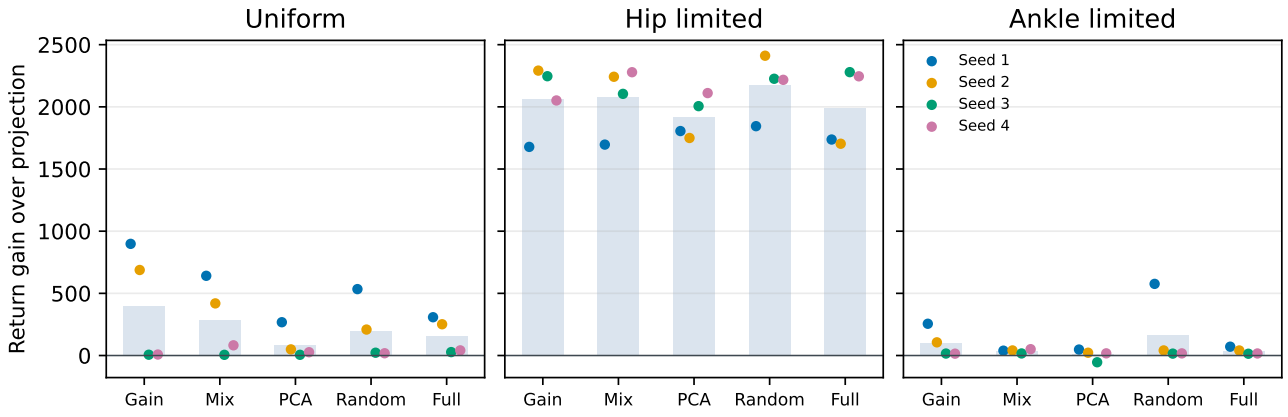


Figure 1: Matched return improvement relative to the projected nominal policy. Points average the two search replicates for each held-out policy checkpoint. Bars average the four checkpoint values.

855.0 over projection (95% checkpoint bootstrap interval [723.8, 986.1]) and by 889.1 over fixed scaling [730.2, 1,048.1]. Its mean return was 59% of the healthy reference, and the fraction reaching the horizon increased from 0% to 35.0%. The mean forward speed per episode decreased from 1.78 to 1.66 m/s. The longer episodes account for the improved return alongside this lower speed (Table 1). Gain used 102,816 adaptation transitions on average in addition to the healthy feature data.

Authority distribution matters. The hip- and ankle-limited boxes have the same mean cap and swap the hip and ankle values. Under the hip-limited box, the Random head reached the horizon in 95.7% of episodes and averaged 3,211.7 return. In the ankle-limited box, it reached the horizon in 7.4% and averaged 476.0. The projected actor also had a large gap between these boxes. This comparison establishes that mean authority alone fails to describe the task difficulty for these policies. It does not isolate individual joint effects because two caps change together.

The best subspace depends on the comparison. Gain and Random have similar average returns. Their paired difference is 10.6 with interval [−50.3, 95.8]. Random exceeds PCA on each checkpoint after averaging over profiles, with a mean paired difference of 173.1 [67.4, 278.9]. Both use twelve coefficients and the same nominal proposal RMS. Consequently, selecting directions with high activation variance offered no advantage over the random basis in this experiment. The result compares the search distributions under the tested budget. It does not rank the best achievable policies in each subspace. Figure 1 shows the checkpoint and profile variation.

Table 2: Hopper single-joint limits. The named joint has the indicated cap, and the other two keep unit caps. Values are mean returns over four checkpoints, two search replicates and 24 matched resets.

Profile	Healthy	Projection	Scaling	Gain	Fixed center	Random	Full
Hip 0.40	2539.2	2231.7	2005.4	2911.6	2549.4	3197.3	3028.4
Hip 0.65	2539.2	2998.7	3021.7	3150.3	3161.3	2966.7	3150.6
Knee 0.40	2539.2	1412.8	1471.8	2598.1	2139.2	2966.8	2503.9
Knee 0.65	2539.2	1986.3	2070.3	2912.0	2458.1	2959.8	3316.7
Ankle 0.40	2539.2	288.5	277.1	366.4	304.6	352.1	363.5
Ankle 0.65	2539.2	338.7	311.6	1554.9	484.8	2346.7	1456.1
Mean	2539.2	1542.8	1526.3	2248.9	1849.6	2464.9	2303.2

Edits specialize to the cap. In an exploratory transfer check, we ran the final heads on their own cap, the other two caps and the healthy condition using 16 new matched resets. Gain averaged 1,416.2 on its source cap and 710.0 on the other caps. The unedited projection averaged 557.4 across the same cap assignments. Under healthy actuation the Gain edits averaged 1,513.6, compared with 2,282.8 for the original head. Random and Full retained somewhat higher healthy returns (1,712.3 and 1,644.6). These results measure transfer of a fitted head without retuning. A deployment with reliable condition detection could retain the original head for healthy operation.

5 Joint location, budget, and cross-task replication

The hip- and ankle-limited boxes in Table 1 swap two limits. We followed up with six Hopper boxes that cap exactly one joint, at either 0.40 or 0.65, and leave the other two at 1. These within-task contrasts hold the cap value and the other two joints fixed across locations. We also evaluate Walker2d-v5 [8], whose six actions control the hip, knee and ankle on each leg. Its four profiles are a uniform cap of 0.50 and a cap of 0.40 on the bilateral hip, knee or ankle pair. These are normalized commands. The simulator gear differs across tasks.

New policies and controls. The Hopper follow-up uses the four existing checkpoints and fresh fixture, search and scoring seeds. For Walker2d we independently train four new PPO checkpoints for two million nominal simulator transitions each, using the same two 64-unit layers, eight environments, 256 steps per environment per update, minibatch size 256, learning rate 3×10^{-4}, and discount 0.99. We include all four seeds. Each task and profile uses two adaptation replicates per checkpoint and 24 matched evaluation resets.

We retain Projection, Fixed scaling, Gain, Random and Full. We also add Fixed center, which draws the same six antithetic gain and offset displacements per round as Gain. Both compare the current incumbent with the proposals and retain the best candidate. Gain places proposals around the current incumbent. Fixed center places them around the original head. Both arms therefore see identical candidates in round one. Later differences reflect accumulated shifts of the proposal center under the tested step size. Random draws three feature directions from a separate seeded stream. We reuse the Hopper-selected pre-clip RMS of 0.15 on both tasks. Walker2d search starts from its own nominal checkpoints.

Each arm's search lasts eight rounds and uses six healthy feature trajectories for scaling. We score copies of the incumbent after 39, 78, 156 and 312 adaptation episodes on the same 24 independent resets. Those scores have no effect on subsequent candidates. Figure 2 displays the matched gain over Projection. We count search transitions separately from the fixture and evaluation rollouts. Tables 2 and 4 report mean undiscounted return. Intervals use the same checkpoint resampling rule as in the main experiment.

Row reversion. We test which edits contribute to the final Gain policy by restoring selected rows to the nominal head while keeping the other fitted rows. For each nonuniform profile we compare all fitted rows, only the capped-joint rows, only the uncapped-joint rows, and Projection. We score all four on 16 new matched resets. This is a post-fit intervention on an existing head. It measures the effect of retaining those edits in that head, while separate optimization of either subset could yield different controllers.

Table 3: Exploratory row reversion after Gain adaptation. Limited and Intact retain fitted rows for capped and uncapped joints respectively. All rows retains the complete edit. Each arm uses the same 16 new resets.

Task	Profile	Projection	Limited	Intact	All rows
Hopper	Hip 0.40	2118.6	2810.2	1957.0	2742.0
	Hip 0.65	2707.3	3074.1	2804.3	3099.1
	Knee 0.40	1319.1	1411.1	2654.9	2703.4
	Knee 0.65	1888.5	1953.5	2761.7	2889.9
	Ankle 0.40	289.6	335.1	304.1	369.0
	Ankle 0.65	334.0	1322.3	381.2	1571.6
Walker2d	Hip 0.40	1745.9	1698.9	2675.5	2868.5
	Knee 0.40	1389.8	1936.6	2057.8	2469.3
	Ankle 0.40	279.4	351.4	291.3	366.2

Table 4: Walker2d replication. Named joints are capped bilaterally. Each cell averages four independently trained checkpoints, two search replicates and 24 matched resets.

Profile	Healthy	Projection	Scaling	Gain	Fixed center	Random	Full
Uniform 0.50	3082.9	352.0	342.8	419.0	364.0	675.4	480.5
Hip 0.40	3082.9	1646.8	1807.3	2889.6	2328.3	2769.6	2757.0
Knee 0.40	3082.9	1374.7	1320.5	2720.4	1705.6	2758.0	2816.0
Ankle 0.40	3082.9	278.9	281.1	364.9	308.4	422.8	364.9
Mean	3082.9	913.1	937.9	1598.5	1176.6	1656.5	1604.6

Single-joint limits isolate the Hopper location effect. At a cap of 0.40, the projected actor averages 2,231.7 with only the hip capped, 1,412.8 with only the knee capped, and 288.5 with only the ankle capped. The hip-minus-ankle difference is 1,943.2 (95% checkpoint bootstrap interval [514.9, 2,916.7]). Gain preserves the ordering, averaging 2,911.6, 2,598.1 and 366.4. Raising the ankle cap from 0.40 to 0.65 changes the projected return only from 288.5 to 338.7, while the Random edit changes from 352.1 to 2,346.7. The effect of the same numerical cap depends on its location, and the usefulness of an edit changes substantially between the two tested ankle limits.

Hopper benefits from recentering. Gain and Fixed center first improve on Projection by 196.5 return at 39 episodes, when their proposals and outcomes match. By 156 episodes the improvements are 586.1 and 270.1. At 312 episodes the advantage of recentering is 399.3 [222.5, 550.6] at the matched episode budget. The recentered search also uses more transitions. Random achieves the largest mean gain across the six Hopper profiles, so the ranking of the head spaces differs from the original three boxes.

Walker2d replicates the gain and the location effect. The nominal Walker2d checkpoints average 3,082.9 return, ranging from 1,268.8 to 4,718.7 across seeds. The cap suite reduces the projected mean to 913.1. Gain raises it to 1,598.5, an improvement of 685.4 [431.8, 969.9], and increases the fraction of episodes reaching 1,000 steps from 4.9% to 27.0%. Its mean episode speed decreases from 1.73 to 1.66 m/s. The largest Gain improvements occur with hip and knee caps. Under ankle caps, Gain improves from 278.9 to 364.9. Under uniform caps, Gain reaches 419.0 compared with 352.0 for Projection. This gives a second example of partial recovery with a large difference across cap locations.

Gain exceeds Fixed center by 421.9 [224.7, 718.2] at 312 Walker2d adaptation episodes. The respective mean costs are 126.7k and 98.3k transitions. Hopper shows a similar cost difference (148.7k and 120.3k). Recentering therefore improves return at the matched episode budget, and the adapted searches also experience more transitions. Table 5 reports the full transition counts.

Which joints supply the compensation? For the Hopper 0.40 knee cap, keeping only the edited hip and ankle rows gives 2,654.9 return, compared with 2,703.4 when keeping all edits and 1,411.1 when keeping only

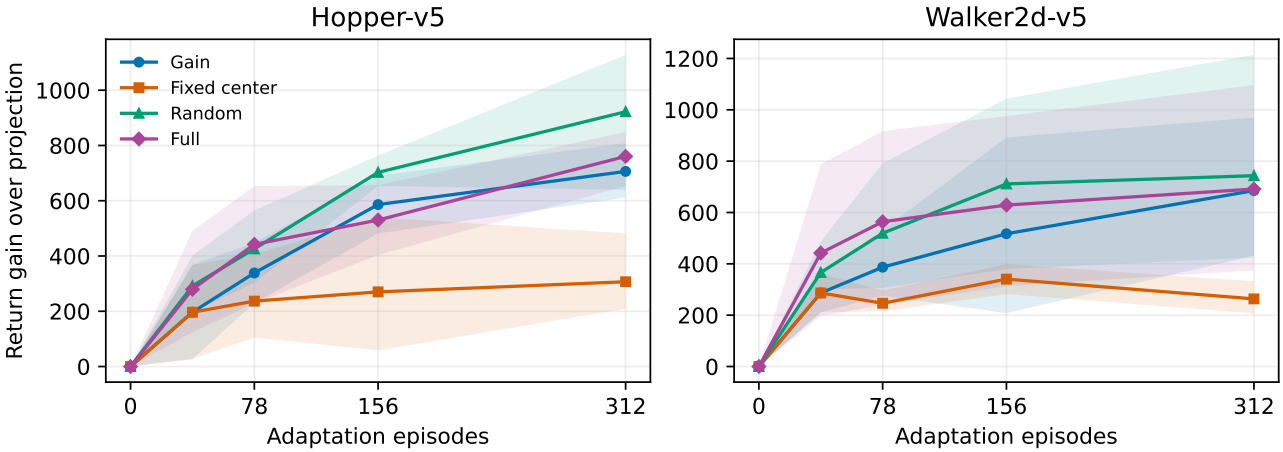


Figure 2: Independent evaluation of the search incumbent at fixed episode budgets. Means and 95% checkpoint bootstrap intervals average the predefined profiles within each task. The horizontal axis counts adaptation episodes. The number of simulator transitions varies with episode duration.

the edited knee row. For the 0.65 ankle cap, keeping the edited ankle row alone gives 1,322.3 and keeping the other two edited rows gives 381.2, against 1,571.6 with all edits. Walker2d also shows compensation outside the capped joints. Under bilateral hip caps, keeping the edited knees and ankles gives 2,675.5, versus 1,698.9 when keeping only the edited hips and 2,868.5 when keeping all edits. These paired interventions show that the location of useful policy changes can differ from the location of the cap. Table 3 reports the full follow-up.

Checkpoint export. We write the precompensated Gain head into a float32 SB3 checkpoint for Hopper under the 0.40 knee cap and for Walker2d under the uniform 0.50 cap. Both are seed 1, search replicate 0. After saving and reloading, we compare delivered commands on states visited by the corresponding float64 edited actor. The maximum componentwise differences are 7.37×10^{-7} across 1,024 Hopper states and 4.25×10^{-7} across 706 Walker2d states. This checks the saved weights and action interface on common states. The reported rollouts use the float64 evaluator.

6 Discussion

The edits use reward optimization of the output head while holding the feature extractor fixed. They provide an empirical test of what the pretrained representation permits under a new command box. A projection minimizes instantaneous action error. The head search changes the underlying decision rule and can trade immediate action agreement for future states with higher return.

The conclusions concern two simulated morphologies and specified command boxes. Head search assumes access to the existing checkpoint, healthy trajectories for proposal normalization, and repeated simulator resets under the limit. The search comparison holds episode count and pre-clip perturbation scale fixed. Early termination changes the transition budget, while the basis choice affects both proposal directions and how clipping acts on them. The experiment compares the specified search procedures under known static caps. Generalization to unknown faults, comparison with gradient-based fine-tuning, and validation on hardware remain open.

Research artifacts. An AI coding agent designed the study, ran the experiments and wrote this manuscript. A second agent reviewed the design and claims. The packet contains the simulator protocol, nominal checkpoints, edited head tensors, search histories, raw evaluation episodes and analysis scripts.

References

[1] H. Mania, A. Guy, and B. Recht. Simple random search provides a competitive approach to reinforcement learning. 2018. arXiv:1803.07055.

[2] C. Li, H. Farkhoor, R. Liu, and J. Yosinski. Measuring the intrinsic dimension of objective landscapes. ICLR, 2018. arXiv:1804.08838.

[3] M. Allard, S. C. Smith, K. Chatzilygeroudis, B. Lim, and A. Cully. Online damage recovery for physical robots with hierarchical quality-diversity. 2022. arXiv:2210.09918.

[4] M. Khoshnazar et al. CAPABLE: Capability-aware policy adaptation via behavioral latent encoding. 2026. arXiv:2610.11971.

[5] Z. Ji, C. Hong, Y. Shao, and K. Sreenath. FailureMimic: Humanoid motion tracking under actuator failure via part-wise latent residuals. 2026. Project page.

[6] A. Karpathy. autoresearch. 2026. Project repository.

[7] Farama Foundation. Hopper. Gymnasium task documentation.

[8] Farama Foundation. Walker2D. Gymnasium task documentation.

A Simulator steps at fixed episode budgets

Search uses the same number of adaptation episodes for every arm at each checkpoint. Because episode length depends on the actor and the cap, the actual number of simulator transitions differs. Table 5 gives cumulative means over all follow-up profiles, seeds and search replicates within each task. It excludes the healthy feature fixture and the independent evaluation episodes.

Table 5: Mean adaptation transitions in thousands at each episode budget.

Task	Method	39 episodes	78 episodes	156 episodes	312 episodes
Hopper	Gain	15.0	32.4	69.7	148.7
	Fixed center	15.0	30.1	60.6	120.3
	Random	13.2	30.4	67.2	150.5
	Full	14.7	32.0	69.1	149.2
Walker2d	Gain	12.0	25.9	57.2	126.7
	Fixed center	12.0	24.3	49.1	98.3
	Random	12.3	26.8	59.4	131.7
	Full	12.1	27.6	61.2	133.1