SC-PPO: SECANT-CORRECTED PROXIMAL POLICY OPTIMIZATION FOR EXECUTION-ROBUST NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper investigates execution-robust navigation in confined spaces, where a learned policy must remain collision-free even when its actions are executed with small errors. While canonical reinforcement learning (RL) algorithms such as proximal policy optimization (PPO) target the maximal expected return under exact execution, we argue that a deployed robot should trade a small amount of nominal return for insensitivity to execution errors, since limited clearance and closed-loop feedback turn these errors into collisions. However, the mismatch between the parameter space where PPO updates the policy and the action space where execution errors enter makes this goal difficult to pursue: any action sensitivity in the kernel of the policy Jacobian is invisible to the parameter gradient, so a policy can be flat in parameter space yet sharp in action space. To address the challenge, we propose a model-free RL algorithm, namely secant-correction PPO (SC-PPO), which compares two gradient evaluations within PPO's frozen evaluation context and uses their difference to correct the policy update without any disturbance model. Its action-space correction operates directly where execution errors enter and yields an action-risk bound under a measurable critic-fidelity condition, and the same operator supplies a gap-based criterion that replaces KL-divergence early stopping. SC-PPO is compared with PPO and flatness-based baselines on the BARN benchmark under persistent-bias and random-noise execution faults, and achieves the highest nominal success rate and the highest average success rate under faults. We further demonstrate the trained policy on a full-size humanoid robot.

1 INTRODUCTION

Autonomous mobile robots are increasingly deployed in cluttered spaces: warehouses with narrow aisles, corridors shared with pedestrians, and densely vegetated outdoor trails. Reinforcement learning (RL) has become a popular way to train navigation policies for such spaces, and learned policies now achieve high success rates on challenging simulated benchmarks (Perille et al., 2020; Xiao et al., 2022). However, anyone who has deployed such a policy on a real robot has likely seen it scrape a shelf or get stuck at a doorway it passed easily in simulation. The cause is often not a wrong decision but an imperfect execution: wheel slip, actuator lag, or uneven ground makes the executed velocity differ slightly from the commanded one, and in a passage only slightly wider than the robot, closed-loop feedback can amplify this small difference into a collision (Figure 1). As a result, robust navigation through confined spaces has emerged as an important research topic and has attracted increasing attention from both the robotics and RL communities.

While a detailed review of related work will be presented in Section 2, here, we wish to note that most existing learning-based navigation methods focus solely on maximizing the expected return, assuming that every selected action is executed exactly as commanded, and overlook how sensitive this return is to small execution errors. A policy trained in this way may clear a narrow gap by only a few centimeters, with no incentive to prefer a trajectory that keeps its margin when the executed action deviates from the commanded one. In practice, a deployed robot is better served by a policy with a slightly lower nominal return whose performance degrades gracefully under such deviations. Common ways of adding caution do not provide this. A fixed safety margin around obstacles, as

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.

used by classical planners (Lu et al., 2014), cannot be both large enough to absorb the error and small enough to keep passages only slightly wider than the robot open, and the margin actually needed varies with speed, heading, and the closed-loop response. Risk-sensitive objectives such as variance or CVaR (Tamar et al., 2012; Chow et al., 2015) penalize the spread of the return, but under exact execution a policy that clears a gap by centimeters succeeds in every training rollout, leaving nothing to penalize: the fragility lies in the sensitivity of the return to the executed action, not in its distribution. Robust RL methods pursue this goal, but they typically require an explicit model of the disturbance, such as a noise distribution, an uncertainty set, or a trained adversary (Pinto et al., 2017; Tessler et al., 2019), which is rarely available and often inaccurate for a real robot. This motivates the study of model-free execution-tolerant policy optimization, which aims to make a learned policy robust to action perturbations without assuming any model of how these perturbations arise.

However, the gap between the space where the policy is updated and the space where execution errors enter makes this problem difficult to solve within canonical policy gradient methods such as proximal policy optimization (PPO). PPO updates the network parameters, whereas execution errors perturb the actions. The two spaces are linked by the policy Jacobian, and any component of the action-cost gradient that lies in the kernel of this Jacobian is invisible to the parameter gradient. Consequently, a policy can be perfectly flat in parameter space while remaining highly sensitive to action perturbations, so promoting parameter-space flatness (Lee and Yoon, 2025) does not, by itself, guarantee robustness at execution time.

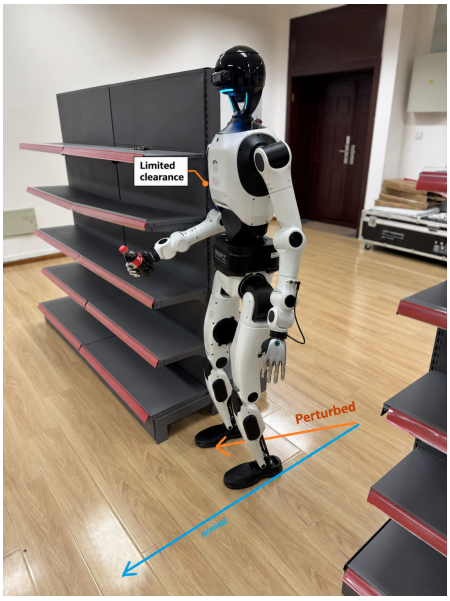


Figure 1: Execution sensitivity near shelving. Blue and orange arrows schematically indicate nominal and perturbed motion; small deviations can exhaust the limited clearance.

In this paper, we address execution tolerance within the PPO framework and propose a model-free algorithm, namely SC-PPO, built on a secant-correction (SC) operator. The SC operator compares two gradient evaluations under a frozen evaluation context and uses their difference to correct the update direction. Since PPO's inner optimization loop reuses a fixed batch of data over several epochs, it naturally provides this frozen context without collecting additional samples. SC-PPO consists of (1) an Action-Space Secant Correction (ASC), which applies the SC operator to a Q-head directly in the action space where execution errors enter, (2) a Parameter-Space Secant Correction (PSC), which applies the SC operator to the clipped surrogate objective, and (3) a Revision-Gap Monitor (RGM), which replaces KL-based early stopping with a quantity measured by the SC operator itself.

The paper's primary contributions can be summarized as follows: (1) We identify a structural obstruction between parameter-space and action-space robustness: action-space flatness always implies parameter-space flatness, whereas the converse degrades as the action-cost gradient approaches the kernel of the policy Jacobian and fails once it lies there (Proposition 1); (2) We propose SC-PPO as a model-free solution, whose correction is derived from a quantity measured during training rather than posited in advance, and which requires no disturbance model; its action-space correction yields an unconditional action-risk bound under a measurable critic-fidelity condition (Theorem 1); (3) We benchmark SC-PPO against PPO and two flatness-based baselines on BARN under four families of execution faults, isolate the contribution of each component through matched ablations, and demonstrate the policy on a full-size humanoid robot.

2 Related Work

To the best of our knowledge, no prior work improves the robustness of PPO to action execution errors without a disturbance model while providing a guarantee that can be checked during training.

Navigation. Classical local planners search dynamically admissible velocity commands (Fox et al., 1997) and obtain clearance by inflating obstacles in a costmap (Lu et al., 2014). Learning-based planners map local observations directly to actions (Tai et al., 2017; Long et al., 2018), combine learned control with sampling-based planning (Faust et al., 2018), or learn from synthetic obstacle configurations (Xiao et al., 2021); the BARN benchmark compares such methods in 300 highly constrained environments (Perille et al., 2020; Xiao et al., 2022). Some account for uncertainty indirectly, through noisy sensing (Faust et al., 2018) or predicted collision risk (Xiao et al., 2021), but none targets errors in executing the command. SC-PPO changes only the policy update and can be combined with any of these pipelines.

Robustness in RL. *Risk-sensitive* objectives penalize the variance (Tamar et al., 2012) or the tail (Chow et al., 2015; 2018) of the return; as discussed in Section 1, they act on the return distribution induced by the training dynamics rather than on the sensitivity to execution errors. *Model-based* robustness makes the perturbation explicit: robust MDPs optimize over an uncertainty set of transitions (Iyengar, 2005; Nilim and El Ghaoui, 2005), adversarial training uses a learned adversary (Pinto et al., 2017), domain randomization samples simulated dynamics (Peng et al., 2018), and perturbation-robust training perturbs the observations (Zhang et al., 2020) or the executed actions (Tessler et al., 2019); the resulting robustness covers only the assumed perturbations. Smoothness regularization (Mysore et al., 2021) penalizes output changes across nearby states, which targets jerky control rather than execution errors. *Flatness-based* methods avoid a model: Lee and Yoon (2025) train PPO with sharpness-aware minimization, arguing that parameter-space flatness implies action robustness, and Rahman et al. (2025) perturb the parameters during PPO optimization. Proposition 1 shows that this link degrades as the action-cost gradient approaches the kernel of the policy Jacobian and cannot be verified during training. SC-PPO needs no disturbance model and acts on action-space sensitivity directly, under a condition that is measurable during training.

Improvements to the PPO update. PPO (Schulman et al., 2017) approximates the trust region of TRPO (Schulman et al., 2015) by clipping. Later work adds rollback and trust-region-triggered clipping (Wang et al., 2020), KL-based early stopping of the inner epochs (Huang et al., 2022), better advantage and critic estimation (Schulman et al., 2016; Cobbe et al., 2021), and tuned implementation choices (Engstrom et al., 2020; Andrychowicz et al., 2021). These improve stability and sample efficiency under the nominal return and do not address execution errors. SC-PPO is complementary: it reuses the fixed-batch inner loop they share, and RGM replaces KL-based early stopping.

3 SC-PPO: Secant-Corrected PPO

SC-PPO modifies only the inner optimization loop of PPO; Figure 2 shows how PSC (Section 3.2), ASC (Section 3.3), and RGM (Section 3.4) fit together. Proofs are deferred to the appendices.

3.1 Navigation under Imperfect Action Execution

Let ξ specify a navigation task. A policy maps a history $\mathbf{h}_t$ of observations, goal information, and past actions to an action,

$$\mathbf{a}_t = \pi_{\boldsymbol{\theta}}(\mathbf{h}_t), \qquad \widetilde{\mathbf{a}}_t = \mathbf{a}_t + \mathbf{e}_t, \qquad \mathbf{s}_{t+1} = F_\xi(\mathbf{s}_t, \widetilde{\mathbf{a}}_t), \tag{1}$$

where $\mathbf{e}_t$ is the deviation between the commanded and executed action. The policy $\pi_{\boldsymbol{\theta}} = A_{\boldsymbol{\theta}_\mathrm{a}} \circ E_{\boldsymbol{\theta}_\mathrm{e}}$ is composed of an encoder E and an action head A; the optimized block $\boldsymbol{\theta}$ may contain either or both. The nominal task cost is $\mathcal{L}(\boldsymbol{\theta}; \xi) = \mathbb{E}[\sum_t c_\xi(\mathbf{s}_t, \mathbf{a}_t) + c_{\xi,T}(\mathbf{s}_H)]$ and the population version is $J_\mathcal{D}(\boldsymbol{\theta}) = \mathbb{E}_{\xi\sim\mathcal{D}}\mathcal{L}(\boldsymbol{\theta}; \xi)$.

The learner optimizes a surrogate $J_k(\boldsymbol{\theta}; Z_k)$, where Z_k collects everything other than $\boldsymbol{\theta}$ on which it depends. In PPO, the clipped objective is

$$L_k^{\mathrm{CLIP}}(\boldsymbol{\theta}) = \frac{1}{|B_k|} \sum_{(\mathbf{h},\mathbf{a})\in B_k} \min\big(r_k(\boldsymbol{\theta})\hat{A}_k,\ \mathrm{clip}(r_k(\boldsymbol{\theta}), 1-\epsilon, 1+\epsilon)\hat{A}_k\big), \tag{2}$$

where $r_k(\boldsymbol{\theta}) = \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{h})/\pi_{\boldsymbol{\theta}_\mathrm{old}}(\mathbf{a}|\mathbf{h})$ is the probability ratio and $\hat{A}_k$ is the advantage estimate.

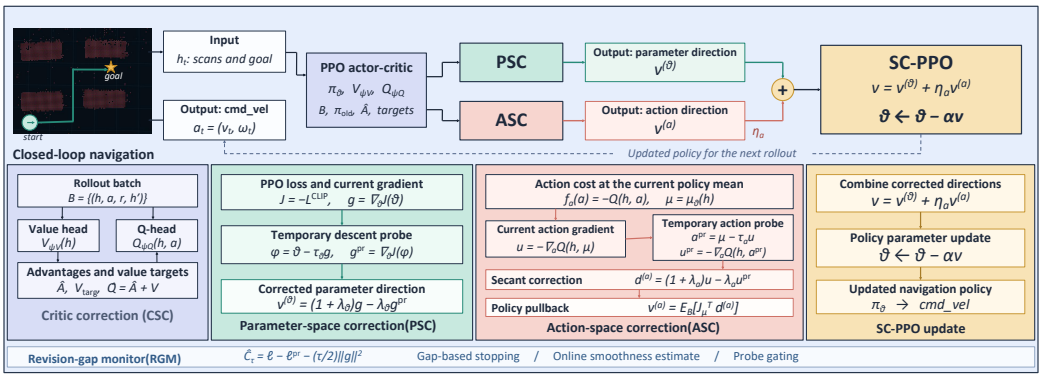


Figure 2: **SC-PPO in one iteration.** Data enters at the left; the committed policy exits at the right; probes never leave training. The direction branch (top) combines the parameter-space correction PSC (purple) and the action-space correction ASC (red). The measurement branch, RGM (blue), certifies the revision gap and gates the probe. The critic-side correction CSC is evaluated only as an ablation. Removing ASC reduces SC-PPO to PSC with gap monitoring; removing both corrections recovers PPO with gap monitoring. Each row of Table 2 is one substitution in this diagram.

Sign convention. PPO *maximizes* L^{CLIP}, whereas the secant correction introduced below is stated for a function to be *minimized*. To keep one convention throughout, every correction is applied to a *loss*: we write

$$J_k(\boldsymbol{\theta}) \;=\; -\,L_k^{\mathrm{CLIP}}(\boldsymbol{\theta}), \qquad f_a(\mathbf{a}) \;=\; -\,Q(\mathbf{h}, \mathbf{a}), \tag{3}$$

so that all gradients below are descent gradients and every update has the form $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \mathbf{v}$.

3.2 Parameter-Space Secant Correction (PSC)

Frozen evaluation context. Within iteration k, PPO runs K epochs over the same batch B_k with the old policy, the advantages, the value targets, and all random seeds held fixed; these form Z_k. Throughout the inner loop, $J_k(\cdot\,; Z_k)$ is therefore a fixed deterministic function of $\boldsymbol{\theta}$, and two of its gradients evaluated at different parameters differ only through the change in parameters. PSC exploits this property to extract curvature information from a second gradient evaluation without any additional environment interaction. The correction is applied only inside the inner loop, never across iterations, where Z_k changes.

Definition 1 (Secant-correction operator)**.** *For a differentiable $f : \mathbb{R}^d \to \mathbb{R}$ with β-Lipschitz gradient and $0 < \tau < 1/\beta$, the* secant-correction operator *is*

$$\mathrm{SC}_\tau[f](x) \;=\; 2\nabla f(x) - \nabla f\big(x - \tau \nabla f(x)\big). \tag{4}$$

Update. With $J_k = -L_k^{\mathrm{CLIP}}$ as fixed in Eq. (3), the PSC update within one PPO epoch is

$$\boxed{\begin{aligned} &\mathbf{g}_k = \nabla_{\boldsymbol{\theta}} J_k(\boldsymbol{\theta}_k), \qquad \boldsymbol{\phi}_k = \boldsymbol{\theta}_k - \tau_\theta \mathbf{g}_k, \qquad \mathbf{g}_k^{\mathrm{pr}} = \nabla_{\boldsymbol{\theta}} J_k(\boldsymbol{\phi}_k), \\ &\mathbf{v}_k^{(\theta)} = (1 + \lambda_\theta)\mathbf{g}_k - \lambda_\theta \mathbf{g}_k^{\mathrm{pr}}, \qquad \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha_k \mathbf{v}_k^{(\theta)}. \end{aligned}} \tag{5}$$

The weight $\lambda_\theta = 1$ recovers Definition 1 and $\lambda_\theta = 0$ recovers PPO. The probe $\boldsymbol{\phi}_k$ is discarded after its gradient is computed, and no derivative is taken through its construction. Algorithm 1 (Appendix A) places this step within the full iteration.

What the correction optimizes. For f as in Definition 1, the *Moreau envelope* and *revision gap* are

$$M_\tau(x) = \min_y \Big\{ f(y) + \frac{\|y - x\|^2}{2\tau} \Big\}, \qquad C_\tau(x) = f(x) - M_\tau(x) \geq 0, \tag{6}$$

so C_τ measures how much f could still be reduced by a nearby revision; the envelope is a standard tool in weakly convex optimization (Davis and Drusvyatskiy, 2019). Lemma 1 (Appendix B) establishes that $\mathrm{SC}_\tau[f]$ approximates the gradient of the *revision objective* $F_\tau = 2f - M_\tau = f + C_\tau$,

with guaranteed descent on F_τ when $\beta\tau \le 1/4$; that $\arg\min F_\tau = \arg\min f$; that C_τ is two-sided equivalent to $\frac{\tau}{2}\|\nabla f\|^2$; and that $\mathrm{SC}_\tau[f](x) = (\mathbf{I}+\tau\overline{\mathbf{H}}_\tau)\nabla f(x)$ with $\overline{\mathbf{H}}_\tau$ the Hessian averaged over the probe segment. To first order, $\mathrm{SC}_\tau[f] \approx \nabla\big(f + \frac{\tau}{2}\|\nabla f\|^2\big)$. PSC therefore descends a surrogate that adds to the PPO loss a penalty on its remaining revision gap, equivalently on its gradient norm, without changing the set of minimizers.

Relation to gradient-norm regularization. This first-order form places PSC among gradient-norm penalties (Drucker and LeCun, 1992; Barrett and Dherin, 2021; Zhao et al., 2022; Karakida et al., 2023). Sharpness-aware minimization (Foret et al., 2021; Kwon et al., 2021; Andriushchenko and Flammarion, 2022; Dai et al., 2023), applied to PPO by Lee and Yoon (2025), uses an ascent probe instead; Entropy-SGD (Chaudhari et al., 2017) and the extragradient method (Korpelevich, 1976) are further relatives. Descent-probe and ascent-probe updates agree to $O(\tau^2)$ (Appendix H), so we claim no better update direction. The difference lies in measurement: the two loss values of the descent probe certify C_τ from below (Lemma 1(ii)), whereas the ascent-probe analogue is at most $-\tau\|\mathbf{g}\|^2$ and thus vacuous; RGM is built on this certificate. Because L^{CLIP} is only piecewise smooth, the conditions under which Lemma 1 applies to J_k are given in Appendix C.

From parameter flatness to action robustness. PSC controls $\|\nabla_{\boldsymbol{\theta}} J_k\|$, a parameter-space quantity. Turning this into robustness to execution errors requires a *visibility constant* κ that bounds the sensitivity of the closed-loop cost to action deviations by its sensitivity to parameters (Assumption 3); under it, PSC yields a conditional action-risk bound (Theorem 2, Appendix C). Because κ involves the gradient of the closed-loop cost with respect to action deviations, it cannot be estimated without a dynamics model. The next section shows that this limitation is structural and removes it.

3.3 Action-Space Secant Correction (ASC)

Proposition 1 (Action flatness versus parameter flatness)**.** *Let $\mu_{\boldsymbol{\theta}}(\mathbf{h})$ be a differentiable policy mean with Jacobian $\mathbf{J} = \partial\mu_{\boldsymbol{\theta}}/\partial\boldsymbol{\theta}$, and write $\mathbf{u} = \nabla_{\mathbf{a}} Q$ at $\mathbf{a} = \mu_{\boldsymbol{\theta}}(\mathbf{h})$, so that $\nabla_{\boldsymbol{\theta}} Q = \mathbf{J}^\top\mathbf{u}$. Decompose $\mathbf{u} = \mathbf{u}_\parallel + \mathbf{u}_\perp$ with $\mathbf{u}_\perp \in \ker(\mathbf{J}^\top)$ and $\mathbf{u}_\parallel \perp \ker(\mathbf{J}^\top)$, and let $\varrho = \|\mathbf{u}_\perp\| / \|\mathbf{u}\|$. Then:* (i) $\|\nabla_{\boldsymbol{\theta}} Q\| \le \|\mathbf{J}\|\,\|\mathbf{u}\|$*, so action flatness implies parameter flatness unconditionally.* (ii) $\nabla_{\boldsymbol{\theta}} Q = \mathbf{J}^\top\mathbf{u}_\parallel$ *does not depend on $\mathbf{u}_\perp$. Any constant κ with $\|\mathbf{u}\| \le \kappa\|\nabla_{\boldsymbol{\theta}} Q\|$ satisfies $\kappa \ge 1/(\|\mathbf{J}\|\sqrt{1-\varrho^2})$; it diverges as $\varrho \to 1$, and none exists when $\mathbf{u} \in \ker(\mathbf{J}^\top) \setminus \{\mathbf{0}\}$, where the parameter gradient vanishes while the action gradient does not.*

Proposition 1 locates the quantity of interest in the action space. Both corrections ultimately update $\boldsymbol{\theta}$ through $\mathbf{J}^\top$; they differ in *what they penalize*. A parameter-space penalty sees the action gradient only through $\mathbf{J}^\top\mathbf{u}$ and is blind to $\mathbf{u}_\perp$, whereas ASC penalizes $\|\mathbf{u}\|^2$ itself, the full action gradient, and certifies it with an action-space revision gap that is computed from two critic evaluations.

Q-head. We extend the critic with a Q-head that shares the encoder $E_{\boldsymbol{\theta}_e}$. Let $\mathbf{h} = E(\mathbf{h}_t)$ be the encoder output and $\mu_{\boldsymbol{\theta}}(\mathbf{h})$ the mean action from the policy head. The Q-head takes $(\mathbf{h}, \mathbf{a})$ as input:

$$Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mathbf{a}) = W_2\,\sigma\big(W_1[\mathbf{h}; \mathbf{a}] + b_1\big) + b_2, \tag{7}$$

where the action enters after the shared encoder, so $\nabla_{\mathbf{a}} Q$ traverses only the Q-head weights. The existing value head $V_{\boldsymbol{\psi}_V}(\mathbf{h})$ is retained, and the Q-head is trained alongside it by regression onto $\hat{Q}_k = \hat{A}_k + V_{\boldsymbol{\psi}_V}(\mathbf{h})$, which reuses PPO's advantage estimates: $L_Q(\boldsymbol{\psi}_Q) = \frac{1}{|B_k|}\sum_{(\mathbf{h},\mathbf{a})\in B_k}(Q_{\boldsymbol{\psi}_Q}(\mathbf{h},\mathbf{a}) - \hat{Q}_k)^2$.

Update. Following Eq. (3), the action cost at a fixed history is $f_a(\mathbf{a}) = -Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mathbf{a})$ with gradient $\mathbf{u} = \nabla_{\mathbf{a}} f_a$. Applying Definition 1 to f_a at the current mean action $\mu_k = \mu_{\boldsymbol{\theta}}(\mathbf{h})$ gives

$$\begin{aligned} &\mathbf{u}_k = -\nabla_{\mathbf{a}} Q(\mathbf{h}, \mu_k), \qquad \mathbf{a}_k^{\mathrm{pr}} = \mu_k - \tau_a\mathbf{u}_k, \qquad \mathbf{u}_k^{\mathrm{pr}} = \nabla_{\mathbf{a}} f_a(\mathbf{a}_k^{\mathrm{pr}}), \\ &\mathbf{d}_k^{(a)} = (1+\lambda_a)\mathbf{u}_k - \lambda_a\mathbf{u}_k^{\mathrm{pr}}, \end{aligned} \tag{8}$$

and the corrected action-cost gradient is pulled back through $\mathbf{J}_\mu = \partial\mu_{\boldsymbol{\theta}}/\partial\boldsymbol{\theta}$:

$$\mathbf{v}_k^{(a)} = \mathbb{E}_{(\mathbf{h},\mathbf{a})\in B_k}\big[\mathbf{J}_\mu^\top\,\mathbf{d}_k^{(a)}\big]. \tag{9}$$

The pullback has the form of a deterministic policy gradient (Silver et al., 2014) with a secant-corrected action gradient, and costs one vector–Jacobian product. To first order, $\mathbf{d}_k^{(a)} \approx \nabla_{\mathbf{a}}\big(f_a + \frac{\lambda_a \tau_a}{2}\|\nabla_{\mathbf{a}} f_a\|^2\big)$ at μ_k, so the update moves the mean action toward regions where the learned Q is both high and flat.

Action-risk bound. Because the action-space operator controls $\|\nabla_{\mathbf{a}} Q\|$ directly, the visibility constant κ is no longer needed.

Assumption 1 (Critic fidelity)**.** $\big\|\nabla_{\mathbf{a}} Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mu_{\boldsymbol{\theta}}(\mathbf{h})) - \nabla_{\mathbf{a}} Q^{\pi}(\mathbf{h}, \mu_{\boldsymbol{\theta}}(\mathbf{h}))\big\| \le \delta_Q$ *on the training batch, where* Q^{π} *is the true action-value function.*

Unlike the visibility constant, the critic fidelity δ_Q is measurable during training.

Theorem 1 (Action-risk bound at the Q level)**.** *Fix a history* $\mathbf{h}$ *and let* $f_a(\mathbf{a}) = -Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mathbf{a})$ *satisfy* β_a*-smoothness on the deviation ball, with* $\mathbf{u} = \nabla_{\mathbf{a}} f_a(\mu)$ *and action-level revision gap* $C_{\tau_a}^{(a)}$*. Define* $\mathcal{R}_\varepsilon(\mathbf{h}) = \sup_{\|\mathbf{e}\| \le \varepsilon}\{-Q^{\pi}(\mathbf{h}, \mu_{\boldsymbol{\theta}}(\mathbf{h}) + \mathbf{e})\}$*. Then, under Assumption 1,*

$$\begin{aligned}\mathcal{R}_\varepsilon(\mathbf{h}) &\le -Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mu) + \varepsilon\,\|\mathbf{u}\| + \delta_Q \varepsilon + \frac{\beta_a \varepsilon^2}{2} + \Delta_Q \\ &\le -Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mu) + \varepsilon\sqrt{\tfrac{2(1+\beta_a\tau_a)}{\tau_a} C_{\tau_a}^{(a)}} + \delta_Q \varepsilon + \frac{\beta_a \varepsilon^2}{2} + \Delta_Q,\end{aligned}$$

where $\Delta_Q = \sup_{\|\mathbf{e}\| \le \varepsilon} |Q^{\pi} - Q_{\boldsymbol{\psi}_Q}|$*. No condition on the policy Jacobian is used.*

The unmeasurable κ of the conditional bound is replaced by the critic errors (δ_Q, Δ_Q), both of which shrink as the Q-head improves. Lifting to trajectory risk adds the usual $(1-\gamma)^{-1}$ horizon factor (Proposition 2, Appendix D).

3.4 Revision-Gap Monitor (RGM)

Each application of Definition 1 already evaluates the loss at the current point and at the probe. By Lemma 1(ii), these two values give a certified lower bound $\widehat{C}_\tau = f(x) - f(x - \tau\mathbf{g}) - \frac{\tau}{2}\|\mathbf{g}\|^2 \le C_\tau(x)$ at no extra cost. RGM uses the parameter-space certificate in three ways.

Gap-based early stopping. Define the normalized gap $\rho_k = \widehat{C}_{\tau,k}/(\frac{\tau}{2}\|\mathbf{g}_k\|^2)$ whenever $\|\mathbf{g}_k\|^2 > \varsigma$, a fixed floor that guards against $0/0$ at stationarity. We stop the inner loop when $\rho_k > T_\rho$ with $T_\rho = 1/(1-\bar{t})$ and $\bar{t} = 1/4$. By Proposition 3 (Appendix E), a firing certifies that the local $\beta\tau_\theta$ exceeds $1/4$, the regime in which the descent guarantee of Lemma 1(iii) no longer applies. The threshold is thus fixed by the analysis rather than tuned as a tolerance on a distributional distance.

Online smoothness estimate. Rearranging the gap bounds gives a certified lower bound $\hat{\beta}_k = \frac{1}{\tau}\max\{1-\rho_k, 1-1/\rho_k\} \le \beta$ (Eq. (11), Appendix E). We average $\hat{\beta}_k$ over a sliding window and use it to keep τ_θ inside the $\beta\tau \le 1/4$ regime with a fixed safety factor c_β (Appendix I).

Gated probe. When $\widehat{C}_\tau$ remains below a threshold $C_{\min}$, the landscape is already flat; the probe gradient is skipped and the nominal gradient is used, which recovers part of the added cost.

3.5 Assembly and Cost

The combined update within one PPO epoch is

$$\mathbf{v}_k = \mathbf{v}_k^{(\theta)} + \eta_a\,\mathbf{v}_k^{(a)}, \qquad \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha_k \mathbf{v}_k, \tag{10}$$

where $\mathbf{v}_k^{(\theta)}$ is the PSC direction (Eq. 5) and $\mathbf{v}_k^{(a)}$ the ASC direction (Eq. 9). The mixing weight η_a is kept distinct from the secant weight λ_a of Eq. (8): λ_a sets how much correction is applied inside the action-space operator, and η_a sets how much the corrected action term contributes to the step. Setting $\eta_a{=}0$ removes ASC while keeping the Q-head; setting $\lambda_a{=}0$ with $\eta_a{>}0$ keeps the action term but removes its correction; both appear as controls in Table 2. Counting actor forward–backward passes per epoch with PPO as $1\times$, the PSC probe adds one pass and the ASC pullback

one vector–Jacobian product, while both $\nabla_{\mathbf{a}} Q$ evaluations traverse only the Q-head; full SC-PPO therefore costs about $3\times$ the actor compute of PPO, less when the probe is gated. Environment interaction and the deployed policy are unchanged.

4 EXPERIMENTS

4.1 SETUP

Training. All methods share the actor architecture, value critic, environment budget, and optimizer; they differ only in the update rule. The base learner is PPO with the clipped surrogate, GAE (Schulman et al., 2016), and a shared encoder with separate policy and value heads; the Q-head (Eq. 7) is added only for methods that use ASC. Unless stated otherwise, $\lambda_\theta{=}\lambda_a{=}1$, τ_θ is adapted online by RGM (Section 3.4), τ_a is fixed, η_a is selected on the validation split, and $K{=}10$ PPO epochs are used. Every method is trained for 4M environment steps with five training seeds, and the reported checkpoint is selected on the fault-validation criterion of Appendix J.

Execution faults. Faults are injected at test time only, at every control step, into the commanded linear and angular velocities (v_t, ω_t) before they reach the base; the resulting command is clipped to the robot's velocity limits. We use four fault families, two persistent and two random: (i) a *linear-velocity bias* $b_v \in \{-0.2, -0.1, +0.1, +0.2\}$ m/s and (ii) an *angular-velocity bias* $b_\omega \in \{-0.3, -0.15, +0.15, +0.3\}$ rad/s, each held for the whole episode; and zero-mean *Gaussian noise*, resampled every step, on (iii) the linear velocity with $\sigma_v \in \{0.05, 0.1, 0.2\}$ m/s or (iv) the angular velocity with $\sigma_\omega \in \{0.1, 0.2, 0.4\}$ rad/s. No method is trained on any of these faults (Appendix I).

Metrics. An episode succeeds if the robot reaches the goal without collision within the time limit. We report the nominal success rate (SR), the SR of each fault family averaged over its nonzero levels, their equally weighted average (Avg.), and the mean traversal time T_{succ} of successful nominal episodes. All values are means over the five training seeds; the primary comparison is Avg. against PPO.

Baselines. **PPO**: unmodified PPO with approximate-KL early stopping (Huang et al., 2022). **SAM+PPO** (Lee and Yoon, 2025): sharpness-aware minimization of the PPO surrogate, i.e. the flatness-based method whose parameter-to-action link Proposition 1 examines. **GR+PPO**: PPO with an explicit gradient-norm penalty on the clipped surrogate (Barrett and Dherin, 2021; Zhao et al., 2022), the first-order objective that PSC approximates (Section 3.2); it separates the effect of the secant correction from that of gradient-norm regularization. Implementation details and tuning grids are given in Appendix J.

4.2 LARGE-SCALE SIMULATION ON BARN

The primary testbed is BARN, whose 300 environments we divide into a 210/45/45 train/validation/test split stratified by difficulty (Perille et al., 2020; Barn Challenge Contributors, 2026). Table 1 reports the success rate of every method under nominal execution and under each fault family.

SC-PPO matches the nominal SR of PPO (41.78% versus 41.33%), whereas SAM+PPO and GR+PPO lose 4.0 and 7.1 points. Both baselines score higher under faults than nominally, which suggests that checkpoint selection on fault validation trades their nominal performance for tolerance; SC-PPO avoids this trade-off. Under faults, SC-PPO attains the highest average SR (39.84%), 2.4 points above PPO and 0.6 above SAM+PPO, and it improves on PPO in all four families: by 3.8 points under velocity noise, 2.4 under angular bias, 2.1 under velocity bias, and 1.2 under angular noise. Only under velocity bias is SAM+PPO ahead, by 0.3 points. With five seeds and seed standard deviations of 2–3 points (Table 2), no single per-family gap is conclusive; the evidence is that the gain over PPO has the same sign in every family. The gain is not obtained by driving more slowly: successful SC-PPO traversals are 0.17 s faster than those of PPO.

Table 1: BARN success rates (%), 4M training steps, five training seeds, 210/45/45 train/validation/test maps. Checkpoints are selected on fault validation. Each fault column averages its family's nonzero levels; Avg. weights the four families equally. T_{succ} is mean successful nominal traversal time (s, lower is better), averaged equally over seeds. Best value per column in bold.

		Persistent bias		Gaussian noise			
Method	Nominal	Velocity	Angular	Velocity	Angular	Avg.	T_{succ}
PPO	41.33	37.00	36.78	36.92	39.16	37.47	5.87
SAM+PPO	37.33	**39.44**	37.11	40.50	39.76	39.20	**5.65**
GR+PPO	34.22	34.22	35.78	34.01	37.24	35.31	5.88
SC-PPO	**41.78**	39.11	**39.22**	**40.71**	**40.33**	**39.84**	5.70

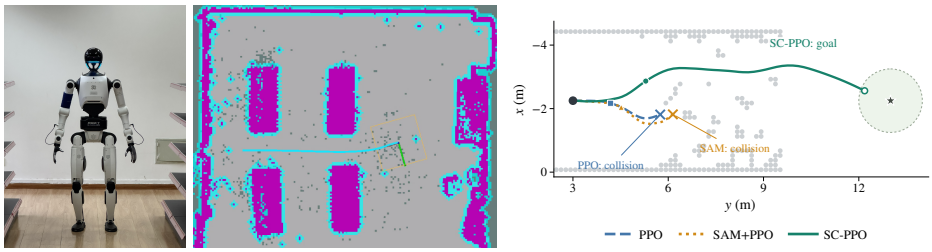


Figure 3: Physical-robot demonstration. *Left:* the Tiangong 3.0 Pro humanoid. *Middle:* map of the test course with the executed SC-PPO path (cyan). *Right:* one run under a persistent angular bias $b_\omega = -0.3$ rad/s from the same start pose; PPO and SAM+PPO collide, whereas SC-PPO reaches the goal.

4.3 Physical-Robot Demonstration

We deploy the simulation-trained SC-PPO policy without retraining on a Tiangong 3.0 Pro humanoid robot (X-Humanoid; about 1.7 m tall, 43 degrees of freedom; Figure 3). The command (v_t, ω_t) is sent as a forward-velocity and yaw-rate target to the robot's onboard locomotion controller, which tracks it through a gait. Foot placement and body sway make the executed velocity deviate from the command at every step, so the robot itself supplies an execution error $\mathbf{e}_t$ as in Eq. (1). The range observation is a planar scan projected from the point cloud of the head-mounted LiDAR (Livox Mid-360), and the goal is given in the robot frame by LiDAR-inertial odometry. The robot traverses narrow passages between shelves and boxes, with nominal commands and with the persistent angular bias of Section 4.1 added in software; PPO and SAM+PPO are run on the same courses from the same start poses, and all runs are shown side by side in the supplementary video. Under the bias, PPO and SAM+PPO drift toward the obstacles and collide shortly after the start, whereas SC-PPO counteracts the drift and reaches the goal (Figure 3, right).

4.4 Ablations and Diagnostics

Each row of Table 2 changes one element of SC-PPO: $\lambda_\theta = 0$ removes PSC; $\eta_a = 0$ removes the action term but keeps the Q-head; $\lambda_a = 0$ uses the plain action gradient $\mathbf{u}_k$ instead of $\mathbf{d}_k^{(a)}$; removing RGM restores KL stopping; and CSC also corrects the value loss. Faulted SR is reported at the fault-validation checkpoint (Selected) and at the final checkpoint of the same run (Final).

Removing the action-space term ($\eta_a = 0$) causes the largest loss among the removals: -5.8 points nominal and -2.3 faulted. Since the Q-head is still trained on the shared encoder in this row, the Q-head helps only when its action gradient drives the policy. Using the plain action gradient ($\lambda_a = 0$) performs on par with full SC-PPO, within one seed standard deviation, so the benefit of ASC here comes mainly from optimizing the learned Q in the action space; a separate contribution of the secant correction inside it is not resolved at five seeds. Removing PSC ($\lambda_\theta = 0$) changes the results by less than the seed spread. Removing RGM lowers nominal SR by 3.1 points and nearly doubles the seed standard deviation of faulted SR (6.47 versus 3.37), so the gap-based stop mainly stabilizes training. Adding CSC is clearly harmful (-7.6 points nominal and -8.6 faulted), which supports

Table 2: Matched component ablations: success rate (%), mean ± sample standard deviation over five training seeds, at 4M steps. Selected uses fault validation; Final uses the final checkpoint of the same training trajectory. Removing RGM jointly disables adaptation, gating and gap stopping, and restores KL stopping.

	Nominal	Faulted SR	
Setting	SR	Selected	Final
PPO	41.33	37.47 ± 2.38	38.08 ± 2.39
SC-PPO	41.78	39.84 ± 3.37	39.83 ± 3.39
SC-PPO: $\lambda_\theta = 0$	40.44	38.89 ± 4.31	39.87 ± 5.43
SC-PPO: $\eta_a = 0$ (Q-head retained)	36.00	37.51 ± 2.45	39.32 ± 2.05
SC-PPO: $\lambda_a = 0$ (plain action gradient)	40.89	40.68 ± 3.21	40.78 ± 2.97
SC-PPO: without RGM	38.67	38.32 ± 6.47	39.61 ± 4.60
SC-PPO: with CSC	34.22	31.21 ± 5.13	31.34 ± 5.09

restricting the correction to the actor. Selected and final checkpoints of each variant differ by at most 1.8 points, so these conclusions do not depend on checkpoint selection.

5 Discussion and Limitations

Lemma 1 and Proposition 1 hold for a fixed smooth surrogate with exact gradients; they do not cover changing critics, gradient clipping, adaptive optimizers—with which gradient regularization is known to interact unfavourably (Zhao et al., 2024)—or the large-τ regime. On a scalar quadratic the SC operator is gradient descent with effective step $\alpha(1+\tau)$ (Appendix G), so step sizes must be controlled when comparing methods. Theorem 1 holds at every parameter vector however it was reached: it identifies the mechanism and does not imply that SC-PPO dominates gradient descent on the bound. Its trajectory-level version (Proposition 2) pays a $(1-\gamma)^{-1}$ horizon factor, so the theory supports the mechanism rather than a numerically useful certificate. The Q-head's fidelity (Assumption 1) is measurable but not trivially small; a poorly trained Q-head provides no useful action-level signal. Robustness need not be free at deployment, since a policy that keeps larger margins may take longer to reach the goal; we did not observe this on BARN (Table 1), but report traversal time for this reason. Finally, the physical-robot deployment is a qualitative demonstration; a controlled hardware study with repeated trials is left to future work.

6 Conclusion

This paper proposed SC-PPO, which instantiates a single secant-correction operator at two levels of PPO—the parameter space (PSC) and the action space (ASC)—and reuses its loss evaluations in a revision-gap monitor (RGM) that replaces KL-based early stopping. The action-space correction is the central component: it acts where execution errors enter, removes the dependence on the unmeasurable visibility constant, and yields an action-risk bound under a critic-fidelity condition that can be checked during training. On BARN, SC-PPO matches the nominal success rate of PPO and attains the highest success rate averaged over four families of execution faults, improving on PPO in every family without slower traversal. The ablations attribute most of this benefit to the action-space term and the revision-gap monitor. We also demonstrated the policy on a full-size humanoid robot, whose gait makes execution errors intrinsic.

Reproducibility statement

Algorithm 1 and Section 3 specify the update and its state-management contract. The SC operator (Definition 1) and Lemma 1 are stated for any β-smooth function; each component specializes one instance. Appendices B–H contain complete proofs. Section 4 and Appendices I–J specify the evaluation protocol and the baseline implementations.

ETHICS AND SAFETY STATEMENT

The analysis does not certify collision avoidance and does not justify removing independent safety measures. All quantitative results in this paper are from simulation; the physical-robot deployment is a qualitative demonstration, and every run was supervised by an operator with an emergency stop.

AI USE STATEMENT

We used generative AI tools to aid and polish the writing, including drafting and revising the text of this paper and checking the reported numbers for consistency. We also used them for retrieval and discovery of related work, and for research execution: they helped write the training and evaluation code, set up the BARN simulation environment, and deploy the policy on the humanoid robot. We have reviewed all AI-assisted work, and all numbers reported in the paper were cross-checked against the experiment logs. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Maksym Andriushchenko and Nicolas Flammarion. Towards understanding sharpness-aware minimization. In *ICML*, 2022.

Marcin Andrychowicz, Anton Raichuk, Piotr Stańczyk, Manu Orsini, Sertan Girgin, Raphaël Marinier, Léonard Hussenot, Matthieu Geist, Olivier Pietquin, Marcin Michalski, Sylvain Gelly, and Olivier Bachem. What matters for on-policy deep actor-critic methods? A large-scale study. In *ICLR*, 2021.

BARN Challenge Contributors. The BARN Challenge: navigation benchmark implementation. `https://github.com/Daffan/the-barn-challenge`, 2026.

David G. T. Barrett and Benoit Dherin. Implicit gradient regularization. In *ICLR*, 2021.

Peter L. Bartlett and Shahar Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *JMLR*, 3:463–482, 2002.

Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-SGD: Biasing gradient descent into wide valleys. In *ICLR*, 2017.

Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: A CVaR optimization approach. In *NeurIPS*, 2015.

Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *JMLR*, 18(167):1–51, 2018.

Karl Cobbe, Jacob Hilton, Oleg Klimov, and John Schulman. Phasic policy gradient. In *ICML*, pp. 2020–2027, 2021.

Yan Dai, Kwangjun Ahn, and Suvrit Sra. The crucial role of normalization in sharpness-aware minimization. In *NeurIPS*, 2023.

Damek Davis and Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

Harris Drucker and Yann LeCun. Improving generalization performance using double backpropagation. *IEEE Trans. Neural Networks*, 3(6):991–997, 1992.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep policy gradients: A case study on PPO and TRPO. In *ICLR*, 2020.

Aleksandra Faust, Oscar Ramirez, Marek Fiser, Kenneth Oslund, Anthony Francis, James Davidson, and Lydia Tapia. PRM-RL: Long-range robotic navigation tasks by combining reinforcement learning and sampling-based planning. In *ICRA*, pp. 5113–5120, 2018.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *ICLR*, 2021.

Dieter Fox, Wolfram Burgard, and Sebastian Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33, 1997.

Shengyi Huang, Rousslan Fernand Julien Dossa, Antonin Raffin, Anssi Kanervisto, and Weixun Wang. The 37 implementation details of proximal policy optimization. In *ICLR Blog Track*, 2022.

Garud N. Iyengar. Robust dynamic programming. *Math. of Operations Research*, 30(2):257–280, 2005.

Ryo Karakida, Tomoumi Takase, Tomohiro Hayase, and Kazuki Osawa. Understanding gradient regularization in deep learning: Efficient finite-difference computation and implicit bias. In *ICML*, pp. 15809–15827, 2023.

G. M. Korpelevich. The extragradient method for finding saddle points and other problems. *Ekonomika i Matematicheskie Metody*, 12:747–756, 1976.

Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. ASAM: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *ICML*, pp. 5905–5914, 2021.

Hyun Kyu Lee and Sung Whan Yoon. Flat reward in policy parameter space implies robust reinforcement learning. In *ICLR*, 2025.

Pinxin Long, Tingxiang Fan, Xinyi Liao, Wenxi Liu, Hao Zhang, and Jia Pan. Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning. In *ICRA*, 2018.

David V. Lu, Dave Hershberger, and William D. Smart. Layered costmaps for context-sensitive navigation. In *IROS*, pp. 709–715, 2014.

Siddharth Mysore, Bassel Mabsout, Renato Mancuso, and Kate Saenko. Regularizing action policies for smooth control with reinforcement learning. In *ICRA*, 2021.

Arnab Nilim and Laurent El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *ICRA*, pp. 3803–3810, 2018.

Daniel Perille, Abigail Truong, Xuesu Xiao, and Peter Stone. Benchmarking metric ground navigation. In *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2020.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *ICML*, 2017.

Md Masudur Rahman, Juan Wachs, and Yexiang Xue. Robust policy gradient optimization through parameter perturbation in reinforcement learning. In *NeurIPS Workshop on Aligning Reinforcement Learning Experimentalists and Theorists*, 2025.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, pp. 627–635, 2011.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *ICML*, pp. 1889–1897, 2015.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *ICLR*, 2016.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.

David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *ICML*, pp. 387–395, 2014.

Lei Tai, Giuseppe Paolo, and Ming Liu. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation. In *IROS*, 2017.

Aviv Tamar, Dotan Di Castro, and Shie Mannor. Policy gradients with variance related risk criteria. In *ICML*, 2012.

Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In *ICML*, pp. 6215–6224, 2019.

Yuhui Wang, Hao He, and Xiaoyang Tan. Truly proximal policy optimization. In *Proceedings of the 35th Conference on Uncertainty in Artificial Intelligence (UAI)*, PMLR 115, pp. 113–122, 2020.

Xuesu Xiao, Bo Liu, Garrett Warnell, and Peter Stone. Toward agile maneuvers in highly constrained spaces: Learning from hallucination. *IEEE RA-L*, 6(2):1503–1510, 2021.

Xuesu Xiao, Zifan Xu, Zizhao Wang, Yunlong Song, Garrett Warnell, Peter Stone, et al. Autonomous ground navigation in highly constrained spaces: Lessons learned from the BARN Challenge at ICRA 2022. *IEEE Robotics & Automation Magazine*, 2022.

Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. In *NeurIPS*, 2020.

Yang Zhao, Hao Zhang, and Xiuyuan Hu. Penalizing gradient norm for efficiently improving generalization in deep learning. In *ICML*, pp. 26982–26992, 2022.

Yang Zhao, Hao Zhang, and Xiuyuan Hu. When will gradient regularization be harmful? In *ICML*, 2024.

A PSEUDOCODE OF SC-PPO

Algorithm 1 gives one SC-PPO epoch; the notation follows Section 3.

Algorithm 1 One SC-PPO epoch (within PPO's K-epoch inner loop). All objectives are losses (Eq. 3); every step is a descent step.

Require: $\boldsymbol{\theta}$; frozen context $Z = (B, \pi_{\text{old}}, \hat{A}, V_{\text{targ}})$; probe steps τ_θ, τ_a; step size α; secant weights $\lambda_\theta, \lambda_a$; mixing weight η_a; Q-head $Q_{\boldsymbol{\psi}_Q}$; gradient floor ς

— RGM: *revision-gap monitor* —

$J \leftarrow -L^{\text{CLIP}}(\cdot\,; Z)$ ▷ loss convention, Eq. (3)

$(\ell, \mathbf{g}) \leftarrow (J(\boldsymbol{\theta}), \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}))$

$\ell^{\text{pr}} \leftarrow J(\boldsymbol{\theta} - \tau_\theta \mathbf{g})$; $\quad \widehat{C}_\tau \leftarrow \ell - \ell^{\text{pr}} - \frac{\tau_\theta}{2} \|\mathbf{g}\|^2$ ▷ gap certificate

if $\|\mathbf{g}\|^2 > \varsigma$ **then**

  $\rho \leftarrow \widehat{C}_\tau / (\frac{\tau_\theta}{2} \|\mathbf{g}\|^2)$; $\quad \hat{\beta} \leftarrow \frac{1}{\tau_\theta} \max\{1-\rho,\, 1 - 1/\rho\}$

  if $\rho > T_\rho$ **then**

    stop the inner loop ▷ gap-based stop

  end if

end if

— PSC: *parameter-space correction* —

if $\widehat{C}_\tau > C_{\min}$ **then** ▷ gated probe

  $\boldsymbol{\phi} \leftarrow \boldsymbol{\theta} - \tau_\theta \mathbf{g}$; $\quad \mathbf{g}^{\text{pr}} \leftarrow \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\phi})$; $\quad$ discard $\boldsymbol{\phi}$

  $\mathbf{v}^{(\theta)} \leftarrow (1+\lambda_\theta)\mathbf{g} - \lambda_\theta \mathbf{g}^{\text{pr}}$

else

  $\mathbf{v}^{(\theta)} \leftarrow \mathbf{g}$ ▷ landscape already flat

end if

— ASC: *action-space correction* —

$\mathbf{v}^{(a)} \leftarrow \mathbf{0}$

for $(\mathbf{h}, \mathbf{a}) \in B$ **do**

  $\mu \leftarrow \mu_{\boldsymbol{\theta}}(\mathbf{h})$; $\quad \mathbf{u} \leftarrow -\nabla_{\mathbf{a}} Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mu)$ ▷ action-*cost* gradient

  $\mathbf{a}^{\text{pr}} \leftarrow \mu - \tau_a \mathbf{u}$; $\quad \mathbf{u}^{\text{pr}} \leftarrow -\nabla_{\mathbf{a}} Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mathbf{a}^{\text{pr}})$

  $\mathbf{d}^{(a)} \leftarrow (1+\lambda_a)\mathbf{u} - \lambda_a \mathbf{u}^{\text{pr}}$

  $\mathbf{v}^{(a)} \mathrel{+}= \mathbf{J}_\mu^\top \mathbf{d}^{(a)} / |B|$ ▷ one VJP; batched in practice

end for

— *commit* —

$\mathbf{v} \leftarrow \mathbf{v}^{(\theta)} + \eta_a \mathbf{v}^{(a)}$; $\quad \boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \mathbf{v}$

return $\boldsymbol{\theta}$, $\widehat{C}_\tau$, ρ, $\hat{\beta}$

B SC OPERATOR: PROPERTIES AND PROOFS

This appendix states and proves the full properties of the SC operator referenced in the main text.

Assumption 2 (Fixed smooth surrogate)**.** *$f : \mathbb{R}^d \to \mathbb{R}$ is continuously differentiable with a globally β-Lipschitz gradient, $\beta > 0$, and $0 < \tau < 1/\beta$.*

Lemma 1 (Properties of the SC operator)**.** *Under Assumption 2, let $\mathbf{g} = \nabla f(x)$ and $\mathbf{v} = \text{SC}_\tau[f](x)$. Then:*

(i) ***Gap bounds:*** $\dfrac{\tau \|\mathbf{g}\|^2}{2(1+\beta\tau)} \le C_\tau(x) \le \dfrac{\tau \|\mathbf{g}\|^2}{2(1-\beta\tau)}$.

(ii) ***Certificate:*** *Define $\widehat{C}_\tau = f(x) - f(x - \tau\mathbf{g}) - \frac{\tau}{2}\|\mathbf{g}\|^2$. Then $\frac{\tau}{2}(1-\beta\tau)\|\mathbf{g}\|^2 \le \widehat{C}_\tau \le C_\tau(x)$.*

(iii) ***Approximation and descent:*** $\|\mathbf{v} - \nabla F_\tau(x)\| \le \frac{\beta^2\tau^2}{1-\beta\tau}\|\mathbf{g}\|$. *If $\beta\tau \le 1/4$ and $0 < \alpha \le 1/(2L_F)$ with $L_F = 2\beta + \beta/(1-\beta\tau)$, then $F_\tau(x - \alpha\mathbf{v}) \le F_\tau(x) - \frac{\alpha}{2}\|\nabla F_\tau(x)\|^2$. Moreover, $\langle \mathbf{g}, \mathbf{v} \rangle \ge (1-\beta\tau)\|\mathbf{g}\|^2$, so sufficiently small steps also decrease f.*

(iv) ***Solution set:*** $C_\tau(x) = 0 \Leftrightarrow \nabla F_\tau(x) = 0 \Leftrightarrow \mathbf{g} = 0$, *and* $\arg\min F_\tau = \arg\min f$.

(v) ***Secant identity:*** *If f is twice differentiable on the probe segment, then* $\mathbf{v} = (\mathbf{I} + \tau\overline{\mathbf{H}}_\tau)\mathbf{g}$, *where* $\overline{\mathbf{H}}_\tau = \int_0^1 \nabla^2 f(x - t\tau\mathbf{g})\, dt$.

Part (i) shows the gap is a two-sided proxy for $\frac{\tau}{2}\|\nabla f\|^2$; part (ii) shows the two loss evaluations certify a non-vacuous lower bound; part (iii) establishes that the SC operator approximates ∇F_τ; part (iv) ensures the correction does not alter the set of optimal solutions.

The key comparison with the ascent probe of Karakida et al. (2023) and SAM (Foret et al., 2021) is the certificate. Let $\widehat{C}_\tau^+ = f(x) - f(x+\tau\mathbf{g}) - \frac{\tau}{2}\|\mathbf{g}\|^2$; then $\widehat{C}_\tau^+ \le -\tau\|\mathbf{g}\|^2$, which is strictly negative and hence vacuous as a bound on $C_\tau \ge 0$. The separation is $\Theta(\tau\|\mathbf{g}\|^2)$: first order in τ, unlike the $O(\tau^2\|\mathbf{g}\|^2)$ difference between the two update directions.

Under Assumption 2, the objective in Eq. (6) is $(1/\tau - \beta)$-strongly convex, so a unique minimizer $\phi_\tau^\star$ exists. The first-order condition is $\nabla f(\phi_\tau^\star) + (\phi_\tau^\star - x)/\tau = 0$, giving the envelope identity $\nabla M_\tau(x) = \nabla f(\phi_\tau^\star)$.

Proof of Lemma 1(i): Gap bounds. Let $g = \nabla f(x)$ and $s = \phi - x$. Smoothness gives $f(x) + g^\top s - \frac{\beta}{2}\|s\|^2 \le f(x+s) \le f(x) + g^\top s + \frac{\beta}{2}\|s\|^2$. Adding $\|s\|^2/(2\tau)$ and minimizing the quadratics over s gives $f(x) - \frac{\tau\|g\|^2}{2(1-\beta\tau)} \le M_\tau(x) \le f(x) - \frac{\tau\|g\|^2}{2(1+\beta\tau)}$. Subtracting from $f(x)$ proves the claim. □

Proof of Lemma 1(ii): Certificate. The feasible point $\phi = x - \tau g$ gives $M_\tau(x) \le f(x-\tau g) + \frac{\tau}{2}\|g\|^2$, so $\widehat{C}_\tau \le C_\tau(x)$. The descent lemma gives $f(x-\tau g) \le f(x) - \tau\|g\|^2 + \frac{\beta\tau^2}{2}\|g\|^2$, so $\widehat{C}_\tau \ge \frac{\tau}{2}(1-\beta\tau)\|g\|^2 \ge 0$. For the ascent monitor, $f(x+\tau g) \ge f(x) + \tau\|g\|^2 - \frac{\beta\tau^2}{2}\|g\|^2$, so $\widehat{C}_\tau^+ \le -\frac{\tau}{2}(3-\beta\tau)\|g\|^2 \le -\tau\|g\|^2$. □

Proof of Lemma 1(iii): Approximation and descent. Write $g^\star = \nabla f(\phi_\tau^\star)$, $g = \nabla f(x)$, and $\phi^{\mathrm{pr}} = x - \tau g$. The optimality condition gives $\phi_\tau^\star = x - \tau g^\star$, so $\|g^\star - g\| \le \beta\tau\|g^\star\|$, $\|g^\star\| \le \|g\|/(1-\beta\tau)$, and $\|\phi_\tau^\star - \phi^{\mathrm{pr}}\| \le \frac{\beta\tau^2}{1-\beta\tau}\|g\|$. Applying smoothness once more: $\|v - \nabla F_\tau(x)\| \le \frac{\beta^2\tau^2}{1-\beta\tau}\|g\|$.

Subtracting optimality conditions for two points gives $\|\phi_\tau^\star(x_1) - \phi_\tau^\star(x_2)\| \le \|x_1 - x_2\|/(1-\beta\tau)$, so ∇M_τ is $\beta/(1-\beta\tau)$-Lipschitz and $\nabla F_\tau = 2\nabla f - \nabla M_\tau$ is L_F-Lipschitz.

Let $t = \beta\tau$ and $h = \nabla F_\tau(x) = 2g - g^\star$. Then $\|h\| \ge \frac{1-2t}{1-t}\|g\|$. For $t \le 1/4$, $\|v - h\| \le \frac{1}{8}\|h\|$, so $\langle h, v\rangle \ge \frac{7}{8}\|h\|^2$ and $\|v\| \le \frac{9}{8}\|h\|$. The descent lemma yields $F_\tau(x - \alpha v) \le F_\tau(x) - \frac{7\alpha}{8}\|h\|^2 + \frac{81L_F\alpha^2}{128}\|h\|^2$. For $\alpha \le 1/(2L_F)$ the decrease coefficient is at least $\alpha/2$.

Nominal-objective alignment: $\langle g, v\rangle \ge (1-\beta\tau)\|g\|^2$ and $\|v\| \le (1+\beta\tau)\|g\|$, so f decreases under sufficiently small steps when $g \ne 0$. □

Proof of Lemma 1(iv): Solution set. *(a)* If $g = 0$, then $\phi = x$ is optimal, $C_\tau = 0$. Conversely, $C_\tau = 0$ implies $\phi_\tau^\star = x$, so $g = 0$. *(b)* The lower bound on $\|h\|$ has a positive coefficient when $t < 1/2$, so $\nabla F_\tau = 0 \Leftrightarrow g = 0$. *(c)* $C_\tau \ge 0$ gives $F_\tau \ge f$; the gap bounds give $F_\tau - f^\star \le (f - f^\star)/(1-t)$. Hence F_τ attains its minimum value $f^\star$ exactly on $\arg\min f$. □

Proof of Lemma 1(v): Secant identity. Follows from the fundamental theorem of calculus applied to $\nabla f(x) - \nabla f(x - \tau g) = \int_0^1 \nabla^2 f(x - t\tau g)\, dt \cdot \tau g$. □

C Supplementary Material for PSC

Smoothness of the clipped loss. Assumption 2 asks for a globally β-Lipschitz gradient, which L^{CLIP} does not satisfy: the $\min$ and the clip introduce kinks at $r_k(\boldsymbol{\theta}) = 1 \pm \epsilon$ and where the two

branches cross, so J_k is piecewise smooth rather than smooth. Three remarks delimit what this costs. *(a)* On each piece the gradient is Lipschitz, and Lemma 1 applies verbatim whenever the probe segment $[\boldsymbol{\theta}_k, \boldsymbol{\theta}_k - \tau_\theta \mathbf{g}_k]$ stays inside one piece; β is then the maximum Lipschitz constant over the pieces met. *(b)* The fraction of samples whose probe segment crosses a kink is observable—it is the fraction whose clip indicator differs at $\boldsymbol{\theta}_k$ and at $\boldsymbol{\phi}_k$—and we log it (Appendix I); for the τ_θ values we use, the probe displacement is small relative to the clip width. *(c)* Where a crossing does occur, the clipped branch is locally constant in $\boldsymbol{\theta}$, so the affected sample contributes zero to both $\mathbf{g}_k$ and $\mathbf{g}_k^{\mathrm{pr}}$ and drops out of the secant difference rather than corrupting it. ASC does not inherit this issue: the Q-head (Eq. 7) is a smooth network with no clipping.

Conditional action-risk bound. PSC flattens the loss landscape in parameter space. To connect this to action-level robustness, we need a condition relating the two spaces. Let $J(\boldsymbol{\theta}, \mathbf{e})$ denote a smooth closed-loop cost under additive action deviations $\mathbf{e} = (\mathbf{e}_0, \ldots, \mathbf{e}_{H-1})$, and let $\mathbf{q} = \nabla_{\mathbf{e}} J(\boldsymbol{\theta}, 0)$, $\mathbf{g} = \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}, 0)$.

Assumption 3 (Visibility)**.** $\|\mathbf{q}\| \le \kappa \|\mathbf{g}\|$ *for some finite* $\kappa \ge 0$*, and* $\nabla_{\mathbf{e}} J(\boldsymbol{\theta}, \cdot)$ *is* β_a*-Lipschitz on the deviation ball.*

Theorem 2 (Conditional action-risk bound)**.** *Let Assumptions 2 and 3 hold, and let the action risk be* $R_\varepsilon(\boldsymbol{\theta}) = \sup_{\|\mathbf{e}\| \le \varepsilon} J(\boldsymbol{\theta}, \mathbf{e})$*. Then*

$$R_\varepsilon(\boldsymbol{\theta}) \le J(\boldsymbol{\theta}) + \kappa\varepsilon \|\mathbf{g}\| + \frac{\beta_a \varepsilon^2}{2} \quad \textit{and} \quad R_\varepsilon(\boldsymbol{\theta}) \le F_\tau(\boldsymbol{\theta}) + \frac{\kappa^2 \varepsilon^2 (1 + \beta\tau)}{2\tau} + \frac{\beta_a \varepsilon^2}{2}.$$

The visibility constant κ is not directly measurable: it depends on the policy Jacobian's behavior over action directions weighted by future costs, and no finite-sample estimate of it is available during training.

D Action-Risk Proofs

D.1 Proof of Theorem 2

Smoothness in the deviation argument gives $J(\boldsymbol{\theta}, \mathbf{e}) \le J(\boldsymbol{\theta}, 0) + \mathbf{q}^\top \mathbf{e} + \frac{\beta_a}{2} \|\mathbf{e}\|^2$. Taking the supremum and using $\|\mathbf{q}\| \le \kappa \|\mathbf{g}\|$ proves the first bound. Lemma 1(i) gives $\|\mathbf{g}\| \le \sqrt{2(1 + \beta\tau) C_\tau / \tau}$; substituting yields the second. □

D.2 Proof of Theorem 1

Fix $\mathbf{h}$ and write $\mu = \mu_{\boldsymbol{\theta}}(\mathbf{h})$, $f_a(\mathbf{a}) = -Q_{\boldsymbol{\psi}_Q}(\mathbf{h}, \mathbf{a})$, $\mathbf{u} = \nabla_{\mathbf{a}} f_a(\mu)$. Smoothness of f_a on the deviation ball gives, for $\|\mathbf{e}\| \le \varepsilon$,

$$f_a(\mu + \mathbf{e}) \;\le\; f_a(\mu) + \mathbf{u}^\top \mathbf{e} + \tfrac{\beta_a}{2} \|\mathbf{e}\|^2 \;\le\; f_a(\mu) + \varepsilon \|\mathbf{u}\| + \tfrac{\beta_a}{2} \varepsilon^2.$$

This bounds the *learned* critic's perturbed cost. To reach Q^π, add and subtract: $-Q^\pi(\mathbf{h}, \mu + \mathbf{e}) \le f_a(\mu + \mathbf{e}) + \sup_{\|\mathbf{e}\| \le \varepsilon} |Q^\pi - Q_{\boldsymbol{\psi}_Q}| = f_a(\mu + \mathbf{e}) + \Delta_Q$, and replace $\mathbf{u}$ by the true action-cost gradient at the cost of $\delta_Q \varepsilon$ via Assumption 1. Taking the supremum over $\|\mathbf{e}\| \le \varepsilon$ gives the first inequality. For the second, apply Lemma 1(i) to f_a at μ, which gives $\|\mathbf{u}\| \le \sqrt{2(1 + \beta_a \tau_a) C_{\tau_a}^{(a)} / \tau_a}$, and substitute. No property of $\mathbf{J}$ is used anywhere. □

D.3 Lifting to trajectory risk

Proposition 2 (Lifting to trajectory risk)**.** *Let the execution deviation satisfy* $\|\mathbf{e}_t\| \le \varepsilon$ *at every step, let the discounted occupancy of* $\pi_{\boldsymbol{\theta}}$ *be* d^π*, and suppose Theorem 1 holds* d^π*-almost surely. Then the discounted trajectory risk obeys*

$$R_\varepsilon(\boldsymbol{\theta}) \;\le\; J(\boldsymbol{\theta}) + \frac{1}{1 - \gamma} \mathbb{E}_{\mathbf{h} \sim d^\pi} \Big[\varepsilon \|\mathbf{u}(\mathbf{h})\| + \delta_Q \varepsilon + \tfrac{\beta_a \varepsilon^2}{2} + \Delta_Q \Big].$$

The factor $(1 - \gamma)^{-1}$ *(or the horizon* H *in the undiscounted case) is the usual simulation-lemma cost of compounding one-step deviations and is not removable.*

Proof. Let $\pi_{\mathbf{e}}$ denote the executed policy, which at every visited history plays $\mu_{\boldsymbol{\theta}}(\mathbf{h}) + \mathbf{e}(\mathbf{h})$ with $\|\mathbf{e}(\mathbf{h})\| \le \varepsilon$. The performance-difference (simulation) lemma gives

$$J(\pi_{\mathbf{e}}) - J(\pi_{\boldsymbol{\theta}}) = \frac{1}{1-\gamma}\, \mathbb{E}_{\mathbf{h}\sim d^{\pi_{\mathbf{e}}}}\big[\, -Q^{\pi}(\mathbf{h}, \mu_{\boldsymbol{\theta}}(\mathbf{h}) + \mathbf{e}(\mathbf{h})) + Q^{\pi}(\mathbf{h}, \mu_{\boldsymbol{\theta}}(\mathbf{h}))\,\big].$$

Bounding the bracket pointwise by Theorem 1 and taking the supremum over admissible $\mathbf{e}$ yields the claim, with the occupancy taken under $\pi_{\mathbf{e}}$; replacing $d^{\pi_{\mathbf{e}}}$ by d^{π} is exact in the on-policy limit and otherwise contributes a further total-variation term of order ε, which we absorb into Δ_Q. The factor $(1-\gamma)^{-1}$ is the standard compounding cost and is tight for the chain counterexample of Ross et al. (2011). □

D.4 Proof of Proposition 1

(i) The chain rule gives $\nabla_{\boldsymbol{\theta}} Q = \mathbf{J}^{\top}\mathbf{u}$, so $\|\nabla_{\boldsymbol{\theta}} Q\| \le \|\mathbf{J}\|\,\|\mathbf{u}\|$, and Lemma 1(i) applied to f_a bounds $\|\mathbf{u}\|$ by $\sqrt{2(1+\beta_a\tau_a)\eta/\tau_a}$ when $C_{\tau_a}^{(a)} \le \eta$. *(ii)* Since $\mathbf{J}^{\top}\mathbf{u}_{\perp} = \mathbf{0}$, $\nabla_{\boldsymbol{\theta}} Q = \mathbf{J}^{\top}\mathbf{u}_{\parallel}$, which does not depend on $\mathbf{u}_{\perp}$. Hence $\|\nabla_{\boldsymbol{\theta}} Q\| \le \|\mathbf{J}\|\,\|\mathbf{u}_{\parallel}\| = \|\mathbf{J}\|\sqrt{1-\varrho^2}\;\|\mathbf{u}\|$, and any admissible κ satisfies $\kappa \ge \|\mathbf{u}\| / \|\nabla_{\boldsymbol{\theta}} Q\| \ge 1/(\|\mathbf{J}\|\sqrt{1-\varrho^2})$. If $\varrho = 1$, then $\nabla_{\boldsymbol{\theta}} Q = \mathbf{0}$ while $\|\mathbf{u}\| > 0$, so no finite κ exists. Conversely, if $\varrho = 0$, then $\mathbf{u}$ lies in the orthogonal complement of $\ker(\mathbf{J}^{\top})$, on which $\mathbf{J}^{\top}$ acts injectively with smallest singular value $\sigma_+(\mathbf{J}) > 0$, so $\|\nabla_{\boldsymbol{\theta}} Q\| \ge \sigma_+(\mathbf{J})\,\|\mathbf{u}\|$ and $\kappa = 1/\sigma_+(\mathbf{J})$ suffices. □

E Revision-Gap Monitor: Formal Results

Proposition 3 (Gap-stop validity and calibration)**.** *Under Assumption 2 and* $\|\mathbf{g}_k\|^2 > \varsigma$*, define* $\rho_k = \widehat{C}_{\tau,k}/(\frac{\tau}{2}\|\mathbf{g}_k\|^2)$*. Then:* (i) $\rho_k \in [\,1-\beta\tau,\ 1/(1-\beta\tau)\,]$*;* (ii) *the criterion* $\rho_k > T_\rho$ *can fire only if* $T_\rho < 1/(1-\beta\tau)$*, i.e. only if* $\beta\tau > 1 - 1/T_\rho$*, so a threshold* $T_\rho \ge 1/(1-\beta\tau)$ *never fires and any firing certifies that the local curvature is large relative to the probe step;* (iii) *choosing* $T_\rho = 1/(1-\bar{t})$ *makes the criterion fire only when the local* $\beta\tau$ *exceeds the design value* $\bar{t}$*, which for* $\bar{t} = 1/4$ *is the regime in which Lemma 1(iii) ceases to guarantee descent.*

Proof. (i) Lemma 1(ii) gives $\widehat{C}_\tau \ge \frac{\tau}{2}(1-\beta\tau)\,\|\mathbf{g}\|^2$, hence $\rho \ge 1-\beta\tau$; and $\widehat{C}_\tau \le C_\tau \le \frac{\tau\|\mathbf{g}\|^2}{2(1-\beta\tau)}$ by Lemma 1(i), hence $\rho \le 1/(1-\beta\tau)$. Both require $\|\mathbf{g}\|^2 > \varsigma > 0$ for ρ to be defined. *(ii)* If $T_\rho \ge 1/(1-\beta\tau)$ then $\rho \le T_\rho$ always, so the test never fires; contrapositively, a firing implies $T_\rho < 1/(1-\beta\tau)$, i.e. $\beta\tau > 1 - 1/T_\rho$. *(iii)* Substituting $T_\rho = 1/(1-\bar{t})$ into (ii) gives firing $\Rightarrow \beta\tau > \bar{t}$. Taking $\bar{t} = 1/4$ reproduces the hypothesis of Lemma 1(iii). □

Certified smoothness lower bound. From Proposition 3(i), $\rho \ge 1-\beta\tau$ gives $\beta\tau \ge 1-\rho$, and $\rho \le 1/(1-\beta\tau)$ gives $1-\beta\tau \le 1/\rho$, i.e. $\beta\tau \ge 1 - 1/\rho$. Both hold simultaneously, so $\beta \ge \frac{1}{\tau}\max\{1-\rho,\, 1-1/\rho\} = \hat{\beta}$, which is

$$\hat{\beta}_k = \frac{1}{\tau}\max\Big\{\, 1-\rho_k,\ 1-\tfrac{1}{\rho_k}\,\Big\} \;\le\; \beta. \tag{11}$$

For $\rho \in (0,1)$ the first branch is the active (positive) one and the second is negative; for $\rho > 1$ the roles reverse; at $\rho = 1$ both vanish and the bound is $\hat{\beta} = 0$, correctly reporting that a gap exactly equal to $\frac{\tau}{2}\|\mathbf{g}\|^2$ certifies nothing about curvature. □

Local risk descent.

Proposition 4. *If* $\|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}, \mathbf{e}) - \mathbf{g}\| \le L_{\boldsymbol{\theta}\mathbf{e}}\,\|\mathbf{e}\|$*,* $\beta\tau \le 1/4$*,* $L_{\boldsymbol{\theta}\mathbf{e}}\varepsilon \le \|\mathbf{g}\|/4$*, and* $0 < \alpha \le 1/(4\beta)$*, then* $R_\varepsilon(\boldsymbol{\theta} - \alpha\mathbf{v}) \le R_\varepsilon(\boldsymbol{\theta}) - \frac{\alpha}{8}\|\mathbf{g}\|^2$.

Proof. The bound $\beta\tau \le 1/4$ gives $\|\mathbf{v} - \mathbf{g}\| \le \|\mathbf{g}\|/4$. The cross-sensitivity assumption gives $\|\mathbf{g}_{\mathbf{e}} - \mathbf{g}\| \le \|\mathbf{g}\|/4$, so $\langle \mathbf{g}_{\mathbf{e}}, \mathbf{v}\rangle \ge \frac{7}{16}\|\mathbf{g}\|^2$. The descent lemma for each fixed $\mathbf{e}$ yields the result. □

F Finite-Sample Conditions

Let $S = \{\xi_i\}_{i=1}^n$ be independent tasks, each loss in $[0, M]$, and define $\mathcal{G} = \{\xi \mapsto \mathcal{L}(\boldsymbol{\theta}, \mathrm{e}; \xi) : \boldsymbol{\theta} \in \Theta, \|\mathrm{e}\| \le \varepsilon\}$. A Rademacher bound (Bartlett and Mendelson, 2002) gives, with probability $\ge 1-\delta$, $J_{\mathcal{D}}(\boldsymbol{\theta}, \mathrm{e}) \le J_S(\boldsymbol{\theta}, \mathrm{e}) + \Gamma_n$ with $\Gamma_n = 2\widehat{\mathfrak{R}}_S(\mathcal{G}) + 3M\sqrt{\log(2/\delta)/(2n)}$. If the empirical cost satisfies Theorem 1, then $R_{\mathcal{D},\varepsilon}(\boldsymbol{\theta}) \le F^{(a)}_{\tau,S}(\boldsymbol{\theta}) + \Gamma_n + \frac{\varepsilon^2(1+\beta_a\tau_a)}{2\tau_a} + \delta_Q\varepsilon + \frac{\beta_a\varepsilon^2}{2}$.

G Step-Size Effects in a Scalar Example

Let $J(\theta) = \frac{1}{2}(\theta - y)^2$ with test target zero and $\theta_0 = 0$. Then $\theta_{\mathrm{GD}} = \alpha y$ and $\theta_{\mathrm{SC}} = \alpha(1+\tau)y$, so the expected test error of the SC update is $(1+\tau)^2$ times that of gradient descent at equal α.

H Relationship Between Probe Directions

If $\nabla^2 f$ is L_H-Lipschitz on the probe segments, then $\|2g - \nabla f(x-d) - \nabla f(x+d)\| \le L_H \|d\|^2$. With $d = \tau g$, the descent-probe and ascent-probe directions differ by at most $L_H\tau^2\|g\|^2$; for quadratic f they coincide. The update directions are $O(\tau^2)$-close, but the monitors they produce differ by $\Theta(\tau)$ (Lemma 1(ii)).

I Reproducibility and Protocol Specification

Paired evaluation. Use the same batch, evaluator version, targets, and seed within a pair. Recompute every quantity that depends on the probed policy. A zero-displacement probe must reproduce the nominal loss and gradient to numerical precision.

Gradient scope. Declare whether $\boldsymbol{\theta}$ includes the encoder, the action head, or both. Freeze evaluator weights without detaching their action derivatives. Report SGD and adaptive-optimizer runs separately.

Q-head specification. The Q-head architecture (Eq. 7) uses a two-layer MLP with the action concatenated after the shared encoder output. Report hidden dimensions, activation function, and whether the Q-head gradient flows into the encoder.

Compute accounting. Report environment steps, actor and critic backward passes, Q-head evaluations, wall-clock time, and peak memory for every method. Compare interaction-matched and actor-compute-matched regimes.

Fault injection. Faults are added to the commanded (v_t, ω_t) at the control rate, after the policy output and before clipping to the velocity limits. Gaussian noise is drawn from a seeded generator that is shared across methods, so every method faces the same noise sequence on the same episode. These values are fixed before evaluation.

Diagnostics logged. For SC-PPO runs we log $\widehat{C}_\tau$, ρ_k, $\hat{\beta}_k$, the clip-crossing fraction of Appendix C, the visibility diagnostic ϱ, and an estimate of the critic error δ_Q.

Evaluation records. Publish map splits, seeds, starts and goals, simulator settings, action scaling, control rate, termination criteria, per-condition confidence intervals, and all failed runs.

J Baseline Implementations

All baselines share the network, the PPO hyperparameters, the environment budget, and the evaluation protocol of SC-PPO; only the components listed below differ. Baseline-specific hyperparameters, and η_a for SC-PPO, are selected on the validation split by the mean of the nominal SR and the SR under a validation perturbation (uniform action noise) that is not among the test faults, so that no method is tuned on the test faults.

PPO. Clipped PPO with GAE, advantage normalization, and approximate-KL early stopping at threshold 0.03 (Huang et al., 2022).

SAM+PPO. Following Lee and Yoon (2025), each minibatch step computes $\mathbf{g} = \nabla_{\boldsymbol{\theta}} J_k(\boldsymbol{\theta})$, evaluates $\mathbf{g}^+ = \nabla_{\boldsymbol{\theta}} J_k(\boldsymbol{\theta} + \rho\, \mathbf{g}/\,\|\mathbf{g}\|)$ on the same minibatch, and updates $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \mathbf{g}^+$; only the actor loss is perturbed. The radius grid is $\rho \in \{0.001, 0.005, 0.01, 0.05\}$ together with the median of $\tau_\theta \,\|\mathbf{g}_k\|$ from SC-PPO runs, so that one grid point matches the probe displacement of PSC. Its actor compute is $2\times$ that of PPO.

GR+PPO. The actor loss is $J_k(\boldsymbol{\theta}) + \frac{\lambda_{\mathrm{GR}}}{2} \,\|\nabla_{\boldsymbol{\theta}} J_k(\boldsymbol{\theta})\|^2$ (Barrett and Dherin, 2021; Zhao et al., 2022), the first-order objective approximated by PSC (Section 3.2), evaluated on the same minibatch; only the actor loss is penalized. λ_{GR} is selected from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$.