CONSTRAINT-ENHANCED REINFORCEMENT LEARNING BASED ON DYNAMIC DECOUPLED SPHERICAL RADIAL SQUASHING

Anonymous authors
Paper under double-blind review

ABSTRACT

When deploying reinforcement learning policies to physical robots, actuator rate constraints—hard limits on how fast each joint command may change per control step—must be respected. These limits can vary substantially across joints, producing a high-dimensional box in action-increment space. Existing solver-based projections introduce runtime optimization and training–execution inconsistency, whereas single-radius spherical parameterizations inscribe an isotropic ℓ_2 ball and increasingly under-cover the true feasible set as dimensionality and rate-limit heterogeneity grow. This paper proposes Dynamic Decoupled Spherical Radial Squashing (DD-SRad), which computes a position-adaptive effective radius independently for each actuator and tightly aligns the closure of its reachable set with the heterogeneous ℓ_∞ feasible box. For every latent action, DD-SRad deterministically satisfies the per-step rate and position constraints. Its analytic, per-dimension mapping has a diagonally positive Jacobian at finite latent values, polynomial gradient decay, and requires no runtime optimization solver. Across MuJoCo benchmarks, DD-SRad attains the highest return among the constrained methods with zero constraint violations and improves constraint-space utilization by 30%–50% over spherical baselines. High-fidelity IsaacLab experiments on Unitree H1 and G1, using specification-consistent rate limits and native task reward configurations, provide hardware-relevant simulation evidence while not constituting physical-robot validation. Code and rollout videos are available at `https://anonymous.4open.science/r/DD-SRad`.

1 INTRODUCTION

Physical deployment of reinforcement-learning policies requires every commanded action to be actuator-executable, not merely reward-maximizing. Per-step joint-command rate limits arise from motor inertia, amplifier bandwidth, and transmission stiffness; violations can be discarded, trigger protection, or damage hardware. These limits are heterogeneous—Unitree H1 hip rates are 2.2$\times$ its ankle rates, and industrial manipulators may exceed a 5:1 proximal-to-distal ratio Wensing et al. (2017)—so deployment-oriented RL must handle an axis-aligned rather than isotropic action-increment set.

The constraint $|a_t^i - a_{t-1}^i| \le \delta^i$ defines the moving feasible box $\mathcal{F}_t = \prod_i [a_{t-1}^i - \delta^i, a_{t-1}^i + \delta^i]$. MPC/QP enforces it through runtime optimization but separates learned and executed actions Grandia et al. (2023); Bledt et al. (2018); Kim et al. (2019); Pandala et al. (2022); Williams et al. (2017); Nagabandi et al. (2018); CMDP methods generally bound expected costs rather than every step Achiam et al. (2017); Tessler et al. (2019); Zhang et al. (2020); Yang et al. (2022); Zhang et al. (2022); Altman (1999); and existing parameterizations either inscribe an isotropic ℓ_2 ball Kasaura et al. (2023) or assume a static feasible set Brahmanage et al. (2023); Hung et al. (2025); Stolz et al. (2025). The spherical reachable-volume fraction $(\min_i \delta^i)^d / \prod_i \delta^i$ decays exponentially with dimension and heterogeneity. No prior method jointly provides deterministic per-step satisfaction, exact coverage of the moving ℓ_∞ box, non-truncated finite-latent per-dimension gradients, and solver-free end-to-end off-policy training (Table 1).

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.

Table 1: Comparison of representative rate-constraint handling paradigms.

Method	Hard per step	Heterog. limits	Exact box	Moving set	Solver free	Off-policy E2E
RL+MPC/QP	✓	Partial	✗	✓	✗	✗
CMDP	✗	✗	–	–	✓	✓
TN-SAC	✓	✓	✓	✗	✓	✓
SRad	✓	✗	✗	✗	✓	✓
FlowPG / ARAM	✓	✓	✓	✗	✓	✗
DD-SRad (Ours)	✓	✓	✓	✓	✓	✓

Hard per step: every control step satisfies the actuator rate limits. *Heterogeneous limits*: different joints may have different rate limits. *Moving set*: the valid action region changes with the previous action.

To fill this gap, we propose **Dynamic Decoupled Spherical Radial Squashing (DD-SRad)**, an analytic action parameterization that simultaneously provides deterministic per-step constraint satisfaction, tight ℓ_∞ coverage, and end-to-end gradient backpropagation without a runtime solver. Rather than constructing an arbitrary surjection onto a hyperrectangle, DD-SRad computes a *position-adaptive effective radius* R^i_{eff} independently for each dimension and applies a polynomial-decay squashing map. Its reachable-set closure equals the full rate–position feasible box $\mathcal{F}_t$, while its diagonal Jacobian retains nonzero per-coordinate sensitivity at every finite differentiable latent action. The layer plugs directly into SAC and TD3 without modifying their actor–critic architectures.

The contributions are:

- **Geometric alignment**: independent position-adaptive radii tightly cover the heterogeneous ℓ_∞ box and recover the volume lost by single-radius ℓ_2 maps.
- **Deterministic guarantees**: every latent action satisfies the rate and position constraints; the mapping is $\|\boldsymbol{\delta}\|_\infty$-Lipschitz with an explicit bounded-latent conditioning characterization.
- **Gradient compatibility**: the diagonal positive Jacobian avoids clipping's exact finite-latent dead zones and decays polynomially rather than exponentially, without runtime optimization.
- **Empirical validation**: DD-SRad attains the highest mean return among constrained methods with zero violations across MuJoCo and specification-consistent IsaacLab H1/G1 simulations.

2 Methodology

2.1 Preliminaries

Spherical Radial Squashing (SRad). SRad Kasaura et al. (2023) maps latent $u \in \mathbb{R}^d$ to

$$a = c_s + R \cdot \frac{u}{\sqrt{1 + \|u\|^2}} \tag{1}$$

where c_s and R are the ball center and global radius. Although $\|a - c_s\| < R$ and $\nabla_u a \neq 0$, rate constraints require $R = \min_i \delta^i$, coupling all dimensions inside an ℓ_2 ball. DD-SRad instead uses independent R^i_{eff}, exactly aligning the reachable-set closure with $\mathcal{F}_t$ (Theorem 2.7) while retaining finite-latent sensitivity (Proposition 2.5).

For state $s_t \in \mathcal{S} \subseteq \mathbb{R}^n$ and bounded action space $\mathcal{A} = \prod_i [a^i_{\min}, a^i_{\max}]$, the heterogeneous rate constraint is

$$|a^i_t - a^i_{t-1}| \leq \delta^i, \quad \forall i \in \{1, \ldots, d\},\ \forall t \geq 1 \tag{2}$$

where $\boldsymbol{\delta} \in \mathbb{R}^d_{++}$ may be heterogeneous. Augmenting the state as $\tilde{s}_t = (s_t, a_{t-1})$ restores Markovianity (Appendix B.1) but does not enforce the constraint; DD-SRad supplies that output mechanism. We assume bounded rewards, compact $\mathcal{A}$, and Lipschitz dynamics.

2.2 Dynamic Decoupled Spherical Radial Squashing

2.2.1 The DD-SRad Mapping

Definition 2.1 (Per-Dimension Effective Radius). *Given per-dimension rate limits $\boldsymbol{\delta} \in \mathbb{R}^d_{++}$, previous action $a_{prev} \in \mathcal{A}$, and position bounds $[a^i_{\min}, a^i_{\max}]$, define the effective radius for dimension i as:*

$$R^i_{eff}(u^i, a^i_{prev}) = \begin{cases} \min\left(\delta^i,\ a^i_{\max} - a^i_{prev}\right), & \text{if } u^i > 0 \\ \min\left(\delta^i,\ a^i_{prev} - a^i_{\min}\right), & \text{if } u^i < 0 \\ \delta^i, & \text{if } u^i = 0 \end{cases} \tag{3}$$

Definition 2.2 (DD-SRad Mapping). *For latent $u \in \mathbb{R}^d$ and previous action $a_{prev} \in \mathcal{A}$, DD-SRad acts independently per dimension:*

$$a^i = a^i_{prev} + R^i_{eff}(u^i, a^i_{prev}) \cdot \frac{u^i}{\sqrt{1 + (u^i)^2}}, \quad \forall i \in \{1, \dots, d\} \tag{4}$$

Equivalently, $\mathbf{a} = \mathbf{a}_{\text{prev}} + \mathbf{R}_{\text{eff}} \odot \mathbf{f}(\mathbf{u})$, where $f^i(u) = u^i/\sqrt{1 + (u^i)^2}$ and $\odot$ is element-wise multiplication.

Remark 2.3 (Geometric Comparison). *SRad constructs an ℓ_2 ball with $R = \min_i \delta^i$, whereas DD-SRad's reachable-set closure is the ℓ_∞ box $\prod_i [a^i_{prev} - R^i_{eff}, a^i_{prev} + R^i_{eff}]$ (Theorem 2.7). Hard ℓ_∞ clipping has the same closure but truncates gradients at active boundaries; DD-SRad retains nonzero sensitivity at every finite differentiable latent value.*

An optional global ℓ_2 numerical safeguard triggers in less than 0.1% of steps and is specified in Appendix C; the per-dimension map supplies the guarantee.

2.3 Theoretical Guarantees

Proofs are deferred to Appendix B.

2.3.1 Hard Constraint Satisfaction

The executed action must satisfy both the rate limit and physical position interval for every latent output, rather than only in expectation.

Theorem 2.4 (Heterogeneous Rate Constraint Satisfaction). *For any $a_{prev} \in \mathcal{A}$, any latent action $u \in \mathbb{R}^d$, and any per-dimension rate limits $\boldsymbol{\delta} \in \mathbb{R}^d_{++}$, the DD-SRad mapping (4) deterministically guarantees*

$$|a^i - a^i_{prev}| \le \delta^i \quad \textit{and} \quad a^i \in [a^i_{\min}, a^i_{\max}], \qquad \forall i. \tag{5}$$

Furthermore, when $a^i_{prev} \in (a^i_{\min}, a^i_{\max})$, both constraints hold with strict inequalities.

Proof: Appendix B.2.

2.3.2 Jacobian Structure and Bounded-Latent Conditioning

The squashing choice affects policy optimization even when two maps share the same reachable set. DD-SRad's derivative decays polynomially as $O(|u^i|^{-3})$, whereas the controlled D-Tanh ablation decays exponentially. The following proposition makes explicit that conditioning is local to a bounded latent region, not global over $\mathbb{R}^d$.

Proposition 2.5 (Jacobian Structure and Gradient Norm Bound). *Let $\kappa \triangleq \max_i \delta^i / \min_i \delta^i \ge 1$. At differentiable points with $a^i_{prev} \in (a^i_{\min}, a^i_{\max})$, the Jacobian $J(u) = \partial a / \partial u$ satisfies:*

(i) $J(u)$ is diagonal, and for every $u^i \neq 0$,

$$\frac{\partial a^i}{\partial u^i} = \frac{R^i_{eff}(u^i, a^i_{prev})}{(1 + (u^i)^2)^{3/2}} > 0; \tag{6}$$

(ii) The global spectral-norm bound

$$\|J(u)\|_2 \le \max_i \delta^i \tag{7}$$

holds for every differentiable u. For any $\|u\|_\infty \le B$, define

$$\rho_{\min}(u) \triangleq \min_i R^i_{\textit{eff}}(u^i, a^i_{\textit{prev}}), \qquad \rho_{\max}(u) \triangleq \max_i R^i_{\textit{eff}}(u^i, a^i_{\textit{prev}}).$$

Whenever $\rho_{\min}(u) > 0$,

$$\mathrm{cond}_2(J(u)) \le \frac{\rho_{\max}(u)}{\rho_{\min}(u)}(1+B^2)^{3/2}. \tag{8}$$

Away from position boundaries, where $R^i_{\textit{eff}} = \delta^i$ on the active sign branches,

$$\mathrm{cond}_2(J(u)) \le \kappa(1+B^2)^{3/2}, \qquad \mathrm{cond}_2(J(\mathbf{0})) = \kappa. \tag{9}$$

(iii) For the TD3 backbone, the gradient of the pointwise actor objective satisfies

$$\|\nabla_u \mathcal{L}_{\textit{actor}}\|_2 \le \max_i \delta^i \, \|\nabla_a Q(\tilde{s}, a)\|_2 + 2\lambda_{\textit{base}}\|u\|_2, \tag{10}$$

where $\lambda_{\textit{base}}$ is the latent-action regularization coefficient formally defined in the actor objective in Eq. (14). For SAC, the entropy contribution $\alpha\|\nabla_u \log \pi(u \mid \tilde{s})\|_2$ is additionally included.

Proof: Appendix B.3. Under bounded actor-gradient and Lipschitz-critic assumptions, a pointwise critical point obeys $\|u\|_\infty \le \max_i \delta^i G/(2\lambda_{\text{base}})$, yielding a finite stationary-point conditioning bound for every $\lambda_{\text{base}} > 0$, but not a global training-trajectory bound. At $u^i = 0$, unequal position margins give valid one-sided derivatives; full derivations and autodiff handling are in Appendix C.

Proposition 2.6 (Lipschitz Continuity)**.** *For fixed $a_{\textit{prev}} \in \mathcal{A}$, the DD-SRad mapping $\phi(\cdot, a_{\textit{prev}})$ is globally $\|\boldsymbol{\delta}\|_\infty$-Lipschitz under the Euclidean norm:*

$$\|\phi(u, a_{\textit{prev}}) - \phi(\hat{u}, a_{\textit{prev}})\|_2 \le \|\boldsymbol{\delta}\|_\infty \|u - \hat{u}\|_2. \tag{11}$$

The constant $\|\boldsymbol{\delta}\|_\infty$ is tight when the previous action is at least δ^i away from both position boundaries in a dimension attaining $\max_i \delta^i$.

Proof: Appendix B.4.

2.3.3 Tight ℓ_∞-Coverage of the Reachable Set

Hard satisfaction alone does not distinguish DD-SRad from clipping or other feasible-set maps. The following theorem isolates the geometric contribution: DD-SRad recovers the complete joint rate–position box in closure, whereas a single-radius map remains inside an inscribed ℓ_2 ball.

Theorem 2.7 (Tight Coverage of the Reachable Set with the ℓ_∞ Feasible Set)**.** *Fix $a_{\textit{prev}} \in \mathcal{A}$, and let the joint rate-position feasible set be $\mathcal{F} = \{a \in \mathbb{R}^d : |a^i - a^i_{\textit{prev}}| \le \delta^i\} \cap \mathcal{A}$. The closure of the reachable set of the DD-SRad mapping satisfies*

$$\overline{\mathcal{R}_{\textit{DD}}} = \prod_{i=1}^{d} \left[a^i_{\textit{prev}} - R^{-,i},\, a^i_{\textit{prev}} + R^{+,i}\right] = \mathcal{F}, \tag{12}$$

with Lebesgue measure $V_{\textit{DD}} = \prod_{i=1}^d (R^{-,i} + R^{+,i})$. Relative to ℓ_2 spherical parameterization (SRad) with global radius $R = \min_i \delta^i$, the volume ratio away from position boundaries ($R^i_{\textit{eff}} = \delta^i$) is

$$\frac{V_{\textit{DD}}}{V_{\textit{SRad}}} = \frac{2^d \Gamma(d/2+1)}{\pi^{d/2}} \cdot \frac{\prod_i \delta^i}{(\min_i \delta^i)^d}, \tag{13}$$

where $V_{\textit{SRad}} = \frac{\pi^{d/2}}{\Gamma(d/2+1)}(\min_i \delta^i)^d$ is the d-dimensional ℓ_2 ball volume and $\Gamma(\cdot)$ is the Gamma function.

Proof: Appendix B.5.

2.4 Policy Optimization with DD-SRad Parameterization

Upon embedding the DD-SRad mapping into the off-policy actor network, the actor loss for SAC/TD3 backbones takes the unified form

$$\mathcal{L}_{\text{actor}} = \mathbb{E}_{(s,a_{\text{prev}})\sim\mathcal{D}}[\alpha \log \pi_\theta(a|\tilde{s}) - Q_\phi(\tilde{s}, a)] + \lambda_{\text{base}}\, \mathbb{E}[\|u\|^2], \tag{14}$$

where $\mathcal{D}$ is the replay buffer, α is the SAC temperature ($\alpha = 0$ for TD3), and $\lambda_{\text{base}} > 0$ regularizes the latent action. For SAC with $u \sim \mathcal{N}(\mu_\theta, \sigma_\theta^2)$, change of variables gives

$$\log \pi_\theta(a|\tilde{s}) = \log \mathcal{N}(u; \mu_\theta(s), \sigma_\theta^2(s)) - \sum_{i=1}^{d} \log \frac{\partial a^i}{\partial u^i}, \tag{15}$$

with the closed-form log determinant

$$\sum_{i=1}^{d} \log \frac{\partial a^i}{\partial u^i} = \sum_{i=1}^{d} \left[\log R^i_{\text{eff}}(u^i, a^i_{\text{prev}}) - \frac{3}{2} \log\big(1 + (u^i)^2\big) \right]. \tag{16}$$

This expression is smooth in θ at $u \neq 0$ and adds no iterative overhead. The regularizer confines u to the effective-gradient regime—Jacobian entries approach zero as $\|u\|$ grows—and mitigates cross-dimensional imbalance under heterogeneous limits (Eq. 10); ğ3.3 verifies its robustness. The complete procedure is given in Appendix Algorithm 1.

2.5 Gradient Propagation Through Off-Policy Backbones

DD-SRad is compatible with action-conditioned off-policy critics such as SAC Haarnoja et al. (2018) and TD3 Fujimoto et al. (2018). For map $\phi_{\tilde{s}}$ conditioned on a_{prev}, the actor gradient is

$$\nabla_\theta \mathcal{L}_{\text{actor}} = \mathbb{E}_{\tilde{s}} \left[\nabla_a Q(\tilde{s}, a)\big|_{a=\phi_{\tilde{s}}(u_\theta)} \frac{\partial \phi_{\tilde{s}}}{\partial u} \frac{\partial u}{\partial \theta} \right]. \tag{17}$$

At finite differentiable latent values, $\partial\phi_{\tilde{s}}/\partial u$ is diagonal with strictly positive entries (Proposition 2.5(i)); hence no coordinate is exactly removed from the critic-gradient signal, although its magnitude decays as $|u^i|$ grows. TD3 uses the same chain rule without entropy. For on-policy methods such as PPO Schulman et al. (2017), the mapping still guarantees feasibility after augmenting observations with a_{t-1}, but scalar-advantage estimates can have higher variance than the per-dimension $\nabla_a Q$ signal under high heterogeneity (Appendix K).

3 MuJoCo Benchmarks

Three five-seed experiment groups test ğ2.3: ğ3.1 evaluates deterministic satisfaction and smooth versus gradient-truncating maps; ğ3.2 tests ℓ_∞ coverage through per-dimension utilization; and ğ3.3 varies λ_{base}. We report Return↑, Utilization $\mathbb{E}[\|\Delta a\|_1]/\|\boldsymbol{\delta}\|_1$, and per-step constraint-violation rate (CVR); stability analyses additionally use $\text{CV} = \sigma/|\mu| \times 100\% \downarrow$ across seeds. Experiments use MuJoCo v2.3.7/Gymnasium v0.29; Appendix D gives all hyperparameters.

We evaluate SAC Haarnoja et al. (2018) and TD3 Fujimoto et al. (2018). The controlled **D-Tanh** ablation retains DD-SRad's per-dimension R^i_{eff} and box geometry but substitutes $\tanh$ to isolate Jacobian decay. **BoxPre+** uses critic-aware ℓ_∞ clipping Kasaura et al. (2023); **Post(QP)** is constraint-unaware post-processing; **SRad-Strict** uses $R = \min_i \delta^i$; and **SRad-QP** appends QP projection to SRad.

3.1 Benchmark Performance

Experiments cover Ant-v5 (8D), Humanoid-v5 (17D), HalfCheetah-v5 (6D), and Hopper-v5 (3D), each trained for 1.5M–2M steps under wide homogeneous and tight heterogeneous limits (Appendix Table 5).

Table 2 gives tight-heterogeneous results and Appendix Table 8 the wide case. Parameterization-based methods have CVR$= 0$ by construction; critic-unaware SAC-Post(QP) may violate tight limits

Table 2: Tight-heterogeneous MuJoCo results (5-seed mean±std; best in **bold**; † see note).

(a) SAC Backbone

	Ant-v5 ($\boldsymbol{\delta} = [0.2^4, 0.5^4]$, $\kappa = 2.5$)		**Humanoid-v5** ($\boldsymbol{\delta} = [0.8^6, 0.5^6, 0.2^5]$, $\kappa = 4.0$)	
Method	Return ↑	Util	Return ↑	Util
BoxPre+	1998 ± 125	0.453 ± 0.026	5204 ± 176	0.159 ± 0.033
Post(QP)†	1517 ± 137	0.651 ± 0.015	5191 ± 346	0.249 ± 0.045
SRad-Strict	1448 ± 996	0.141 ± 0.029	2742 ± 1350	0.024 ± 0.007
SRad-QP	1703 ± 232	0.583 ± 0.035	4558 ± 500	0.053 ± 0.024
D-Tanh	2719 ± 174	0.570 ± 0.037	4868 ± 267	0.662 ± 0.069
DD-SRad (Ours)	$\mathbf{3112 \pm 788}$	0.492 ± 0.051	$\mathbf{5620 \pm 215}$	0.489 ± 0.035
	HalfCheetah-v5 ($\boldsymbol{\delta} = [0.2^3, 0.5^3]$, $\kappa = 2.5$)		**Hopper-v5** ($\boldsymbol{\delta} = [0.2, 0.5, 0.5]$, $\kappa = 2.5$)	
Method	Return ↑	Util	Return ↑	Util
BoxPre+	3796 ± 59	0.600 ± 0.016	2319 ± 208	0.267 ± 0.015
Post(QP)†	1798 ± 246	0.767 ± 0.022	1278 ± 165	0.353 ± 0.014
SRad-Strict	1432 ± 147	0.213 ± 0.014	2195 ± 735	0.155 ± 0.020
SRad-QP	2482 ± 433	0.557 ± 0.018	1624 ± 332	0.370 ± 0.009
D-Tanh	4205 ± 231	0.662 ± 0.069	2334 ± 276	0.480 ± 0.011
DD-SRad (Ours)	$\mathbf{4290 \pm 682}$	0.626 ± 0.095	$\mathbf{2610 \pm 183}$	0.461 ± 0.007

(b) TD3 Backbone

	Ant-v5 ($\boldsymbol{\delta} = [0.2^4, 0.5^4]$, $\kappa = 2.5$)		**Humanoid-v5** ($\boldsymbol{\delta} = [0.8^6, 0.5^6, 0.2^5]$, $\kappa = 4.0$)	
Method	Return ↑	Util	Return ↑	Util
BoxPre+	2917 ± 289	0.518 ± 0.018	5291 ± 119	0.179 ± 0.030
Post(QP)†	1998 ± 282	0.689 ± 0.029	4420 ± 117	0.465 ± 0.036
SRad-Strict	1246 ± 376	0.176 ± 0.015	2270 ± 1126	0.053 ± 0.013
SRad-QP	2147 ± 195	0.554 ± 0.011	3810 ± 332	0.015 ± 0.003
D-Tanh	3484 ± 781	0.565 ± 0.017	5313 ± 341	0.184 ± 0.039
DD-SRad (Ours)	$\mathbf{4260 \pm 176}$	0.548 ± 0.021	$\mathbf{5498 \pm 203}$	0.203 ± 0.024
	HalfCheetah-v5 ($\boldsymbol{\delta} = [0.2^3, 0.5^3]$, $\kappa = 2.5$)		**Hopper-v5** ($\boldsymbol{\delta} = [0.2, 0.5, 0.5]$, $\kappa = 2.5$)	
Method	Return ↑	Util	Return ↑	Util
BoxPre+	3641 ± 681	0.607 ± 0.081	3256 ± 196	0.303 ± 0.014
Post(QP)†	1484 ± 1042	0.695 ± 0.060	849 ± 251	0.487 ± 0.023
SRad-Strict	1518 ± 185	0.211 ± 0.011	2394 ± 322	0.166 ± 0.015
SRad-QP	3211 ± 128	0.582 ± 0.027	1020 ± 139	0.333 ± 0.019
D-Tanh	4000 ± 318	0.672 ± 0.035	2712 ± 228	0.496 ± 0.003
DD-SRad (Ours)	$\mathbf{4329 \pm 529}$	0.668 ± 0.028	$\mathbf{3312 \pm 384}$	0.462 ± 0.023

Notes: Parameterization methods have CVR$= 0$ (Theorem 2.4). †Post(QP)'s critic does not observe projection; boundary dwelling inflates Util, so Util is comparable only among critic-aware methods.

because its critic never observes the projected action. DD-SRad has the highest mean Return against every constrained baseline in all 16 environment–backbone–constraint configurations; paired tests and effect sizes are in Appendix Table 9.

DD-SRad leads all eight tight cases. Although D-Tanh has identical R^i_{eff} and box geometry, it trails by 5%–14% under SAC despite sometimes higher Util, consistent with exponential rather than polynomial derivative decay. On Humanoid ($d = 17$, $\kappa = 4.0$), D-Tanh falls below BoxPre+ while DD-SRad remains best. SRad-Strict collapses (Ant/Humanoid-SAC CV$= 68.8/49.3\%$), and its ankle utilization (≈ 0.081) is far below the geometric ceiling 0.40 from Theorem 2.7. BoxPre+ has the same rectangular closure yet reaches only 1998 versus 3112 on Ant-SAC, isolating clipping's

exact boundary dead zones from DD-SRad's nonzero finite-latent sensitivity. Appendices E and F provide all curves and radars.

3.2 Analysis of Action Optimization Under Heterogeneous Constraints

Per-dimension utilization separates policy conservatism from parameterization-induced truncation and directly tests Theorem 2.7. Figure 1 shows stable training in all eight heterogeneous cases; HalfCheetah degrades most ($\approx 40\%$ under SAC) because locomotion requires large joint outputs.

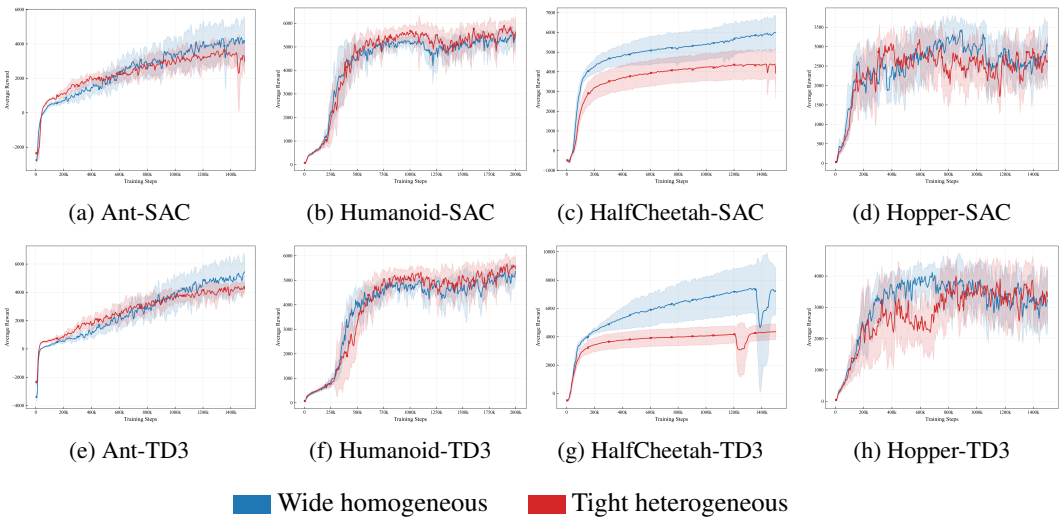


Figure 1: Mean return curves of DD-SRad under wide homogeneous and tight heterogeneous constraints with SAC/TD3 backbones.

Under tight limits, Ant hip ($\delta = 0.2$) and ankle ($\delta = 0.5$) groups approach their own boundaries rather than $\min_i \delta^i$, directly validating the ℓ_∞ reachable-set prediction. Appendix G provides all radars.

3.3 Sensitivity Analysis of λ_{BASE}

λ_{base} is the sole optimization-specific hyperparameter: it controls unnecessary latent magnitude while the boundary is set by $f(u) = u/\sqrt{1+u^2}$ ($f = 0.90/0.95$ at $|u| \approx 2.06/3.04$). On Ant-v5 we test $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$ under wide $\boldsymbol{\delta} = [1]^8$ and tight $\boldsymbol{\delta} = [0.2^4, 0.5^4]$ limits.

Figure 2 shows stability over 10^{-3}–10^{-1}; under wide limits, 10^{-3}–10^{0} converge comparably, whereas 10^{-4} loses $\approx 20\%$ Return with irregular utilization.

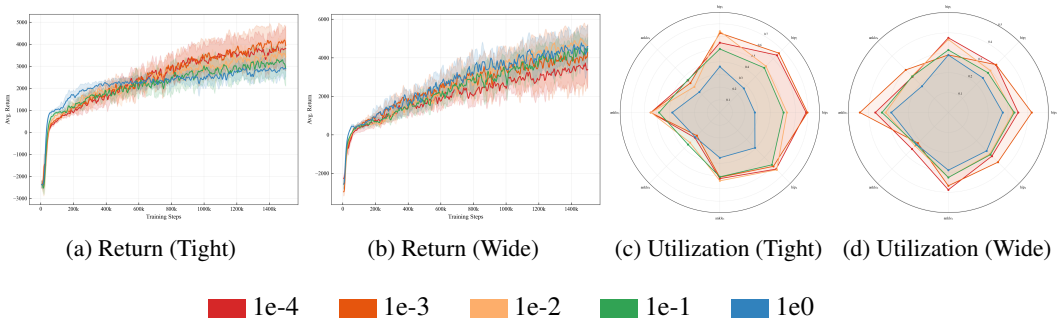


Figure 2: Mean return and per-dimension utilization of DD-SRad (SAC) on Ant-v5 under varying λ_{base}.

Under tight limits, the response peaks at 10^{-2}: 10^{-4} causes runaway imbalance (ankle Util> 0.65 while hips collapse), whereas 10^{0} suppresses all Util to ≈ 0.2. Wide and tight regimes favor 10^{-3} and 10^{-2}; their geometric midpoint $\lambda_{\text{base}} = 0.005$ works consistently without per-environment tuning.

4 IsaacLab Humanoid Simulations

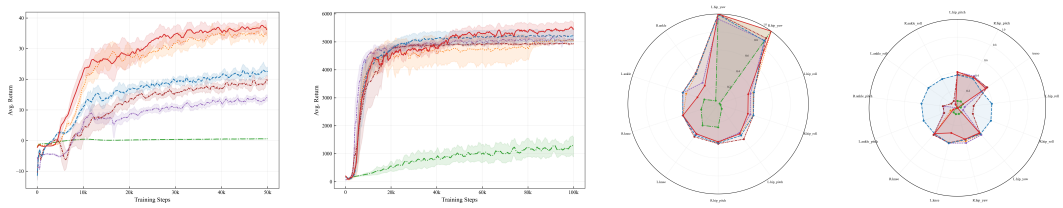
IsaacLab tests whether the ℓ_∞ behavior persists under contact dynamics, sensor noise, and temporal coupling absent from MuJoCo. Public Unitree specifications determine the limits, making this a controlled hardware-relevant simulation bridge, not physical-robot validation; H1/G1 videos are in the anonymous repository and supplement.

4.1 Experimental Setup

We use Unitree H1 (10 joints, rough terrain, $\kappa \approx 2.2$) and G1 (13 joints including torso, flat terrain, $\kappa = 4.0$). Official limits are normalized and tightened to stress-test the guarantee; Appendix Table 6 gives all settings.

All methods use FastSAC Seo et al. (2025) with $N_{\text{env}} = 4096$; its standard action-conditioned SAC gradient is compatible by Proposition 2.5. Baselines are D-Tanh, BoxPre+, Post(QP), SRad-QP, and SRad-Strict. Metrics are Return, CVR, velocity-tracking error (VTE↓), fall rate (Fall↓), episode length (Steps↑), and proximal/distal utilization and gap; Appendix Figure 8 shows the distributions.

4.2 Simulation Results



(a) H1 Mean Return (b) G1 Mean Return (c) H1 Radar Chart (d) G1 Radar Chart

Figure 3: Learning curves and radar charts for H1 and G1 (mean ± std over 5 seeds; official constraints).

On H1 (Table 3(a)), DD-SRad leads at 37.14 ± 2.84 (Fall$= 48.7\%$, Steps$= 748$, $d = 5.84$ versus BoxPre+), ahead of D-Tanh (34.56), BoxPre+ (23.11 ± 1.87), Post(QP) (19.53), SRad-QP (13.64), and SRad-Strict (0.83); BoxPre+/SRad-Strict Fall$= 70.2/100\%$. On G1, DD-SRad reaches 5473 ± 238 (Fall$= 0.3\%$, Steps$= 998$) and VTE 0.138 ± 0.009 (95% interval $[0.120, 0.156]$), versus 0.224 ± 0.028 ($[0.168, 0.280]$) for BoxPre+; SRad-Strict obtains 1312 Return and 91% falls.

Table 3(b) shows task-adaptive allocation: DD-SRad is H1-balanced (Gap$= 0.10$) but G1 hip-dominant (Util-P/D/Gap$= 0.40/0.02/0.38$). BoxPre+'s Gap≈ 0.01 accompanies 62% higher G1 VTE, consistent with clipping dead zones (Proposition 2.5), whereas SRad-Strict's Util-P$= 0.08$ is below its ≈ 0.45 geometric ceiling (Theorem 2.7).

Scope and limitations. DD-SRad certifies only modeled per-step rate/position bounds for independent joint-rate boxes and favors action-conditioned off-policy critics; scalar-advantage on-policy estimates may be heterogeneity-sensitive. Coupled power/torque/contact-force/CBF constraints require shared-budget multipliers or external safety filters. IsaacLab is a specification-consistent simulation bridge; deployment also requires validating actuator latency, torque tracking, thermal/contact dynamics, communication delay, and state-estimation error (Appendix K).

Table 3: IsaacLab results (FastSAC Seo et al. (2025); **bold**: best; constrained CVR$= 0$).

(a) Task-Level Performance (Return↑; VTE↓: velocity-tracking error; Fall↓; Steps↑)

	Unitree H1(Humanoid,Rough,$\kappa\approx2.2$)				**Unitree G1**(Humanoid,Flat,$\kappa=4.0$)			
Method	Return↑	VTE↓	Fall↓	Steps↑	Return↑	VTE↓	Fall↓	Steps↑
BoxPre+	23.11 ± 1.87	$\mathbf{0.385\pm0.026}$	70.2	526	5213 ± 241	0.224 ± 0.028	5.7	970
Post(QP)	19.53 ± 2.43	0.465 ± 0.064	88.0	339	4954 ± 42	0.139 ± 0.006	**0.0**	**1000**
SRad-QP	13.64 ± 1.33	0.390 ± 0.044	76.5	431	5166 ± 98	0.187 ± 0.011	0.3	998
SRad-Strict	0.83 ± 0.11	0.624 ± 0.008	100.0	139	1312 ± 344	0.360 ± 0.036	91.0	403
D-Tanh	34.56 ± 3.04	0.462 ± 0.041	58.7	679	5145 ± 91	0.207 ± 0.016	2.2	989
DD-SRad (Ours)	$\mathbf{37.14\pm2.84}$	0.459 ± 0.023	**48.7**	**748**	$\mathbf{5473\pm238}$	$\mathbf{0.138\pm0.009}$	0.3	998

(b) Constraint Utilization (Util-P/D: proximal/distal groups; Gap $=|$Util-P$-$Util-D$|$)

	Unitree H1(Humanoid,Rough,$\kappa\approx2.2$)			**Unitree G1**(Humanoid,Flat,$\kappa=4.0$)		
Method	Util-P	Util-D	Average Gap	Util-P	Util-D	Average Gap
BoxPre+	0.42 ± 0.00	0.44 ± 0.01	0.02	0.37 ± 0.02	0.38 ± 0.02	0.01
Post(QP)	0.45 ± 0.01	0.48 ± 0.01	0.03	0.39 ± 0.01	0.09 ± 0.01	0.30
SRad-QP	0.37 ± 0.01	0.42 ± 0.01	0.06	0.41 ± 0.01	0.03 ± 0.04	0.38
SRad-Strict	0.08 ± 0.00	0.50 ± 0.01	0.42	0.08 ± 0.01	0.01 ± 0.00	0.08
D-Tanh	0.41 ± 0.00	0.62 ± 0.05	0.21	0.40 ± 0.01	0.02 ± 0.05	0.37
DD-SRad (Ours)	0.38 ± 0.01	0.48 ± 0.03	0.10	0.40 ± 0.01	0.02 ± 0.04	0.38

Notes: Five-seed mean$\pm$std; 1000 test episodes/seed; curves/radars share data. Groups: H1 hip/ankle ($\delta = 0.40/0.18$), G1 hip_pitch/ankle ($\delta = 0.40/0.18$–0.25); H1 falls reflect rough terrain.

5 Related Work

Solver-based control and constrained RL. MPC/QP rate-limit controllers require online optimization and may separate learned from executed actions Williams et al. (2017); Nagabandi et al. (2018); Grandia et al. (2023); Kim et al. (2019); Pandala et al. (2022); Bledt et al. (2018). CMDPs treat expected costs or constrained updates Altman (1999); Achiam et al. (2017); Tessler et al. (2019); Zhang et al. (2020); Yang et al. (2022); Zhang et al. (2022), while recent theory studies anytime, instantaneous, and sample-efficient constraints McMahan and Zhu (2024); Shi et al. (2023); Ghosh et al. (2022). With state augmented by a_{t-1}, DD-SRad enforces the heterogeneous per-step rate box at policy output without an online solver.

Action parameterization and safety filters. Static squashing covers fixed intervals but not inter-step limits Haarnoja et al. (2018); Fujimoto et al. (2018), while SRad's radius $\min_i \delta^i$ under-covers heterogeneous boxes Kasaura et al. (2023). CBF-QP and differentiable projection handle broader constraints but incur runtime optimization or projection Ames et al. (2017); Pham et al. (2018); Dalal et al. (2018); Singletary et al. (2022). Unlike supervised HardNet Min and Azizan (2024), DD-SRad gives the moving rate box a closed-form separable map with a positive diagonal Jacobian.

Generative feasible-action methods. FlowPG, CV-Flows, ARAM, and TN-SAC address static or state-dependent feasible sets Brahmanage et al. (2023; 2025); Hung et al. (2025); Stolz et al. (2025), whereas rate-limit support moves with a_{t-1}. DD-SRad constructs it analytically with deterministic per-step satisfaction, tight heterogeneous coverage, finite-latent sensitivity, and solver-free execution; Appendix A gives the full comparison.

6 Conclusion

DD-SRad maps heterogeneous ℓ_∞ rate boxes through position-adaptive radii, guaranteeing rate/position limits and exact joint-box coverage with spectral, Lipschitz, and bounded-latent conditioning bounds. It yields the highest constrained-method mean Return in every MuJoCo environment–backbone case with zero violations and transfers to IsaacLab H1/G1; physical-robot validation and broader closed-loop safety remain future work.

AI USE STATEMENT

In preparing this manuscript, we used OpenAI's ChatGPT[1] solely for language editing and improving clarity of expression. All AI-assisted edits were reviewed and approved by the authors. The authors take full responsibility for the final content of the manuscript.

ETHICS STATEMENT

The authors have read and adhered to the ICLR Code of Ethics. This work does not involve human participants, animal subjects, personal or private data, or sensitive datasets. All experiments were conducted in public simulation environments using publicly documented robot specifications. The released code, configurations, and rollout videos contain no personally identifiable or sensitive information.

REPRODUCIBILITY STATEMENT

The material supporting reproducibility is organized as follows. The DD-SRad mapping, optimization objective, and Jacobian characterization are specified in Section 2, with pseudocode in Algorithm 1; all assumptions and complete proofs are provided in Appendices B and C. Network architectures, environment and software versions, training hyperparameters, rate-limit configurations, and evaluation protocols are documented in Appendix D, while the IsaacLab reward definitions are reported in Appendix H. Sections 3 and 4 report the seed counts, evaluation metrics, and principal results, with additional learning curves, utilization analyses, and aggregate statistical tests in Appendices E, E.1, and F. The anonymous repository linked in the abstract and the supplementary repository archive contain the implementation, configuration files, and H1/G1 rollout videos. No private training data or inaccessible evaluation dataset is required.

REFERENCES

J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017, pp. 22–31.

E. Altman, *Constrained Markov Decision Processes*. CRC Press, 1999.

A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.

D. P. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete Time Case*. Athena Scientific, 1996.

G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "MIT Cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, 2018, pp. 2245–2252.

J. C. Brahmanage, J. Ling, and A. Kumar, "FlowPG: Action-constrained policy gradient with normalizing flows," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

J. C. Brahmanage, J. Ling, and A. Kumar, "Leveraging constraint violation signals for action-constrained reinforcement learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Y. Chebotar et al., "Closing the sim-to-real loop: Adapting simulation randomization with real world experience," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8973–8979.

G. Dalal, K. Dvijotham, M. Vecerik, T. Hester, C. Paduraru, and Y. Tassa, "Safe exploration in continuous action spaces," *arXiv preprint arXiv:1801.08757*, 2018.

S. Fujimoto, H. van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International Conference on Machine Learning (ICML)*, 2018, pp. 1587–1596.

A. Ghosh, X. Zhou, and N. Shroff, "Provably efficient model-free constrained RL with linear function approximation," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022, pp. 13303–13315.

R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, "Perceptive locomotion through nonlinear model-predictive control," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3402–3421, Oct. 2023.

[1] https://chatgpt.com/

S. Gros and M. Zanon, "Data-driven economic NMPC using reinforcement learning," *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 636–648, Feb. 2020.

S. Gu et al., "A review of safe reinforcement learning: Methods, theories, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 11216–11235, Dec. 2024.

T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning (ICML)*, 2018, pp. 1861–1870.

W.-T. Hung et al., "Efficient action-constrained reinforcement learning via acceptance-rejection method and augmented MDPs," in *International Conference on Learning Representations (ICLR)*, 2025.

T. Johannink et al., "Residual reinforcement learning for robot control," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6023–6029.

K. Kasaura, S. Miura, T. Kozuno, R. Yonetani, K. Hoshino, and Y. Hosoe, "Benchmarking actor-critic deep reinforcement learning algorithms for robotics control with action constraints," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4449-4456, Aug. 2023.

D. Kim et al., "Highly dynamic quadruped locomotion via whole-body impulse control and model predictive control," *arXiv preprint arXiv:1909.06586*, 2019.

J. Lee et al., "Learning quadrupedal locomotion over challenging terrain," *Science Robotics*, vol. 5, no. 47, 2020.

J. Li, L. Sun, J. Chen, M. Tomizuka, and W. Zhan, "A safe hierarchical planning framework for complex driving scenarios based on reinforcement learning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, 2021, pp. 2660–2666.

T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," in *Proceedings of the Fourth International Conference on Learning Representations (ICLR)*, 2016.

J. McMahan and X. Zhu, "Anytime-constrained reinforcement learning," in *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024, pp. 4321–4329.

Y. Min and N. Azizan, "HardNet: Hard-constrained neural networks with universal approximation guarantees," *arXiv preprint arXiv:2410.10807*, 2024.

A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Australia, 2018, pp. 7559–7566.

A. Pandala, R. T. Fawcett, and K. A. Hamed, "Robust predictive control for quadrupedal locomotion: Learning to close the gap between reduced and full-order models," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6622–6629, 2022.

T.-H. Pham, G. De Magistris, and R. Tachibana, "OptLayer – Practical constrained optimization for deep reinforcement learning in the real world," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Australia, 2018, pp. 6236–6243.

A. Rajeswaran et al., "Learning complex dexterous manipulation with deep reinforcement learning and demonstrations," *arXiv preprint arXiv:1709.10087*, 2017.

A. Ray, J. Achiam, and D. Amodei, "Benchmarking safe exploration in deep reinforcement learning," 2019. [Online]. Available: `https://cdn.openai.com/safexp-short.pdf`

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

Y. Seo, C. Sferrazza, J. Chen, G. Shi, R. Duan, and P. Abbeel, "Learning sim-to-real humanoid locomotion in 15 minutes," *arXiv preprint arXiv:2512.01996*, 2025.

M. Shi, Y. Liang, and N. Shroff, "A near-optimal algorithm for safe reinforcement learning under instantaneous hard constraints," in *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023, pp. 31243–31268.

A. Singletary, S. Kolathaya, and A. D. Ames, "Safety-critical kinematic control of robotic systems," *IEEE Control Systems Letters*, vol. 6, pp. 139–144, 2022.

P. Stolz et al., "Improving Stochastic Action-Constrained Reinforcement Learning via Truncated Distributions," *arXiv preprint arXiv:2511.22406*, 2025.

C. Tessler, D. J. Mankowitz, and S. Mannor, "Reward constrained policy optimization," in *International Conference on Learning Representations (ICLR)*, 2019.

M. Vecerík et al., "Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards," *arXiv preprint arXiv:1707.08817*, 2017.

A. Wachi, X. Shen, and Y. Sui, "A survey of constraint formulations in safe reinforcement learning," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2024.

P. M. Wensing, A. Wang, S. Seok, D. Otten, J. Lang, and S. Kim, "Proprioceptive actuator design in the MIT Cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 509–522, June 2017.

G. Williams et al., "Information theoretic MPC for model-based reinforcement learning," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, 2017, pp. 1714–1721.

L. Yang et al., "Constrained update projection approach to safe policy optimization," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Y. Zhang, Q. Vuong, and K. W. Ross, "First order constrained optimization in policy space," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

L. Zhang et al., "Penalized proximal policy optimization for safe reinforcement learning," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2022.

A Extended Related Work

The most mature engineering solution for handling actuator constraints is the hierarchical control architecture, which decouples the high-level RL policy from the low-level constraint handling. Williams et al. Williams et al. (2017) proposed information-theoretic MPPI, solving constrained optimization within the control period via large-scale parallel sampling and demonstrating strong online planning capability on racing trajectory tracking tasks; however, the computational cost of MPPI grows steeply with state dimension, making it difficult to maintain the required control frequency on high-DOF humanoid robots. Nagabandi et al. Nagabandi et al. (2018) learned system dynamics with a neural network and ran MPC on top, achieving online adaptation to changes in environment parameters; however, the accuracy of the learned dynamics model degrades away from the training distribution, causing constraint satisfaction to deteriorate—particularly evident in contact-rich locomotion tasks. Grandia et al. Grandia et al. (2023) embedded perception into nonlinear MPC to achieve dynamic locomotion on rough terrain, demonstrating MPC's constraint-handling capability in complex environments; however, the per-step solve time of approximately 15 ms, together with perception processing overhead, makes generalization to higher-frequency control scenarios difficult. Kim et al. Kim et al. (2019) proposed a combined whole-body impulse control and MPC framework achieving highly dynamic gaits on a quadruped; the method's reliance on precise dynamics models limits its robustness to parametric uncertainty and external disturbances. Pandala et al. Pandala et al. (2022) introduced robust predictive control to reduce the gap between simplified and full-order models; however, its conservative design further compresses the reachable action space under tight rate constraints, affecting motion performance.

Johannink et al. Johannink et al. (2019) proposed residual reinforcement learning, stacking an RL policy on top of a classical controller to enable safe exploration with fewer samples in robotic manipulation; however, the performance ceiling of this framework is limited by the quality of the base controller, and the high-level–low-level objective inconsistency is especially pronounced near constraint boundaries. Lee et al. Lee et al. (2020) completed end-to-end locomotion learning on challenging terrain with a real quadruped, demonstrating the potential of pure RL under complex dynamics; however, that work does not explicitly handle joint rate limits, relying on reward engineering for soft constraints on abnormal actions without providing hard guarantees. Li et al. Li et al. (2021) designed an RL-based safe hierarchical planning framework for complex driving scenarios; the high-level policy cannot perceive the low-level constraint boundaries at inference time, leading

to biased policy gradient estimates near constraint regions, and empirical studies show that actual execution performance degrades by approximately 18% compared to end-to-end methods Rajeswaran et al. (2017). Gros and Zanon Gros and Zanon (2020) proposed a data-driven economic nonlinear MPC framework that theoretically unifies RL and MPC; however, their framework requires differentiable dynamics models and incurs high computational complexity, limiting real-time deployment feasibility.

The above hierarchical approaches share three structural difficulties: objective inconsistency between the high-level RL and low-level constraint handler—the high-level optimizes long-term cumulative return while the low-level minimizes tracking error, and these two objectives conflict fundamentally when constraints are active; computational overhead of the low-level solver (MPC, QP) in high-dimensional systems significantly compresses control bandwidth, with the convex MPC solve on MIT Cheetah 3 reducing the control frequency from an ideal 500 Hz to approximately 40 Hz in practice Bledt et al. (2018); and training-deployment inconsistency making sim-to-real transfer difficult Vecerík et al. (2017); Chebotar et al. (2019)—the policy observes idealized execution during training, while low-level corrections change the action effect at deployment, with performance degradation of 25–40%.

From a constrained RL perspective, Altman Altman (1999) established the theoretical foundation of constrained Markov decision processes (CMDP), formalizing constraint satisfaction as expected cumulative constraints $\mathbb{E}_\pi[\sum_t c_t] \leq d$; the core limitation of this framework is that it permits transient violations as long as the long-term average is satisfied, whereas rate constraints require $|\Delta a_t^i| \leq \delta^i, \forall t$ to hold strictly at every time step—two constraint types with essentially different mathematical structures. Achiam et al. Achiam et al. (2017) proposed Constrained Policy Optimization (CPO), maintaining constraint feasibility during policy updates via second-order trust-region methods and establishing an important reference point for constrained RL on the Safety Gym benchmark; however, CPO handles expected-form constraints, exhibits high gradient estimate variance in sparse violation scenarios, and requires solving a linear system at each step with considerable computational cost. Tessler et al. Tessler et al. (2019) proposed RCPO, converting constraints into penalty terms via dynamic Lagrangian multiplier adjustment; this method cannot eliminate single-step violations at test time even after convergence, lacking hard guarantees. Zhang et al. Zhang et al. (2020) proposed FOCOPS, approximating CPO's constrained policy update with first-order optimization to substantially reduce computational cost; however, its theoretical guarantees are equally limited to expected constraints and are ineffective for per-step rate limits. Yang et al. Yang et al. (2022) proposed CUP, decoupling policy improvement from constraint satisfaction via constrained update projection; however, this method's conservatism near constraint boundaries reduces exploration efficiency, leading to insufficient constraint utilization under high-κ heterogeneous constraints. Zhang et al. Zhang et al. (2022) proposed P3O, balancing task performance and constraint satisfaction with adaptive penalty coefficients; its online tuning mechanism is sensitive to hyperparameters, with questionable stability under rapidly varying rate constraints. Ray et al. Ray et al. (2019) systematically evaluated the performance of multiple safe RL algorithms, revealing an inherent trade-off between constraint satisfaction rate and task performance in existing CMDP methods, but providing no specialized solutions for the temporally dependent rate-constraint scenario. Survey works by Wachi et al. Wachi et al. (2024) and Gu et al. Gu et al. (2024) further confirm that existing safe RL theoretical frameworks cannot mathematically cover the per-step hard guarantee requirements of rate constraints.

At the policy parameterization level, Schulman et al. Schulman et al. (2017) proposed PPO, which has found wide application in on-policy settings; however, PPO itself does not handle action-domain constraints, allowing actions to exceed physical limits during training, and requiring additional constraint handling at deployment. Fujimoto et al. Fujimoto et al. (2018) proposed TD3, improving critic estimate accuracy via target policy smoothing; similarly, the standard TD3 framework has no native support for rate constraints. Lillicrap et al. Lillicrap et al. (2016) introduced a $\tanh$ output layer in DDPG to restrict actions to $[-1, 1]$; this method hard-codes the action range as a uniform symmetric interval and cannot handle dynamic shifting feasible sets centered at a_{t-1}. Haarnoja et al. Haarnoja et al. (2018) proposed SAC, adopting a squashed Gaussian policy to explicitly model action boundaries while maintaining entropy maximization; however, squashed Gaussians are equally applicable only to static position-domain constraints and cannot express inter-step action differential relationships. Kasaura et al. Kasaura et al. (2023) systematically studied the effect of action constraints on actor-critic algorithms and proposed Spherical Radial Squashing (SRad), constraining policy output within a ball of center c_s and radius R via $a = c_s + R \cdot u/\sqrt{1 + \|u\|^2}$, outperforming

naive clip and QP post-processing on robotic control constraint benchmarks; SRad performs well under homogeneous constraints, but its global radius design faces a structural trade-off under heterogeneous constraints—if $R = \min_i \delta^i$, the reachable set for high-budget dimensions is compressed to a fraction $\min_i \delta^i / \delta^i$ of the original radius; if $R > \min_i \delta^i$, low-budget dimensions are violated; no globally valid and tight radius selection exists. Singletary et al. Singletary et al. (2022) proposed a safety-critical kinematics control framework using control barrier functions (CBF) to guarantee safety constraints at the kinematic level; the CBF approach requires high model accuracy and faces computational bottlenecks with real-time CBF-QP solving in high-dimensional joint spaces. Pham et al. Pham et al. (2018) proposed OptLayer, inserting a differentiable optimization layer between the policy and environment to project infeasible actions to the nearest feasible point, achieving hard constraint satisfaction; however, QP solve time under high-dimensional non-convex constraints grows with dimension, and the Jacobian computation of the differentiable projection layer introduces additional latency in real-time control. Dalal et al. Dalal et al. (2018) proposed Safe Layer, learning a safe projection via linear approximation of constraint functions to achieve low-cost action correction; linear approximation accuracy degrades near constraint boundaries, causing protection failures under high-frequency rate constraint activation. Ames et al. Ames et al. (2017) proposed a CBF-based QP safety filter, guaranteeing forward invariance in control input space via QP solving with rigorous continuous-time safety guarantees; however, adapting CBF-QP to discrete-time RL differential action constraints requires conservative margin design, excessively restricting the action range during normal motion.

Recent generative model-based methods attempt to fundamentally change the constraint handling paradigm. Brahmanage et al. Brahmanage et al. (2023) proposed FlowPG, sampling directly from the feasible set via normalizing flows, achieving excellent performance on static action constraint benchmarks; however, FlowPG requires pre-building a flow model of the global feasible set, while the feasible set $\mathcal{F}_t$ of rate constraints drifts with each step's a_{t-1}, making it impossible to establish a static distribution, rendering FlowPG inapplicable to rate-constraint scenarios. Conditional normalizing flows can theoretically model time-varying feasible sets conditioned on a_{t-1}, but executing a full L-layer flow forward pass at each inference step introduces unacceptable real-time latency at 50 Hz control frequency; DD-SRad's analytic parameterization incurs zero inference overhead. Brahmanage et al. Brahmanage et al. (2025) further proposed CV-Flows, augmenting flow model training with constraint violation signals to improve sampling quality near boundaries; however, its assumption of a static feasible set equally prevents it from handling temporally dependent rate constraints. Hung et al. Hung et al. (2025) proposed ARAM, efficiently realizing constraint-feasible action sampling via acceptance-rejection sampling and augmented MDPs, with sample efficiency significantly superior to MCMC methods under static constraints; however, the acceptance-rejection framework equally requires relatively stable feasible sets during training and cannot directly accommodate rate constraints where the feasible set shifts dynamically with a_{t-1} at each step. Stolz et al. Stolz et al. (2025) proposed defining a truncated Gaussian distribution policy directly over constrained intervals, resolving the abnormal probability density of squashed Gaussians at constraint boundaries via closed-form/approximate normalization constants and entropy corrections, achieving significant performance improvements on static and state-dependent action constraint benchmarks; the original work does not cover the temporally dependent feasible set $\mathcal{F}_t$ induced by a_{t-1}, and adapting it to rate constraints requires state augmentation and recomputing truncation endpoints at each step. This approach and DD-SRad represent two parallel design routes within action-constrained RL: the former focuses on correct density modeling of the policy distribution, the latter on analytic parameterization of ℓ_∞ geometry; the two share the same reachable set under an ℓ_∞ feasible set, differing in that one handles the density function for stochastic sampling while the other uses a deterministic mapping such that entropy and log-probability computations in u-space remain consistent with standard SAC.

Synthesizing the above analysis, existing methods share a common geometric mismatch problem when handling rate constraints. The rate constraint $|a_t^i - a_{t-1}^i| \le \delta^i$ defines an ℓ_∞ hyperrectangle feasible set $\mathcal{F}_t = \{a : |a^i - a_{t-1}^i| \le \delta^i,\ \forall i\}$ in the differential action space, while ℓ_2 spherical parameterization (SRad) and the static feasible set assumption of generative model methods are both incompatible with this ℓ_∞ geometric structure. Under heterogeneous constraints ($\kappa \gg 1$), the effective exploration radius of ℓ_2 spherical parameterization is limited to $R \le \min_i \delta^i$, compressing the reachable set for high-budget dimensions to a fraction $\min_i \delta^i / \delta^i$ of the original radius—corresponding to approximately 60% radius loss at $\kappa = 2.5$. The reachable set of ℓ_∞ projection methods (such

as per-dimension clip) is consistent with $\mathcal{F}_t$, but clip truncates the gradient at the boundary of the feasible set, causing the policy gradient to lose directional information as the constraint approaches saturation.

B Proofs of Theoretical Results

This appendix collects complete proofs of all theoretical results stated in §2.3. Notation follows §2 throughout. Two elementary facts are used repeatedly throughout the proofs.

- **Squashing bound.** For $f(t) \triangleq t/\sqrt{1+t^2}$, one has $|f(t)| < 1$ for all $t \in \mathbb{R}$ and $|f(t)| \leq |t|$ (since $\sqrt{1+t^2} \geq 1$).
- **Squashing derivative.** $f'(t) = (1+t^2)^{-3/2}$, which satisfies $0 < f'(t) \leq 1$ for all $t \in \mathbb{R}$.

B.1 Proof of Augmented MDP Validity

Proof. The three properties are verified in turn.

(i) Markov property. By construction, $\tilde{s}_{t+1} = (s_{t+1}, a_t)$, where s_{t+1} is drawn from the original transition kernel $P(\cdot \mid s_t, a_t)$ and the action component is set deterministically to a_t. Hence the augmented transition kernel satisfies

$$\tilde{P}(\tilde{s}_{t+1} \mid \tilde{s}_t, a_t) = P(s_{t+1} \mid s_t, a_t)\,.$$

The right-hand side depends only on (s_t, a_t), and s_t is a component of $\tilde{s}_t$, so the transition probability is a function of $(\tilde{s}_t, a_t)$ alone. The Markov property therefore holds on the augmented state space $\tilde{\mathcal{S}} = \mathcal{S} \times \mathcal{A}$.

(ii) Policy validity. The DD-SRad mapping (4) takes $a_{\text{prev}} = a_{t-1}$ as input, which is the second component of the augmented state $\tilde{s}_t = (s_t, a_{t-1})$. By Theorem 2.4, the output $a_t = \phi(u_t, a_{t-1})$ satisfies $|a_t^i - a_{t-1}^i| \leq \delta^i$ and $a_t^i \in [a_{\min}^i, a_{\max}^i]$ for every i and every latent realization u_t. Therefore the stochastic policy $\pi : \tilde{\mathcal{S}} \to \Delta(\mathcal{A})$ is well-defined and maps every augmented state to a distribution over $\mathcal{A}$ satisfying constraint (2).

(iii) Existence of the optimal value function. The augmented MDP $\tilde{\mathcal{M}}$ inherits all standard regularity conditions from the original MDP: the reward r is bounded, the augmented action space $\mathcal{A}$ is compact (unchanged), and the augmented transition kernel $\tilde{P}$ is Lipschitz continuous in $(\tilde{s}, a)$ because P is Lipschitz in (s, a) and the action component of $\tilde{s}_{t+1}$ is set deterministically. Under these conditions, the augmented Bellman operator

$$(\mathcal{T}Q)(\tilde{s}, a) \triangleq r(s, a) + \gamma\, \mathbb{E}_{\tilde{s}' \sim \tilde{P}(\cdot|\tilde{s},a)} \Big[\max_{a'} Q(\tilde{s}', a') \Big]$$

is a γ-contraction in $\|\cdot\|_\infty$ on the space of bounded functions Bertsekas and Shreve (1996). By Banach's fixed-point theorem, $\mathcal{T}$ admits a unique fixed point Q^*, which is the optimal action-value function of $\tilde{\mathcal{M}}$. □

B.2 Proof of Theorem 2.4 (Heterogeneous Rate Constraint Satisfaction)

Proof. Fix any dimension $i \in \{1, \ldots, d\}$ and any $u^i \in \mathbb{R}$. Let $\Delta a^i \triangleq a^i - a^i_{\text{prev}} = R^i_{\text{eff}}(u^i, a^i_{\text{prev}}) \cdot f(u^i)$.

Three exhaustive cases are considered according to the sign of u^i.

Case 1: $u^i > 0$**.** Then $R^i_{\text{eff}} = \min\big(\delta^i,\, a^i_{\max} - a^i_{\text{prev}}\big) \geq 0$ and $f(u^i) \in (0, 1)$.

1. *Rate constraint.* $|\Delta a^i| = R^i_{\text{eff}} \cdot f(u^i) < R^i_{\text{eff}} \leq \delta^i$, where the strict inequality uses $f(u^i) < 1$. (When $a^i_{\text{prev}} = a^i_{\max}$ one has $R^i_{\text{eff}} = 0$, giving $\Delta a^i = 0$, which also satisfies $|\Delta a^i| \leq \delta^i$.)

2. *Position constraint.* Since $\Delta a^i = R^i_{\text{eff}} f(u^i) \geq 0$, it follows that $a^i = a^i_{\text{prev}} + \Delta a^i \geq a^i_{\text{prev}} \geq a^i_{\min}$. Moreover, $\Delta a^i < R^i_{\text{eff}} \leq a^i_{\max} - a^i_{\text{prev}}$, so $a^i < a^i_{\max}$. Hence $a^i \in [a^i_{\min}, a^i_{\max}]$.

If additionally $a^i_{\text{prev}} \in (a^i_{\min}, a^i_{\max})$, then $R^i_{\text{eff}} > 0$, so $|\Delta a^i| < \delta^i$ and $a^i \in (a^i_{\min}, a^i_{\max})$ hold strictly.

Case 2: $u^i < 0$**.** Then $R^i_{\text{eff}} = \min\big(\delta^i,\, a^i_{\text{prev}} - a^i_{\min}\big) \geq 0$ and $f(u^i) \in (-1, 0)$. A symmetric argument to Case 1 (with the roles of $a^i_{\min}$ and $a^i_{\max}$ exchanged) yields $|\Delta a^i| \leq \delta^i$ and $a^i \in [a^i_{\min}, a^i_{\max}]$, with strict inclusions when $a^i_{\text{prev}} \in (a^i_{\min}, a^i_{\max})$.

Case 3: $u^i = 0$**.** Then $\Delta a^i = R^i_{\text{eff}} \cdot f(0) = 0$, so $|\Delta a^i| = 0 \leq \delta^i$ and $a^i = a^i_{\text{prev}} \in \mathcal{A}^i$.

The three cases exhaust all $u^i \in \mathbb{R}$, including $u^i = 0$. Applying the argument independently to every dimension proves that $|a^i - a^i_{\text{prev}}| \leq \delta^i$ and $a^i \in [a^i_{\min}, a^i_{\max}]$ hold deterministically for every $u \in \mathbb{R}^d$. □

B.3 Proof of Proposition 2.5 (Jacobian Structure and Gradient Norm Bound)

Proof. **(i) Diagonal structure and positivity.** Fix dimension i and suppose $u^i \neq 0$. On the open half-lines $\{u^i > 0\}$ and $\{u^i < 0\}$, the effective radius $R^i_{\text{eff}}(u^i, a^i_{\text{prev}})$ is piecewise constant (it equals $\min(\delta^i, a^i_{\max} - a^i_{\text{prev}})$ for $u^i > 0$ and $\min(\delta^i, a^i_{\text{prev}} - a^i_{\min})$ for $u^i < 0$). Differentiating $a^i = a^i_{\text{prev}} + R^i_{\text{eff}} \cdot f(u^i)$ with respect to u^i on each open half-line via the chain rule gives

$$\frac{\partial a^i}{\partial u^i} = R^i_{\text{eff}} \cdot f'(u^i) = \frac{R^i_{\text{eff}}}{(1 + (u^i)^2)^{3/2}}.$$

Since a_j depends only on u^j for $j \neq i$, all off-diagonal entries of the Jacobian $J(u) = \partial a / \partial u$ are zero. Hence $J(u)$ is diagonal. When $a^i_{\text{prev}} \in (a^i_{\min}, a^i_{\max})$, $R^i_{\text{eff}} > 0$, so $\partial a^i / \partial u^i > 0$.

(ii) Spectral norm and bounded-latent condition number. Since $J(u)$ is diagonal, its spectral norm is

$$\|J(u)\|_2 = \max_i \frac{R^i_{\text{eff}}}{(1 + (u^i)^2)^{3/2}} \leq \max_i R^i_{\text{eff}} \leq \max_i \delta^i.$$

This spectral-norm bound is global.

For the condition number, positivity of the diagonal entries gives the exact expression

$$\text{cond}_2(J(u)) = \frac{\max_i \dfrac{R^i_{\text{eff}}}{(1 + (u^i)^2)^{3/2}}}{\min_i \dfrac{R^i_{\text{eff}}}{(1 + (u^i)^2)^{3/2}}}.$$

Let $\|u\|_\infty \leq B$ and define $\rho_{\min}(u) = \min_i R^i_{\text{eff}}$ and $\rho_{\max}(u) = \max_i R^i_{\text{eff}}$. If $\rho_{\min}(u) > 0$, then

$$\max_i \frac{R^i_{\text{eff}}}{(1 + (u^i)^2)^{3/2}} \leq \rho_{\max}(u),$$

while

$$\min_i \frac{R^i_{\text{eff}}}{(1 + (u^i)^2)^{3/2}} \geq \frac{\rho_{\min}(u)}{(1 + B^2)^{3/2}}.$$

Consequently,

$$\text{cond}_2(J(u)) \leq \frac{\rho_{\max}(u)}{\rho_{\min}(u)} (1 + B^2)^{3/2}.$$

Away from position boundaries, $R^i_{\text{eff}} = \delta^i$ on the active sign branches, yielding

$$\text{cond}_2(J(u)) \le \frac{\max_i \delta^i}{\min_i \delta^i}(1+B^2)^{3/2} = \kappa(1+B^2)^{3/2}.$$

At $u = \mathbf{0}$ in this interior regime, the positive and negative branch derivatives both equal δ^i, and hence $\text{cond}_2(J(\mathbf{0})) = \kappa$. No finite global condition-number bound exists over arbitrary $u \in \mathbb{R}^d$, because different latent coordinates may approach saturation at different rates.

(iii) Gradient norm bound. For the TD3 backbone, the actor loss is $\mathcal{L}_{\text{actor}}(u) = -Q(\tilde{s}, \phi(u)) + \lambda_{\text{base}}\|u\|_2^2$. Differentiating with respect to u and applying the triangle inequality:

$$\|\nabla_u \mathcal{L}_{\text{actor}}\|_2 = \left\| \nabla_a Q(\tilde{s}, a)\big|_{a=\phi(u)} \cdot J(u) + 2\lambda_{\text{base}} u \right\|_2 \le \|J(u)\|_2 \cdot \|\nabla_a Q(\tilde{s}, a)\|_2 + 2\lambda_{\text{base}}\|u\|_2.$$

Substituting the spectral norm bound from (ii) yields inequality (10). For the SAC backbone, the entropy term $\alpha \log \pi_\theta(a|\tilde{s})$ contributes an additional gradient term $\alpha \nabla_u \log \pi_\theta(a|\tilde{s})$, which enters linearly and is accounted for as stated. □

B.4 Proof of Proposition 2.6 (Lipschitz Continuity of the Mapping)

Proof. Fix $a_{\text{prev}} \in \mathcal{A}$ and $u, \hat{u} \in \mathbb{R}^d$. For each dimension i, define $g^i(v) \triangleq R^i_{\text{eff}}(v, a^i_{\text{prev}}) \cdot f(v)$ so that $a^i - \hat{a}^i = g^i(u^i) - g^i(\hat{u}^i)$. Denote $R^+ \triangleq \min(\delta^i, a^i_{\max} - a^i_{\text{prev}})$ and $R^- \triangleq \min(\delta^i, a^i_{\text{prev}} - a^i_{\min})$; both satisfy $R^{\pm} \le \delta^i$.

Case 1: $u^i, \hat{u}^i > 0$**.** Both map under R^+, so

$$|g^i(u^i) - g^i(\hat{u}^i)| = R^+ |f(u^i) - f(\hat{u}^i)|.$$

By the Mean Value Theorem and $f'(t) = (1+t^2)^{-3/2} \le 1$:

$$|f(u^i) - f(\hat{u}^i)| \le |u^i - \hat{u}^i|.$$

Hence $|g^i(u^i) - g^i(\hat{u}^i)| \le R^+ |u^i - \hat{u}^i| \le \delta^i |u^i - \hat{u}^i|$.

Case 2: $u^i, \hat{u}^i < 0$**.** Both map under R^-; the identical argument gives $|g^i(u^i) - g^i(\hat{u}^i)| \le \delta^i |u^i - \hat{u}^i|$.

Case 3: $u^i \ge 0$ **and** $\hat{u}^i \le 0$ **(or vice versa).** Assume without loss of generality $u^i \ge 0 \ge \hat{u}^i$. Then $g^i(u^i) = R^+ f(u^i) \ge 0$ and $g^i(\hat{u}^i) = R^- f(\hat{u}^i) \le 0$, so

$$\begin{aligned} |g^i(u^i) - g^i(\hat{u}^i)| &= R^+ f(u^i) + R^- |f(\hat{u}^i)| \\ &\le \delta^i \big[f(u^i) + |f(\hat{u}^i)| \big] \\ &\le \delta^i \big[u^i + |\hat{u}^i| \big] = \delta^i |u^i - \hat{u}^i|, \end{aligned}$$

where the second inequality uses $|f(t)| \le |t|$, and the last equality holds because $u^i \ge 0 \ge \hat{u}^i$.

Combining all three cases, $|a^i - \hat{a}^i| \le \delta^i |u^i - \hat{u}^i|$ holds for every i. Therefore,

$$\begin{aligned} \|\phi(u) - \phi(\hat{u})\|_2^2 &= \sum_{i=1}^d |a^i - \hat{a}^i|^2 \\ &\le \sum_{i=1}^d (\delta^i)^2 (u^i - \hat{u}^i)^2 \\ &\le \Big(\max_i \delta^i \Big)^2 \sum_{i=1}^d (u^i - \hat{u}^i)^2 \\ &= \|\boldsymbol{\delta}\|_\infty^2 \|u - \hat{u}\|_2^2. \end{aligned}$$

Taking square roots proves the stated $\|\boldsymbol{\delta}\|_\infty$-Lipschitz bound.

Tightness. Let $i^\star \in \arg\max_i \delta^i$ and choose $a^{i^\star}_{\text{prev}}$ sufficiently far from both position boundaries so that $R^{+,i^\star} = R^{-,i^\star} = \delta^{i^\star}$. For $\varepsilon > 0$, set $u = \varepsilon e_{i^\star}$ and $\hat{u} = -\varepsilon e_{i^\star}$. Then

$$\frac{\|\phi(u) - \phi(\hat{u})\|_2}{\|u - \hat{u}\|_2} = \delta^{i^\star} \frac{f(\varepsilon)}{\varepsilon} \longrightarrow \delta^{i^\star} = \|\boldsymbol{\delta}\|_\infty \qquad \text{as } \varepsilon \to 0^+.$$

Hence no smaller global Lipschitz constant is valid. □

B.5 Proof of Theorem 2.7 (Tight ℓ_∞-Coverage of the Reachable Set)

Proof. Fix $a_{\text{prev}} \in \mathcal{A}$. For each dimension i, define the direction-dependent effective radii

$$R^{+,i} \triangleq \min\big(\delta^i,\ a^i_{\max} - a^i_{\text{prev}}\big)\,, \qquad R^{-,i} \triangleq \min\big(\delta^i,\ a^i_{\text{prev}} - a^i_{\min}\big)\,,$$

corresponding to R^i_{eff} when $u^i > 0$ and $u^i < 0$, respectively.

Step 1: Characterization of $\mathcal{R}_{\text{DD}}$.

Inclusion (image is contained in the open box). By Theorem 2.4, for every $u \in \mathbb{R}^d$ and every i:

- If $u^i > 0$: $\Delta a^i = R^{+,i} f(u^i) \in (0, R^{+,i})$, so $a^i \in (a^i_{\text{prev}},\ a^i_{\text{prev}} + R^{+,i})$.
- If $u^i < 0$: $\Delta a^i = R^{-,i} f(u^i) \in (-R^{-,i}, 0)$, so $a^i \in (a^i_{\text{prev}} - R^{-,i},\ a^i_{\text{prev}})$.
- If $u^i = 0$: $a^i = a^i_{\text{prev}}$.

In all cases, $a^i \in (a^i_{\text{prev}} - R^{-,i},\ a^i_{\text{prev}} + R^{+,i})$.

Surjectivity (every interior point is achieved). Fix any target $a^* = (a^{*1}, \ldots, a^{*d})$ satisfying $a^{*i} \in (a^i_{\text{prev}} - R^{-,i},\ a^i_{\text{prev}} + R^{+,i})$ for all i. Set $\Delta^{*i} \triangleq a^{*i} - a^i_{\text{prev}}$ and consider two subcases.

If $\Delta^{*i} > 0$: the goal is to find $u^{*i} > 0$ satisfying $R^{+,i} f(u^{*i}) = \Delta^{*i}$. Since $\Delta^{*i} < R^{+,i}$, the ratio $\Delta^{*i}/R^{+,i} \in (0,1)$. The equation $f(u) = c$ for $c \in (0,1)$ has the unique positive solution $u = c/\sqrt{1-c^2}$, giving

$$u^{*i} = \frac{\Delta^{*i}}{\sqrt{(R^{+,i})^2 - (\Delta^{*i})^2}} > 0.$$

For this u^{*i}, $R^i_{\text{eff}}(u^{*i}, a^i_{\text{prev}}) = R^{+,i}$ by definition, so $\phi(u^*)^i = a^i_{\text{prev}} + R^{+,i} f(u^{*i}) = a^{*i}$ as required.

If $\Delta^{*i} < 0$: a symmetric construction with $R^{-,i}$ yields the unique negative preimage $u^{*i} = \Delta^{*i}/\sqrt{(R^{-,i})^2 - (\Delta^{*i})^2} < 0$.

If $\Delta^{*i} = 0$: $u^{*i} = 0$ gives $a^{*i} = a^i_{\text{prev}}$.

Since dimensions are decoupled, setting $u^* = (u^{*1}, \ldots, u^{*d})$ achieves $\phi(u^*) = a^*$.

Conclusion. The reachable set (image) is

$$\mathcal{R}_{\text{DD}} = \prod_{i=1}^{d} \big(a^i_{\text{prev}} - R^{-,i},\ a^i_{\text{prev}} + R^{+,i}\big)\,,$$

and its closure is the closed product interval $\overline{\mathcal{R}_{\text{DD}}} = \prod_{i=1}^d [a^i_{\text{prev}} - R^{-,i},\ a^i_{\text{prev}} + R^{+,i}]$.

Step 2: Agreement with $\mathcal{F}$. Observe that

$$a^i_{\text{prev}} + R^{+,i} = a^i_{\text{prev}} + \min\big(\delta^i, a^i_{\max} - a^i_{\text{prev}}\big) = \min\big(a^i_{\text{prev}} + \delta^i,\ a^i_{\max}\big)\,,$$

$$a^i_{\text{prev}} - R^{-,i} = a^i_{\text{prev}} - \min\big(\delta^i, a^i_{\text{prev}} - a^i_{\min}\big) = \max\big(a^i_{\text{prev}} - \delta^i,\ a^i_{\min}\big)\,.$$

Therefore, for each dimension i:

$$\big[a^i_{\text{prev}} - R^{-,i},\ a^i_{\text{prev}} + R^{+,i}\big] = \big[a^i_{\text{prev}} - \delta^i,\ a^i_{\text{prev}} + \delta^i\big] \cap \big[a^i_{\min},\ a^i_{\max}\big]\,.$$

Taking the Cartesian product over all dimensions:

$$\overline{\mathcal{R}_{\mathrm{DD}}} = \prod_{i=1}^{d} \left(\left[a^i_{\mathrm{prev}} - \delta^i, a^i_{\mathrm{prev}} + \delta^i \right] \cap [a^i_{\min}, a^i_{\max}] \right) = \mathcal{F},$$

where the last equality uses the product structure of both the rate-constraint set and $\mathcal{A}$. This establishes (12).

Step 3: Volume ratio. When a^i_{prev} is sufficiently far from the position boundaries so that $R^{+,i} = R^{-,i} = \delta^i$ for all i, the Lebesgue measure of $\overline{\mathcal{R}_{\mathrm{DD}}}$ is

$$V_{\mathrm{DD}} = \prod_{i=1}^{d} (R^{+,i} + R^{-,i}) = \prod_{i=1}^{d} 2\delta^i.$$

The d-dimensional ℓ_2 ball with radius $R = \min_i \delta^i$ has volume

$$V_{\mathrm{SRad}} = \frac{\pi^{d/2}}{\Gamma(d/2+1)} \left(\min_i \delta^i \right)^d.$$

Dividing:

$$\frac{V_{\mathrm{DD}}}{V_{\mathrm{SRad}}} = \frac{\prod_{i=1}^{d} 2\delta^i}{\frac{\pi^{d/2}}{\Gamma(d/2+1)} \left(\min_i \delta^i \right)^d} = \frac{2^d \, \Gamma(d/2+1)}{\pi^{d/2}} \cdot \frac{\prod_i \delta^i}{\left(\min_i \delta^i \right)^d},$$

which establishes (13). □

C Additional Optimization Details

C.1 Numerical Safety Redundancy

After per-dimension squashing, the optional global ℓ_2 safeguard scales $\Delta\mathbf{a} = \mathbf{a} - \mathbf{a}_{\mathrm{prev}}$ as

$$\widetilde{\Delta\mathbf{a}} = \Delta\mathbf{a} \cdot \min\left(1, \frac{\|\boldsymbol{\delta}\|_2}{\|\Delta\mathbf{a}\|_2}\right). \tag{18}$$

Its empirical trigger rate is below 0.1%; the primary per-dimension guarantee remains Theorem 2.4.

C.2 Latent Regularization and Bounded-Latent Conditioning

Let

$$G \triangleq \sup_{\tilde{s},a} \left\| \alpha \nabla_a \log \pi_\theta(a \mid \tilde{s}) - \nabla_a Q_\phi(\tilde{s}, a) \right\|_2 < \infty$$

under the standard bounded actor-gradient and Lipschitz-critic assumptions. At a pointwise critical point of the latent actor objective, the first-order condition and Proposition 2.5(ii) give

$$2\lambda_{\text{base}} \|u\|_2 \le \|J(u)\|_2 G \le \max_i \delta^i G,$$

and hence

$$\|u\|_\infty \le \|u\|_2 \le B(\lambda_{\text{base}}) \triangleq \frac{\max_i \delta^i G}{2\lambda_{\text{base}}}.$$

Away from position boundaries, substitution into Eq. 8 yields

$$\mathrm{cond}_2(J(u)) \le \kappa \left[1 + \left(\frac{\max_i \delta^i G}{2\lambda_{\text{base}}} \right)^2 \right]^{3/2}.$$

This stationary-point bound is finite for every $\lambda_{\text{base}} > 0$ and grows as $\lambda_{\text{base}} \to 0^+$; it is not a global bound along arbitrary training trajectories.

At $u^i = 0$, the mapping is continuous but can be non-differentiable when the two position margins differ. Its one-sided derivatives are $R^{-,i} = \min(\delta^i, a^i_{\text{prev}} - a^i_{\min})$ and $R^{+,i} = \min(\delta^i, a^i_{\max} - a^i_{\text{prev}})$. Automatic differentiation selects a valid one-sided value; away from position boundaries both equal δ^i. The asymmetric case has measure zero under a continuous latent policy and caused no observed convergence issue (ğ3.3).

D STRUCTURE AND HYPERPARAMETERS

The complete DD-SRad algorithm is presented in Algorithm 1.

Algorithm 1: DD-SRad (Off-Policy Backbone)

Input: Per-dimension rate limits $\boldsymbol{\delta}$, latent action regularization coefficient λ_{base}, position bounds $[a_{\min}, a_{\max}]$, off-policy backbone $\mathcal{B}$

Initialize augmented replay buffer $\mathcal{D}$, policy network π_θ, action-value function Q_ϕ, and target network $Q_{\phi'}$;

for *each training step* **do**

- **Environment interaction:** Read s_t from the environment, retrieve previous action a_{t-1} from the augmented state, construct augmented state $\tilde{s}_t = (s_t, a_{t-1})$;
- **Constrained action generation (DD-SRad core):** Sample latent action $u_t \sim \pi_\theta(\cdot \mid \tilde{s}_t)$;
- Compute effective radius $R^i_{\text{eff}}(u^i_t, a^i_{t-1})$ independently per dimension i (Eq. 3); apply per-dimension spherical squashing to obtain a_t (Eq. 4);
- Execute a_t, obtain r_t and s_{t+1}, store $(\tilde{s}_t, a_t, r_t, \tilde{s}_{t+1})$ in $\mathcal{D}$;
- **Q-function update (determined by backbone $\mathcal{B}$):** Sample a batch from $\mathcal{D}$; update Q_ϕ according to the Bellman target of backbone $\mathcal{B}$;
- **Policy gradient update:** Minimize $\mathcal{L}_{\text{actor}}$ per Eq. (14); backpropagate gradient through the DD-SRad Jacobian (Proposition 2.5) via the chain rule to θ; under the SAC backbone, compute $\log \pi_\theta(a|\tilde{s})$ via change of variables per Eq. (15);
- Update target network $\phi' \leftarrow \tau\phi + (1-\tau)\phi'$;

All MuJoCo benchmark experiments were run on a single NVIDIA RTX A5000 GPU (24 GB VRAM), with 1.5–2M training steps per environment-backbone-constraint configuration; single-run training time per environment is approximately 2–3 hours. IsaacLab simulation experiments were also conducted on a single A5000 using 4096 parallel simulation environments: H1 rough terrain was trained for 50k gradient steps with approximately 6 hours per run; G1 flat terrain was trained for 100k gradient steps with approximately 12 hours per run; 5 seeds were run serially, with complete H1 and G1 experiments taking approximately 30 hours and 60 hours respectively. Code is based on PyTorch 3.11 and IsaacLab 5.1.0, with Ubuntu 22.04 as the server operating system.

Table 4 lists the complete network and training hyperparameters used in all experiments. Unless otherwise specified, all experiments use the same network architecture and training configuration.

Table 5 lists the constraint parameter settings for each MuJoCo experiment.

Table 6 lists the per-joint-group constraint parameters for each robot in the simulation experiments.

Table 7 lists the hyperparameters for each baseline method.

Note: the network architecture, learning rates, and buffer sizes of all baseline methods are identical to DD-SRad to ensure fair comparison.

E EXTENDED BENCHMARK PERFORMANCE

Figure 4 presents the mean return learning curves across four MuJoCo environments (Ant-v5, Humanoid-v5, HalfCheetah-v5, Hopper-v5) under tight heterogeneous constraints, for both SAC and TD3 backbones, covering all six methods. Shaded regions indicate mean $\pm$ standard deviation over 5 random seeds. Under wide homogeneous constraints all methods converge to similar levels; the curves under tight heterogeneous constraints are shown here to visualize the training stability differences discussed in §3.1, in particular the systematic collapse of SRad-Strict (high CV) versus the stable convergence of DD-SRad.

Table 8 presents the quantitative performance from the ğ3.1 standard benchmark experiments under wide homogeneous constraints (Humanoid-v5: $\boldsymbol{\delta} = [0.4]^d$; all others: $\boldsymbol{\delta} = [1.0]^d$), for reference alongside the tight heterogeneous constraint results in the main Table 2. Under wide constraints, the rate limits exert weak actual constraint force on the policy, with CVR approaching 0 for all methods

Table 4: Network architecture and training hyperparameters (common to all experiments)

Category	Parameter	Value
Network Architecture	Actor hidden layers	[256, 256]
	Critic hidden layers	[256, 256]
	Activation function	ReLU
	Output activation	Tanh (Actor mean), Identity (log_std)
Optimizer	Actor learning rate	3×10^{-4}
	Critic learning rate	3×10^{-4}
	Alpha learning rate	3×10^{-4}
	Optimizer type	Adam
	Gradient clipping	1.0
Training parameters	Batch size	256
	Replay buffer size	1,000,000
	Learning starts	10,000
	Train frequency	1
	Gradient steps	1
	Target update τ	0.005
	Discount γ	0.99
Other	Evaluation frequency	Every 5,000 steps
	Evaluation episodes	5
	Random seeds	12345, 22345, 32345, 42345, 52345

Table 5: Constraint parameter settings (MuJoCo experiments, by experiment group)

Experiment	Environment	Constraint type	$\boldsymbol{\delta}$
ğ3.1 (Exp. 1/2)	Ant-v5	Wide homogeneous	$[1.0]^8$
		Tight heterogeneous	$[0.2^4, 0.5^4]$
	Humanoid-v5	Wide homogeneous	$[0.4]^{17}$
		Tight heterogeneous	$[0.8^6, 0.5^6, 0.2^5]$
	Hopper-v5	Wide homogeneous	$[1.0]^3$
		Tight heterogeneous	$[0.2, 0.5, 0.5]$
	HalfCheetah-v5	Wide homogeneous	$[1.0]^6$
		Tight heterogeneous	$[0.2^3, 0.5^3]$
ğ3.3 (Exp. 3)	Ant-v5 (wide)	All methods	$[1.0]^8$
	Ant-v5 (tight)	All methods	$[0.2^4, 0.5^4]$

Table 6: Constraint parameters for simulation experiments (from official technical specifications; normalized to joint position increment units, $\delta_i = \omega^i_{\max} \times \Delta t \times \eta$, $\Delta t = 0.02\,\text{s}$, η is the safety factor)

Robot	Terrain	Joint group	No. joints	δ (normalized)
Unitree H1	Rough	Hip yaw/roll/pitch (proximal)	6	0.40
		Knee (mid)	2	0.25
		Ankle (distal)	2	0.18
Unitree G1	Flat	Hip pitch/roll (proximal)	4	0.40
		Hip yaw/knee (mid)	4	0.35
		Ankle pitch	2	0.25
		Ankle roll/torso	3	0.18–0.25

Table 7: Baseline method hyperparameters

Method	Parameter	Value
DD-SRad	λ_{base}	0.005
SAC-BoxPre+	Projection	Per-dimension clip (ℓ_∞ hyperrectangle)
	Critic training	Pre-projected actions + penalty
	Penalty coefficient	Equivalent to DD-SRad λ_{base}
SAC-Post(QP)	Projection	Per-dimension clip post-processing
	Critic training	Pre-projected actions (no penalty)
SRad-Strict	Global radius R	$\min_i \delta^i$
	Other parameters	Same as DD-SRad
SRad-QP	Mapping	SRad ($R = \min_i \delta^i$)
	Post-processing	QP projection to $\mathcal{F}_t$
	Other parameters	Same as DD-SRad

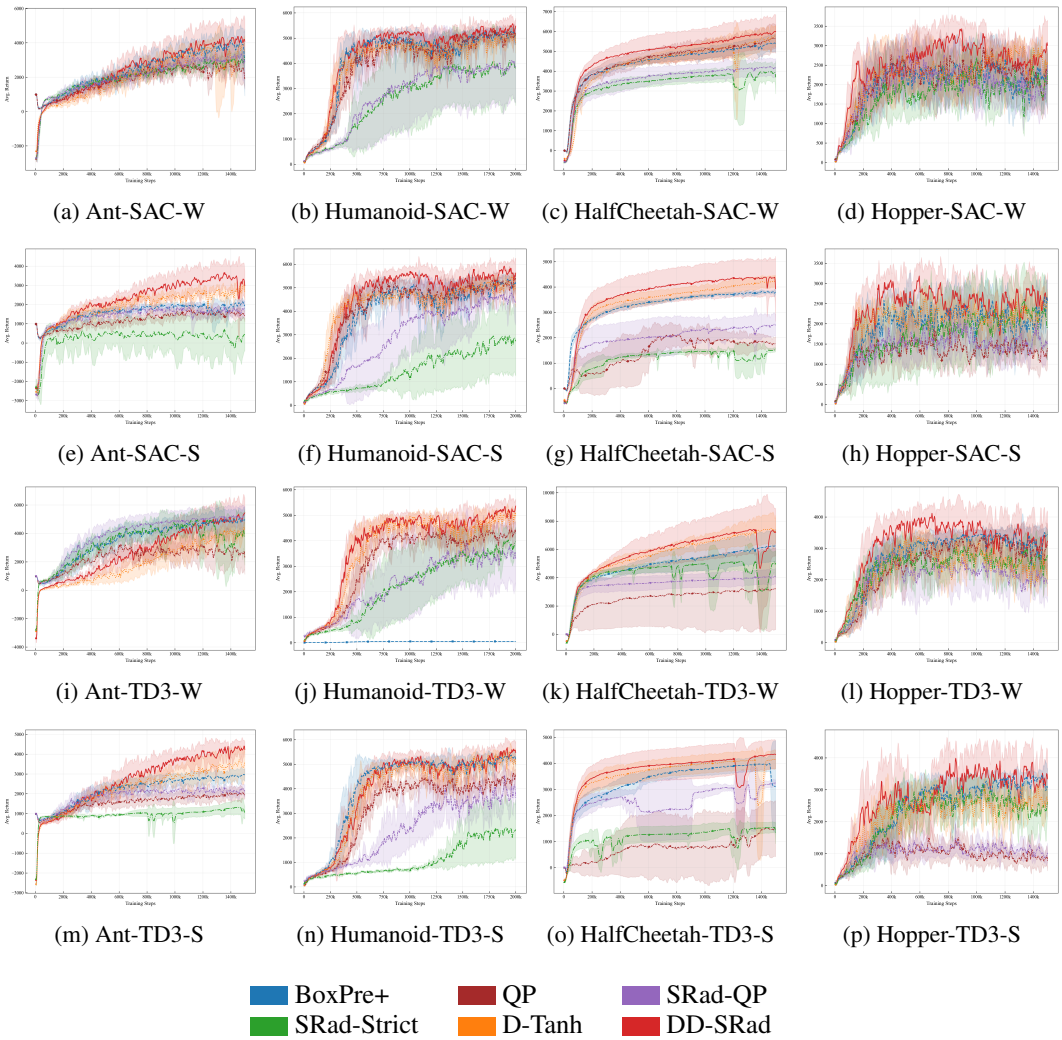


Figure 4: SAC/TD3-Based benchmark performance comparison across four environments.

and Return gaps narrowing substantially—this pattern itself validates the design expectation that DD-SRad introduces no additional performance loss when constraints are non-binding.

E.1 Aggregate Statistical Significance

Because each individual configuration contains only 5 seeds, isolated per-configuration tests have limited power. We therefore additionally compare DD-SRad with each baseline across the 16 paired environment–backbone–constraint configurations. Table 9 reports the number of positive mean-Return differences, the number of per-configuration Welch tests reaching $p < 0.05$, a one-sided paired Wilcoxon signed-rank test across configurations, the median relative Return gain, and Cohen's d over the tight-heterogeneous configurations.

F Per-Dimension Utilization Radar Charts Under Benchmark

To quantify how fully each method exploits its per-dimension rate budget, define the constraint utilization of dimension i as

$$u^i \;=\; \frac{1}{T}\sum_{t=1}^{T}\frac{|\Delta a_t^i|}{\delta^i}, \tag{19}$$

where $\Delta a_t^i = a_t^i - a_{t-1}^i$ is the executed action increment at step t and δ^i is the per-dimension rate limit. By construction $u^i \in [0,1]$, with $u^i = 1$ indicating that the constraint is saturated on average and $u^i = 0$ indicating the dimension is never moved.

Figure 5 presents the post-convergence per-dimension utilization radar charts across four MuJoCo environments under tight heterogeneous constraints, for both SAC and TD3 backbones. Each axis of each polygon corresponds to one action dimension, with the radial value indicating the 5-seed mean utilization $\mathbb{E}[|\Delta a^i|]/\delta^i$ for that dimension. The DD-SRad polygon (red) hugs the boundary corresponding to each dimension's δ^i; BoxPre+ (blue) exhibits systematic inward contraction on high-budget dimensions (axes with larger δ^i); SRad-Strict (green) shows the most severe collapse on high-budget dimensions, consistent with Theorem 2.7's prediction that ℓ_2 parameterization achieves utilization of approximately $\min_j \delta^j/\delta^i$ on those dimensions; QP Post (dark red) exhibits inflated Util due to boundary-dwelling behavior but carries no geometric coverage meaning (see ğ3.1 footnote). Under wide homogeneous constraints (panels (a)(c)(i)(k)), performance gaps across methods narrow, consistent with the expectation that constraints are non-binding.

G Per-Dimension Utilization Radar Charts Under Heterogeneous Constraints

Figure 6 presents the per-dimension constraint utilization radar charts of DD-SRad under wide homogeneous (blue) and tight heterogeneous (red) constraints, for all four environments with SAC and TD3 backbones. Each axis corresponds to one action dimension, with radial value indicating $\mathbb{E}[|\Delta a^i|]/\delta^i$. The layered polygon structure under tight constraints—where each dimension independently approaches its own δ^i boundary rather than being uniformly truncated to $\min_i \delta^i$—provides visual corroboration of the ℓ_∞-tight coverage property established in Theorem 2.7 and the quantitative Gap analysis of §3.2.

H Simulation Reward Settings

Both simulation robots use the native reward implementations and weights supplied by their corresponding open-source IsaacLab task configurations; no reward term or weight is customized specifically for DD-SRad. Table 10 reproduces the principal configured components for completeness. The reward is written as $r_t = \sum_k w_k \phi_k$, where the native curriculum coefficient is $c_f = \min(\bar{L}/L^*, 1)$, $\bar{L}$ is the exponential moving average episode length, and L^* is the target episode length.

Table 8: Supplementary SAC results under wide homogeneous constraints (5-seed mean±std; best in **bold**; † see note).

(a) Ant-v5: $\boldsymbol{\delta} = [1.0]^8$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	3737 ± 729	19.5	0.323 ± 0.014
Post(QP)†	2437 ± 467	19.1	0.341 ± 0.011
SRad-Strict	2952 ± 385	**13.0**	0.244 ± 0.009
SRad-QP	3122 ± 719	23.0	0.253 ± 0.012
D-Tanh	4039 ± 725	18.0	0.301 ± 0.009
DD-SRad (Ours)	$\mathbf{4199 \pm 957}$	22.8	0.334 ± 0.026

(b) Humanoid-v5: $\boldsymbol{\delta} = [0.4]^{17}$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	5144 ± 145	2.8	0.158 ± 0.014
Post(QP)†	5163 ± 175	3.4	0.289 ± 0.018
SRad-Strict	3918 ± 1350	34.5	0.057 ± 0.016
SRad-QP	3903 ± 1324	33.9	0.060 ± 0.015
D-Tanh	5212 ± 135	**2.6**	0.253 ± 0.029
DD-SRad (Ours)	$\mathbf{5384 \pm 261}$	4.8	0.241 ± 0.049

(c) HalfCheetah-v5: $\boldsymbol{\delta} = [1.0]^6$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	5383 ± 412	**7.7**	0.528 ± 0.009
Post(QP)†	5582 ± 704	12.6	0.578 ± 0.052
SRad-Strict	3921 ± 469	12.0	0.273 ± 0.020
SRad-QP	4186 ± 439	10.5	0.270 ± 0.006
D-Tanh	5646 ± 517	9.2	0.626 ± 0.018
DD-SRad (Ours)	$\mathbf{5925 \pm 782}$	13.2	0.555 ± 0.103

(d) Hopper-v5: $\boldsymbol{\delta} = [1.0]^3$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	2226 ± 437	19.6	0.145 ± 0.009
Post(QP)†	2037 ± 425	20.9	0.167 ± 0.009
SRad-Strict	1992 ± 145	**7.3**	0.193 ± 0.016
SRad-QP	1915 ± 188	9.8	0.185 ± 0.017
D-Tanh	2649 ± 194	**7.3**	0.361 ± 0.016
DD-SRad (Ours)	$\mathbf{2764 \pm 405}$	14.7	0.325 ± 0.012

Table 8, continued: TD3 backbone.

(e) Ant-v5: $\boldsymbol{\delta} = [1.0]^8$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	4906 ± 708	14.4	0.307 ± 0.021
Post(QP)†	2613 ± 1011	38.7	0.309 ± 0.033
SRad-Strict	3756 ± 1287	34.3	0.236 ± 0.015
SRad-QP	4973 ± 716	14.4	0.214 ± 0.009
D-Tanh	4058 ± 483	**11.9**	0.353 ± 0.005
DD-SRad (Ours)	$\mathbf{5088 \pm 881}$	17.3	0.362 ± 0.031

(f) Humanoid-v5: $\boldsymbol{\delta} = [0.4]^{17}$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	5018 ± 144	2.9	0.154 ± 0.025
Post(QP)†	4289 ± 111	**2.6**	0.537 ± 0.044
SRad-Strict	3884 ± 404	10.4	0.127 ± 0.020
SRad-QP	3483 ± 1009	29.0	0.018 ± 0.005
D-Tanh	5027 ± 144	2.9	0.242 ± 0.047
DD-SRad (Ours)	$\mathbf{5107 \pm 343}$	6.7	0.207 ± 0.020

(g) HalfCheetah-v5: $\boldsymbol{\delta} = [1.0]^6$, $\kappa = 1.0$

Method	Return ↑	CV ↓	Util
BoxPre+	6198 ± 1093	17.6	0.510 ± 0.059
Post(QP)†	3163 ± 2881	91.1	0.589 ± 0.102
SRad-Strict	4119 ± 1665	40.4	0.264 ± 0.035
SRad-QP	4051 ± 521	**12.9**	0.339 ± 0.023
D-Tanh	6596 ± 942	14.3	0.611 ± 0.064
DD-SRad (Ours)	$\mathbf{6650 \pm 2477}$	37.2	0.523 ± 0.119

(h) Hopper-v5: $\boldsymbol{\delta} = [1.0]^3$, $\kappa = 1.0$

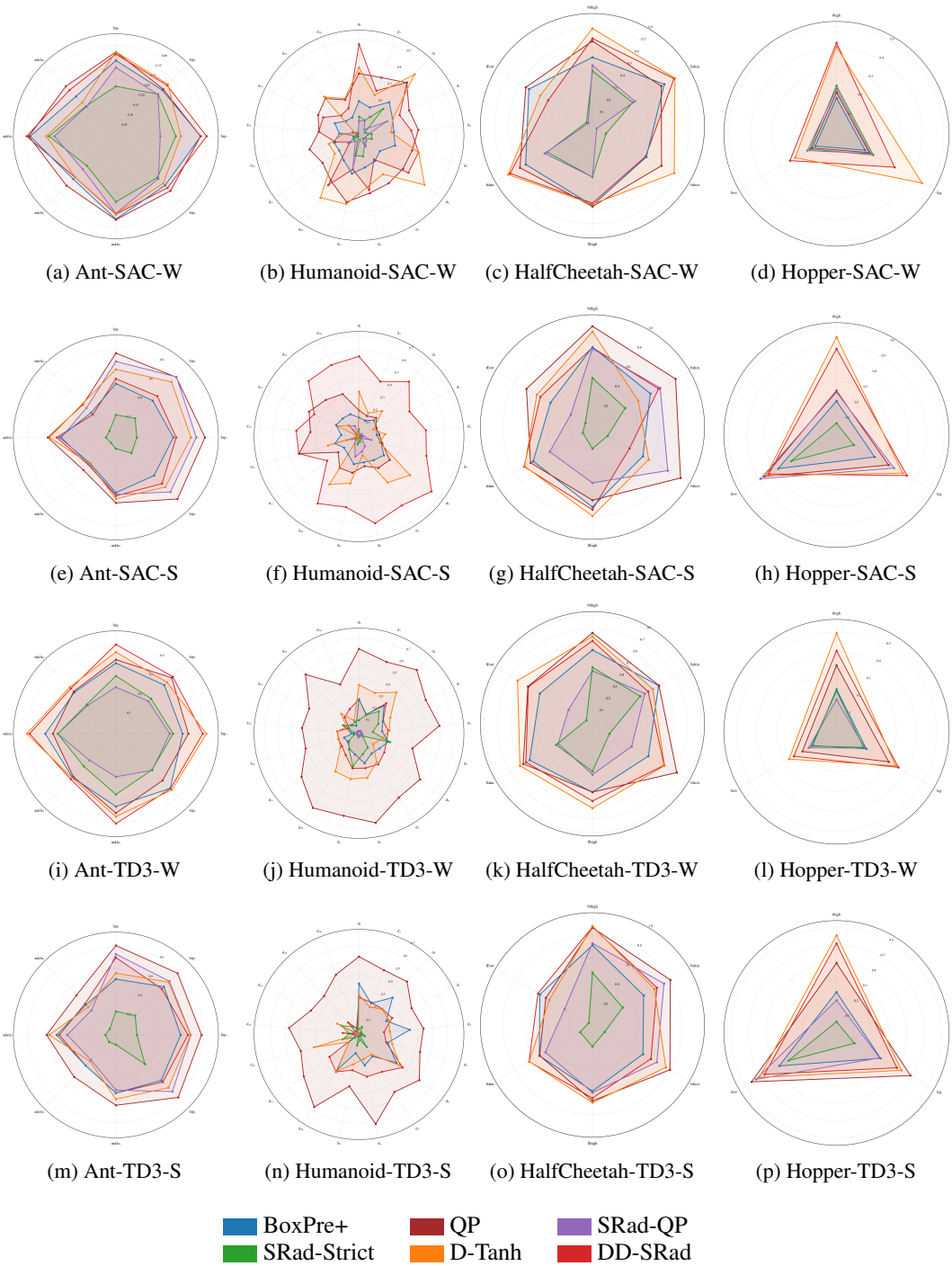
Method	Return ↑	CV ↓	Util
BoxPre+	3157 ± 126	**4.0**	0.178 ± 0.006
Post(QP)†	2915 ± 473	16.2	0.281 ± 0.023
SRad-Strict	2598 ± 556	21.4	0.168 ± 0.017
SRad-QP	2030 ± 406	20.0	0.163 ± 0.018
D-Tanh	2303 ± 122	5.3	0.371 ± 0.020
DD-SRad (Ours)	$\mathbf{3216 \pm 478}$	14.9	0.337 ± 0.017

Notes: CVR$= 0$ for all parameterization-based methods (Theorem 2.4). Return gaps narrow relative to tight constraints, as expected when constraints are non-binding. †Post(QP) Util is inflated by clip boundary dwelling (Table 2).

Table 9: Aggregate comparison of DD-SRad against each constrained baseline over 16 paired configurations.

Baseline	**Sign wins**	**Welch** $p < 0.05$	**Wilcoxon** W (p)	**Median gain**	**Cohen's** d **(tight)**
BoxPre+	16/16	6/16	136 (1.5×10^{-5})	+9.0%	1.85
D-Tanh	16/16	5/16	136 (1.5×10^{-5})	+6.6%	1.23
SRad-Strict	16/16	15/16	136 (1.5×10^{-5})	+46.7%	4.42
SRad-QP	16/16	15/16	136 (1.5×10^{-5})	+45.5%	5.05
Post(QP)	16/16	13/16	136 (1.5×10^{-5})	+83.5%	5.50

Notes: Welch tests use the 5 seed-level Returns within each configuration. The Wilcoxon test is a one-sided paired signed-rank test over the 16 configuration-level mean Returns; $W = 136$ is the maximum attainable statistic for 16 positive paired differences.



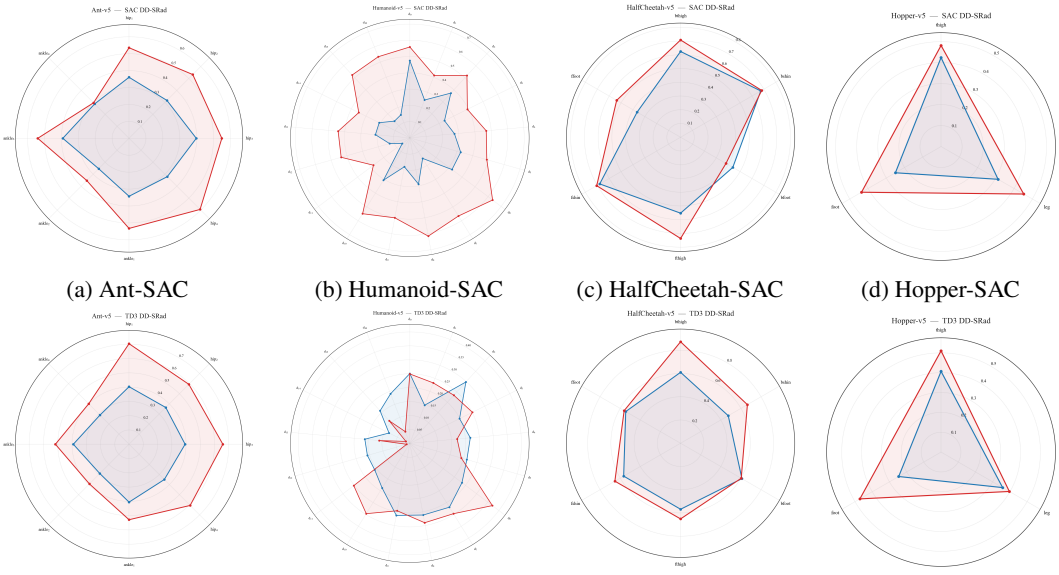
(a) Ant-SAC-W (b) Humanoid-SAC-W (c) HalfCheetah-SAC-W (d) Hopper-SAC-W

(e) Ant-SAC-S (f) Humanoid-SAC-S (g) HalfCheetah-SAC-S (h) Hopper-SAC-S

(i) Ant-TD3-W (j) Humanoid-TD3-W (k) HalfCheetah-TD3-W (l) Hopper-TD3-W

(m) Ant-TD3-S (n) Humanoid-TD3-S (o) HalfCheetah-TD3-S (p) Hopper-TD3-S

Figure 5: Per-dimension constraint utilization radar charts of all methods on four environments under tight heterogeneous constraints (SAC/TD3 backbones).



(a) Ant-SAC (b) Humanoid-SAC (c) HalfCheetah-SAC (d) Hopper-SAC

(e) Ant-TD3 (f) Humanoid-TD3 (g) HalfCheetah-TD3 (h) Hopper-TD3

Figure 6: Per-dimension constraint utilization radar charts of DD-SRad under wide homogeneous (blue) and tight heterogeneous (red) constraints with SAC/TD3 backbones.

Table 10: Reward function definitions for Unitree H1 Rough.

Reward term	Formula	Weight
Linear velocity tracking	$\exp\big(-\|(v_{xy} - v_{xy}^{\text{cmd}})\|^2/0.25\big)$	1.5
Angular velocity tracking	$\exp\big(-(w_z - w_z^{\text{cmd}})^2/0.25\big)$	1.0
Survival bonus	constant	0.02
Orientation penalty	$-\|g_{xy}\|^2 \cdot c_f$	3.0
Joint default position	$-\|\theta - \theta_{\text{default}}\|^2 \cdot c_f$	0.05

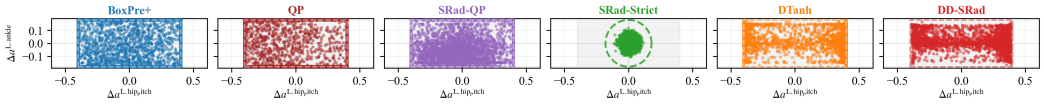
Table 10, continued: Unitree G1 Flat (HolosomaRewardComputerV3).

Reward term	Formula	Weight
Linear velocity tracking	$\exp\big(-\|(v_{xy} - v_{xy}^{\text{cmd}})\|^2/0.25\big)$	1.5
Angular velocity tracking	$\exp\big(-(w_z - w_z^{\text{cmd}})^2/0.25\big)$	1.0
Survival bonus	constant	0.02
Orientation penalty	$-\|g_{xy}\|^2 \cdot \max(c_f, 0.3)$	3.0
Joint default position (non-knee)	$-\|\theta_{\neg k} - \theta_{\text{def}}^{\neg k}\|^2 \cdot c_f$	0.05
Joint default position (knee)	$-\|\theta_k - \theta_{\text{def}}^{k}\|^2 \cdot c_f$	0.05×0.5
Foot height	$\exp\big(-\sum(h_f - h_f^{\text{swing}})^2/(2\sigma_h^2)\big)$	1.0
Knee flexion	$\exp\big(-(\theta_k - \theta_{\text{def}}^k - \kappa_{\text{flex}} \sin\phi_k)^2/(2\sigma_k^2)\big)$	1.5
Minimum knee angle	$-\operatorname{clamp}(\theta_{\min} - \theta_k,\, 0,\, \infty)^2 \cdot c_f$	2.0
Stride length	$\exp\big(-\sum(x_f^b - L\sin\phi_f)^2/(2\sigma_s^2)\big)$	1.5
Foot parallelism	$-(\psi_{f,L} - \psi_{f,R})^2 \cdot c_f$	1.0
Foot clearance	$-\operatorname{clamp}(d_{\min} - \lvert y_{f,L} - y_{f,R}\rvert,\, 0,\, \infty)^2 \cdot c_f$	4.0
Anti-crossing	$-\operatorname{clamp}(y_{f,R} - y_{f,L},\, 0,\, \infty)^2 \cdot c_f$	5.0
Hip roll	$-\|\theta_{\text{hip_roll}} - \theta_{\text{def}}^{\text{hr}}\|^2 \cdot c_f$	3.0
Hip yaw	$-\|\theta_{\text{hip_yaw}} - \theta_{\text{def}}^{\text{hy}}\|^2 \cdot c_f$	3.0
Torso joint	$-\|\theta_{\text{torso}} - \theta_{\text{def}}^{\text{torso}}\|^2 \cdot \max(c_f, 0.3)$	2.0
Ankle pitch	$-\|\theta_{\text{ap}} - \theta_{\text{def}}^{\text{ap}}\|^2 \cdot c_f$	2.0
Foot slip	$-\sum_f \|\dot{p}_f^{xy}\| \cdot \mathbf{1}[\|F_f\| > 1\,\text{N}]$	0.5

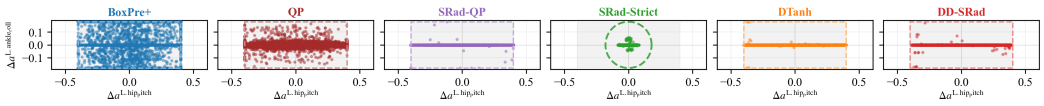
ϕ_f: gait phase; L: stride length target; $d_{\min}$: minimum foot clearance; σ: Gaussian widths for each term.

I Reachable Set Scatter of H1 and G1

Figure 7 presents the empirical reachable sets of each method in the $(\Delta a^{L.\text{hip_pitch}}, \Delta a^{L.\text{ankle_roll}})$ plane for H1 and G1, providing execution-level geometric verification complementary to the Gap metric in Table 3(b).



(a) H1 Empirical Reachable Sets



(b) G1 Empirical Reachable Sets

Figure 7: Empirical Reachable Sets

On H1, DD-SRad scatters uniformly within the constraint rectangle $[-\delta^{\text{hip}}, \delta^{\text{hip}}] \times [-\delta^{\text{ankle}}, \delta^{\text{ankle}}]$, with both dimensional rate budgets independently and fully utilized, consistent with the balanced utilization of Util-P$= 0.38$, Util-D$= 0.48$; D-Tanh scatter morphology closely resembles DD-SRad, reflecting that the shared R_{eff}^i mechanism produces similar coverage structures under the same geometric constraints; SRad-Strict scatter concentrates within the disk of radius $R = \min_i \delta^i$, with the high-budget dimension (hip_pitch) reachable set uniformly compressed to the tightest constraint magnitude, directly corresponding to the utilization collapse Util-P$= 0.08$ and Fall$= 100\%$.

On G1, DD-SRad scatter distributes broadly along the $\Delta a^{L.\text{hip_pitch}}$ axis and is highly concentrated near zero along $\Delta a^{L.\text{ankle_roll}}$, forming a near-horizontal band structure—a direct manifestation of the position-adaptive R^i_{eff} mechanism interacting with the flat terrain task structure: hip_pitch joints bear gait propulsion (large δ, fully utilized), while ankle_roll joints have extremely low lateral stability demands under flat terrain's low perturbation, and the policy converges to near-zero ankle_roll through independent per-dimension optimization. D-Tanh exhibits the same morphology, confirming that this solution structure is jointly determined by ℓ_∞ geometry and task rewards, independent of the specific squashing function; the task-metric gap (VTE: 0.138 vs 0.207, Return: 5473 vs 5145) reflects the effective gradient region advantage of spherical radial squashing over $\tanh$ (Proposition 2.5). BoxPre+ (Gap$= 0.01$) forces uniform coverage—a geometric side effect of ℓ_∞ clip projection—preventing task-aligned allocation and directly manifesting as VTE$= 0.224$ ($+62\%$ vs DD-SRad). SAC-Post(QP) covers the rectangle interior with reduced boundary density, consistent with QP's conservative margin at feasible-set vertices (Return$= 4954$, $\approx 10\%$ below DD-SRad). SRad-Strict concentrates within the ℓ_2 disk of radius $R = \min_i \delta^i$, directly corresponding to Util-P$= 0.08$ and Return$= 1312$.

The scatter morphology is fully consistent with the Gap metric in Table 3(b), forming cross-platform verification from MuJoCo to IsaacLab: ℓ_∞-exact parameterization guarantees geometric coverage and enables task-gradient-guided dimensional allocation—balancing rate budgets under high-perturbation rough terrain (H1) and focusing on propulsion joints for precise velocity tracking on flat terrain (G1)—an adaptive capability neither BoxPre+'s clip nor SRad-Strict's global radius can achieve.

J Per-Dimension Action Increment Probability Density of H1 and G1

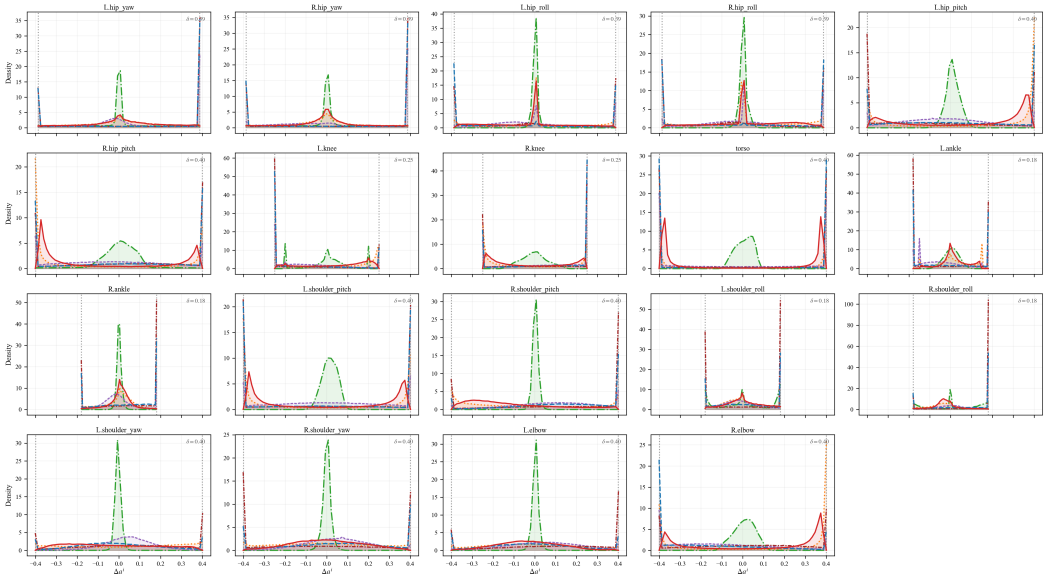
To visualize the empirical distribution of executed action increments, the probability density of Δa^i for each controlled dimension is plotted. Given N observed samples $\{\Delta a^i_k\}_{k=1}^N$, the density is estimated via a normalized histogram with B uniform bins over $[-\delta^i,\, \delta^i]$:

$$\hat{p}(x) \;=\; \frac{c_b}{N \cdot w}, \qquad x \in \text{bin } b, \tag{20}$$

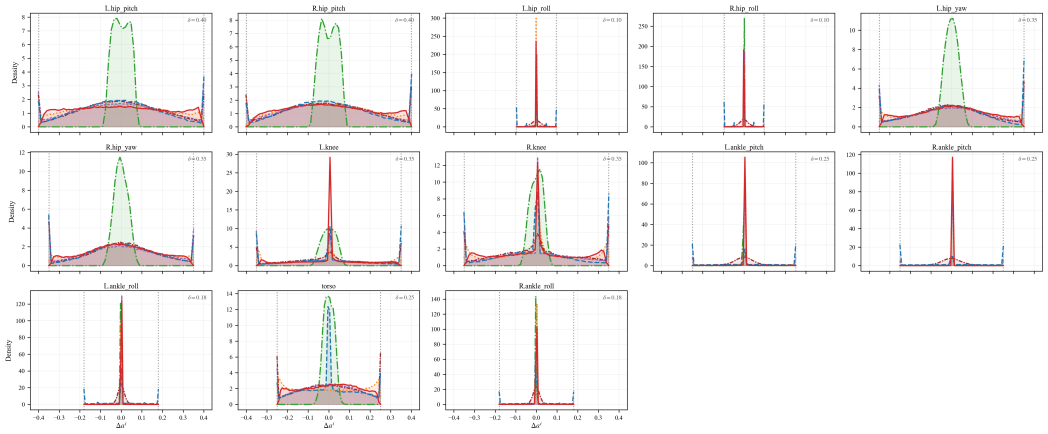
where c_b is the count of samples in bin b and $w = 2\delta^i/B$ is the bin width. This satisfies $\int_{-\delta^i}^{\delta^i} \hat{p}(x)\, dx = 1$, so $\hat{p}$ is a valid probability density function. Values of $\hat{p} \gg 1$ are physically meaningful and indicate that samples concentrate in a narrow sub-interval of $[-\delta^i, \delta^i]$.

Figure 8 presents per-dimension probability density $\hat{p}(\Delta a^i)$ for each method on H1 and G1, forming a three-layer complementary verification alongside the mean utilization metric of the radar charts and the geometric morphology of the scatter plots. Density values far exceeding 1 indicate that action increments are highly concentrated in a narrow sub-interval of the constraint range, which is the probability density manifestation of the policy actively choosing a conservative solution in a specific dimension.

H1 (Rough Terrain, 10 Dimensions). DD-SRad exhibits a broad unilateral distribution on the hip_pitch group ($\delta = 0.40$) with probability mass clearly shifted toward the positive constraint boundary, reflecting the demand of rough terrain gait propulsion for sustained forward hip joint rate output—the policy fully exploits the rate budget of the large-δ dimensions to resist dynamic perturbations and maintain forward momentum. D-Tanh (orange) shows nearly identical distribution morphology in the same dimensions, consistent with the shared R^i_{eff} geometry; the systematic return gap (34.56 vs 37.14) arises from the difference in squashing function Jacobian decay rate, not from reachable set geometry. SRad-Strict (green) exhibits a clear bimodal structure in the hip_pitch dimension with peaks far from the constraint boundary, directly corresponding to the mechanism by which the global ℓ_2 radius $R = \min_i \delta^i$ uniformly compresses the hip joint large-δ step size to the ankle joint magnitude, blocking effective gait propulsion; its Fall$= 100\%$ is the inevitable consequence of this geometric mismatch on rough terrain. In the ankle dimensions ($\delta = 0.18$), DD-SRad shows a moderately spread distribution, indicating that the ankle joint rate budget receives substantive utilization under rough terrain's dynamic balance demands; SRad-Strict ankle dimensions also show extreme concentration, because global radius compression prevents the policy from producing



(a) H1 per-dimension action distribution



(b) G1 per-dimension action distribution

BoxPre+ QP SRad-QP
SRad-Strict D-Tanh DD-SRad

Figure 8: H1 and G1 per-dimension action distribution

effective recovery actions. BoxPre+ (blue) shows relatively flat distributions across all dimensions, reflecting the uniform projection property of ℓ_∞ clip within the feasible set, but this flat distribution corresponds to Return$= 23.11$ significantly lower than DD-SRad, validating that uniform activation is not equivalent to task efficiency (ğ4.2). In the arm and shoulder joint dimensions (shoulder/elbow, $\delta = 0.18$–0.40), SRad-Strict similarly shows bimodal or concentrated distributions in large-δ dimensions, while DD-SRad and D-Tanh maintain broad distributions matched to task demands, with the overall pattern consistent with the leg joints.

G1 (Flat Terrain, 13 Dimensions). The G1 distribution plots reveal the task-adaptive structural differences across methods under flat terrain's low-perturbation conditions, forming a multi-perspective verification that closes layer by layer with the scatter plots and radar charts.

In the hip_pitch dimension ($\delta = 0.40$, the main gait propulsion joint), DD-SRad (red) exhibits a broad distribution covering most of the constraint interval with probability mass well spread out, directly corresponding to the high utilization of Util-P$= 0.40$. D-Tanh (orange) shows nearly identical distribution morphology to DD-SRad, further confirming that this broad utilization structure is jointly determined by the R^i_{eff} mechanism and task rewards, not a specific product of the squashing function. SRad-Strict (green) exhibits a bimodal distribution in the hip_pitch dimension with peaks concentrated near $\pm \min_i \delta^i = \pm 0.10$, directly manifesting the mechanism by which the global ℓ_2 radius uniformly compresses the rate of all dimensions to the tightest hip_roll constraint magnitude, causing the actual reachable step size of the gait propulsion joint to be only 25% of DD-SRad's.

In distal joint dimensions such as ankle_pitch ($\delta = 0.25$) and ankle_roll ($\delta = 0.18$), both DD-SRad and D-Tanh exhibit highly concentrated zero-mean spikes ($\hat{p} \gg 1$), i.e., probability mass highly concentrated in a narrow band around $\Delta a^i \approx 0$. This is the optimal solution to which the policy actively converges through task gradient optimization under flat terrain's low-perturbation conditions: flat ground imposes extremely low demands on ankle joint lateral stabilization and pitch adjustment, so the policy compresses distal joint actions to near zero to reduce ineffective consumption, focusing on hip_pitch and other large-δ propulsion joints for precise velocity tracking, ultimately achieving the lowest velocity tracking error of all methods at VTE$= 0.138$. The R^i_{eff} mechanism allows each dimension to independently respond to this task structure—the adaptive tightening of the effective radius near position boundaries for distal joints does not affect the full exploration of the hip_pitch dimension, with the two completely decoupled at the parameterization level. BoxPre+ also shows some zero-vicinity concentration in ankle dimensions, but with broader spread than DD-SRad, reflecting that ℓ_∞ clip projection within the feasible set still forces retention of non-zero activation components in each dimension; the cost is quantified as VTE$= 0.224$. In the hip_roll dimension ($\delta = 0.10$, G1's tightest constraint), all methods exhibit extremely high zero-mean spikes, reflecting both the strict constraint limitation and the fact that flat terrain imposes almost no demand on hip lateral roll—consistent with the task characteristics. SRad-Strict shows bimodal distributions in intermediate-δ dimensions such as torso, consistent with the expectation that the global ℓ_2 radius produces varying degrees of mismatch across dimensions with different constraint magnitudes; the severe degradation of Return$= 1312$ is the cumulative result of geometric mismatch across all dimensions.

K LIMITATIONS

DD-SRad deterministically enforces the modeled per-step joint-command rate and position limits. This guarantee concerns the policy-output layer and the specified box constraint; it does not by itself establish closed-loop collision avoidance, trajectory-level safety, or task success.

The IsaacLab experiments use public Unitree specifications, native task rewards, contact dynamics, and hardware-relevant metrics, but remain controlled simulations rather than physical-robot validation. Full deployment additionally depends on actuator latency, low-level torque tracking, thermal limits, unmodeled contact dynamics, communication delay, and state-estimation error. Accordingly, the present results should be interpreted as a step toward hardware-informed policy learning rather than a complete real-world safety validation.

The diagonal Jacobian is naturally compatible with off-policy algorithms such as SAC and TD3, whose action-conditioned critic provides a per-dimension gradient signal. Extension to on-policy

methods such as PPO may require an action-conditioned advantage estimator or other variance-reduction mechanisms under strongly heterogeneous constraints.

The current formulation targets independent per-dimension boxes. Several extensions are structurally possible but are not certified by the present theorems: a shared multiplier may enforce cumulative energy or torque budgets, producing a diagonal-plus-low-rank Jacobian; state-dependent limits $\delta^i(s, a_{\mathrm{prev}})$ preserve the diagonal action Jacobian when the limits are treated as state-conditioned quantities; and DD-SRad may be composed with a CBF-QP so that the analytic layer handles rate limits continuously while the QP activates only when additional state-safety constraints are threatened. Establishing guarantees and empirical performance for these coupled settings is left to future work.