SAFEOT: A UNIFIED SAFETY-PRICE VIEW OF SAFE REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Safe reinforcement learning (safe RL) is commonly formulated in constrained Markov decision processes (CMDPs) with explicit constraints on expected safety costs, or optimized through scalarized penalty and Lagrangian objectives. Although both address the same reward–safety trade-off, they are usually treated as distinct families of methods rather than within a single operational framework, and common multi-constraint extensions either tune one Lagrange multiplier per cost or rely on scalarized penalties, which require per-constraint weighting and are prone to oscillation as the number of costs grows. This makes it difficult to jointly represent and control multiple safety constraints with distinct cost functions and bounds. We propose SafeOT, which recasts safe policy optimization as a budget-constrained optimal-flow problem over the transition graph of a CMDP. This representation places penalty-based and constraint-based safe RL on one spectrum and handles all constraints in a single projection. Coupling this projection with a PPO-like per-iteration policy update yields SafeOT, a GPU-efficient algorithm whose projection takes under 0.2 seconds per update. On 16 Safety-Gymnasium and Bullet-Safety-Gym tasks, SafeOT reduces cumulative constraint violation relative to TRPO-Lagrangian by 63% and 70%, respectively, lowering it on every task with comparable return on Safety-Gymnasium. With six to eight per-joint budgets on MuJoCo robots, its unclipped variant also incurs less violation than Lagrangian and PID Lagrangian updates.

1 INTRODUCTION

Embodied AI is moving from instruction following in curated demonstrations toward long-horizon control in environments where mistakes have physical consequences. A robot may complete its task while colliding with obstacles, damaging objects or entering a hazardous region, and average task success does not capture these failures (Zhang et al., 2025a; 2026b). The requirements concern distinct events and need not share one tolerance: exposure to a hazard may admit a cost budget (a soft constraint), whereas an irreversible event calls for a hard constraint. A deployable policy therefore has to control several semantically distinct costs rather than a single scalarized reward–cost objective.

The standard formalism for this requirement is the constrained Markov decision process (CMDP), which maximizes reward subject to expected safety costs $J_{c_k}(\pi) \leq b_k$ (Altman, 1999). Safe reinforcement learning (safe RL) implements it in two families. Penalty-based methods fold the costs into the reward with fixed weights (Zhang et al., 2022). Constraint-based methods enforce the budgets, by adapting one multiplier per budget from observed costs with Lagrangian or proportional–integral–derivative (PID) updates (Tessler et al., 2019; Stooke et al., 2020) or by constrained trust-region and projection updates (Achiam et al., 2017; Yang et al., 2020; Zhang et al., 2020; Yang et al., 2022). The two families are usually developed separately, although a fixed penalty and a Lagrangian update differ only in how they set the cost weights of the same penalized update. A fixed weight must be set in advance and does not reveal which budget is active. A weight updated from observed violations reacts late: with a fixed step size, it exceeds its initial value only after violations have accumulated (Lemma 1), and each budget is priced from its own error history although all share one policy.

Optimal transport (OT) compares distributions through a minimum-cost coupling. The exact problem is a linear program (LP), and entropic regularization makes it solvable by Sinkhorn matrix scaling (Cuturi, 2013). Classical OT moves mass between two sets of nodes with fixed marginals, whereas robot

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.

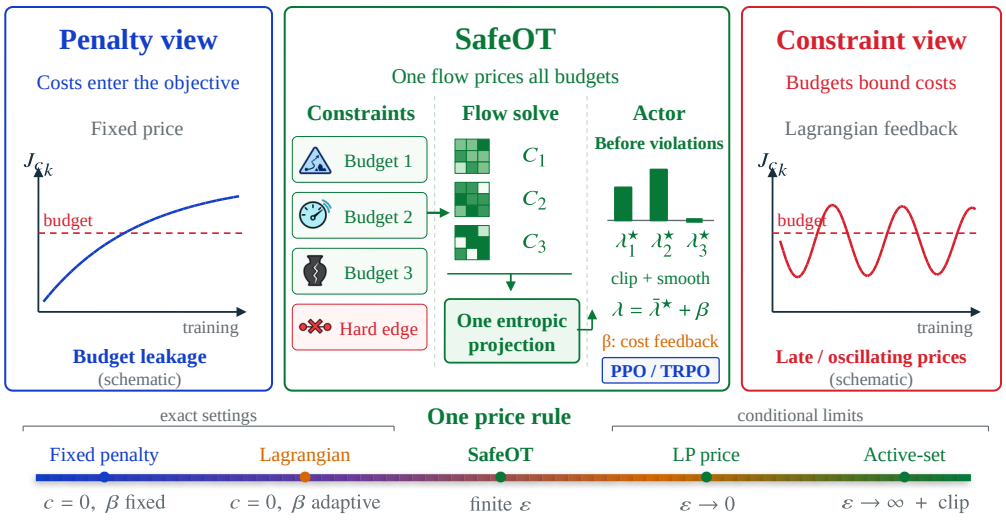

Figure 1: **SafeOT between penalty-based and constraint-based safe RL.** Side panels: schematic failure modes of a fixed penalty (budget leakage) and of a Lagrangian multiplier (oscillation near the budget). Center: all soft constraints bound one flow on the transition graph, and hard constraints remove edges. The budget duals of one entropic projection are the safety prices $\lambda^\star$. Clipped and smoothed into $\bar{\lambda}^\star$ and combined with observed-cost feedback β, they weight the cost advantages of the actor update. Bottom: fixed penalties and Lagrangian updates are settings of this price rule (Table 1), and linear-programming and active-set prices are its limits (Proposition 2).

dynamics form a directed graph on which mass passes through intermediate states and recirculates. Optimal Flow Transport (OFT) replaces the marginal constraints with graph flow conservation, again with entropic matrix iterations (Shi et al., 2025). This raises the central question of this paper:

Can a single flow projection price every safety budget by where the policy is heading, rather than by where it has already failed?

The bridge is the occupancy-measure linear program for discounted CMDPs (Altman, 1999; Puterman, 1994). The discounted occupancy $\rho^\pi(s,a)$ of a policy π records how much probability mass passes through each state–action pair. Expected return and every expected cost are linear in it, and Bellman flow conservation (2) characterizes the occupancies of stationary policies. Aggregating it through the dynamics gives a discounted transition flow (Section 2), so a safe policy update becomes a constrained flow problem on the transition graph.

In this paper, we propose SafeOT (Figure 1), which lifts this flow view into a safe RL algorithm. At each policy iteration, SafeOT solves a single entropic projection that moves the observed flow toward higher advantage under all cost budgets at once, and reads its budget duals as safety prices. These prices, together with a light feedback on observed costs, weight the cost advantages of a PPO-like policy update. All budgets are thus priced jointly inside the update, and a budget can be priced before the policy violates it. In summary, we make three contributions:

- **A unified optimal-flow formulation.** We place penalty-based and constraint-based safe RL in one budget-constrained optimal-flow formulation, in which fixed penalties and Lagrangian updates are special cases of a single price rule.
- **Safety prices from a flow projection.** Viewing discounted CMDP occupancies as graph flows, we turn every cost budget into a linear constraint on the flow and show that the budget duals of one entropic projection act as safety prices, with a bound on the resulting constraint violation.
- **A scalable SafeOT algorithm.** We embed the projection into a PPO-like per-iteration policy update, yielding SafeOT as a GPU-efficient implementation that lowers cumulative constraint violation relative to TRPO-Lagrangian (TRPO-Lag; Ray et al., 2019) on all 16 Safety-Gymnasium and Bullet-Safety-Gym tasks, with comparable return on Safety-Gymnasium.

2 PRELIMINARIES

This section fixes only the notation needed for the flow view. The point is not to introduce a second RL formalism, but to expose one conservation law that is already present in the CMDP occupancy linear program. Section 3 adds the cost budgets and the entropic projection on top of this law, and Section 4 turns the resulting prices into a PPO-like policy update.

CMDP occupancies. A CMDP has states $\mathcal{S}$, actions $\mathcal{A}$, transition kernel P, reward r, nonnegative costs c_k, discount $\gamma \in (0,1)$ and initial distribution $\boldsymbol{\mu}_0$. Its normalized discounted occupancy is $\rho^\pi(s,a) = (1-\gamma)\sum_{t\geq 0}\gamma^t \Pr_\pi(s_t = s, a_t = a)$, so $J_f(\pi) = \sum_{s,a}\rho^\pi(s,a)f(s,a)$. In finite spaces, the CMDP is the occupancy LP (Altman, 1999; Puterman, 1994)

$$\max_{\rho\geq 0} \quad \sum_{s,a}\rho(s,a)r(s,a) \tag{1}$$

$$\text{s.t.} \quad \sum_a \rho(s',a) = (1-\gamma)\boldsymbol{\mu}_0(s') + \gamma\sum_{s,a}P(s'|s,a)\rho(s,a), \quad \forall s', \tag{2}$$

$$\sum_{s,a}\rho(s,a)c_k(s,a) \leq b_k, \quad \forall k. \tag{3}$$

Bellman flow conservation (2) characterizes stationary-policy occupancies. On visited states, the policy is recovered as $\pi(a|s) = \rho(s,a)/\sum_{a'}\rho(s,a')$.

From occupancies to transition flows. Aggregating occupancy through the dynamics gives

$$F^\pi_{ss'} = \sum_a \rho^\pi(s,a)P(s'|s,a), \qquad F^\pi\mathbf{1} - \gamma(F^\pi)^\top\mathbf{1} = (1-\gamma)\boldsymbol{\mu}_0. \tag{4}$$

Initial mass enters through $\boldsymbol{\mu}_0$ and γ discounts incoming flow. This is the bridge used throughout the paper: discounted CMDP occupancies are represented as graph flows with discounted flow conservation, with the converse requiring compatibility with P. On positive-mass edges, the conditional cost $C^\pi_k(s,s') = \sum_a \rho^\pi(s,a)P(s'|s,a)c_k(s,a)/F^\pi_{ss'}$ satisfies $J_{c_k}(\pi) = \langle C^\pi_k, F^\pi\rangle$, where $\langle X, Y\rangle := \sum_{ij}X_{ij}Y_{ij}$. Thus $C^\pi_k(s,s')$ is the average immediate cost conditional on traversing edge (s,s'). Lemma 2 accounts for its change after a policy update.

Optimal Flow Transport. Classical optimal transport moves mass between two fixed marginals. Optimal Flow Transport (OFT) (Shi et al., 2025) instead routes mass over a directed graph: a nonnegative edge flow F must meet a supply–demand vector $\boldsymbol{\sigma}$ through flow conservation, optionally under edge capacities S, at minimum edge cost D with an entropic regularizer,

$$\min_{F\geq 0}\ \langle D, F\rangle - \varepsilon\,\mathcal{H}(F) \qquad \text{s.t.} \qquad F\mathbf{1} - F^\top\mathbf{1} = \boldsymbol{\sigma}, \quad F \leq S, \tag{5}$$

with $\mathcal{H}(F) = -\sum_{ij}F_{ij}(\log F_{ij} - 1)$. The entropy makes the solution an exponential of edge costs and node potentials, which matrix-scaling iterations compute efficiently (Shi et al., 2025; Cuturi, 2013). Our projection instantiates this template on the transition graph of the CMDP: nodes are states, edges are observed transitions, and the flow is the discounted transition flow of (4), with actions entering through the advantage and cost estimates attached to each edge. Four substitutions complete the mapping: the edge cost D becomes the negative edge advantage $-A$, flow conservation becomes the discounted conservation of (4) with supply $(1-\gamma)\boldsymbol{\mu}_0$, the capacities $F \leq S$ become one linear budget $\langle C_k, F\rangle \leq b_k$ per constraint as in (3), and the entropy is taken relative to the observed flow $\hat{F}$ through $\mathrm{KL}(F\|\hat{F}) = \sum_{ij}[F_{ij}\log(F_{ij}/\hat{F}_{ij}) - F_{ij} + \hat{F}_{ij}]$.

3 OPTIMAL-FLOW GEOMETRY OF SAFE RL

A safe policy update is an allocation problem. The update moves visitation toward transitions with high advantage, and each such move spends the cost budgets, often several of them at once. For each constraint $J_{c_k}(\pi) \leq b_k$ (Section 2), the *budget* b_k bounds the expected cost, and the *price* $\lambda_k \geq 0$ is the reward advantage that the update gives up per unit of this cost. Our central idea is to price each budget by how scarce it becomes under the visitation shift that the update proposes. Optimal flow makes this shift explicit and returns all prices jointly as its dual variables. Penalty-based methods instead fix the prices in advance, and constraint-based methods adapt them from observed costs or from a local model of each constraint. Section 3.1 writes the shift as a flow problem, Section 3.2 derives the prices from its dual, and Section 3.3 revisits existing safe RL methods through them.

3.1 Allocating Flow under Budgets

We aggregate the visited states into N clusters with k-means, a standard form of state aggregation (Li et al., 2006), and build the empirical transition graph over these clusters: a directed edge (i, j) connects cluster i to cluster j whenever the rollouts contain a transition between them. This is the transition graph of Section 2, with states replaced by their clusters. Each edge records (i) the observed flow $\hat{F}_{ij}$, the normalized frequency of its transitions, (ii) the mean reward advantage A_{ij}, and (iii) the mean cost $C_k(i, j)$ for each soft constraint $k \in \mathcal{K}_\mathrm{s}$, which estimates C_k^π of Section 2. Edges whose transitions violate a hard constraint form the excluded set H. We seek a target flow that increases the advantage while staying close to the observed flow, obeys flow conservation and every budget, and gives zero flow to H.

Definition 1 (Entropic flow projection)**.** For $\varepsilon > 0$, the target flow is

$$\begin{aligned} F_\varepsilon^\star = \underset{F \geq 0,\ F_H = 0}{\arg\min} \quad & -\langle A, F\rangle + \varepsilon\, \mathrm{KL}(F \,\|\, \hat{F}) \\ \text{s.t.} \quad & F\mathbf{1} - \gamma F^\top \mathbf{1} = (1-\gamma)\boldsymbol{\mu}_0, \qquad \langle C_k, F\rangle \leq b_k \ \ (k \in \mathcal{K}_\mathrm{s}). \end{aligned} \tag{6}$$

The objective trades advantage against departure from the observed flow, and the entropic regularization ε sets the strength of this trust region. Flow conservation keeps F a valid discounted flow (4), and each budget bounds the expected cost of its constraint over the whole graph (3). Because a budget limits the total cost rather than the cost of each edge, the projection itself decides how to allocate visitation within it.

3.2 Safety Prices from the Flow Dual

The solution of (6) is a target flow on this graph, not a policy update, and a flow need not be realizable by any policy (Section 4.3). Its prices, however, can weight the policy update directly. Proposition 1 identifies them as the Lagrange multipliers of the flow projection.

Proposition 1 (Dual form, proved in Appendix B)**.** *Suppose some flow F° satisfies flow conservation, meets every budget strictly, is positive on every edge with $\hat{F}_{ij} > 0$ outside H, and is zero on all other edges. Then* (6) *has a unique solution $F^\star$, and there exist node potentials $u \in \mathbb{R}^N$ and multipliers $\lambda^\star \in \mathbb{R}_+^{|\mathcal{K}_\mathrm{s}|}$ such that, on every edge with $\hat{F}_{ij} > 0$ outside H,*

$$F_{ij}^\star = \hat{F}_{ij} \exp\Big(\tfrac{1}{\varepsilon}\big(A_{ij} - \textstyle\sum_{k \in \mathcal{K}_\mathrm{s}} \lambda_k^\star C_k(i,j) + u_i - \gamma u_j\big)\Big), \tag{7}$$

and $\lambda_k^\star > 0$ only if budget k is active, that is, $\langle C_k, F^\star\rangle = b_k$.

Definition 2 (Safety price)**.** The *safety price* of budget k is its multiplier $\lambda_k^\star \geq 0$ in Proposition 1, the rate at which the flow projection trades reward advantage against cost k.

Safety prices are shadow prices in the classical sense: loosening b_k by a small δ improves the optimal value of (6) by about $\lambda_k^\star \delta$. What is new is where they come from: they are solved jointly for all budgets from the visitation shift that the update proposes, so they measure how scarce each budget is under that shift. By (7), the target reweights every edge by the exponential of $A_{ij} - \sum_k \lambda_k^\star C_k(i,j)$, its reward advantage after each cost is charged at its safety price, and the potentials u only enforce flow conservation. Penalized policy updates charge costs in the same way, but with prices set by hand or by past violations.

3.3 Safe RL Through the Lens of Prices

Table 1 compares safe RL methods by the source of their prices. Fixed penalties use constants, Lagrangian and PID Lagrangian methods use past costs, and CPO and PCPO compute a local price at each step from a linear model of each constraint (Appendix A.2). SafeOT combines the safety price with observed-cost feedback. Its combined price for budget k is $\lambda_k = \bar{\lambda}_k^\star + \beta_k$ (Section 4.2), where $\bar{\lambda}_k^\star$ is the safety price of Definition 2, clipped at a level $c \geq 0$ and smoothed over iterations, and the feedback term β_k integrates the observed budget error. With $c = 0$ only the feedback term remains, which gives a fixed penalty or a Lagrangian method, depending on how β_k is updated.

The difference matters most while the current policy still meets its budgets. Suppose the transitions with the highest advantage are also the most costly, so that following the advantage alone would

Table 1: Safe RL methods grouped by how they set the price of a budget. FOCOPS and CUP update a Lagrangian price inside a trust-region step, so they share the Lagrangian row.

Price rule	Past-cost feedback	Solved for next step	Examples
Fixed penalty	none	none	P3O (Zhang et al., 2022)
Lagrangian	integral	none	PPO-Lag, TRPO-Lag, FOCOPS, CUP
PID Lagrangian	PID	none	CPPO-PID, TRPO-PID
Trust region	none	linearized constraint	CPO, PCPO
SafeOT	**integral**	**flow projection**	**Ours**

overspend budget k. The projection can then make this budget active and give it a positive safety price before any violation is observed. The feedback term still sees a cost within budget, and with a fixed step it cannot grow until violations have accumulated (Lemma 1).

4 THE SAFEOT ALGORITHM

SafeOT retains the policy update of a Lagrangian method such as TRPO-Lag and changes only how its Lagrange multipliers are computed. The multiplier of each budget, its combined price, adds the safety price of the flow projection to the usual feedback on observed violations, so it responds to how scarce the budget becomes under the proposed update, not only to violations that have already occurred. Algorithm 1 (Appendix A) shows one training iteration in three stages (Figure 2), which Sections 4.1–4.3 explain. Algorithm 2 in Appendix A.6 gives the detailed version with every hyperparameter.

4.1 STAGE 1: BUILDING THE TRANSITION GRAPH

Stage 1 builds the empirical transition graph of Section 3.1 from one batch of rollouts (lines 2–3 of Algorithm 1), with advantages from generalized advantage estimation (GAE) and the per-step budget $b_k = B_k/L$, where B_k is the episodic budget of constraint k and L is the mean episode length (Appendix A.7 gives the details). It also collects the excluded edges H of any hard constraints (line 3). Because the graph has only N nodes, the projection is cheap to solve through its concave dual (Appendix C.1 describes the solver). With $N = 128$, building the graph and solving the projection take 0.12 s per iteration, 0.18% of the training time (Table 5 in Appendix A.8).

4.2 STAGE 2: PRICING EVERY BUDGET

The combined price of each budget has two complementary parts (line 6 of Algorithm 1).

Anticipatory safety price. Solving the projection (6) gives one safety price $\lambda_k^\star$ per budget, the multiplier of Definition 2. It can turn positive before any violation is observed (Section 3.3). SafeOT clips it at level c and smooths it with weight ξ (line 4 of Algorithm 1), which limits the influence of a poorly estimated graph. With $c = 0$, SafeOT reduces to the base Lagrangian learner.

Reactive feedback term. The feedback term follows the multiplier update of a standard Lagrangian method (line 5 of Algorithm 1), so it grows with the observed budget violation. Our benchmark code uses Adam instead of a fixed step (Appendix A.7).

Complementary roles. The feedback term rises only after violations have occurred. Lemma 1 bounds its growth under a fixed step, for one budget with the index k dropped and the budget error $e_t = \bar{J}(t) - B$ at iteration t.

Lemma 1 (Feedback lag)**.** *For a step $\alpha > 0$ and $\beta_0 \geq 0$, the update $\beta_t = [\beta_{t-1} + \alpha e_t]_+$ satisfies*

$$\beta_T \leq \beta_0 + \alpha V_T^e, \qquad V_T^e = \textstyle\sum_{t=1}^{T} [e_t]_+,$$

where V_T^e is the violation accumulated up to iteration T. Hence β_T reaches a level $\ell > \beta_0$ only after $V_T^e \geq (\ell - \beta_0)/\alpha$.

The safety price does not wait for violations, but it sees only the target flow, which the policy may not follow (Section 4.3). Each part thus covers what the other misses.

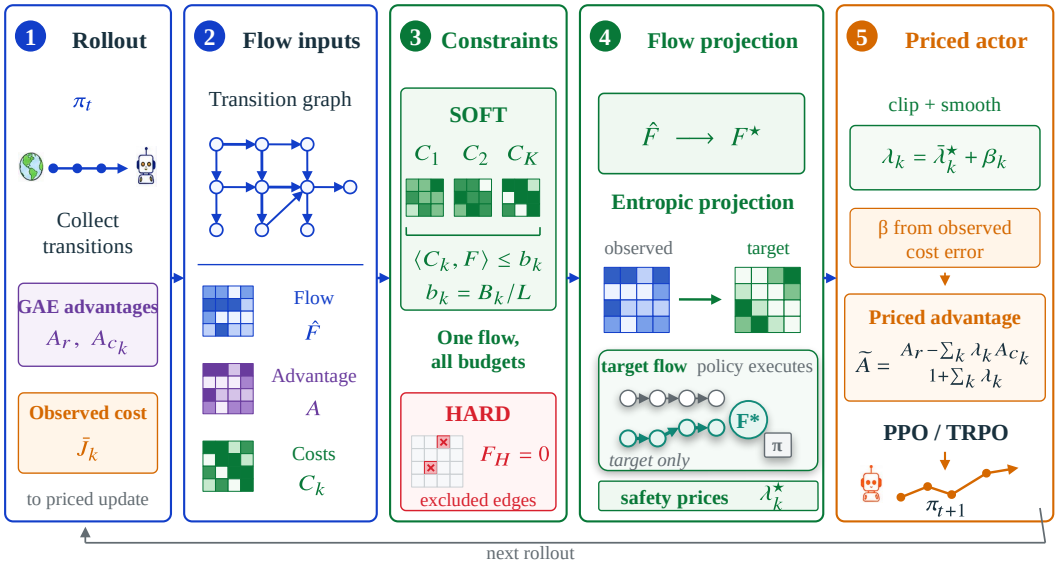

Figure 2: **One SafeOT iteration** (Algorithm 1). Stage 1 builds the transition graph (panels 1–3), Stage 2 prices every budget (panel 4 and the top of panel 5), and Stage 3 takes the priced policy step (bottom of panel 5).

4.3 Stage 3: Taking the Priced Policy Step

Stage 3 forms the priced advantage of TRPO-Lag with the combined prices as multipliers and takes an unmodified PPO or TRPO step on it (line 7 of Algorithm 1). A larger price λ_k thus shifts weight from reward toward avoiding cost k.

The updated policy need not realize the target flow $F^\star$, because a flow can split mass at a node in proportions that no policy produces, and one trust-region step moves the policy only part of the way. Lemma 2 in Appendix B.5 bounds the resulting budget violation by three terms: the target residual, the flow-to-policy gap and the cost-estimation error. The projection controls only the first term. The other two show up only in the observed cost, which the feedback term tracks, so graph feasibility alone cannot certify policy safety (Appendix F measures all three terms during training).

Hard constraints. No finite price can enforce a constraint that must never be violated, because finite prices leave positive target flow on every observed edge that is not excluded (Corollary 1 in Appendix A). Algorithm 1 therefore excludes such edges (line 3). Exclusion alone left our on-policy learners unsafe or with little return (Appendix I). Our hard-constraint experiments (Section 5.3 and Appendix I.2) thus use the combined price only for the soft budget and handle the hard constraint with a separate soft actor-critic learner (Appendix A.9), which uses the feasibility critic of feasible dual policy iteration (FDPI) (Yang et al., 2026).

5 Experiments

We ask whether the safety price lowers the constraint violation accumulated during training without lowering return (Section 5.2) and whether it extends to multiple constraints (Section 5.3). Appendix H.5 ablates the price rule, and Appendix D.4 shows when the reduction occurs.

5.1 Experimental Setup

Benchmarks. We evaluate on ten Safety-Gymnasium tasks under the SafePO protocol (Ji et al., 2023), six level-1 navigation and four velocity tasks with cost budget $B = 25$, and on six Bullet-Safety-Gym tasks (Gronauer, 2022) with $B = 10$, which we run without retuning to test transfer. Appendix C lists all tasks and settings, and Appendix D.7 adds six harder Safety-Gymnasium tasks.

Baselines. We compare with eight SafePO baselines. The Lagrangian methods PPO-Lag and TRPO-Lag (Ray et al., 2019) raise a Lagrange multiplier while the observed cost exceeds the budget, and the PID Lagrangian methods CPPO-PID and TRPO-PID (Stooke et al., 2020) set it by PID feedback on the cost. The constrained policy optimization methods CPO (Achiam et al., 2017), PCPO (Yang et al., 2020), FOCOPS (Zhang et al., 2020) and CUP (Yang et al., 2022) solve a constrained problem at each policy update.

Metrics. As in Safety Gym (Ray et al., 2019), safe exploration (Thananjeyan et al., 2021; Bharadhwaj et al., 2021) and online constrained RL (Efroni et al., 2020; Ding et al., 2021; Müller et al., 2024), we measure safety during training by the cumulative constraint violation

$$\mathrm{CV} = \frac{1}{B} \sum_{t=1}^{T} \left[\bar{J}(t) - B \right]_+,$$

the area of the training-cost curve $\bar{J}(t)$ above the budget over the T training epochs. We also report the return and cost of the final policy. Because a method can lower CV by giving up return, we also compare the methods on both objectives at once, by Pareto dominance and by the hypervolume indicator, the standard measures for comparing trade-offs in multi-objective optimization (Zitzler & Thiele, 1999; Zitzler et al., 2003) and multi-objective RL (Xu et al., 2020; Hayes et al., 2022). Beats/beaten counts the task–method pairs in which a method is better than another in both CV and final return, or worse in both. On each task, we divide CV by the largest CV and rescale the final return to $[0, 1]$ over the nine methods. The hypervolume of a method is then the area of the unit square that its point dominates, which is 1 only at zero CV and the highest return. Across tasks, we report medians of per-task ratios (Agarwal et al., 2021) and means of per-task ranks (Demšar, 2006). Appendix D.3 gives the computation of every metric and the per-task values, and Appendix D.5 reports the standard Safety Gym metrics.

Implementation. SafeOT differs from the TRPO-Lag script only by the safety price, and on Bullet-Safety-Gym also by the cost input of the observed-cost feedback (Appendix C.3), so TRPO-Lag is the controlled comparison. It uses one configuration on all tasks ($\varepsilon = 1$, $c = 1$, $N = 128$), selected on three development tasks. All methods share the SafePO training loop, training length and five seeds, with default baseline hyperparameters (Appendices C and D.8).

5.2 Main results

The safety price lowers violation on every benchmark task. Relative to TRPO-Lag, the controlled comparison, SafeOT lowers CV on all 16 tasks, by 63% on Safety-Gymnasium and 70% on Bullet-Safety-Gym (sign test). The reduction holds on five further seeds and at each of four budgets, the standard training cost rate is lower on 15 of 16 tasks (Table 18, Appendices E.2 and D.5), and the difference in final return is not significant (Appendix E.1).

SafeOT gives the best balance of violation and return. On Safety-Gymnasium, its hypervolume is 0.50, against at most 0.39 for any baseline, and it stays the largest in 99% of seed resamples and on five further seeds (Table 18). Pooling all ten seeds of the nine methods, its hypervolume is 0.49 against at most 0.44, the largest in 90% of resamples, and its mean return rank is the best. It beats another method in both violation and return in 30 task–method pairs and is beaten in 2, and it is Pareto-optimal on nine of the ten tasks, more than any baseline (Table 2). In mean CV and return ranks, it lies on the Pareto front of the nine methods on both benchmarks (Figure 3a, b), and on Safety-Gymnasium in 99.9% of seed resamples and with all ten seeds pooled.

Among the low-violation methods, SafeOT earns the highest return. The PID Lagrangian methods, CPO and PCPO have a similar or lower CV but rank far lower in return, and SafeOT earns the higher final return against each of them on eight to ten of the ten tasks (Figure 4a, Table 10). At the violation of each of these methods, and at the final cost of CPO, CPPO-PID and PCPO, SafeOT also reaches their return on seven to nine of the ten tasks (Appendix E.2).

One price rule spans the trade-off. Figure 4b varies the entropic regularization ε and the clip level c of SafeOT. With $c = 0$, SafeOT is TRPO-Lag, and larger clip levels move it toward and past CPO. A conservative setting, unclipped ($c = \infty$) with a halved feedback step and selected on the development tasks by a pre-registered rule (Appendix D.6), violates less than CPO on nine of the ten tasks and also earns a higher final return on seven. On five further seeds, it is Pareto-optimal on all ten tasks, whereas the default setting is on six.

Table 2: Results on ten Safety-Gymnasium and six Bullet-Safety-Gym tasks (five seeds). Joint columns score CV and final return together (Section 5.1): mean hypervolume over tasks, task–method pairs won or lost in both, and tasks on which a method is Pareto-optimal. CV: median over tasks of CV relative to SafeOT. Return rank: mean rank of the final return. Bold: best in a column, ties included. Per-task values: Tables 10 and 12.

	Safety-Gymnasium (10 tasks)					Bullet-Safety-Gym (6 tasks)				
	Joint			Separate		Joint			Separate	
Method	Hyper-volume	Beats/ beaten	Pareto	CV	Return rank	Hyper-volume	Beats/ beaten	Pareto	CV	Return rank
SafeOT	**0.50**	**30/2**	**9/10**	1×	**3.5**	**0.28**	10/6	3/6	1×	4.8
TRPO-Lag	0.13	7/23	5/10	2.8×	3.9	0.07	1/12	3/6	3.3×	3.5
PPO-Lag	0.02	1/20	2/10	2.7×	3.9	0.06	4/8	3/6	2.8×	2.8
CPPO-PID	0.25	14/15	3/10	0.99×	6.4	0.20	12/5	**4/6**	0.85×	5.2
TRPO-PID	0.39	19/5	5/10	0.86×	5.5	0.25	12/4	**4/6**	0.74×	6.0
CPO	0.39	17/5	5/10	0.82×	5.2	0.17	**13/2**	**4/6**	0.67×	6.2
PCPO	0.21	10/4	8/10	**0.58×**	7.4	0.02	4/10	3/6	**0.56×**	8.2
FOCOPS	0.21	8/22	4/10	2.2×	5.0	0.11	2/20	1/6	2.3×	5.8
CUP	0.20	9/19	3/10	2.0×	4.3	**0.28**	11/2	**4/6**	1.3×	**2.5**

The advantage transfers with a smaller margin. On Bullet-Safety-Gym, without retuning, SafeOT lies between the Lagrangian methods and CUP, which reach a higher return at a higher CV, and the PID Lagrangian methods, CPO and PCPO, which violate less at a lower return. Its hypervolume ties that of CUP (0.28), and it again lies on the Pareto front of the mean ranks (Figure 3b).

5.3 Multiple constraints

Table 3: Multiple constraints on MuJoCo robots with one speed budget per joint (K constraints, five seeds): cost above the budgets, summed over joints and training epochs ($\times 10^3$), and satisfied constraints (of K) and return of the final policy. Lag.: Lagrangian, PID: PID Lagrangian, SafeOT: unclipped. Bold: best mean. Underlined: not significantly different from the best by Welch's t-test (Welch, 1947) at $p \geq 0.05$. Best or tied: robots on which a method is bold or underlined.

		Excess cost ↓			Satisfied constraints ↑			Final return ↑		
Robot	K	Lag.	PID	SafeOT	Lag.	PID	SafeOT	Lag.	PID	SafeOT
HalfCheetah	6	26.8	16.4	**14.3**	4.0	5.0	**5.8**	**1842**	1236	1715
Ant	8	11.8	5.3	**0.8**	6.2	6.0	**6.8**	**2639**	2032	2415
Hopper	3	6.3	**1.3**	1.5	**2.6**	2.2	2.2	458	839	**881**
Walker2d	6	1.7	1.1	**0.1**	5.0	**5.2**	**5.2**	**1276**	800	708
Best or tied		0/4	2/4	**4/4**	3/4	**4/4**	**4/4**	3/4	2/4	**4/4**

With several constraints, SafeOT prices all of them in one flow projection, whereas the Lagrangian and PID Lagrangian methods update one multiplier per constraint. Table 3 gives every joint of four MuJoCo robots its own speed budget, so that the policy must satisfy three to eight constraints at once. SafeOT accumulates significantly less excess cost than the Lagrangian method on all four robots and than the PID Lagrangian method on HalfCheetah and Ant, up to 16 times less (Welch's t-test, $p < 0.05$). Neither the Lagrangian nor the PID Lagrangian method ends with significantly more satisfied constraints or a significantly higher final return than SafeOT on any robot, so SafeOT is the only method that is best or tied on every robot in all three measures. On exact tabular CMDPs with one, four or eight budgets and on learned grid worlds with four or eight, SafeOT violates less than both the Lagrangian and the PID Lagrangian methods on every instance (Table 31).

Hard and soft constraints together. On the Goal2 tasks, SafeOT prices a soft hazard budget and leaves contacts, which must never occur, to a learned executor (Section 4.3). It then has a lower contact rate and a higher return than SDAC (Kim et al., 2023) and SRCPO (Kim et al., 2024) under their own shields, Sauté RL (Sootla et al., 2022) and the Safety Layer (Dalal et al., 2018) on all three tasks (Appendix I.2). ET-MDP (Sun et al., 2021) has a lower contact rate on RacecarGoal2 but a return at or below zero, whereas SafeOT earns the highest return of these methods on all three.

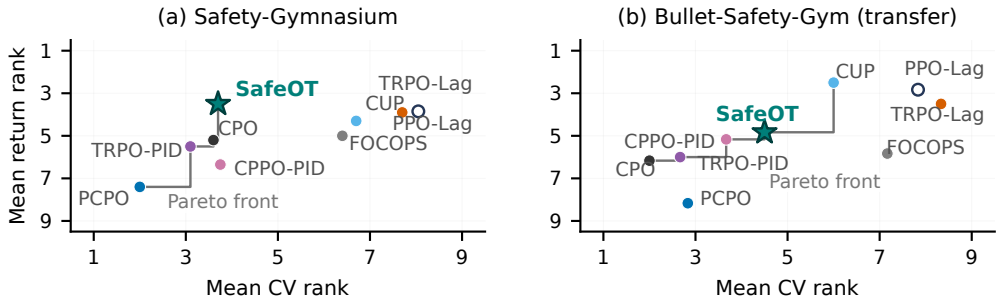

Figure 3: Mean ranks in cumulative constraint violation (CV) and final return (lower is better) on (a) Safety-Gymnasium and (b) Bullet-Safety-Gym. Gray: Pareto front of the nine methods, which includes SafeOT on both.

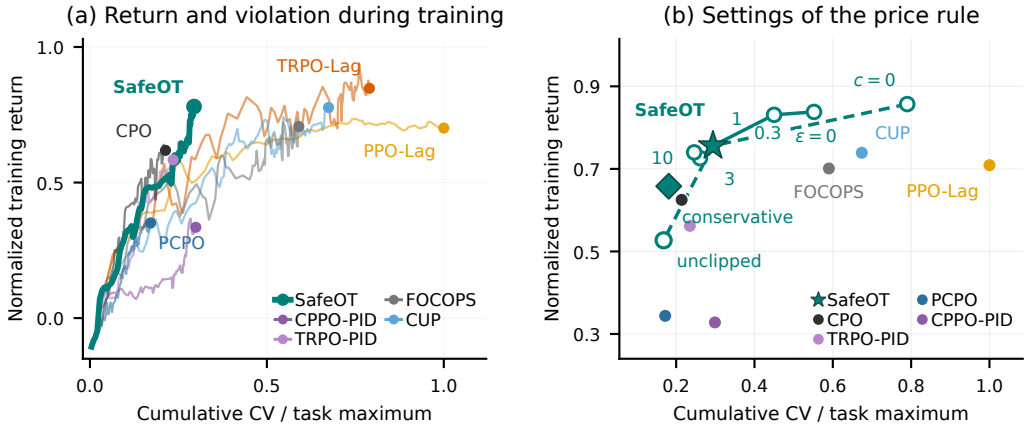

Figure 4: Trade-offs on the ten Safety-Gymnasium tasks (medians over tasks, five seeds, CV and training return divided by their largest final values). (a) Training paths of the methods. (b) Settings of the price rule: ε (solid) and clip level c (dashed) around the default SafeOT (star), where $c = 0$ is TRPO-Lag, and a conservative setting (diamond, Appendix D.6).

6 Conclusion

A Lagrange multiplier prices a budget from violations that have already happened, whereas SafeOT prices it from the flow that the next update is predicted to take. This gives one price view of safe RL. Lagrangian methods are its special case, and its settings tune how strongly return is traded for safety, from the Lagrangian update to settings more conservative than CPO. At both ends it improves on the base methods. It lowers the violation during training relative to the Lagrangian update without a significant loss in return, and its conservative setting violates less than CPO while earning more return on most tasks. Overall, SafeOT gives the best balance of violation and return, and one projection prices many constraints at once.

Limitations and outlook. The price anticipates violation only as far as the transition graph predicts the next flow, so SafeOT gives no per-seed guarantee for the final policy. It needs only a model of the next update, so we plan to extend it to world models and to train vision-language-action policies (Appendix I.6) with it, compared with SafeVLA (Zhang et al., 2025a).

AI USE STATEMENT

LLM-based assistants were used under human direction for mathematical formulation and proof drafting, method and experiment implementation, orchestration and analysis, result interpretation, text revision, figure preparation, literature search and reference formatting. The authors selected the research goals and the framing of this revision. AI-generated prose is not experimental evidence: the empirical claims refer to archived runs and their analysis artifacts. Final responsibility for the manuscript rests with the human authors.

ETHICS STATEMENT

This work studies algorithms for reinforcement learning under explicit safety budgets on standard simulated benchmarks (Safety-Gymnasium). It involves no human subjects or sensitive data. The intended positive impact is a policy update that makes safety costs explicit and can reduce violations during learning, which matters wherever exploration itself has a cost. The same expected-cost machinery can be misused if an expected-cost budget is presented as a per-episode guarantee. We therefore report irreversible-event channels as per-episode leak rates with confidence bounds, report the cases in which the method does not win, and make no claim about real-robot deployment, which requires domain-specific hazard analysis, independent validation and runtime monitoring beyond the scope of this paper.

REPRODUCIBILITY STATEMENT

The algorithm is specified in Section 4 and Algorithm 1, with implementation details in Appendix C.1 and its proofs in Appendix B. Tasks, channel definitions, budgets, protocol, hyper-parameters and metrics are in Appendix C, and the means of the SafePO runs (benchmark, controls, sweeps and 10^7 runs) are backed by the per-seed values in the supplementary data package (coverage: Appendix D). The reference baselines and matched controller variants follow the settings specified in Appendix B. The off-policy executor is the unmodified FDPI trainer with the wrapper described in Appendix C. The accompanying anonymous source package contains the training and evaluation code, frozen evidence exports, and analysis scripts for the reported tables. Appendix G.1 specifies the controlled CMDP study, its selection protocol and recording limits.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 22–31, 2017.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29304–29320, 2021.

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 2669–2678, 2018. doi: 10.1609/aaai.v32i1.11797.

Eitan Altman. *Constrained Markov Decision Processes*. Chapman & Hall/CRC, 1999.

Homanga Bharadhwaj, Aviral Kumar, Nicholas Rhinehart, Sergey Levine, Florian Shkurti, and Animesh Garg. Conservative safety critics for exploration. In *International Conference on Learning Representations*, 2021.

Richard Cheng, Gábor Orosz, Richard M. Murray, and Joel W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3387–3395, 2019. doi: 10.1609/aaai.v33i01.33013387.

Roberto Cominetti and Jaime San Martín. Asymptotic analysis of the exponential penalty trajectory in linear programming. *Mathematical Programming*, 67:169–187, 1994. doi: 10.1007/BF01582220.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pp. 2292–2300, 2013.

Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. arXiv preprint arXiv:1801.08757, 2018.

Valentin De Bortoli, Iryna Korshunova, Andriy Mnih, and Arnaud Doucet. Schrödinger bridge flow for unpaired data translation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 103384–103441, 2024. doi: 10.52202/079017-3285.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30, 2006.

F. d'Epenoux. A probabilistic production and inventory problem. *Management Science*, 10(1):98–108, 1963. doi: 10.1287/mnsc.10.1.98.

Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, Zhaoran Wang, and Mihailo R. Jovanović. Provably efficient safe exploration via primal-dual policy optimization. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3304–3312, 2021.

Yonathan Efroni, Shie Mannor, and Matteo Pirotta. Exploration-exploitation in constrained MDPs. arXiv preprint arXiv:2003.02189, 2020.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2160–2169, 2019.

Arash Givchi, Pei Wang, Junqi Wang, and Patrick Shafto. Distributionally-constrained policy optimization via unbalanced optimal transport. arXiv preprint arXiv:2102.07889, 2021.

Sven Gronauer. Bullet-Safety-Gym: A framework for constrained reinforcement learning. Technical report, Technical University of Munich, 2022. URL `https://doi.org/10.14459/2022md1639974`.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870, 2018.

Conor F. Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M. Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1):26, 2022. doi: 10.1007/s10458-022-09552-y.

Weidong Huang, Jiaming Ji, Chunhe Xia, Borong Zhang, and Yaodong Yang. SafeDreamer: Safe reinforcement learning with world models. In *International Conference on Learning Representations*, pp. 53839–53869, 2024.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Juntao Dai, and Yaodong Yang. Safety-Gymnasium: A unified safe reinforcement learning benchmark. In *Advances in Neural Information Processing Systems*, volume 36, pp. 18964–18993, 2023. doi: 10.52202/075280-0831.

Dohyeong Kim, Kyungjae Lee, and Songhwai Oh. Trust region-based safe distributional reinforcement learning for multiple constraints. In *Advances in Neural Information Processing Systems*, volume 36, pp. 19908–19939, 2023. doi: 10.52202/075280-0874.

Dohyeong Kim, Taehyun Cho, Seungyub Han, Hojun Chung, Kyungjae Lee, and Songhwai Oh. Spectral-risk safe reinforcement learning with convergence guarantees. In *Advances in Neural Information Processing Systems*, volume 37, pp. 92465–92498, 2024. doi: 10.52202/079017-2936.

Jongmin Lee, Wonseok Jeon, Byung-Jun Lee, Joelle Pineau, and Kee-Eung Kim. OptiDICE: Offline policy optimization via stationary distribution correction estimation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6120–6130, 2021.

Jongmin Lee, Cosmin Paduraru, Daniel J. Mankowitz, Nicolas Heess, Doina Precup, Kee-Eung Kim, and Arthur Guez. COptiDICE: Offline constrained reinforcement learning via stationary distribution correction estimation. In *International Conference on Learning Representations*, 2022.

Lihong Li, Thomas J. Walsh, and Michael L. Littman. Towards a unified theory of state abstraction for MDPs. In *Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics*, 2006. URL http://anytime.cs.umass.edu/aimath06/proceedings/P21.pdf.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Zuxin Liu, Zhepeng Cen, Vladislav Isenbaev, Wei Liu, Zhiwei Steven Wu, Bo Li, and Ding Zhao. Constrained variational policy optimization for safe reinforcement learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13644–13668, 2022.

Alan S. Manne. Linear programming and sequential decisions. *Management Science*, 6(3):259–267, 1960. doi: 10.1287/mnsc.6.3.259.

Adrian Müller, Pragnya Alatur, Volkan Cevher, Giorgia Ramponi, and Niao He. Truly no-regret learning in constrained MDPs. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36605–36653, 2024.

Gergely Neu, Vicenç Gómez, and Anders Jonsson. A unified view of entropy-regularized Markov decision processes. arXiv preprint arXiv:1705.07798, 2017.

Andrew Y. Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pp. 278–287, 1999.

OmniSafe Team. On-policy algorithms. OmniSafe documentation, 2024. URL https://omnisafe.readthedocs.io/en/latest/benchmark/on-policy.html. Table 2; accessed September 25, 2026.

David Pfau, Ian Davies, Diana Borsa, João G. M. Araújo, Brendan Tracey, and Hado van Hasselt. Wasserstein policy optimization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 49128–49149, 2025.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994. doi: 10.1002/9780470316887.

James Queeney, Erhan Can Ozcan, Ioannis Ch. Paschalidis, and Christos G. Cassandras. Optimal transport perturbations for safe reinforcement learning with robustness guarantees. *Transactions on Machine Learning Research*, 2024.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. OpenAI, 2019. URL https://cdn.openai.com/safexp-short.pdf.

Zahra Shahrooei and Ali Baheri. Optimal transport-guided safety in temporal difference reinforcement learning. arXiv preprint arXiv:2502.16328, 2025.

Liangliang Shi, Yufeng Li, Kaipeng Zeng, Yihui Tu, and Junchi Yan. Optimal flow transport and its entropic regularization: a GPU-friendly matrix iterative algorithm for flow balance satisfaction. In *International Conference on Learning Representations*, pp. 31304–31322, 2025.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion Schrödinger bridge matching. In *Advances in Neural Information Processing Systems*, volume 36, pp. 62183–62223, 2023. doi: 10.52202/075280-2717.

Aivar Sootla, Alexander I. Cowen-Rivers, Taher Jafferjee, Ziyan Wang, David Mguni, Jun Wang, and Haitham Bou-Ammar. Sauté RL: Almost surely safe reinforcement learning using state augmentation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 20423–20443, 2022.

Lindsay Spoor, Álvaro Serra-Gómez, Aske Plaat, and Thomas Moerland. Towards a practical understanding of Lagrangian methods in safe reinforcement learning. arXiv preprint arXiv:2510.17564, 2026.

Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by PID Lagrangian methods. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9133–9143, 2020.

Hao Sun, Ziping Xu, Meng Fang, Zhenghao Peng, Jiadong Guo, Bo Dai, and Bolei Zhou. Safe exploration by solving early terminated MDP. arXiv preprint arXiv:2107.04200, 2021.

Ke Sun, Yingnan Zhao, Wulong Liu, Bei Jiang, and Linglong Kong. Distributional reinforcement learning with regularized Wasserstein loss. In *Advances in Neural Information Processing Systems*, volume 37, pp. 63184–63221, 2024. doi: 10.52202/079017-2018.

Chen Tessler, Daniel J. Mankowitz, and Shie Mannor. Reward constrained policy optimization. In *International Conference on Learning Representations*, 2019.

Brijen Thananjeyan, Ashwin Balakrishna, Suraj Nair, Michael Luo, Krishnan Srinivasan, Minho Hwang, Joseph E. Gonzalez, Julian Ibarz, Chelsea Finn, and Ken Goldberg. Recovery RL: Safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters*, 6(3):4915–4922, 2021. doi: 10.1109/LRA.2021.3070252.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

B. L. Welch. The generalization of 'Student's' problem when several different population variances are involved. *Biometrika*, 34(1–2):28–35, 1947. doi: 10.1093/biomet/34.1-2.28.

Jie Xu, Yunsheng Tian, Pingchuan Ma, Daniela Rus, Shinjiro Sueda, and Wojciech Matusik. Prediction-guided multi-objective reinforcement learning for continuous robot control. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10607–10616, 2020.

Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. In *Advances in Neural Information Processing Systems*, volume 35, pp. 9111–9124, 2022. doi: 10.52202/068431-0662.

Tsung-Yen Yang, Justinian Rosca, Karthik Narasimhan, and Peter J. Ramadge. Projection-based constrained policy optimization. In *International Conference on Learning Representations*, 2020.

Yujie Yang, Jinglin Teh, Ziyu Lin, Kaicheng Yu, Tao Zhang, and Shengbo Eben Li. Breaking safety paradox with feasible dual policy iteration. In *International Conference on Learning Representations*, pp. 85030–85053, 2026.

Borong Zhang, Yuhao Zhang, Jiaming Ji, Yingshan Lei, Josef Dai, Yuanpei Chen, and Yaodong Yang. SafeVLA: Towards safety alignment of vision-language-action model via constrained learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. doi: 10.52202/085713-5128.

Jichen Zhang, Liqun Zhao, Antonis Papachristodoulou, and Jack Umenberger. Constrained diffusers for safe planning and control. In *Advances in Neural Information Processing Systems*, volume 38, 2025b. doi: 10.52202/085713-1174.

Linrui Zhang, Li Shen, Long Yang, Shixiang Chen, Xueqian Wang, Bo Yuan, and Dacheng Tao. Penalized proximal policy optimization for safe reinforcement learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pp. 3744–3750, 2022. doi: 10.24963/ijcai.2022/520.

Mingxu Zhang, Huicheng Zhang, Jiaming Ji, Yaodong Yang, and Ying Sun. Enhance the safety in reinforcement learning by ADRC Lagrangian methods. *Transactions on Machine Learning Research*, 2026a.

Yiming Zhang, Quan Vuong, and Keith W. Ross. First order constrained optimization in policy space. In *Advances in Neural Information Processing Systems*, volume 33, pp. 15338–15349, 2020.

Yuhao Zhang, Borong Zhang, Jiaming Fan, Jiachen Shen, Yishuai Cai, Yaodong Yang, and Jiaming Ji. RedVLA: Physical red teaming for vision-language-action models. arXiv preprint arXiv:2604.22591, 2026b.

Weiye Zhao, Feihan Li, Yifan Sun, Yujie Yang, Rui Chen, Tianhao Wei, and Changliu Liu. Absolute state-wise constrained policy optimization: High-probability state-wise constraints satisfaction. arXiv preprint arXiv:2410.01212, 2024.

Eckart Zitzler and Lothar Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999. doi: 10.1109/4235.797969.

Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003. doi: 10.1109/TEVC.2003.810758.

Appendix

A Method Details and Full Statements **17**

A.1 Notation . . . 17
A.2 Price Rules of the Compared Methods . . . 17
A.3 Limits of the Entropic Regularization . . . 18
A.4 Excluded Edges and Finite Prices . . . 18
A.5 Observed-Cost Feedback . . . 19
A.6 Detailed Algorithm . . . 19
A.7 From Rollout Data to Combined Prices . . . 19
A.8 Time per Iteration . . . 20
A.9 Hard-Constraint Executor . . . 21

B Proofs and Policy Realization **21**

B.1 Proof of Proposition 1 . . . 21
B.2 Excluded Edges and Finite Entropic Prices . . . 22
B.3 Proof of Proposition 2 . . . 23
B.4 Growth of the Feedback Term . . . 24
B.5 From a Target Flow to Policy Cost . . . 25
B.6 Graph Surrogate and Policy Realization . . . 26
B.7 Proof of Proposition 3 . . . 26

C Experimental Details **27**

C.1 Implementation and Evaluation Protocol . . . 27
C.2 Bullet-Safety-Gym Protocol . . . 29
C.3 Bullet Controller Diagnostics . . . 30
C.4 Study Coverage . . . 30

D Benchmark Results and Seed Coverage **31**

D.1 Learning Curves on Every Task . . . 31
D.2 Suite-Level Learning Curves and Median Trade-offs . . . 31
D.3 Per-Task Results . . . 31
D.4 Price Anticipation and Transient Violation . . . 36
D.5 Standard Safety Gym Metrics . . . 39
D.6 Tuning the Lagrangian Baseline . . . 40
D.7 Six Harder Safety-Gymnasium Tasks . . . 43
D.8 Comparison with Published Benchmark Results . . . 43

E Return–Safety Trade-offs and Deployment **44**

E.1 Operating Points at Four Budget Levels 44
E.2 Cost-Matched Comparisons 44
E.3 Final-Policy Feasibility 46

F Empirical Target-to-Policy Decomposition **47**

G Multiple Budgets and Controlled Studies **49**
G.1 Known-Model Policy Updates 49
G.2 Multiple Budgets with Known Dynamics 50
G.3 Learned Grid Worlds with One Budget 51
G.4 Multiple Budgets with Learned Policies 51
G.5 Extension to Hopper and Walker2d 53
G.6 Merging the Robot Budgets 55
G.7 Additional Controls 56
G.8 Single-Budget Controller Comparisons 56

H Hyperparameter Sensitivity and Compute **57**
H.1 Single-Parameter Sensitivity on Ten Tasks 57
H.2 Entropy Sweep on All Ten Tasks 57
H.3 Controller and Solver Settings 58
H.4 Solver Timing 59
H.5 Feedback and Safety-Price Ablations 59

I Hard and Soft Constraints in the Executor **60**
I.1 Fixed Margins and Margin Schedules 60
I.2 Uniform-Schedule Diagnostic 61
I.3 Own-Critic Shield Controls 62
I.4 Random Policy-Sample Control 62
I.5 RacecarGoal2: Native-Contact Evaluation 62
I.6 Exploratory Deployment on a Vision-Language-Action Policy 63

J Related Work **63**

K Numerical Checks of the Flow Projection **64**

A METHOD DETAILS AND FULL STATEMENTS

This section collects the full limit and excluded-edge statements, the feedback qualifications, and the implementation details of Algorithm 1. The proofs remain in Appendix B.

A.1 NOTATION

Table 4 lists the symbols of the main text, the name that the text uses for each, and where each is defined.

Table 4: Notation. Each object has one symbol and one name throughout the paper.

Symbol	Meaning	Defined in
Constrained MDP and flows		
$\mathcal{S}, \mathcal{A}, P, r, \gamma, \boldsymbol{\mu}_0$	states, actions, transition kernel, reward, discount and initial distribution	Section 2
$c_k,\ J_{c_k}(\pi),\ b_k$	cost of constraint k, its expected cost, and its *budget*	Eq. (3)
ρ^π	normalized discounted occupancy of policy π	Section 2
$F^\pi,\ C_k^\pi$	discounted transition flow of π and its edge-conditional cost	Eq. (4)
$\langle X, Y \rangle$	$\sum_{ij} X_{ij} Y_{ij}$	Section 2
Empirical transition graph and entropic flow projection		
N	number of clusters, the nodes of the empirical transition graph	Section 3.1
$\hat{F}$	*observed flow*, the normalized frequency of transitions between clusters	Section 3.1
$A_{ij},\ C_k(i,j)$	mean reward advantage and mean cost of the transitions on edge (i, j)	Section 3.1
$\mathcal{K}_\mathrm{s},\ H$	set of soft constraints, and set of *excluded edges* that violate a hard constraint	Section 3.1
ε	*entropic regularization* of the projection	Definition 1
$F^\star$	*target flow*, the solution of the projection	Eq. (6)
u	node potentials, which enforce flow conservation	Proposition 1
Prices		
$\lambda_k^\star$	*safety price* of budget k	Definition 2
$\bar{\lambda}_k^\star,\ c,\ \xi$	safety price clipped at level c and smoothed with weight ξ	Algorithm 2, line 7
$\beta_k,\ \alpha$	*feedback term* and its fixed step	Algorithm 2, line 8
λ_k	*combined price* applied by the policy update, $\lambda_k = \bar{\lambda}_k^\star + \beta_k$	Algorithm 2, line 9
$B_k,\ L,\ \bar{J}_k$	episodic budget, mean episode length and observed mean episode cost, with $b_k = B_k / L$	Section 4.1
Policy update		
$A_r,\ A_{c_k}$	GAE reward and cost advantages of a rollout sample	Algorithm 2, line 3
$\widetilde{A}$	*priced advantage* used by the actor step	Algorithm 2, line 10

A.2 PRICE RULES OF THE COMPARED METHODS

Table 1 groups safe RL methods by how they set the price λ that weights the cost advantage in the policy update, $A_r - \lambda A_c$, with one price per budget when there are several. With the observed budget error $e_t = \bar{J}(t) - B$ at iteration t, the rows of Table 1 correspond to the following rules.

- *Fixed penalty* (P3O, Zhang et al., 2022): a constant penalty weight κ on a clipped cost surrogate, so the price is κ while the constraint surrogate is active and 0 otherwise. Past costs do not change it, and nothing is solved for the next step.
- *Lagrangian* (PPO-Lag and TRPO-Lag, Ray et al., 2019): $\lambda_{t+1} = [\lambda_t + \alpha e_t]_+$, the integral of the observed budget error. FOCOPS (Zhang et al., 2020) and CUP (Yang et al., 2022) update their Lagrange multiplier from the observed error in the same way and use it inside a trust-region step.
- *PID Lagrangian* (CPPO-PID and TRPO-PID, Stooke et al., 2020): $\lambda_t = [K_P e_t + K_I I_t + K_D D_t]_+$, a PID controller on the observed error with its integral I_t and derivative D_t.

Algorithm 1 SafeOT: one training iteration

Input: Lagrangian learner (policy π, critics), soft constraints with budgets B_k, hard constraints (if any)

1: **for** each iteration **do**

Stage 1: build the transition graph (Section 4.1)

2: Collect rollouts with π, and compute the advantages A_r, A_{c_k} and the mean episode costs $\bar{J}_k$

3: Build the transition graph over N clusters, with excluded edges H for the hard constraints

Stage 2: price every budget (Section 4.2)

4: $\bar{\lambda}_k^\star \leftarrow$ safety price of the projection (6), clipped at c and smoothed ▷ anticipatory

5: $\beta_k \leftarrow [\beta_k + \alpha\,(\bar{J}_k - B_k)]_+$ ▷ reactive

6: $\lambda_k \leftarrow \bar{\lambda}_k^\star + \beta_k$ ▷ combined price

Stage 3: take the priced policy step (Section 4.3)

7: Take a PPO or TRPO step on $\tilde{A} = \big(A_r - \sum_k \lambda_k A_{c_k}\big)/\big(1 + \sum_k \lambda_k\big)$

8: **end for**

- *Trust region* (CPO, Achiam et al., 2017, and PCPO, Yang et al., 2020): at every update, the cost is linearized around the current policy, $J_c(\theta_t) + g_c^\top(\theta - \theta_t) \le B$, and the step is solved within a KL trust region. CPO's price is the multiplier of this local problem, and PCPO projects the reward step onto the linearized constraint. Neither keeps a memory of past costs.
- *SafeOT*: $\lambda_k = \bar{\lambda}_k^\star + \beta_k$, where $\bar{\lambda}_k^\star$ is the clipped and smoothed safety price that the flow projection solves for the next update (Section 3.2), and β_k is the integral feedback term of the Lagrangian row (Section 4.3).

The benchmark runs use the SafePO implementations of these methods with their default hyperparameters (Appendix D.8).

A.3 Limits of the Entropic Regularization

Proposition 2 (The two endpoints of ε)**.** *Under the assumption of Proposition 1:* (i) *as $\varepsilon \to 0$, $F_\varepsilon^\star$ converges to the solution of the linear program (LP) $\max\{\langle A, F\rangle : F$ feasible in (6)$\}$ of least relative entropy to $\hat{F}$, the multipliers $\lambda_\varepsilon^\star$ stay bounded, and every accumulation point is a dual solution of that LP.* (ii) *As $\varepsilon \to \infty$, $F_\varepsilon^\star \to F^I := \arg\min\{\mathrm{KL}(F\|\hat{F}) : F$ feasible$\}$ and every accumulation point of $\lambda_\varepsilon^\star/\varepsilon$ is a multiplier θ^I of that information projection. If $\hat{F}$ satisfies flow conservation and vanishes on H, a local expansion near $b = \hat{J}$, with a stable active set $\mathcal{K}$ and nonsingular flow-conservation and active residual-covariance matrices, gives*

$$\theta_{\mathcal{K}}^I = \tilde{G}_{\mathcal{K}\mathcal{K}}^{-1}(\hat{J} - b)_{\mathcal{K}} + o(\|\hat{J} - b\|).$$

Here $\hat{J}_k = \langle C_k, \hat{F}\rangle$ and $\tilde{G}$ is the $\hat{F}$-weighted covariance after removing potential differences from the costs (Appendix B.3).

The two limits select an LP-optimal flow and the nearest feasible flow, respectively, with the stated dual and local-expansion conditions. The covariance $\tilde{G}$ couples the budgets. Small ε gives LP multipliers, which need not vanish. A positive limit of $\lambda_k^\star/\varepsilon$ drives $\lambda_k^\star$ to infinity, so the clipped price $\min\{\lambda_k^\star, c\}$ tends to c. A budget that F^I meets strictly has $\lambda_k^\star = 0$ for all large ε. At large ε the clipped prices are therefore c on the budgets with $\theta_k^I > 0$ and zero on the budgets that F^I meets strictly, which are the active-set prices of Figure 1.

With $c = 0$, only the feedback term remains, and freezing it gives a fixed penalty (Table 1). Clipping and smoothing need not preserve projection optimality. Appendix B.3 proves the limits. Appendix K tests them numerically.

A.4 Excluded Edges and Finite Prices

Corollary 1 (Finite entropic prices and excluded edges)**.** *Fix $\varepsilon > 0$ and finite edge advantages and costs. Let H_0 be the current set of excluded edges, $E = \mathrm{supp}(\hat{F}) \setminus H_0$, and $C_h \ge 0$ the mean edge cost of a hard constraint h, with $H_h = \{e \in E : C_h(e) > 0\} \ne \emptyset$. Give h a budget $b_h \ge 0$ and keep the other constraints fixed. Every exponential flow* (7) *with finite potentials and multipliers has*

positive h-cost. Under Proposition 1, if $b_h > 0$ and $\lambda_h^\star > 0$, that cost equals b_h. If the zero-budget problem is feasible, its unique minimizer equals the projection with $H_0 \cup H_h$ removed, and has no finite exponential representation on E.

Finite exponential prices keep positive mass on every supported edge outside H_0, including the edges of H_h. Zero-budget exclusion therefore changes the allowed support.

A.5 Observed-Cost Feedback

Proposition 3 (Limit feasibility of the training cost)**.** *For the feedback term updated with a fixed step $\alpha > 0$ or with Adam at a fixed learning rate, if $\bar{J}_k(t) \to \bar{J}_k^\infty$ and $\beta_k(t) \to \beta_k^\infty < \infty$, then $\bar{J}_k^\infty \le B_k$ for every soft constraint, with equality whenever $\beta_k^\infty > 0$. This holds for any bounded sequence of smoothed safety prices $\bar{\lambda}^\star$.*

If the feedback term and the observed cost converge with a finite feedback term, the limiting training cost meets its budget. Convergence and deployment feasibility are not established. Appendix B.7 proves the conditional statement.

Remark: clipping limits the contribution of the safety price. Under Lemma 1, suppose $\lambda_T = \min\{\Lambda, g_T + \beta_T\}$, where $0 \le g_T \le c$ is the smoothed safety price, possibly scaled by a gain in $[0, 1]$, and $\Lambda \in (0, \infty]$ caps the combined price ($\Lambda = \infty$ in Algorithm 1 and $\Lambda = 100$ in the multiple-budget studies of Appendix G). For any $\ell \ge 0$,

$$\lambda_T \ge \ell \quad \Longrightarrow \quad V_T^e \ge \frac{[\ell - c - \beta_0]_+}{\alpha}. \tag{8}$$

When the safety price and the reactive switch of Table 35 both saturate the clip, their combined prices agree when their smoothing and feedback states agree, and saturation alone does not make them agree. This fixed-step accounting does not transfer unchanged to Adam (Appendix B.4). The bound accounts for price growth and does not define a necessary safety threshold. Table 39 tests the feedback component.

A.6 Detailed Algorithm

Algorithm 2 expands Algorithm 1. It lists the hyperparameters of the price rule, initializes the prices, and separates the solve, the clipping and smoothing, and the feedback update. Table 35 gives the remaining fixed settings.

A.7 From Rollout Data to Combined Prices

The observed flow $\hat{F}$ uses normalized undiscounted, nonterminal transition frequencies on N k-means clusters, while the projection uses $\gamma = 0.99$. The clusters are refit every 10 epochs, and edge statistics from earlier batches decay by 0.5 per epoch (Table 35). These sampled frequencies differ from the normalized discounted flows of Section 2, with a flow-conservation mismatch of about 1.6% of the flow for 1000-step episodes (Appendix K). Graph budgets use $b_k = B_k/L$, with L the observed mean episode length clipped to $[50, 5000]$. Feedback uses the original episodic budget B_k. The edge advantage A_{ij} is the mean standardized reward advantage, and the returned safety price is restored to the units of the input advantage before clipping. Here c is the clip level of Algorithm 2, line 7, and the smoothing is an exponential moving average (EMA),

$$\bar{\lambda}_k^\star(t) = (1 - \xi_t)\bar{\lambda}_k^\star(t-1) + \xi_t \min\{\lambda_k^\star(t), c\}, \qquad \xi_t = 1 \text{ initially, then } \xi.$$

Unlike Algorithm 2, which initializes $\bar{\lambda}_k^\star = 0$, the first update uses $\xi_1 = 1$, so the initial value has no effect. The benchmark implementation updates β with Adam, whereas Algorithm 1 states the fixed-step rule that the analysis uses. Appendix B.4 distinguishes their finite-time behavior. After each solve, a numerical check tests the flow-conservation residual, finite flow and total mass. It does not certify joint budget feasibility or complementary slackness. If the check fails, the previous price is kept. The detailed stopping and fallback rules appear below in Appendix C.1. The default on-policy benchmarks use $H = \emptyset$. Table 35 gives the fixed settings.

Algorithm 2 SafeOT: one training iteration, detailed version of Algorithm 1

Input: Lagrangian learner (policy π, critics, feedback step α, initial feedback β_0)
Constraints: soft constraints with episodic budgets B_k, hard constraints (if any)
Parameters: graph size N, entropic regularization ε, clip level c, smoothing weight ξ

1: Initialize $\bar{\lambda}_k^\star \leftarrow 0$ and $\beta_k \leftarrow \beta_0$ for every budget k
2: **for** each iteration **do**
Stage 1: build the transition graph (Section 4.1)
3: Collect rollouts with π. Compute reward and cost advantages A_r, A_{c_k} by generalized advantage estimation (GAE), mean episode costs $\bar{J}_k$ and mean episode length L
4: Group the visited states into N clusters, estimate $\hat{F}, \boldsymbol{\mu}_0, A, C_k$ (Section 3.1) and set $b_k \leftarrow B_k/L$
5: $H \leftarrow$ edges on which a hard constraint is violated in most transitions ▷ $H = \emptyset$ if none
Stage 2: price every budget (Section 4.2)
6: Solve the projection (6), which gives H zero flow, and let $\lambda^\star$ be its dual solution ▷ safety prices
7: $\bar{\lambda}_k^\star \leftarrow (1-\xi)\,\bar{\lambda}_k^\star + \xi \min\{\lambda_k^\star, c\}$ ▷ clip and smooth
8: $\beta_k \leftarrow [\beta_k + \alpha\,(\bar{J}_k - B_k)]_+$ ▷ feedback term
9: $\lambda_k \leftarrow \bar{\lambda}_k^\star + \beta_k$ ▷ combined price
Stage 3: take the priced policy step (Section 4.3)
10: $\widetilde{A} \leftarrow \big(A_r - \sum_k \lambda_k A_{c_k}\big) \big/ \big(1 + \sum_k \lambda_k\big)$ ▷ priced advantage
11: Update π by a standard PPO or TRPO step on $\widetilde{A}$, and update the critics as usual
12: **end for**

A.8 Time per Iteration

Table 5 splits the training time in the default benchmark configuration: one budget $B = 25$, graph size $N = 128$, entropic regularization $\varepsilon = 1$, clip level $c = 1$ and 2×10^4 environment steps per iteration. It pools the per-iteration times in the training logs of 50 runs of this configuration on the ten Safety-Gymnasium tasks, $7{,}400$ iterations in total. Of these runs, 44 train for 2×10^6 environment steps (100 iterations), and six train for 10^7 steps (500 iterations), which repeat seeds 0–2 on PointGoal1 and CarGoal1. *Pricing* is timed from the assignment of the batch to clusters to the smoothed safety price. It covers the graph construction of line 4 of Algorithm 2, including the periodic refit of the clusters, and lines 6–7. Line 5 is empty in these runs, and lines 8–9 are scalar updates. *Rollouts* covers environment steps, policy evaluation and advantage estimation (line 3), and *Actor and critics* is the rest of the update (lines 8–11). Pricing takes 0.12 s per iteration on average and 0.18% of the training time, and at most 0.38% on any task. Per iteration, its median is 0.10 s, its 95th percentile is 0.26 s, and it never exceeds 3.7% of the iteration. The logs record pricing times to 0.01 s and the other times to three significant digits. The solver runs on the CPU, and absolute times depend on the task and the machine.

Table 5: Training time of SafeOT in the default benchmark configuration (Appendix A.8). Iteration and pricing times are means per iteration. Percentages are shares of the total training time of each task, so the three shares in a row add up to 100% within 0.15 percentage points. The All row pools all $7{,}400$ iterations, including those of the six longer runs. Pricing covers line 4 and lines 6–7 of Algorithm 2: clustering the batch, estimating the edge statistics, solving the projection, and clipping and smoothing the safety price.

Task	Runs	Iteration (s)	Rollouts (%)	Actor and critics (%)	Pricing (s)	Pricing (%)
PointGoal1	8	66.1	85.0	14.8	0.11	0.16
CarGoal1	8	82.2	87.7	12.2	0.12	0.14
PointButton1	5	112.9	90.6	9.2	0.15	0.14
CarButton1	3	113.9	91.7	8.1	0.19	0.17
PointPush1	3	84.5	76.2	23.5	0.20	0.24
CarPush1	3	94.3	80.5	19.3	0.20	0.21
Ant	5	31.2	61.6	38.1	0.11	0.36
HalfCheetah	5	29.7	42.9	56.8	0.11	0.38
Hopper	5	31.7	46.9	52.6	0.12	0.38
Walker2d	5	27.4	55.7	44.0	0.10	0.35
All	50	67.7	82.3	17.5	0.12	0.18

A.9 HARD-CONSTRAINT EXECUTOR

Because on-policy training that only excludes edges was unsafe or earned little return (Appendix I), the hard-constraint experiments use a separate soft actor-critic (SAC) learner (Haarnoja et al., 2018) with the feasibility critic of feasible dual policy iteration (FDPI) (Yang et al., 2026) for hard constraints and priced soft costs. At execution, *critic-guided selection* chooses among 16 policy samples and the deterministic action with this critic. A *shield* applies the same rule with the cost critics of the safe distributional actor-critic (SDAC) (Kim et al., 2023) or of spectral-risk-constrained policy optimization (SRCPO) (Kim et al., 2024) (Appendix I.3). Excluding edges from the graph and critic-guided selection are distinct operations. The contact rates in Section 5.3 are empirical outcomes rather than a zero-contact guarantee. Appendix C.1 specifies the learned critic, the training margins and the evaluation threshold.

B PROOFS AND POLICY REALIZATION

Throughout, $E' = \operatorname{supp}(\hat{F}) \setminus H$ is the set of admissible edges, $\mathcal{K}_{\mathrm{s}}$ the soft constraints, and a flow is feasible when $F \geq 0$, $F_{ij} = 0$ off E', $F\mathbf{1} - \gamma F^{\top}\mathbf{1} = (1-\gamma)\boldsymbol{\mu}_0$ and $\langle C_k, F\rangle \leq b_k$ for $k \in \mathcal{K}_{\mathrm{s}}$. We write the flow-conservation operator as $(MF)_i = (F\mathbf{1} - \gamma F^{\top}\mathbf{1})_i$, and call F *balanced* when it satisfies flow conservation, $MF = (1-\gamma)\boldsymbol{\mu}_0$. Its adjoint acts on node vectors as $(M^{\top}w)_{ij} = w_i - \gamma w_j$. Summing the flow-conservation equation over nodes gives $(1-\gamma)\mathbf{1}^{\top}F\mathbf{1} = (1-\gamma)$, so every balanced flow has mass one.

B.1 PROOF OF PROPOSITION 1

Existence and strong duality. The objective is strictly convex on $\{F \geq 0 : F = 0 \text{ off } E'\}$ (relative entropy is strictly convex there and the linear term does not change curvature), and the feasible set is a non-empty, closed, bounded (mass one) convex set, so a unique minimizer $F^{\star}_{\varepsilon}$ exists. We use the generalized relative entropy $\sum_{E'}[F_{ij}(\log(F_{ij}/\hat{F}_{ij}) - 1) + \hat{F}_{ij}]$. Because feasible flows vanish off E', it equals $\mathrm{KL}(F\|\hat{F})$ minus the constant $\sum_{(i,j)\notin E'} \hat{F}_{ij}$ and yields the same minimizer. The assumed flow F° lies in the relative interior of the feasible set restricted to E' and satisfies the budgets strictly, so Slater's condition holds for the convex program on E': strong duality holds and Karush–Kuhn–Tucker (KKT) multipliers $u \in \mathbb{R}^N$ (flow conservation) and $\lambda^{\star} \in \mathbb{R}_{+}^{|\mathcal{K}_{\mathrm{s}}|}$ (budgets) exist.

Stationarity. The Lagrangian is

$$\mathcal{L}(F,u,\lambda) = \langle -A, F\rangle + \varepsilon \textstyle\sum_{(i,j)\in E'} F_{ij}\big(\log \frac{F_{ij}}{\hat{F}_{ij}} - 1\big) \\ - \langle u, MF - (1-\gamma)\boldsymbol{\mu}_0\rangle + \textstyle\sum_{k\in\mathcal{K}_{\mathrm{s}}} \lambda_k(\langle C_k, F\rangle - b_k),$$

and setting $\partial\mathcal{L}/\partial F_{ij} = 0$ on E' gives (7),

$$\varepsilon \log(F_{ij}/\hat{F}_{ij}) = A_{ij} - \sum_k \lambda_k C_k(i,j) + (M^{\top}u)_{ij}, \qquad (M^{\top}u)_{ij} = u_i - \gamma u_j.$$

The minimizer is strictly positive on E' because the entropy has infinite slope at zero, which is why the positivity in the assumption is needed (a merely feasible flow can force $F_{ij} = 0$ on an edge of E' through flow conservation, in which case no finite u exists). Complementary slackness gives $\lambda^{\star}_k(\langle C_k, F^{\star}\rangle - b_k) = 0$.

The dual problem. Substituting $F^{\star}(u,\lambda)$ into $\mathcal{L}$ gives the dual

$$D(u,\lambda) = (1-\gamma)\langle u, \boldsymbol{\mu}_0\rangle - \textstyle\sum_{k\in\mathcal{K}_{\mathrm{s}}} \lambda_k b_k - \varepsilon \sum_{(i,j)\in E'} \hat{F}_{ij} \exp\big(\frac{1}{\varepsilon}(A_{ij} - \sum_k \lambda_k C_k(i,j) + u_i - \gamma u_j)\big),$$

which is concave (linear terms minus exponentials of affine maps). Its partial derivatives are $\partial_{u_i} D = (1-\gamma)\mu_{0,i} - (MF^{\star})_i$, the flow-conservation residual, and $\partial_{\lambda_k} D = \langle C_k, F^{\star}\rangle - b_k$, the budget residual. The Hessian in u is $-\varepsilon^{-1} M \operatorname{diag}(F^{\star}) M^{\top}$, negative semi-definite, and definite when every weakly connected component of the admissible graph contains a directed cycle. This is what makes damped Newton in u well posed. With u maximized out, $\bar{D}(\lambda) = \max_u D(u,\lambda)$ is

concave with $\partial_{\lambda_k}\bar{D} = \langle C_k, F^\star\rangle - b_k$, so each $\lambda_k \mapsto \langle C_k, F^\star\rangle$ is non-increasing along coordinate lines, which justifies coordinate bisection in λ. The dual is therefore a concave maximization over $(u, \lambda) \in \mathbb{R}^N \times \mathbb{R}_+^{|\mathcal{K}_s|}$, $|\mathcal{K}_s|$-dimensional once u is maximized out. □

Implementation. For the on-policy benchmark, the solver uses $\gamma = 0.99$ in flow conservation, $\hat{F}$ from undiscounted transition counts decayed by a factor 0.5 per epoch, $\boldsymbol{\mu}_0$ as the empirical distribution of episode-start nodes and A as the edge mean of the standardized reward advantage, and it returns $\lambda^\star$ multiplied back by the advantage scale, i.e. in units of reward advantage per unit of per-step cost. The solver adds a self-loop of weight 10^{-7} at every visited node before normalizing $\hat{F}$. A loop carries the statistics of observed self-transitions and $A = C = 0$ otherwise. Admissible self-loops give outgoing edges. If every visited node is reachable from the current start distribution through admissible edges, these loops permit a balanced flow positive on every retained edge. This is only the flow-conservation part of Proposition 1: strict feasibility of all soft budgets must also hold. Observed loops can carry positive cost, the decayed edge statistics can include currently unreachable nodes, and hard-edge removal can destroy reachability, so the implementation does not ensure the proposition's assumptions. Unobserved loops could let the projection hold mass at zero cost, and the LP limit could exploit this. In the LP checks of Appendix K, the added self-loops carry zero flow, and removing them changes neither the solver status nor the zero prices. The ε=0 setting therefore solves the graph LP, not the occupancy LP of Section 2. If all self-loops remain admissible and have positive flow, their coefficients $1 - \gamma$ make $M\,\mathrm{diag}(F)M^\top$ positive definite. When the budget is not met even at the upper end $\lambda_{\max} = 20$ of the bisection interval, the solver returns $\lambda_{\max}$, which the clip then caps. In the training epochs of the five benchmark seeds of the default configuration, this happens in 0–3% of the epochs on most tasks and in 10%, 16% and 45% on CarGoal1, PointButton1 and CarButton1.

The potential term is a shaping term. For any balanced flow, $\sum_{ij} F_{ij}(u_i - \gamma u_j) = \langle u, MF\rangle = (1-\gamma)\langle u, \boldsymbol{\mu}_0\rangle$, independent of F. Hence $\varepsilon \log(F^\star/\hat{F})$ differs from the price-adjusted edge advantage $A - \sum_k \lambda_k^\star C_k$ by a term that does not change the ranking of balanced flows (Ng et al., 1999).

B.2 Excluded Edges and Finite Entropic Prices

Proof of Corollary 1. On E, $\hat{F}_e > 0$ and the exponential of every finite score is positive. Since $C_h > 0$ on the nonempty H_h, every such flow has strictly positive h-cost. Proposition 1 supplies the finite optimal representation under its hypotheses. Complementary slackness gives cost b_h when $\lambda_h^\star > 0$. At $b_h = 0$, nonnegativity yields

$$\langle C_h, F\rangle \le 0 \quad \Longleftrightarrow \quad F_e = 0 \quad (e \in H_h).$$

Keeping every other constraint fixed therefore gives exactly the feasible set obtained by replacing H_0 with $H_0 \cup H_h$ and dropping this budget. By assumption it is nonempty. Flow conservation makes it compact, and strict convexity of the entropic objective on the reference support gives a unique minimizer. Any finite exponential representation on E would assign positive mass to H_h, contradicting feasibility. This proves nonattainment by a finite complete dual representation, not necessarily divergence of the single coordinate λ_h. □

When the hard-constraint price diverges. A stronger scalar conclusion needs additional conditions. Fix the graph, $\varepsilon > 0$, and the compact convex set $\mathcal{F}_{-h}$ defined by all constraints other than the budget for h. Suppose $\mathcal{F}_{-h}$ contains both a flow positive on every edge of E and a flow with zero h-cost. Then any optimal scalar multiplier $\lambda_h(b)$ for a positive budget b diverges as $b \downarrow 0$. To see this, the zero-cost flow strictly satisfies each positive scalar budget. For every finite λ, the minimizer over $\mathcal{F}_{-h}$ of the entropic objective plus $\lambda\langle C_h, F\rangle$ is positive on E: mixing a boundary candidate with the positive flow has a negative infinite initial entropy derivative, whereas the linear terms have finite derivatives. Its h-cost is therefore positive. A bounded multiplier subsequence as $b \downarrow 0$ would, by compactness and continuity, produce a zero-cost minimizer of a finite-penalty problem, a contradiction. The graph and the entropic regularization ε are fixed here.

Policy interpretation and scope. For a finite action set, $\tau > 0$ and finite λ, A_r, A_c, the update $\pi'(a|s) \propto \pi(a|s)\exp((A_r - \lambda A_c)(s,a)/\tau)$ preserves positive probability on every action in the

reference support. This is a property of the specified entropy update. Without entropy, a risky action with reward/cost $(1, 1)$ and a safe action with $(0, 0)$ are optimally resolved in favor of the safe action by any finite penalty $\lambda > 1$. Moreover, for a nonnegative random event cost Z, $\mathbb{E}Z = 0$ implies $Z = 0$ almost surely: expected-cost constraints can express exclusion at zero budget under their specified distribution. Graph support removal is an explicit realization of that boundary constraint. It still requires a realizable policy/executor to yield policy-level exclusion. Expected contact cost is not generally the probability of any contact in an episode.

B.3 Proof of Proposition 2

Let $\mathcal{F}$ be the feasible set, $P_\varepsilon = \min\{\langle -A, F\rangle + \varepsilon \,\mathrm{KL}(F\|\hat{F}) : F \in \mathcal{F}\}$ and $\mathcal{F}^{\mathrm{LP}} \subseteq \mathcal{F}$ the optimal face of the LP $\min\{\langle -A, F\rangle : F \in \mathcal{F}\}$.

Part (i): the primal limit. $\mathcal{F}$ is compact and $\mathrm{KL}(\cdot\|\hat{F})$ is finite and continuous on it. For any $F^{\mathrm{LP}} \in \mathcal{F}^{\mathrm{LP}}$, optimality of $F^\star_\varepsilon$ gives

$$\langle -A, F^\star_\varepsilon\rangle \le \langle -A, F^{\mathrm{LP}}\rangle + \varepsilon(\mathrm{KL}(F^{\mathrm{LP}}\|\hat{F}) - \mathrm{KL}(F^\star_\varepsilon\|\hat{F})),$$

so every accumulation point of $F^\star_\varepsilon$ as $\varepsilon \to 0$ lies in $\mathcal{F}^{\mathrm{LP}}$. The same inequality, restricted to the optimal face, shows that it minimizes $\mathrm{KL}(\cdot\|\hat{F})$ over $\mathcal{F}^{\mathrm{LP}}$ (the standard viscosity-selection argument). That minimizer is unique by strict convexity, so the whole path converges to it.

Part (i): bounded multipliers. The strictly feasible F° bounds the multipliers uniformly in $\varepsilon \in (0, 1]$: evaluating the Lagrangian at F° gives

$$\sum_k \lambda^\star_{\varepsilon,k}(b_k - \langle C_k, F^\circ\rangle) \le \langle -A, F^\circ\rangle - \langle -A, F^\star_\varepsilon\rangle + \varepsilon \,\mathrm{KL}(F^\circ\|\hat{F}),$$

which is bounded. To bound the potentials modulo $\ker M^\top$, write $\psi = M^\top u_\varepsilon$. Stationarity and $F^\star_{ij} \le 1$ give a uniform upper bound on every ψ_{ij}, since λ_ε is bounded, $\varepsilon \le 1$, and $\hat{F}$ is strictly positive on E'. Multiplying stationarity by $F^\star$ gives

$$\langle u_\varepsilon, (1-\gamma)\boldsymbol{\mu}_0\rangle = \varepsilon \sum_{E'} F^\star_{ij} \log(F^\star_{ij}/\hat{F}_{ij}) - \langle A, F^\star\rangle + \sum_k \lambda_{\varepsilon,k}\langle C_k, F^\star\rangle,$$

which is uniformly bounded below. It also equals $\langle F^\circ, \psi\rangle$. Strict positivity of F° and the coordinate-wise upper bound therefore give a lower bound on every coordinate of ψ. Thus $M^\top u_\varepsilon$ is bounded, and so is u_ε after projection onto $(\ker M^\top)^\perp$.

Part (i): limits of the multipliers. Pass to the limit in the KKT system along a convergent subsequence. On edges that leave the LP support, $F^\star_\varepsilon \to 0$, and stationarity holds in its complementary-slackness form. Hence every accumulation point of $\lambda^\star_\varepsilon$ is an LP dual solution. Equivalently, the dual D of Appendix B.1 is, up to constants, the exponential penalty with weights $\hat{F}$ of the dual LP's constraints $A_{ij} - \sum_k \lambda_k C_k(i,j) + u_i - \gamma u_j \le 0$, so (i) is the weaker, accumulation-point form of the exponential-penalty asymptotics of Cominetti & San Martín (1994).

Part (ii): the limit $\varepsilon \to \infty$. Divide the KKT system by ε and set $\theta = \lambda/\varepsilon$ and $v = u/\varepsilon$:

$$F = \hat{F} \odot \exp(A/\varepsilon - \sum_k \theta_k C_k + M^\top v).$$

As $\varepsilon \to \infty$ the reward term vanishes, and $F^\star_\varepsilon$ converges to the minimizer F^I of $\mathrm{KL}(\cdot\|\hat{F})$ over $\mathcal{F}$, because the objective of $P_\varepsilon/\varepsilon$ converges uniformly on the compact $\mathcal{F}$ and the limit has a unique minimizer. By the argument of (i), $\theta^\star_\varepsilon$ is bounded, and every accumulation point is a multiplier θ^I of the information projection.

Part (ii): local expansion. Assume $\hat{F}$ is balanced ($M\hat{F} = (1-\gamma)\boldsymbol{\mu}_0$) and $\hat{F}(H) = 0$, write $W = \mathrm{diag}(\hat{F})$ on E', and assume $MWM^\top$ is invertible (as in Appendix B.1, every weakly connected component of the admissible graph contains a directed cycle). We expand at $b = \hat{J}$ and $\theta = v = 0$,

where the information projection returns $\hat{F}$. Along a sufficiently small budget perturbation $\delta = \hat{J} - b$, assume a stable active set $\mathcal{K}$ and invertible $\tilde{G}_{\mathcal{K}\mathcal{K}}$. Without the support assumption, flow conservation may be lost after deleting H. The expansion must then be around the projection onto balanced flows with admissible support, not around $\hat{F}$. To first order in (θ, v), $F = \hat{F} \odot (1 - \sum_l \theta_l C_l + M^\top v)$, and flow conservation $M(F - \hat{F}) = 0$ forces

$$v = (MWM^\top)^{-1} MW \textstyle\sum_l \theta_l C_l, \qquad F - \hat{F} = -W \textstyle\sum_l \theta_l \tilde{C}_l,$$
$$\tilde{C}_l = C_l - M^\top (MWM^\top)^{-1} MWC_l,$$

the $\hat{F}$-weighted residual of C_l after removing potential differences $w_i - \gamma w_j$. The conservation correction v is thus first order in θ and cannot be dropped. Because $\tilde{C}_l$ is W-orthogonal to the range of $M^\top$, which contains the constant vector ($\mathbf{1} = M^\top \frac{1}{1-\gamma}\mathbf{1}$), each $\tilde{C}_l$ has $\hat{F}$-mean zero and $\tilde{G}$ is a covariance. Moreover $\langle C_k, W\tilde{C}_l \rangle = \langle \tilde{C}_k, W\tilde{C}_l \rangle$, and the budget residual of constraint k is

$$\langle C_k, F \rangle - b_k = (\hat{J}_k - b_k) - \textstyle\sum_l \tilde{G}_{kl}\theta_l + o(\|\theta\|), \qquad \tilde{G}_{kl} = \langle \tilde{C}_k \odot \tilde{C}_l, \hat{F} \rangle.$$

The KKT conditions of the information projection,

$$\theta \geq 0, \qquad \langle C_k, F \rangle - b_k \leq 0, \qquad \theta_k(\langle C_k, F \rangle - b_k) = 0,$$

are, to first order, the optimality conditions of

$$\min_{\theta \geq 0} \tfrac{1}{2}\theta^\top \tilde{G}\theta - \theta^\top(\hat{J} - b).$$

On its active set $\mathcal{K}$ (where $\theta_k > 0$), this gives

$$\theta^I_{\mathcal{K}} = \tilde{G}^{-1}_{\mathcal{K}\mathcal{K}} \delta_{\mathcal{K}} + o(\|\delta\|)$$

under the stated local regularity assumptions. The inverse function theorem applied to the active residual equations gives $\theta^I = O(\|\delta\|)$ and justifies this remainder. □

Remarks. The active set need not equal the initially violated constraints: cross-constraint residual covariances determine which constraints become active after rerouting. A constant cost $C_k \equiv c_0$ is a potential difference ($w \equiv c_0/(1-\gamma)$), so $\tilde{C}_k = 0$: a cost that every balanced flow pays cannot be re-routed, so it does not change the direction of the priced step. A violated constant-cost budget is infeasible, and its zero residual variance precludes the inverse-covariance formula.

B.4 Growth of the Feedback Term

Proof of Lemma 1. For $\beta \geq 0$, $[\beta + \alpha e]_+ \leq \beta + \alpha[e]_+$. Applying this inequality at each update and telescoping gives $\beta_T \leq \beta_0 + \alpha \sum_{t=1}^T [e_t]_+$. Rearranging at any time with $\beta_T \geq \ell > \beta_0$ gives the stated lower bound. If additionally $[e_t]_+ \leq G$, $G > 0$, at least $\lceil (\ell - \beta_0)/(\alpha G) \rceil$ positive-error observations are needed. □

This accounts for the errors supplied to a fixed-step controller, which may be smoothed or estimated. It does not bound the cost incurred in episodes, and it does not identify a price level ℓ that ensures safety. A fixed penalty or a warm start can exceed ℓ at initialization, and then no such violation is needed.

Clipped safety prices. Write $x_t = \min\{c, \lambda^\star_t\}$ and $z_t = (1 - \xi_t) z_{t-1} + \xi_t x_t$, where $\lambda^\star_t \geq 0$, $c \geq 0$ and $\xi_t \in [0, 1]$. If $z_0 \in [0, c]$, induction gives $z_t \in [0, c]$. The implemented first-step initialization $\xi_1 = 1$ instead sets $z_1 = x_1$ directly and removes dependence on z_0. A gain $s_t \in [0, 1]$, including any ramp or gate, gives $g_t = s_t z_t \leq c$. If $\lambda_T = \min\{\Lambda, g_T + \beta_T\} \geq \ell$, then $\ell \leq g_T + \beta_T \leq c + \beta_0 + \alpha V^e_T$. Combining this with $V^e_T \geq 0$ proves Eq. (8). If $\ell > \Lambda$, the antecedent is impossible. The sharper realized-time bound replaces c by g_T. For an actor step using β_{T-1}, the corresponding sum stops at $T - 1$ instead. No stationary graph or time-invariant target threshold is needed for this pointwise implication, but it does not establish a safety threshold or any future controller behavior. The errors are exactly those supplied to the controller. Conversion to actual episode harm requires additional assumptions.

For the reactive switch of Table 35, a raw input $\kappa \mathbf{1}[e_t > 0]$ with $\kappa \geq c$ is clipped to c on an epoch with $e_t > 0$. A safety price with $\lambda_t^\star \geq c$ has the same clipped input. If both smoothed values have equal previous states and use the same ξ_t, then their new smoothed states agree. Equal gains, feedback states and total caps then give equal combined prices. Equal clipped inputs at a single epoch do not erase unequal smoothing histories: their difference is multiplied by $1 - \xi_t$. Different rollout histories may also produce different feedback states. Consequently saturation alone is not a theorem that the two learning trajectories coincide.

Proportional, derivative and switch terms. For the explicitly defined unfiltered PID rule

$$I_t = [I_{t-1} + K_I e_t]_+, \qquad \lambda_t = [I_t + K_P e_t + K_D(e_t - e_{t-1})]_+,$$

with $I_0 \geq 0$ and nonnegative gains,

$$\lambda_t \leq I_0 + K_I \sum_{s=1}^{t} [e_s]_+ + K_P [e_t]_+ + K_D [e_t - e_{t-1}]_+.$$

This follows from the same induction and subadditivity of the positive part. It is not an identification of this simplified recursion with SafePO's PID implementation, which uses filtered proportional and derivative states. Those actual states must appear in its corresponding inequality. For a switch plus plain integral feedback, $\lambda_t = \kappa \mathbf{1}[e_t > 0] + \beta_t$ is bounded by the current switch term plus $\beta_0 + \alpha V_t^e$. With a smoothed switch, its smoothed value replaces the indicator. In particular, a switch can supply a positive price at its first positive-error observation. No universal multi-epoch delay follows.

Why the finite-time bound does not transfer to Adam. The SafePO experiments use Adam with initial multiplier 0.001 and learning rate 0.035. Adam normalizes a history of gradients, so its increment is not αe_t. For example, with errors $1, 0$, the second update remains positive due to momentum. With errors $e_t = t$, standard Adam moments $(0.9, 0.999)$ and numerical denominator 10^{-8}, the multiplier exceeds 1 at update 27 (1.01953), whereas bounding each Adam step by the learning rate would require $\lceil 0.999/0.035 \rceil = 29$ updates. The fixed-step lemma therefore motivates a measurement rather than a quantitative Adam delay prediction. One qualitative property does transfer: while every error so far is nonpositive, the first Adam moment of the gradient $-e_t$ is nonnegative, so no Adam step raises the multiplier above its initial value before the first violation. The asymptotic conditional statement in Proposition 3 is separate and remains valid under its convergence hypotheses.

B.5 From a Target Flow to Policy Cost

Section 4.3 uses the following bound on the budget violation of the updated policy.

Lemma 2 (Flow-to-policy cost accounting)**.** *On a fixed edge set, let C_k^π be the exact edge-conditional costs of π, so that $J_{c_k}(\pi) = \langle C_k^\pi, F^\pi \rangle$. For any target flow $q \geq 0$ and finite estimated edge costs $\widehat{C}_k$,*

$$\big[J_{c_k}(\pi) - b_k\big]_+ \leq \underbrace{\big[\langle \widehat{C}_k, q \rangle - b_k\big]_+}_{r_k\text{: target residual}} + \underbrace{\|\widehat{C}_k\|_\infty \, \|F^\pi - q\|_1}_{\text{flow-to-policy gap}} + \underbrace{\langle |C_k^\pi - \widehat{C}_k|, F^\pi \rangle}_{a_k\text{: cost-estimation error}}. \tag{9}$$

The first term vanishes for an exact solution $q = F^\star$ of (6) when $\widehat{C}_k$ are the estimated costs C_k of (6), but the other two remain, so graph feasibility alone cannot certify policy safety.

Proof of Lemma 2. Fix the graph partition and an edge universe covering the target and updated-policy flow. Assign exact conditional costs arbitrarily on edges of zero policy mass. Add and subtract the estimated target cost:

$$\begin{aligned} J_{c_k}(\pi) - b_k &= \langle \widehat{C}_k, q \rangle - b_k + \langle \widehat{C}_k, F^\pi - q \rangle + \langle C_k^\pi - \widehat{C}_k, F^\pi \rangle \\ &\leq r_k + \|\widehat{C}_k\|_\infty \|F^\pi - q\|_1 + \langle |C_k^\pi - \widehat{C}_k|, F^\pi \rangle. \end{aligned}$$

The right side is nonnegative, so taking the positive part proves Eq. (9). Summing the inequality over successive updates bounds the summed positive excess by the sums of the target residuals, flow-to-policy gaps and cost-estimation errors. □

No flow-conservation assumption is needed for this inequality. Primal budget feasibility of q makes $r_k = 0$. Complementary slackness alone does not. An approximate solver may leave $r_k > 0$. Even perfect flow realization does not ensure feasibility if costs are biased: $\widehat{C}_k = 0$ can satisfy a budget while true costs are positive. Conversely, using sampled next-policy flows adds sampling discrepancies before one can evaluate this bound for the true policy. Missing edges must be included in the common universe with explicit estimates, not silently dropped. The terms can depend on the entire price and policy history. The inequality holds for any price rule.

Episodic units and budget conversion. Lemma 2 uses a normalized cost and budget b_k. For an episodic occupancy normalized by the actual expected episode length L_{t+1}, write $J^{\mathrm{epi}}_{c_k} = L_{t+1}\langle C^\pi_k, F^\pi\rangle$. Then

$$[J^{\mathrm{epi}}_{c_k} - B_k]_+ \le L_{t+1}\big(r_k + \|\widehat{C}_k\|_\infty \|F^\pi - q\|_1 + a_k\big) + [L_{t+1} b_{k,t} - B_k]_+.$$

The last term accounts for using the previous rollout's estimated length in $b_{k,t} = B_k/\widehat{L}_t$. A discounted normalized flow and an undiscounted episodic flow are also distinct objects. Logged overlapping episode windows add another measurement discrepancy. Accordingly, the tabular realization diagnostic of Appendix G.1 checks normalized cost accounting. Without these conversions, the inequality does not bound the logged unnormalized violation $V = B \cdot \mathrm{CV}$ (Appendix C.1).

Finite-ε prices depend on the reward. Part (ii) of Proposition 2 combines two distinct limits: $\varepsilon \to \infty$ and a local budget perturbation around a balanced observed flow. Their separate remainders do not justify an arbitrary joint $o(\varepsilon\|\hat{J} - b\|)$ remainder for the full reward-dependent price. For example, take a nonconstant cost C, edge advantage $A = C$, a positive balanced observed flow $\hat{F}$, and $b = \langle C, \hat{F}\rangle$ with the stated strict-feasibility conditions. At every $\varepsilon > 0$, $F^\star = \hat{F}$ and $\lambda^\star = 1$ satisfy the KKT system, although the budget error of the observed flow is zero. Thus a zero limit of $\lambda^\star/\varepsilon$ does not imply a zero safety price. Clipping, smoothing and observed-cost feedback further separate the implemented controller from a conditional graph limit.

B.6 Graph Surrogate and Policy Realization

An occupancy ρ^π induces the cluster-level flow $F^\pi_{ij} = \sum_{s\in i}\sum_a \rho^\pi(s,a)\sum_{s'\in j} P(s'|s,a)$. Summing (2) over the states of each cluster shows that F^π is balanced with respect to the aggregated initial distribution. For the *current* policy π_t, if costs are defined by exact discounted-occupancy-weighted edge means and the initial distribution is exact, then $\langle C_k, F^{\pi_t}\rangle = J_{c_k}(\pi_t)$. For any other policy π the same C_k need not equal the ρ^π-weighted edge means, and $\langle C_k, F^\pi\rangle \ne J_{c_k}(\pi)$ in general. Conversely, $\mathcal{F}$ contains flows that no policy realizes: a flow may choose the proportions in which mass leaves a node freely, whereas a policy can only realize proportions in the convex hull of the kernels $P(\cdot|s,a)$ of the states in that node. This gap exists even without aggregation ($N = |\mathcal{S}|$). This relaxes routing, but does not establish an outer relaxation of the original policy problem: empirical support restrictions can exclude feasible policies, and behavior-dependent edge costs can change their budget classification. The implemented statistics are undiscounted rolling counts, so the exact discounted cost identity above does not apply to them. Thus (6) is an empirical graph surrogate. Neither the sign of its policy-cost error nor its removal is guaranteed. Appendix F measures the realization error, and Appendix H.5 the effect of the feedback term.

Action information can be lost even without state aggregation. Consider one state and two actions, both returning to that state. The risky action has reward and cost one. The safe action has reward and cost zero. A behavior policy choosing the risky action with probability $p > b$ induces a single state-to-state edge with flow $F = 1$ and mean cost $C = p$. The graph constraint $CF \le b$ is infeasible, although the CMDP admits the optimal feasible policy choosing the risky action with probability b. Thus edge averaging can erase a feasible action substitution even with exact data. An informative policy-level interpretation needs additional structure preserving action-dependent costs and realizable flow changes. The graph propositions alone do not supply it.

B.7 Proof of Proposition 3

Write $e_k(t) = \bar{J}_k(t) - B_k \to e^\infty_k = \bar{J}^\infty_k - B_k$.

Plain projected ascent. Suppose $e_k^\infty > 0$. For t large, $e_k(t) \geq e_k^\infty/2$, so $\beta_k(t+1) \geq \beta_k(t) + \alpha e_k^\infty/2$ and $\beta_k(t) \to \infty$, contradicting $\beta_k^\infty < \infty$. Hence $\bar{J}_k^\infty \leq B_k$. Suppose $\beta_k^\infty > 0$ and $e_k^\infty < 0$. For t large, $\beta_k(t) > \beta_k^\infty/2 > 0$ and $e_k(t) \leq e_k^\infty/2$, so β_k decreases by at least $\alpha|e_k^\infty|/2$ per step until it reaches zero in finitely many steps, contradicting $\beta_k^\infty > 0$. Hence $\bar{J}_k^\infty = B_k$ when $\beta_k^\infty > 0$.

Adam. With Adam at learning rate η minimizing the loss $-\beta(\bar{J}-B)$, the bias-corrected moments of the gradient $-e_k(t)$ satisfy $\hat{m}_t \to -e_k^\infty$ and $\hat{v}_t \to (e_k^\infty)^2$, so for $e_k^\infty \neq 0$ the unprojected ascent step converges to $\eta e_k^\infty/(|e_k^\infty| + \epsilon_{\text{Adam}})$. Its magnitude is eventually at least $\eta|e_k^\infty|/[2(|e_k^\infty| + \epsilon_{\text{Adam}})] > 0$, so the same two sign arguments apply. The safety-price component does not enter the β-recursion, so the statement holds for any bounded $\bar{\lambda}^\star$. □

C Experimental Details

The baselines run the SafePO scripts with their default hyperparameters, and SafeOT keeps every TRPO-Lag hyperparameter. Appendix H.3 lists the safety-controller settings of every method (Table 35), the shared actor settings and the settings of the flow pricer. GAE uses $\lambda_{\text{GAE}} = 0.95$. The actor and critics have two 64-unit tanh layers.

The off-policy settings below apply to the hard-constraint experiments of Section 5.3 and differ from the on-policy defaults. The main comparison uses the common margin schedule described below. Appendix I.1 reports fixed margins and task-selected margin schedules separately.

C.1 Implementation and Evaluation Protocol

Tasks and costs. All MuJoCo tasks use Safety-Gymnasium (Ji et al., 2023) with native 1000-step episodes. The PyBullet suite is described in Appendix C.2. The single-constraint benchmark uses the native aggregate cost with budget $B = 25$ on the navigation tasks SafetyPointGoal1, SafetyCarGoal1, SafetyPointButton1, SafetyCarButton1, SafetyPointPush1 and SafetyCarPush1 (`-v0`) and on the velocity tasks SafetyAntVelocity, SafetyHalfCheetahVelocity, SafetyHopperVelocity and SafetyWalker2dVelocity (`-v1`). The budget scan of Appendix E adds $B \in \{5, 12.5, 50\}$. The K=3 tasks split the costs of SafetyPointButton1 and SafetyCarButton1 into their native cost signals (`cost_hazards`, `cost_buttons`, `cost_gremlins`) with budgets $(10, 10, 5)$, and the K=2 tasks split those of SafetyPointGoal2 and SafetyCarGoal2 into vase and hazard costs with budgets $(5, 25)$. The hard-constraint experiments split the native cost signals of three level-2 tasks, SafetyCarGoal2, SafetyPointGoal2 and SafetyRacecarGoal2, into a hard contact cost (`cost_vases_contact`) and a soft hazard cost (`cost_hazards`, budget 25). We call these runs *typed*, and *untyped* controls use the aggregate cost. During training a hard constraint also fires when the lidar reading towards the never-touch object exceeds the margin d (0.8 unless stated). Evaluation records native contacts only.

Methods on Safety-Gymnasium. CPO, PCPO, TRPO-Lag, PPO-Lag, CPPO-PID, FOCOPS and CUP use the unmodified SafePO single-agent scripts with one environment, 20,000 steps per epoch, and the stated budgets/horizons. SafeOT replaces that multiplier by $\bar{\lambda}^\star + \beta$, retaining the TRPO step, critics and Adam observed-cost feedback in the default configuration. Feedback minimizes $-\beta(\bar{J} - B)$ using the 50-episode cost window. With clip level $c = 0$ the script reduces to TRPO-Lag. TRPO-PID combines the PID Lagrangian class of SafePO with the TRPO-Lag script (Table 35). The same default $(\varepsilon, c, N) = (1, 1, 128)$ is used on every task. It was selected on three development tasks, PointGoal1, CarGoal1 and PointButton1.

In-house engine. Exploratory comparisons with a PPO actor share the actor, critic and GAE settings of the default configuration. The matched Lagrangian, PID Lagrangian and priced variants differ only in their multiplier source. Their unclipped safety price uses the bisection bound $\lambda_{\max} = 20$. These runs remain distinct from the benchmark implementations.

SafeOT pricer. Appendix H.3 specifies the graph, solver and transfer settings. Graph budgets use $b_k = B_k/L$. The observed flow $\hat{F}$ has a self-loop of mass 10^{-7} on each visited cluster. The entropy sweep includes exact linear-programming solves at $\varepsilon = 0$ with the HiGHS solver. The *no feedback*

ablation of Table 37 fixes $\beta \equiv 0$. Hard constraints on-policy: zero-capacity mode removes edges whose hard-cost rate exceeds 0.5 and applies the constant price $\lambda_{\max}$. The soft mode treats the hard constraint as a soft constraint with its own budget. The production solver uses 2 coordinate sweeps, 12 bisections and at most 60 Newton iterations per flow-conservation solve, with Newton stopping tolerance 10^{-7} and accepted flow-conservation residual below 10^{-5}. Its acceptance check requires a successful flow-conservation solve, finite flow and mass within 10^{-2} of one. It does not certify joint budget feasibility or complementary slackness. A rejected solve retains the previous price. If the flow-conservation solve with excluded edges fails, the solver retries with hard constraints treated as budgeted costs using their supplied budgets. This is a numerical fallback, not a hard-safety certificate. Implementation-specific solver timings are reported separately in Appendix H.4. They exclude graph construction.

Off-policy protocol (FDPI). The hard-constraint experiments use the SAC implementation of FDPI (Yang et al., 2026) with a feasibility wrapper on the hard indicator and its margin, a separate soft constraint, and replay rewards reduced by the current $\lambda_{\text{soft}} c_{\text{soft}}$. SafeOT uses $N = 128$, graph rebuilding every 2×10^4 samples, smoothing weight $\xi = 0.5$, clip level $c = 1$ and integral gain 10^{-3}. Controls use integral feedback alone, fixed price 0.02, or native aggregate feasibility cost without margin/pricing. Settings are two environments, 2M samples, checkpoints every 10^5 updates, networks $[256, 256]$, learning rate 10^{-4}, feasibility threshold 0.95, a threshold $p_f = 0.1$ on the predicted discounted hard cost below which an action counts as feasible, and target KL 5. SDAC (Kim et al., 2023) and SRCPO (Kim et al., 2024) use their released code and are evaluated in the same way. Sauté RL (Sootla et al., 2022), the Safety Layer (Dalal et al., 2018) and ET-MDP (Sun et al., 2021) wrap a SAC learner with a Lagrange multiplier for the hazard budget 25 and train for 2M steps with margin 0.8 and five seeds. Sauté RL adds the remaining contact budget, which starts at 0, to the state and gives reward -1 once it is spent. ET-MDP, the early-terminated MDP, ends a training episode at the first contact. All three are evaluated like the other rows. The graph uses standardized immediate rewards and restores reward units. Feedback pools primal and secondary collection policies, so its mixture budget is not a deployed-primal-policy budget. With two parallel environments, flattening the rollouts can assign episode starts to the wrong steps of the graph. The CarGoal2 and PointGoal2 runs of Table 40 were trained without correcting this (Appendix I.2), and the effect on the safety price was not measured. The single-environment benchmark runs are not affected.

Uniform protocol for Table 40. CarGoal2, PointGoal2 and RacecarGoal2 anneal the margin from 0.95 to 0.85 over 1.5M samples, then hold it fixed. Five seeds per task retain their respective trainer implementations. Evaluation uses the $1{,}990{,}002$-step checkpoints and 16 sampled candidates plus the deterministic action. The selection threshold $p_f = 0.02$ differs from the training value 0.1. Configuration/checkpoint records are in the supplement. Unfiltered counterparts appear in Appendix I.2.

Task-specific schedules. The Car $0.95 \to 0.78$ schedule spans 1M samples, whereas the Point $0.95 \to 0.85$ schedule spans 1.5M. The Point schedule is the common schedule of Appendix I.2, whereas the Car policies of this schedule differ from the common-schedule Car policies.

Evaluation of the hard-constraint runs. Each reported policy is evaluated on three blocks of 104 native-contact episodes, with reset seeds starting at 5000, 9000 and 13000. The same reset instances are reused across trained policies. Unfiltered evaluation takes deterministic policy actions, whereas critic-guided selection uses seeded candidate sampling. Return, hazard cost and contact rate for a system come from the same evaluated policy and episodes. Table 40 and Table 41 report return and hazard-cost means with seed-level 95% t intervals, and contact rates with seed-bootstrap 95% intervals. These are descriptive, unadjusted intervals, not certified safety bounds. Pooled one-sided 95% Clopper–Pearson (CP95) upper bounds and standard errors (s.e.) are labeled separately. Pooled episode calculations ignore shared reset instances and training-seed clustering.

Margin baselines. Uniform systems (Appendix I.2) and fixed-margin/budget controls (Appendix I.1) are separate comparisons. Evaluation scores native contacts, not margin costs.

Metrics. Training return and cost average the last ten logged epochs. Evaluation uses 30 deterministic episodes of the final policy with frozen observation statistics. A seed is feasible when its mean final cost is at most the budget of every constraint, and a method is feasible on a task (Tables 10, 11 and 15) when the unrounded mean of its seed means is at most B. Write $\bar{J}_k(t)$ for the logged cost of constraint k at epoch t, or $\bar{J}(t)$ for a single budget, as in Section 5.1. We report the unnormalized violation $V_k = \sum_t [\bar{J}_k(t) - B_k]_+$, the epoch-summed positive excess of logged cost. For a single budget, the cumulative constraint violation reported in the main text is $\mathrm{CV} = V/B$. It is not the sum of per-episode excesses and has no episode-count weighting. The in-house engine explicitly logs a trailing 50-completed-episode mean. The SafePO reductions use its native logged entries. Because consecutive 50-episode windows overlap, the scale of CV depends on the logging protocol, so each comparison uses one implementation and one logging protocol. Seed selection: for every (task, method, budget, steps) the earliest complete attempt of each seed is used, independently of its outcome. Retries of a seed are listed in the code supplement and never replace it.

Compute. On-policy runs use two CPU threads and take 0.75–5 h per 2M steps (Table 5 for SafeOT) and 10–15 h per 10M, depending on task and load. Off-policy runs share one A100 among three runs, taking approximately 14 h per 2M samples.

Why the update uses prices. An alternative transfers the target by reweighting samples with a clipped ratio $F^\star/\hat{F}$. Nonnegative sample weights cannot reverse an advantage's sign, whereas subtracting the safety price times a cost advantage can. We therefore transfer the safety prices (Proposition 1). For nonnegative costs, an edge capacity $F \le S$ with $\langle C_k, S\rangle \le b_k$ is sufficient but unnecessary for an aggregate budget. Direct budget constraints leave the allocation to the projection.

Decision rules and supplementary data. Where stated, comparison rules were fixed before their outcomes were inspected. Per-seed values and complete numerical displays are provided in the supplementary data package.

C.2 Bullet-Safety-Gym Protocol

Complete predefined task set. The Bullet-Safety-Gym experiment (Gronauer, 2022) comprises Ball, Car and Drone on Run and Circle: six tasks, SafeOT, the eight baselines and the reactive switch (Appendix D), and seeds 0–4, giving 300 complete runs. This is the predefined transfer set, not all environments distributed with the benchmark. Native horizons are 250 steps for Ball and 500 for Car/Drone. Each run uses 1M steps, 50 epochs of 20,000 steps, budget 10 and 30 deterministic final-policy episodes. The suite therefore has 9,000 evaluation episodes. The budget follows the experiment's software-default convention. It is not asserted to be a universal benchmark recommendation. Table 6 lists our baseline results next to the published ones of Gronauer (2022), which report normalized values from a per-task search with other training lengths, so the table documents protocol differences rather than a numerical comparison (Appendix D.8).

Metrics and paired evaluation. The unnormalized violation V sums the positive excess of the logged epoch cost over the 50 epochs. The logged cost is the epoch average of a rolling last-50-completed-episode mean, updated on each episode completion without resetting at epoch boundaries. Final return and final cost share the final checkpoint, the fixed observation normalizer and the reset seeds. A seed is feasible when its mean final cost is at most 10. Training means use the last ten epochs. The original cost and reward units are retained.

Joint comparison with Lagrangian and PID Lagrangian methods. The four family comparators are TRPO-Lag, PPO-Lag, CPPO-PID and TRPO-PID. The primary violation statistic is the equal-task mean difference of seed-mean CV. The primary return statistic is the equal-task mean return difference divided by $\max(1, |\bar{R}_{\mathrm{base}}|)$. The return non-inferiority margin is 5%. A 20,000-replicate task/paired-seed bootstrap shares resampled seed indices across methods and metrics. Eight one-sided 99.375% bounds address the four joint comparisons. The complete-suite positive decision requires a negative upper bound on the CV difference and a lower bound on the return difference above -0.05 for every comparator. A negative decision requires an opposite adjusted bound. Otherwise the result is inconclusive. These percentile bounds are approximate with six tasks and five seeds, not exact finite-sample guarantees.

The decision is negative because the adjusted opposite CV bound against TRPO-PID excludes the desired direction. The ratios below divide the equal-task mean CV of SafeOT by that of the comparator (below one favors SafeOT). Against TRPO-PID the ratio is 1.526 ($[1.068, 2.669]$). Against CPPO-PID it is 1.410 ($[0.971, 2.504]$), so its lower CV is not resolved by that interval. Against TRPO-Lag and PPO-Lag, the ratios are 0.221 ($[0.131, 0.455]$) and 0.275 ($[0.162, 0.468]$). These ratios of equal-task mean CV, rather than medians of task ratios, are distinct from the primary difference tests. None of the four comparisons establishes return non-inferiority. None establishes return inferiority under the adjusted opposite test. Thus the lower CV relative to the Lagrangian methods transfers to Bullet-Safety-Gym, whereas a joint improvement over all four Lagrangian and PID Lagrangian methods is not established.

All tasks and methods are reported with ordinary paired-seed 95% intervals. Supplementary episode records reproduce training and final-policy statistics. This numerical replay is not an independent training replication or safety guarantee.

C.3 Bullet Controller Diagnostics

The supplementary data contain the phase-by-phase table and the controller replay. Across epochs 1–10, 11–25 and 26–50, the ratios CV(SafeOT)/CV(method) for the Lagrangian methods span 0.09–0.47.

On Bullet-Safety-Gym, the observed-cost feedback of SafeOT uses the rolling 50-episode mean cost at the end of each epoch, whereas the SafePO Lagrangian and PID Lagrangian scripts average the rolling means recorded at episode completions within the epoch and receive zero cost in their first update. A replay of $9{,}000$ controller updates confirms this difference, the only one between SafeOT and TRPO-Lag on this suite besides the safety price. On CarRun, final costs 485.5 (TRPO-Lag) and 78.2 (SafeOT) both exceed $B = 10$. Returns are not compared at matched safety. The follow-up interventions appear in Appendix G.8.

C.4 Study Coverage

Table 7 lists the tasks, methods, seeds, training lengths and budgets of the main learned-policy studies. The configuration search and the tuned TRPO-Lag runs (Appendix D.6), the entropy sweep (Appendix H.2) and the 10M-step runs (Appendix E) are described in their appendices.

Table 6: Published counterparts on Bullet-Safety-Gym. T: our final return/cost (1M steps, $B = 10$, five seeds). G: Gronauer (2022), Tables 1 and 3, with the return divided by that of TRPO and the cost divided by their budget 25 (four seeds, best of a per-task search). The G settings differ from ours, and values in different units are not compared.

Task	Source	TRPO-Lag	CPO	PCPO	Steps
BallCircle	T	530.2/2.8	426.1/8.9	309.2/6.9	1M
	G	0.96/0.97	0.72/0.99	0.89/0.93	16M
BallRun	T	1180.2/92.4	558.6/0.0	544.8/0.0	1M
	G	0.25/0.52	0.18/1.05	0.09/0.30	3.2M
CarCircle	T	957.2/51.8	237.5/22.7	183.8/36.6	1M
	G	0.76/0.89	0.34/0.97	0.71/0.93	16M
CarRun	T	3292.1/485.5	691.9/0.5	655.5/0.0	1M
	G	0.23/0.87	0.21/1.02	0.23/0.91	6.4M
DroneCircle	T	55.3/6.8	58.1/10.7	36.9/7.5	1M
	G	0.75/0.93	0.43/0.99	0.05/0.93	64M
DroneRun	T	670.4/50.1	842.3/14.8	516.2/10.5	1M
	G	0.18/1.60	0.35/0.96	0.29/1.01	32M

Table 7: **Study coverage.** The multi-budget grid total includes 70 reused single-budget runs. Robot primary seeds 0–2 precede the five-seed check. The hard-constraint row is the comparison of Table 40 on CarGoal2, PointGoal2 and RacecarGoal2. Appendix I.5 and Appendix I.1 describe RacecarGoal2 and the controls.

Study (simulator)	Tasks	Methods	Seeds	Steps	Budget	Runs
Safety-Gymnasium (MuJoCo)	10	9	5–10	2M	5–50	1450
Bullet-Safety-Gym (PyBullet)	6	10	5	1M	10	300
Harder Safety-Gymnasium tasks	6	9	5	2M	25	270
Goal2 (MuJoCo)	3	6	5	2M	25 + hard	90
Multi-budget grid	2	7	5	2M	$K = 1, 4, 8$	220
Per-joint budgets (MuJoCo)	4	5–6	$3 + 2$	2M	100/250	110

D Benchmark Results and Seed Coverage

Tables 9 and 8 report every task of the comparison of SafeOT with the eight baselines of Table 2 on Safety-Gymnasium and Bullet-Safety-Gym. On Bullet-Safety-Gym, the reactive switch is an additional control. It keeps the clipping, smoothing and observed-cost feedback of SafeOT but replaces the raw safety price by $10^6\,\mathbf{1}[J_{\mathrm{batch}} > B]$, so its clipped price is $c = 1$ whenever the batch cost exceeds B and zero otherwise (Table 35). Final return and final cost always describe the same evaluated policy. CV uses all complete training histories. Evaluation and training sample sizes are separate. Uncertainty intervals are pointwise and do not establish simultaneous superiority across methods or tasks.

Seed coverage. All 90 Safety-Gymnasium and 54 Bullet-Safety-Gym task–method cells of the nine methods in Table 2 have five training and five evaluated seeds (720 policies, 21,600 final episodes). The conservative setting of SafeOT (Figure 4b and Appendix D.6) adds 16 five-seed cells. The reactive switch adds five-seed control cells on the six Bullet-Safety-Gym and the ten Safety-Gymnasium tasks (Tables 13 and 37). Training and final evaluation come from the same selected attempt. The six harder Safety-Gymnasium tasks are reported in Appendix D.7.

D.1 Learning Curves on Every Task

Figure 5 shows the training return and cost on all sixteen tasks. The curves use 250 Safety-Gymnasium and 150 Bullet-Safety-Gym runs, the five methods of Figure 6 (SafeOT, TRPO-Lag, PPO-Lag, CPPO-PID and CPO) and the five seeds of Table 2. Each cost/return pair uses the same logged policy window. Curves show seed means. Costs are shown on axes that are linear below the budget and logarithmic above it, so their tails remain visible.

D.2 Suite-Level Learning Curves and Median Trade-offs

Figure 6 summarizes the curves of Figure 5 over tasks, and Figure 7 places the nine methods by task-median CV and final return, complementing Table 2 and the mean ranks of Figure 3. Both figures use the same five-seed runs as Table 2. In Figure 6, each task contributes a seed-mean trajectory, and cost is divided by the task budget. For the return display, each task's seed-mean return is divided by the best final-policy method mean on that task. Across tasks, lines are medians and bands are interquartile ranges, so the bands describe task variation rather than seed uncertainty.

D.3 Per-Task Results

Tables 9 and 8 give confidence intervals for the per-task results that Table 2 summarizes. The reductions of Section 5.2, 63% and 70%, are one minus the median over tasks of the ratio CV(SafeOT)/CV(TRPO-Lag), whereas Table 2 reports the median of the inverse ratio ($2.8\times$ and $3.3\times$), which is not the reciprocal of this median. The sign test of Section 5.2 pools the 16 tasks: SafeOT has the lower CV on all 16, so the two-sided p-value is $2^{-15} \approx 3.1 \times 10^{-5}$. The supplementary data give all 144 task–method cells, each as a seed mean with a two-sided Student-t interval,

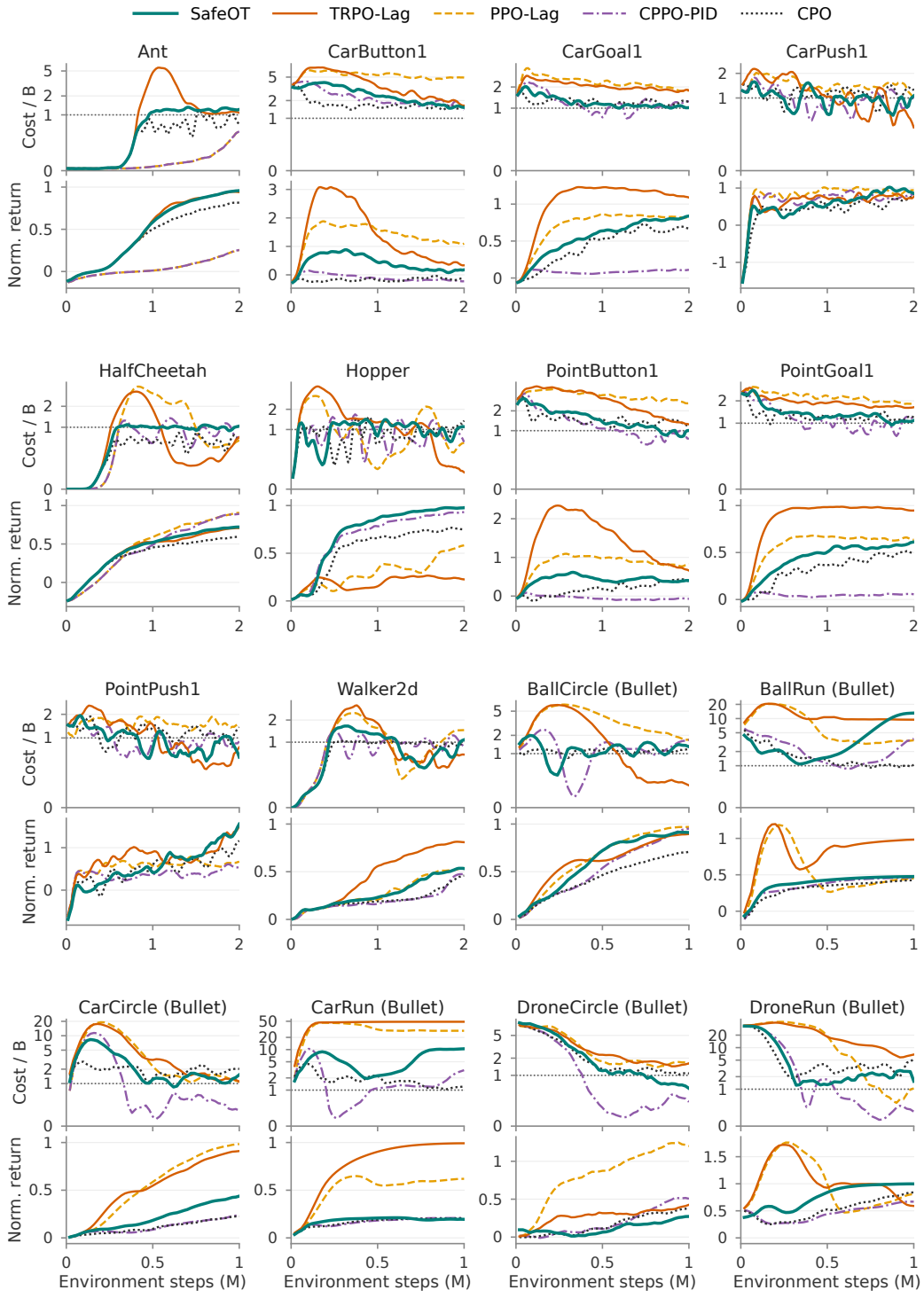

Figure 5: Training cost and return of SafeOT, TRPO-Lag, PPO-Lag, CPPO-PID and CPO on all sixteen tasks (five-seed means, smoothed over five epochs). For each task, the upper panel divides the training cost by the budget, on an axis that is linear below the budget and logarithmic above it (dotted line: the budget). The lower panel divides the training return by the task's best final return among the nine methods, so it can exceed one during training.

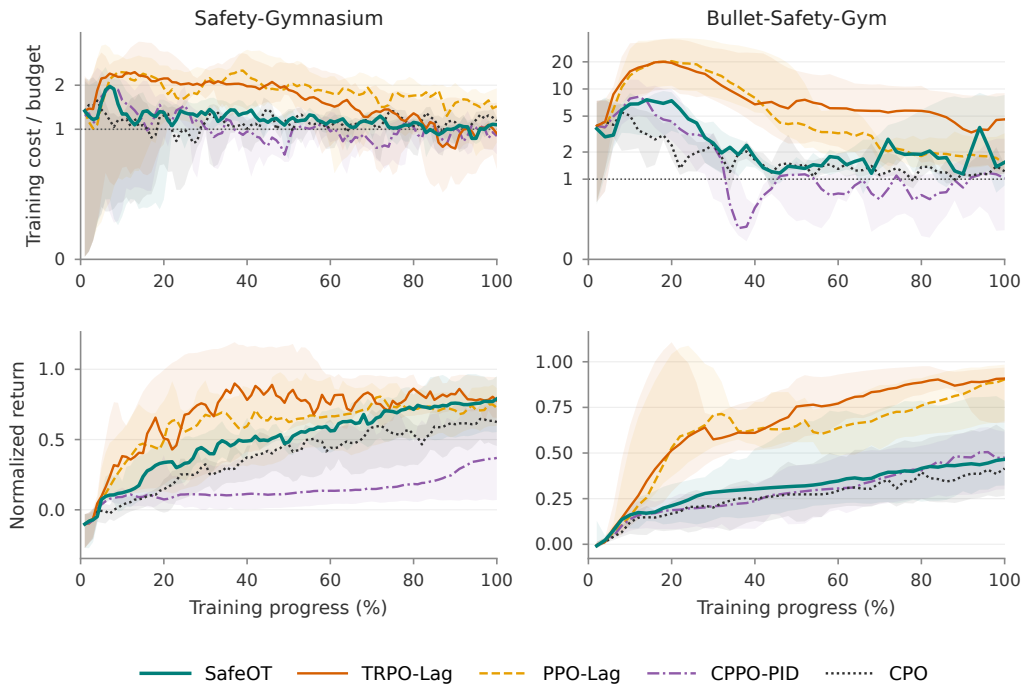

Figure 6: Five-seed suite learning curves of training cost/budget and normalized return (training return divided by the task's best final return among the nine methods) for SafeOT, TRPO-Lag, PPO-Lag, CPPO-PID and CPO on both benchmark suites. Lines are cross-task medians and bands are interquartile ranges. The cost axis is linear below the budget and logarithmic above it.

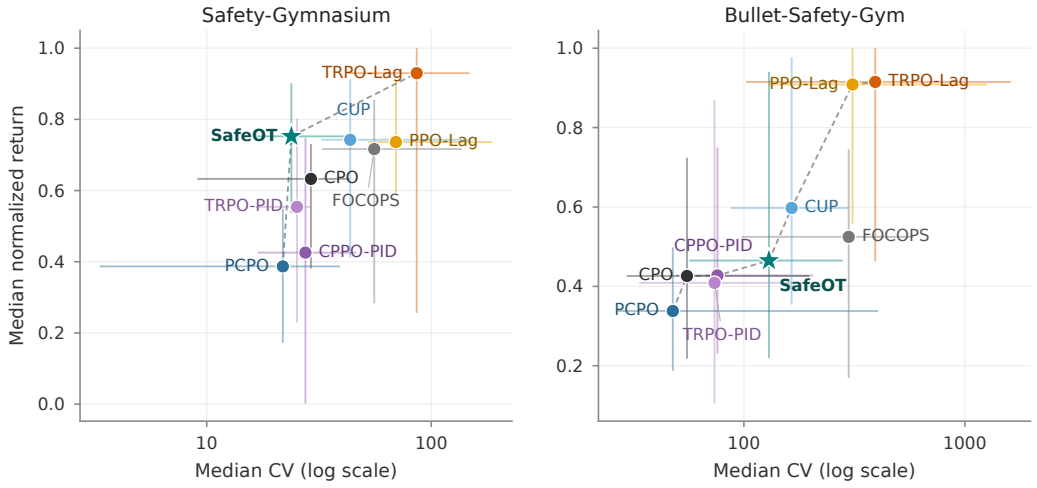

Figure 7: Median trade-off maps for ten Safety-Gymnasium and six Bullet-Safety-Gym tasks. Each method is placed at its task-median CV (logarithmic axis) and task-median final return, with each task's return divided by the best final return among the nine methods. Thin lines are paired task-bootstrap 95% percentile intervals (10,000 resamples), and the dashed line connects the Pareto-optimal methods. Farther left means less violation and higher means more return. These are descriptive medians, not the rank-based Pareto front of Figure 3.

Table 8: Per-task results on Bullet-Safety-Gym ($B = 10$, 1M steps, five seeds). Each cell lists final return, final cost and CV (top to bottom) as mean $\pm$ the half-width of a two-sided seed-t 95% interval. Final return and cost use the same evaluated policies. Intervals are descriptive, unadjusted and not truncated at zero. The reactive switch is an ablation that Table 2 does not rank.

Method	BallRun	BallCircle	CarRun	CarCircle	DroneRun	DroneCircle
SafeOT	579±3.0 149±63.6 164±53.4	509±29.1 4.5±3.7 21.5±3.5	626±182 78.2±184 241±193	461±237 50.3±35.4 91.7±38.3	1186±8.0 0.0±0.0 319±87.3	38.1±62.9 6.2±5.5 95.9±15.6
PPO-Lag	562±649 30.8±85.6 381±211	577±106 19.4±27.2 140±63.6	2108±1947 289±328 1582±942	1049±52.4 32.3±17.5 240±55.5	971±208 0.0±0.0 934±215	153±32.5 17.6±3.3 118±20.0
TRPO-Lag	1180±1058 92.4±157 554±540	530±39.6 2.8±3.1 85.5±25.4	3292±13.5 485±0.1 2185±43.6	957±41.6 51.8±24.2 232±37.7	670±845 50.1±136 1043±638	55.3±56.6 6.8±5.8 119±21.0
CPPO-PID	569±18.5 29.5±57.0 80.6±62.8	577±104 15.6±11.5 29.8±10.8	705±17.7 11.4±29.3 71.2±30.3	260±155 11.6±14.4 90.6±34.5	595±536 2.5±6.9 321±112	57.2±24.7 5.0±3.9 68.9±22.0
TRPO-PID	572±14.8 64.2±114 160±90.4	562±37.5 9.5±7.5 15.8±2.2	558±362 27.1±73.1 59.2±35.8	349±188 23.4±28.2 51.5±15.7	909±411 15.7±43.6 237±77.6	6.3±16.5 7.5±16.9 88.1±23.5
CPO	559±13.0 0.0±0.0 43.7±30.2	426±112 8.9±18.1 15.3±5.3	692±11.2 0.5±0.8 53.6±28.2	238±127 22.7±41.9 56.9±9.4	842±558 14.8±41.1 313±127	58.1±51.4 10.7±12.5 83.4±19.3
PCPO	545±29.2 0.0±0.0 48.4±28.3	309±67.9 6.9±5.0 14.7±5.3	655±38.7 0.0±0.0 46.9±23.9	184±129 36.6±46.4 39.3±19.0	516±332 10.5±29.2 710±593	36.9±42.9 7.5±7.8 99.3±44.9
FOCOPS	646±337 41.0±70.2 387±383	523±94.6 23.0±19.5 68.7±47.5	450±286 32.8±60.2 605±256	527±307 56.1±43.8 209±146	694±741 1.2±3.3 566±184	30.8±46.8 5.6±6.0 126±50.5
CUP	576±5.3 3.5±7.0 171±70.4	578±45.3 19.3±11.0 37.9±12.0	722±7.0 9.8±20.9 280±60.0	693±319 45.5±34.6 158±54.9	1131±58.1 0.0±0.0 316±64.6	82.0±57.1 15.9±9.8 137±37.1
Reactive switch	579±2.8 128±59.3 198±64.1	540±70.7 12.3±8.6 13.4±2.3	616±291 69.4±187 237±73.1	573±227 52.4±25.2 87.5±40.8	1182±11.1 3.5±9.8 323±81.9	38.1±56.5 5.6±3.1 97.0±18.6

$\bar{x} \pm t_{n-1,0.975} s/\sqrt{n}$. Return and cost use the same evaluated seeds, and CV uses all five training seeds. Every cell has $n = 5$.

Metric details. The training cost $\bar{J}(t)$ is the episodic cost logged at epoch t, a running mean over the last 50 training episodes. CV sums its excess over the budget across the T training epochs and divides the sum by B, so the unnormalized violation is $V = B \cdot \mathrm{CV}$. $\mathrm{CV} = 0$ means that the logged training cost never exceeded B. CV counts only cost in excess of the budget, because cost within the budget is permitted, and V has the form of the accumulated violation in Lemma 1, evaluated on the logged cost. Final policies are evaluated over 30 episodes. A method is Pareto-optimal on a task if no other method has a CV at least as low and a return at least as high, with at least one strict inequality. Across tasks with different return scales, we count Pareto-optimal tasks and average the per-task ranks. Appendix D.3 defines the per-task hypervolume and the beats/beaten counts of Table 2. The configuration of SafeOT was selected on PointGoal1, CarGoal1 and PointButton1, and on the seven other Safety-Gymnasium tasks it is Pareto-optimal on 6 of 7. All Safety-Gymnasium runs of Table 2 train for 2M steps, whereas the published Safety-Gymnasium results train for 10M, so returns on the velocity tasks are lower than published (Appendix D.8).

Computation of the reported metrics. Every metric is computed per task from the runs of each method and then aggregated over tasks. For method m on a task, CV_m, R_m and C_m are the means over seeds of its CV, final return and final cost.

- *CV* (Tables 2, 10 and 11), the cumulative constraint violation of the analyses of online constrained RL (Efroni et al., 2020) and a companion to the cost rate of Safety Gym (Ray et al., 2019) (Appendix D.5): for one run, $\mathrm{CV} = \frac{1}{B}\sum_{t=1}^{T}[\bar{J}(t) - B]_+$, where $\bar{J}(t)$ is the

Table 9: Per-task results on Safety-Gymnasium ($B = 25$, 2M steps, five training and five evaluated seeds). Each cell lists final return, final cost and CV (top to bottom) as mean ± the half-width of a two-sided seed-t 95% interval. Final return and cost use the same evaluated policies. Intervals are descriptive, unadjusted and not truncated at zero. Top: navigation tasks, PointGoal1 (PG1), CarGoal1 (CG1), PointButton1 (PB1), CarButton1 (CB1), PointPush1 (PP1) and CarPush1 (CP1). Bottom: velocity tasks. Feasibility in Tables 10 and 11 uses unrounded costs, so CPO on PG1 (25.03) exceeds B, and on Ant PPO-Lag and CPPO-PID coincide because their multipliers stay at zero.

Method	PG1	CG1	PB1	CB1	PP1	CP1
SafeOT	16.1±4.0 27.2±9.8 42.6±6.3	17.6±3.0 25.2±8.2 27.1±7.2	3.2±1.0 31.7±13.4 62.8±17.0	−4.5±3.5 48.8±59.2 162±20.9	0.7±0.3 16.6±31.4 24.3±6.3	0.5±0.2 16.2±16.9 21.6±4.3
PPO-Lag	16.6±6.1 57.7±15.0 130±26.0	20.3±3.0 45.3±13.2 138±44.1	6.0±2.0 74.3±22.5 245±45.7	3.2±1.4 119±29.6 441±75.6	0.4±0.3 25.1±39.0 60.2±22.9	0.5±0.1 20.6±8.7 53.6±11.0
TRPO-Lag	25.3±0.9 41.9±4.5 94.7±5.7	24.3±3.1 38.2±11.1 110±16.6	−0.4±13.9 41.7±21.8 202±50.6	−15.1±6.8 33.4±26.3 308±24.9	0.9±0.7 11.5±13.6 42.5±18.1	0.6±0.3 24.8±15.8 42.1±5.6
CPPO-PID	1.0±2.0 19.4±11.8 38.8±6.3	2.6±1.9 24.6±9.0 39.4±17.9	−0.7±0.6 19.1±7.2 53.6±18.3	−0.8±0.4 31.3±20.6 135±33.7	0.3±0.4 26.3±13.5 25.9±11.4	0.3±0.1 23.5±19.3 23.9±4.0
TRPO-PID	10.4±5.9 28.6±15.5 28.4±10.1	12.9±10.5 21.0±11.3 18.8±3.2	0.3±4.1 25.8±13.0 29.1±3.2	−6.1±7.8 33.4±23.5 62.9±23.9	0.3±0.3 37.0±40.9 25.1±5.9	0.4±0.2 14.8±13.3 25.3±4.7
CPO	9.6±7.9 25.0±10.8 31.5±11.0	14.2±7.5 28.9±8.5 28.1±6.9	2.6±2.0 28.4±10.5 54.9±8.5	−1.0±1.9 54.3±24.9 76.3±8.7	0.6±1.0 27.8±27.7 34.9±16.8	0.4±0.2 20.2±29.4 30.1±8.7
PCPO	11.4±4.6 36.0±16.0 31.8±9.5	11.3±3.1 32.2±8.3 34.9±5.3	1.4±3.3 36.7±5.4 46.5±8.6	−4.7±5.1 50.7±33.2 67.5±28.0	0.2±0.5 19.1±17.7 19.9±5.1	0.1±0.3 26.6±20.7 23.7±2.4
FOCOPS	20.6±6.0 39.4±18.0 92.0±23.8	21.7±10.1 30.1±8.2 82.9±17.4	0.9±7.5 34.1±6.6 191±36.7	−4.8±3.0 43.3±31.5 257±46.4	0.2±0.4 26.6±27.3 32.7±16.7	0.4±0.4 17.0±23.9 30.5±4.6
CUP	20.8±3.3 47.2±8.1 106±20.7	20.7±7.4 43.8±20.5 97.6±28.5	7.3±3.7 72.2±22.1 203±45.1	1.8±2.0 77.1±40.2 334±73.2	0.2±0.4 21.6±15.2 41.0±25.0	0.2±0.3 21.7±20.7 35.7±36.5

Method	Ant	HalfCheetah	Hopper	Walker2d
SafeOT	2945±72.7 15.5±9.6 11.9±4.2	2069±524 16.0±13.7 5.3±3.3	1628±60.9 2.1±3.2 23.4±3.5	1485±1392 23.4±26.1 19.3±5.5
PPO-Lag	1786±295 52.3±38.3 0.03±0.07	2614±190 0.6±1.5 78.5±23.1	980±874 21.3±34.3 60.7±19.5	1486±1391 27.5±33.1 44.7±16.3
TRPO-Lag	2909±104 10.2±7.0 82.2±24.4	2091±540 23.2±26.9 41.6±10.2	417±737 6.0±7.8 90.1±66.5	2231±1124 13.3±11.7 44.1±10.3
CPPO-PID	1786±295 52.3±38.3 0.03±0.07	2601±204 2.6±5.7 16.9±1.6	1564±177 0.7±1.3 29.1±5.2	1368±1318 13.9±15.3 14.3±5.5
TRPO-PID	2920±122 10.4±5.1 13.3±2.4	2072±528 38.2±48.0 19.7±4.1	1550±80.4 3.1±4.1 28.9±4.9	1477±1413 14.9±17.0 13.2±4.4
CPO	2619±150 4.8±10.2 9.1±4.8	1812±282 19.6±17.7 3.9±5.5	1117±634 0.0±0.0 26.1±8.4	1397±1392 15.2±18.0 4.7±2.2
PCPO	2203±243 61.1±93.6 0.08±0.2	1288±128 58.0±28.8 0.7±0.4	1031±609 34.2±77.6 7.6±4.1	830±952 9.5±13.8 3.3±3.3
FOCOPS	2506±338 263±166 6.1±7.9	2056±218 3.0±3.7 60.8±24.2	1070±748 9.3±22.5 50.7±25.1	2558±62.0 0.6±1.2 47.9±6.7
CUP	1948±261 42.0±25.4 0.2±0.5	2649±164 0.7±1.4 36.2±2.7	1342±370 16.6±41.2 46.0±14.9	1010±985 35.3±39.3 28.5±9.8

mean episodic cost of the last 50 training episodes at epoch t and T is the number of logged epochs.

- *Final return and final cost*: the mean episodic return and cost of the final policy over 30 evaluation episodes. *Cost rate*, as in Safety Gym: the total training cost divided by the total number of environment steps.
- *CV ratio* (Table 2): the median over tasks of $\mathrm{CV}_m/\mathrm{CV}_{SafeOT}$. Medians over tasks are a robust aggregate recommended for reporting deep RL results (Agarwal et al., 2021).
- *Return rank* (Table 2): on each task, the nine methods are ranked by R_m, with rank 1 for the highest return and tied methods sharing the mean of their ranks. The table reports the mean rank over tasks, the standard summary for comparing methods over several tasks (Demšar, 2006). The mean CV ranks of Figure 3 are computed in the same way from CV_m, with rank 1 for the lowest CV.
- *Pareto-optimal* (Tables 2, 10 and 11): no other method has a CV at most CV_m and a final return at least R_m, with at least one of the two strict (Zitzler et al., 2003).
- *Beats/beaten* (Table 2): m beats m' on a task if $\mathrm{CV}_m < \mathrm{CV}_{m'}$ and $R_m > R_{m'}$. The table sums over tasks the number of methods that m beats and the number of methods that beat m, the dominance counts of multi-objective comparison (Zitzler et al., 2003).
- *Hypervolume* (Tables 2 and 12), the indicator of Zitzler & Thiele (1999): on each task, let $v_m = \mathrm{CV}_m / \max_{m'} \mathrm{CV}_{m'}$ and $r_m = (R_m - \min_{m'} R_{m'})/(\max_{m'} R_{m'} - \min_{m'} R_{m'})$ over the nine methods. Then $\mathrm{HV}_m = (1 - v_m)\, r_m$ is the area of the unit square that the point (v_m, r_m) dominates with respect to the reference point $(1, 0)$. The table reports the mean over tasks.
- *Seed resamples* (Section 5.2 and Table 18): each resample draws, for every task and method, as many runs with replacement as the method has on that task and recomputes the metric. We report the share of resamples in which a statement holds.
- *Excess cost* (Table 3), the unnormalized violation summed over the K constraints: $\sum_{k=1}^{K} \sum_{t=1}^{T} [\bar{J}_k(t) - B_k]_+$, where $\bar{J}_k(t)$ is the episodic cost of joint k at epoch t and B_k its budget, averaged over seeds. *Satisfied constraints*: the number of joints whose final-policy cost is within its budget, averaged over seeds.
- *Significance* (Table 3): a two-sided Welch's t-test (Welch, 1947) between the five seed values of a method and those of the method with the best mean, at level 0.05.

Joint metrics per task. Table 12 gives the per-task values behind the joint columns of Table 2. On each task, CV is divided by the largest CV of the nine methods, and the final return is rescaled to $[0, 1]$ between the lowest and the highest final return of the nine methods. A method at (v, r) in these units dominates the rectangle $[v, 1] \times [0, r]$ of the unit square, whose corner $(1, 0)$ is the largest CV and the lowest return, and its hypervolume is the area $(1 - v)\, r$ of this rectangle (Zitzler & Thiele, 1999). It is 1 only at zero CV and the highest return, and 0 for the method with the largest CV or the lowest return. A method beats another on a task if it has both a lower CV and a higher final return. It is Pareto-optimal if no method has a CV at least as low and a return at least as high, with at least one strict inequality. Both follow from the per-task CV and final return of Tables 10 and 11. SafeOT has the largest hypervolume on seven of the ten Safety-Gymnasium tasks, more than any other method. On Bullet-Safety-Gym, no method has the largest on more than one task: on CarCircle, CUP (0.201) is slightly ahead of SafeOT (0.198), and the two tie at the printed precision.

D.4 Price Anticipation and Transient Violation

When the reduction occurs. Lemma 1 predicts that a Lagrangian multiplier rises only after violation has accumulated, whereas the safety price can rise before any violation (Section 3.3). On Safety-Gymnasium, the TRPO-Lag multiplier exceeds 0.05 a median of 4 epochs after the first violation and never before it, whereas the price of SafeOT does so no later than the first violation in the median run. TRPO-Lag therefore runs at about twice the budget for much of the first million steps, where most of its excess CV accrues (Figure 8).

Table 10: Final return / cumulative constraint violation (CV), five seeds. Teal: Pareto-optimal in return and CV among nine methods. †: final policy exceeds budget B. CV ratio: task-median CV(method)/CV(SafeOT). Bold: best CV ratio and most feasible tasks among the nine methods of this table and Table 11. Per-task confidence intervals: Tables 9 and 8. Table 2 summarizes these results, and Table 11 continues this table with the constrained policy optimization methods.

		Lagrangian		PID Lagrangian	
Task	**SafeOT**	TRPO-Lag	PPO-Lag	CPPO-PID	TRPO-PID
Safety-Gymnasium					
Ant	2945 / 11.9	2909 / 82.2	1786 / 0.03†	1786 / 0.03†	2920 / 13.3
CarButton1	−4.5 / 162†	−15.1 / 308†	3.2 / 441†	−0.8 / 135†	−6.1 / 62.9†
CarGoal1	17.6 / 27.1†	24.3 / 110†	20.3 / 138†	2.6 / 39.4	12.9 / 18.8
CarPush1	0.46 / 21.6	0.57 / 42.1	0.50 / 53.6	0.29 / 23.9	0.37 / 25.3
HalfCheetah	2069 / 5.3	2091 / 41.6	2614 / 78.5	2601 / 16.9	2072 / 19.7†
Hopper	1628 / 23.4	417 / 90.1	980 / 60.7	1564 / 29.1	1550 / 28.9
PointButton1	3.15 / 62.8†	−0.36 / 202†	5.96 / 245†	−0.75 / 53.6	0.35 / 29.1†
PointGoal1	16.1 / 42.6†	25.3 / 94.7†	16.6 / 130†	1.0 / 38.8	10.4 / 28.4†
PointPush1	0.73 / 24.3	0.87 / 42.5	0.41 / 60.2†	0.29 / 25.9†	0.35 / 25.1†
Walker2d	1485 / 19.3	2231 / 44.1	1486 / 44.7†	1368 / 14.3	1477 / 13.2
CV ratio	1× (ref.)	2.8×	2.7×	0.99×	0.86×
Feasible tasks	6/10	6/10	3/10	**7/10**	5/10
Bullet-Safety-Gym					
BallCircle	509 / 21.5	530 / 85.5	577 / 140†	577 / 29.8†	562 / 15.8
BallRun	579 / 164†	1180 / 554†	562 / 381†	569 / 80.6†	572 / 160†
CarCircle	461 / 91.7†	957 / 232†	1049 / 240†	260 / 90.6†	349 / 51.5†
CarRun	626 / 241†	3292 / 2185†	2108 / 1582†	705 / 71.2†	558 / 59.2†
DroneCircle	38 / 95.9	55 / 119	153 / 118†	57 / 68.9	6 / 88.1
DroneRun	1186 / 319	670 / 1043†	971 / 934	595 / 321	909 / 237†
CV ratio	1× (ref.)	3.3×	2.8×	0.85×	0.74×
Feasible tasks	3/6	2/6	1/6	2/6	2/6

Table 11: Per-task results, continued: SafeOT and the constrained policy optimization methods. Shading, † and bold as in Table 10.

		Constrained policy optimization			
Task	**SafeOT**	CPO	PCPO	FOCOPS	CUP
Safety-Gymnasium					
Ant	2945 / 11.9	2619 / 9.1	2203 / 0.08†	2506 / 6.1†	1948 / 0.2†
CarButton1	−4.5 / 162†	−1.0 / 76.3†	−4.7 / 67.5†	−4.8 / 257†	1.8 / 334†
CarGoal1	17.6 / 27.1†	14.2 / 28.1†	11.3 / 34.9†	21.7 / 82.9†	20.7 / 97.6†
CarPush1	0.46 / 21.6	0.45 / 30.1	0.07 / 23.7†	0.37 / 30.5	0.24 / 35.7
HalfCheetah	2069 / 5.3	1812 / 3.9	1288 / 0.7†	2056 / 60.8	2649 / 36.2
Hopper	1628 / 23.4	1117 / 26.1	1031 / 7.6†	1070 / 50.7	1342 / 46.0
PointButton1	3.15 / 62.8†	2.59 / 54.9†	1.35 / 46.5†	0.89 / 191†	7.28 / 203†
PointGoal1	16.1 / 42.6†	9.6 / 31.5†	11.4 / 31.8†	20.6 / 92.0†	20.8 / 106†
PointPush1	0.73 / 24.3	0.59 / 34.9†	0.19 / 19.9	0.25 / 32.7†	0.25 / 41.0
Walker2d	1485 / 19.3	1397 / 4.7	830 / 3.3	2558 / 47.9	1010 / 28.5†
CV ratio	1× (ref.)	0.82×	**0.58**×	2.2×	2.0×
Feasible tasks	6/10	5/10	2/10	4/10	4/10
Bullet-Safety-Gym					
BallCircle	509 / 21.5	426 / 15.3	309 / 14.7	523 / 68.7†	578 / 37.9†
BallRun	579 / 164†	559 / 43.7	545 / 48.4	646 / 387†	576 / 171
CarCircle	461 / 91.7†	238 / 56.9†	184 / 39.3†	527 / 209†	693 / 158†
CarRun	626 / 241†	692 / 53.6	655 / 46.9	450 / 605†	722 / 280
DroneCircle	38 / 95.9	58 / 83.4†	37 / 99.3	31 / 126	82 / 137†
DroneRun	1186 / 319	842 / 313†	516 / 710†	694 / 566	1131 / 316
CV ratio	1× (ref.)	0.67×	**0.56**×	2.3×	1.3×
Feasible tasks	3/6	3/6	**4/6**	2/6	3/6

Table 12: Hypervolume on each task (five seeds): the area of the unit violation–return square that a method's point dominates, with CV divided by the largest CV and final return rescaled to $[0, 1]$ over the nine methods (Appendix D.3). Bold: largest in a row at the printed precision. The last three rows of each benchmark are the joint columns of Table 2.

Task	SafeOT	TRPO-Lag	PPO-Lag	CPPO-PID	TRPO-PID	CPO	PCPO	FOCOPS	CUP
Safety-Gymnasium									
Ant	**0.86**	0.00	0.00	0.00	0.82	0.64	0.36	0.57	0.14
CarButton1	0.37	0.00	0.00	0.54	0.42	**0.63**	0.48	0.24	0.22
CarGoal1	**0.56**	0.21	0.00	0.00	0.41	0.43	0.30	0.35	0.25
CarPush1	**0.46**	0.21	0.00	0.25	0.32	0.33	0.00	0.26	0.11
HalfCheetah	0.54	0.28	0.00	**0.76**	0.43	0.37	0.00	0.13	0.54
Hopper	**0.74**	0.00	0.15	0.64	0.64	0.41	0.46	0.24	0.37
PointButton1	**0.36**	0.01	0.00	0.00	0.12	0.32	0.21	0.04	0.17
PointGoal1	**0.42**	0.27	0.00	0.00	0.30	0.27	0.32	0.23	0.15
PointPush1	**0.47**	0.29	0.00	0.08	0.13	0.25	0.00	0.04	0.03
Walker2d	0.23	0.06	0.03	0.22	0.27	**0.30**	0.00	0.00	0.04
Mean	**0.50**	0.13	0.02	0.25	0.39	0.39	0.21	0.21	0.20
Beats/beaten	**30/2**	7/23	1/20	14/15	19/5	17/5	10/4	8/22	9/19
Pareto-optimal	**9/10**	5/10	2/10	3/10	5/10	5/10	8/10	4/10	3/10
Bullet-Safety-Gym									
BallCircle	0.63	0.32	0.00	0.78	**0.83**	0.39	0.00	0.41	0.73
BallRun	0.04	0.00	0.01	0.03	0.03	0.02	0.00	**0.05**	0.03
CarCircle	**0.20**	0.03	0.00	0.05	0.15	0.05	0.00	0.05	**0.20**
CarRun	0.06	0.00	**0.16**	0.09	0.04	0.08	0.07	0.00	0.08
DroneCircle	0.06	0.04	0.13	**0.17**	0.00	0.14	0.06	0.01	0.00
DroneRun	**0.69**	0.00	0.07	0.08	0.45	0.34	0.00	0.12	0.64
Mean	**0.28**	0.07	0.06	0.20	0.25	0.17	0.02	0.11	**0.28**
Beats/beaten	10/6	1/12	4/8	12/5	12/4	**13/2**	4/10	2/20	11/2
Pareto-optimal	3/6	3/6	3/6	**4/6**	**4/6**	**4/6**	3/6	1/6	**4/6**

Runs and definitions. Tables 13 and 14 show when the price becomes active and how the violation is spread over training. They use the five runs per task of Table 2 for SafeOT and TRPO-Lag (50 runs each), five seeds per task for the reactive switch, and the CPO runs of Table 2 in Table 14. A price counts as active above 0.05, and the first violation is the first epoch in which the mean cost of the last 50 training episodes exceeds the budget. Twenty-two navigation runs violate in their first logged epoch, so the lead time is informative mainly on the velocity tasks. Summed over the ten tasks, 72% of the excess violation of TRPO-Lag over SafeOT accrues in epochs 1–50, the first million steps (Table 14).

Table 13: Price timing on the ten Safety-Gymnasium tasks (five runs per method). Before: runs whose price is active in an epoch before the first epoch in which the logged training cost exceeds B. Onset: among violation onsets after the first logged epoch, the fraction whose preceding epoch had an active price. The reactive switch is triggered by the cost of the current batch, which can exceed B before the logged 50-episode mean does, so its Before counts show this lead rather than anticipation.

	SafeOT		Reactive switch		TRPO-Lag	
Task	Before	Onset	Before	Onset	Before	Onset
Ant	2/5	90%	4/5	97%	0/5	55%
CarButton1	0/5	100%	0/5	100%	0/5	100%
CarGoal1	2/5	85%	1/5	97%	0/5	0%
CarPush1	1/5	80%	1/5	92%	0/5	96%
HalfCheetah	2/5	76%	5/5	98%	0/5	29%
Hopper	0/5	80%	0/5	81%	0/5	0%
PointButton1	0/5	96%	0/5	92%	0/5	100%
PointGoal1	0/5	87%	0/5	92%	0/5	0%
PointPush1	1/5	74%	1/5	90%	0/5	88%
Walker2d	0/5	45%	1/5	93%	0/5	76%

Table 14: Unnormalized violation V of SafeOT / TRPO-Lag / CPO by epoch window on the ten Safety-Gymnasium tasks, as seed means over the five runs per task of Table 2 (2×10^4 steps per epoch). The three windows add up to $B \cdot \mathrm{CV}$ in Table 9.

Task	Epochs 1–10	Epochs 11–50	Epochs 51–100
Ant	0 / 0 / 0	22.6 / 405 / 21.6	274 / 1650 / 205
CarButton1	601 / 1070 / 494	2360 / 4870 / 669	1100 / 1770 / 745
CarGoal1	199 / 374 / 117	328 / 1280 / 316	151 / 1090 / 269
CarPush1	126 / 264 / 131	240 / 617 / 220	175 / 172 / 402
HalfCheetah	0 / 0 / 0	44.5 / 966 / 27.1	87.1 / 74.4 / 70.3
Hopper	49.1 / 246 / 25.7	271 / 1610 / 273	265 / 392 / 355
PointButton1	436 / 742 / 368	900 / 3060 / 541	235 / 1240 / 463
PointGoal1	395 / 434 / 218	399 / 1010 / 305	271 / 927 / 265
PointPush1	197 / 225 / 145	253 / 773 / 358	156 / 63.5 / 370
Walker2d	0 / 0 / 0	333 / 995 / 48.5	148 / 107 / 68.4

D.5 Standard Safety Gym Metrics

Safety Gym evaluates safe exploration by three metrics (Ray et al., 2019): the return of the final policy, its cost against the budget, and the cost rate, the total training cost divided by the total number of environment steps. Unlike CV, the cost rate also counts cost within the budget. Table 15 ranks the nine methods of Table 2 by these metrics, and Table 16 gives cost-rate ratios. On Safety-Gymnasium, SafeOT has the best mean rank in final return and the second best in final cost, and it is Pareto-optimal in final cost and return on six of the ten tasks, more than any baseline. Its cost rate is lower than that of TRPO-Lag on 9/10 Safety-Gymnasium and 6/6 Bullet-Safety-Gym tasks. PCPO, CPO and the PID Lagrangian methods have a lower cost rate than SafeOT on six to eight of the ten Safety-Gymnasium tasks and worse mean return ranks. Under the cost rate, SafeOT is Pareto-optimal in cost rate and final return on 7/10 Safety-Gymnasium tasks, against 9/10 under CV, and SafeOT lies on the Pareto front of the mean ranks in cost rate and return on both suites. On Bullet-Safety-Gym, SafeOT is in the middle of the nine methods under all three metrics.

Table 15: Standard Safety Gym metrics (Ray et al., 2019) for the methods of Table 2: mean per-task ranks among the nine methods (1 is best) of final return, final cost and training cost rate, the number of tasks whose final policy meets the budget, and the number of tasks on which the method is Pareto-optimal in final cost and return. Best value per column in bold.

Method	Return rank	Final-cost rank	Cost-rate rank	Feasible	Pareto (final)
Safety-Gymnasium (10 tasks)					
SafeOT	**3.5**	3.8	4.4	6/10	**6/10**
PPO-Lag	3.9	7.0	8.3	3/10	3/10
TRPO-Lag	3.9	5.0	7.2	6/10	5/10
CPPO-PID	6.4	**3.4**	3.6	**7/10**	4/10
TRPO-PID	5.5	3.9	2.8	5/10	4/10
CPO	5.2	4.2	4.0	5/10	4/10
PCPO	7.4	6.3	**2.6**	2/10	0/10
FOCOPS	5.0	5.0	5.3	4/10	3/10
CUP	4.3	6.5	6.7	4/10	3/10
Bullet-Safety-Gym (6 tasks)					
SafeOT	4.8	5.0	5.0	3/6	1/6
PPO-Lag	2.8	6.0	7.8	1/6	3/6
TRPO-Lag	3.5	6.5	8.3	2/6	3/6
CPPO-PID	5.2	**3.5**	3.5	2/6	3/6
TRPO-PID	6.0	5.5	**2.2**	2/6	2/6
CPO	6.2	3.9	2.3	3/6	3/6
PCPO	8.2	3.8	2.8	**4/6**	1/6
FOCOPS	5.8	6.0	7.2	2/6	1/6
CUP	**2.5**	4.8	5.8	3/6	**4/6**

Table 16: Cost rate (Ray et al., 2019): total training cost divided by total environment steps (lower is better). On Safety-Gymnasium, the total training cost is approximated from the logged mean cost of the last 50 episodes. On Bullet-Safety-Gym, it uses exact episode totals. Ratios are task medians relative to SafeOT (above $1\times$: more cost). SafeOT lower counts tasks with lower cost rate. Ranks average per-task ranks among nine methods. –: not applicable to the reference.

Suite	Method	Cost-rate ratio	SafeOT lower	Mean cost-rate rank
Safety-Gymnasium	SafeOT	$1\times$ (ref.)	–	4.4
Safety-Gymnasium	TRPO-Lag	$1.5\times$	9/10	7.2
Safety-Gymnasium	PPO-Lag	$1.7\times$	9/10	8.3
Safety-Gymnasium	CPPO-PID	$0.93\times$	4/10	3.6
Safety-Gymnasium	TRPO-PID	$0.95\times$	2/10	2.8
Safety-Gymnasium	CPO	$0.96\times$	3/10	4.0
Safety-Gymnasium	PCPO	$0.9\times$	2/10	2.6
Safety-Gymnasium	FOCOPS	$1.4\times$	8/10	5.3
Safety-Gymnasium	CUP	$1.3\times$	9/10	6.7
Bullet-Safety-Gym	SafeOT	$1\times$ (ref.)	–	5.0
Bullet-Safety-Gym	TRPO-Lag	$2.4\times$	6/6	8.3
Bullet-Safety-Gym	PPO-Lag	$2.4\times$	6/6	7.8
Bullet-Safety-Gym	CPPO-PID	$0.72\times$	1/6	3.5
Bullet-Safety-Gym	TRPO-PID	$0.77\times$	0/6	2.2
Bullet-Safety-Gym	CPO	$0.79\times$	0/6	2.3
Bullet-Safety-Gym	PCPO	$0.74\times$	1/6	2.8
Bullet-Safety-Gym	FOCOPS	$1.9\times$	6/6	7.2
Bullet-Safety-Gym	CUP	$1.1\times$	4/6	5.8

D.6 Tuning the Lagrangian Baseline

The conservative setting of SafeOT was selected on three development tasks (PointGoal1, CarGoal1 and PointButton1) by a selection rule fixed before its runs (Configuration search, below), whereas the default configuration had been chosen earlier on the same tasks by sweeps of ε and the clip level. To give TRPO-Lag a comparable search, we applied the same rule to its multiplier step, trained at one, three and ten times the SafePO default on all ten tasks with five seeds each. The $1\times$ step reproduces the TRPO-Lag results of Table 2. The rule selects the ten-fold step, which lowers the CV of TRPO-Lag on the development tasks to 0.40 times its default at a mean normalized return change of $+0.03$. The three-fold step fails on return. Applied to SafeOT, the same rule selects the conservative setting, unclipped with a halved observed-cost feedback step.

Table 17 compares the tuned TRPO-Lag with the default and the conservative setting of SafeOT. The tuned step removes much of the disadvantage of TRPO-Lag: the default SafeOT keeps the lower CV on seven of the ten tasks (median ratio 1.59, against 2.75 for the default TRPO-Lag) and loses on CarButton1, PointButton1 and PointGoal1. With both methods tuned, SafeOT has the lower CV on all ten tasks (median ratio 1.55), and it is Pareto-optimal on all ten tasks, the only method to be so, with the best mean return rank. The two tuned methods have about the same training cost rate: the conservative setting is lower on six of the ten tasks (median cost-rate ratio SafeOT/TRPO-Lag 0.96). Against a tuned Lagrangian baseline, the advantage of SafeOT is therefore that less of its training cost exceeds the budget, not that its total cost is lower.

The median training-cost curves in Figure 8 show how much less of the training trajectory lies above the budget for SafeOT than for TRPO-Lag.

Figure 9 displays the default and nine other measured configurations of SafeOT and the three TRPO-Lag multiplier steps on the same CV–return axes.

Table 18 repeats the main comparisons on five further seeds (5–9) and reports bootstrap checks. Pooling seeds 0–9 of all nine methods, SafeOT has the largest hypervolume (0.49, against 0.44 for CPO and 0.43 for TRPO-PID), which stays the largest in 90% of seed resamples, and the best mean return rank (3.8, against 4.0 for TRPO-Lag). No method has both a better mean CV rank and a better mean return rank. SafeOT also ranks better than TRPO-Lag in final return, final cost and cost rate

Table 17: SafeOT against TRPO-Lag with its multiplier step tuned by the same pre-registered rule (ten Safety-Gymnasium tasks, five seeds). CV ratio: median over tasks of CV(TRPO-Lag) divided by CV(SafeOT). Pareto and return rank: entries of SafeOT when the listed rows take the place of the defaults among the nine methods of Table 2. Lower CV and lower cost rate: tasks on which SafeOT has the lower CV or the lower training cost rate.

SafeOT	TRPO-Lag	Lower CV	CV ratio	Pareto	Return rank	Lower cost rate
Default	Default	10/10	2.75×	9/10	3.5	9/10
Default	Tuned (10× step)	7/10	1.59×	8/10	3.4	3/10
Conservative	Tuned (10× step)	10/10	1.55×	10/10	3.6	6/10

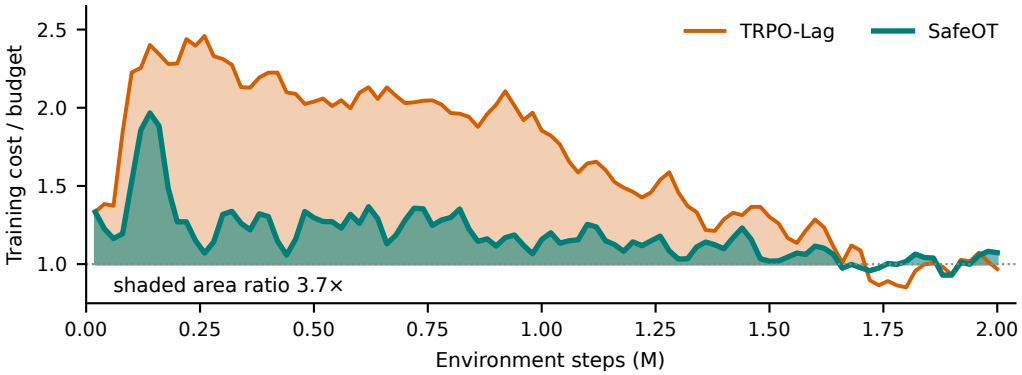

Figure 8: Median training cost relative to the budget across ten Safety-Gymnasium tasks, with the excess over the budget shaded. The ratio of the two shaded areas is 3.7×, computed from these plotted curves. It is distinct from the median of per-task CV ratios.

(3.8, 3.8 and 4.2, against 4.0, 4.7 and 7.2) and is Pareto-optimal in final cost and return on seven of the ten tasks, more than any baseline.

Configuration search. After the main experiments, a pre-registered search tested three changes to the default configuration of SafeOT with the protocol of Table 2 (all sixteen tasks, seeds 0–4): unclipped SafeOT, a halved observed-cost feedback step, and both changes together. On Bullet-Safety-Gym, the unclipped setting reuses 30 earlier runs of the same configuration. Selection used only the three development tasks on which the default had been chosen (PointGoal1, CarGoal1 and PointButton1). Relative to the default, a candidate passed if its median CV ratio was at most 0.85 with a mean normalized return change of at least -0.02, or if its mean normalized return change was at least $+0.05$ with a median CV ratio of at most 1.05. The return change on a task is divided by the larger of the default's absolute return and the spread of the nine methods' returns. Only the combination passed, with a median CV ratio of 0.66 and a mean return change of -0.015. Unclipped SafeOT alone lowered the CV further (0.59) but lost return (-0.13), and the halved feedback step alone raised the CV (1.61) while gaining return ($+0.13$). Each in place of default SafeOT, the three candidates are Pareto-optimal against the eight baselines on 8, 10 and 9 of the ten Safety-Gymnasium tasks (unclipped, both changes, halved step), against 9 for the default, and each is Pareto-optimal on three of the six Bullet-Safety-Gym tasks, as is the default. On the seven Safety-Gymnasium tasks not used for selection, the selected setting is Pareto-optimal on all seven (default: six), with a median CV 0.24 times that of TRPO-Lag (default: 0.44). It trades return for violation. Over the ten Safety-Gymnasium tasks, its median final return divided by the best method mean on each task is 0.66 against 0.75, and its final policy meets the budget on five tasks against six. On Bullet-Safety-Gym, its median final return divided by the best method mean is 0.42, against 0.47 for the default. With seeds 0–4, its CV is lower than that of CPO on nine of the ten Safety-Gymnasium tasks (Table 37), and its final return is higher on seven, all except CarButton1, CarGoal1 and PointPush1. On five further seeds (5–9), the conservative setting is Pareto-optimal on all ten Safety-Gymnasium tasks against all eight baselines (default: six), and its CV is lower than that of TRPO-Lag on all ten (median ratio

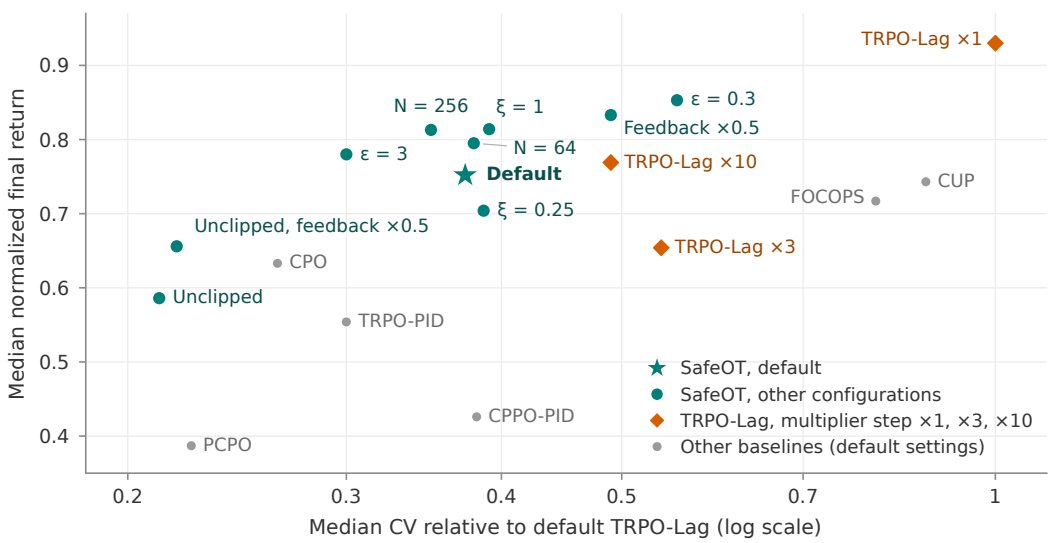

Figure 9: Configuration search versus multiplier-step search on ten Safety-Gymnasium tasks. Each point is a measured configuration. The axes show the median CV relative to default TRPO-Lag (logarithmic scale) and the median over tasks of the final return divided by the best method mean on each task. Teal points: the default SafeOT (star) and nine other measured configurations, labeled by their change from the default: the six single-parameter variants of Appendix H.1 (N: number of clusters, ξ: smoothing weight, ε: entropic regularization) and the three candidates of the configuration search. Orange diamonds: TRPO-Lag at $1\times$, $3\times$, and $10\times$ its multiplier step. Gray points: constrained policy optimization and PID Lagrangian baselines at default settings. Connecting or interpolated frontiers are not asserted.

Table 18: Independent-seed and bootstrap checks on Safety-Gymnasium. The second set of runs uses seeds 5–9 of all nine methods. Ratios after the counts are medians over tasks of CV(SafeOT)/CV(TRPO-Lag), the inverse orientation of the CV ratio of Table 2. Matched-CV counts ask whether SafeOT reaches a comparator's final training return by that comparator's final CV. Matched-final-cost counts use the four-budget threshold-majority rule of Appendix E.2 and all available seeds. Bootstrap entries give the share preserving a rank statement or achieving at least six matched-cost tasks. –: not evaluated.

Check	Seeds 0–4	Seeds 5–9	Bootstrap
Lower CV than TRPO-Lag	10/10, 0.37×	10/10, 0.33×	–
Lower cost rate than TRPO-Lag	9/10	10/10	–
On the Pareto front of mean ranks (CV, return)	Yes	Yes	99.9%
On the Pareto front of mean ranks (final cost, return)	Yes	Yes	97.0%
Largest hypervolume (CV, final return)	Yes	Yes	98.9%
Matched CV: CPO	7/10	7/10	–
Matched CV: CPPO-PID	8/10	9/10	–
Matched CV: PCPO	8/10	7/10	–
Matched CV: TRPO-PID	7/10	–	–
Matched final cost: CPO	8/10 (all seeds)		98.9% $\geq 6/10$
Matched final cost: CPPO-PID	7/10 (all seeds)		93.9% $\geq 6/10$

0.26), than that of CPO on eight and than that of TRPO-PID on six. All other studies use the default configuration.

D.7 Six Harder Safety-Gymnasium Tasks

The extension (Table 19) adds PointGoal2, CarGoal2, PointButton2, CarButton2, Swimmer and Humanoid at $B = 25$, 2M steps and five seeds. All 270 runs and 8,100 final-policy episodes are complete. SafeOT has a lower task-mean CV than TRPO-Lag on all six tasks (median CV ratio TRPO-Lag/SafeOT 2.47). CPO and both PID Lagrangian methods have a lower CV on the four navigation tasks. In final return and CV, SafeOT is Pareto-optimal among the nine methods on three of the six tasks. It is dominated on PointButton2 and CarButton2, and on Swimmer only by the reactive switch. These task-mean comparisons are descriptive, not a superiority test.

D.8 Comparison with Published Benchmark Results

Tables 6 and 20 compare our baseline results with published results on Bullet-Safety-Gym and Safety-Gymnasium. The protocols differ, so these are contextual checks rather than replications. The published Safety-Gymnasium results train for 10M steps, five times as long as our runs. J reports 10 assessment iterations, and our final return and cost are means over 30 deterministic evaluation episodes of the final policy. On the four velocity tasks, our return is below the published one in 58 of 60 comparisons, consistent with the shorter training: at 10M steps (Table 23), CPO comes within 1% of the published returns on Ant, HalfCheetah and Hopper (3085, 2739 and 1705 against 3099–3117, 2738–2787 and 1713–1714). In 19 of these 60 comparisons our final cost exceeds B while the published one does not, most strongly for FOCOPS on Ant (262.7 against 15.1 and 14.8). Its training CV there is only 6.1 (Table 11), so the excess appears in the deterministic final policy rather than during training. On the six navigation tasks, our return is below the published one in 44 of 74 comparisons and above it in 28, 23 of them at a higher cost (two are equal at the printed precision), consistent with costs that have not yet settled at 2M steps. Deterministic evaluation can also differ from the training statistics: TRPO-Lag on CarButton1 has a training return and cost of 1.2 and 46.6

Table 19: Per-task results on six harder Safety-Gymnasium tasks ($B = 25$, 2M steps, five seeds). Cells as in Table 9. The reactive switch replaces PCPO. On Humanoid, CPPO-PID and PPO-Lag coincide on four seeds, and TRPO-PID and TRPO-Lag on two. On CarGoal2, the unrounded final return of SafeOT (3.864) exceeds that of CPO (3.862).

Method	CarButton2	CarGoal2	Humanoid	PointButton2	PointGoal2	Swimmer
SafeOT	−5.5±7.3	3.9±1.5	5827±221	0.2±5.6	3.0±2.0	59.9±61.6
	44.1±16.1	32.9±15.3	0.0±0.0	40.6±27.5	23.7±23.4	16.6±9.4
	176±37.2	94.2±17.5	0.3±0.5	82.7±15.1	92.5±17.8	53.8±24.7
PPO-Lag	3.8±0.7	5.8±1.5	4000±2004	5.2±1.8	4.2±1.8	83.4±35.5
	141±46.5	74.0±25.7	38.2±44.1	87.2±16.0	61.5±15.7	65.6±35.1
	495±54.0	340±41.8	0.7±1.3	293±39.9	329±61.5	235±141
TRPO-Lag	−13.0±7.8	2.9±1.5	5888±157	1.2±4.7	4.6±2.4	49.8±2.1
	41.3±9.2	37.5±17.7	0.0±0.0	37.4±9.2	34.9±24.4	37.6±5.3
	319±17.2	242±22.7	5.6±5.9	193±34.1	270±52.5	128±16.5
CPPO-PID	−1.3±0.7	−0.2±0.6	3988±1985	−0.8±0.5	−0.4±0.3	129±63.8
	42.9±15.9	35.9±22.8	38.0±44.4	33.4±16.6	35.9±24.7	10.7±12.2
	162±55.4	57.9±25.0	0.7±1.3	70.9±25.9	62.6±16.6	57.7±6.7
TRPO-PID	−2.9±1.8	0.6±1.9	5892±184	−1.0±0.7	0.2±1.3	54.3±57.9
	43.8±41.3	22.1±16.8	1.9±5.1	17.2±13.5	32.1±17.4	14.5±9.2
	60.2±14.4	38.9±14.4	3.5±3.3	36.4±3.7	41.2±20.1	57.5±4.3
CPO	−3.0±4.9	3.9±1.8	5641±261	0.5±2.6	2.0±2.3	34.6±3.6
	47.8±61.9	45.6±36.7	1.3±2.3	31.7±12.9	29.9±26.4	25.7±6.6
	94.1±14.4	61.1±9.0	0.9±0.9	58.5±15.7	63.9±12.7	64.8±5.9
FOCOPS	−6.1±4.6	3.5±3.2	1123±352	2.2±3.6	3.4±2.9	47.2±6.9
	42.5±17.5	37.4±22.9	12.4±22.8	50.4±10.7	30.0±19.2	30.9±6.4
	293±24.4	209±40.9	0.0±0.0	202±37.0	186±43.6	91.8±17.9
CUP	2.3±2.5	7.2±1.9	1441±558	4.9±1.4	8.6±2.1	41.0±4.2
	87.9±30.6	78.8±15.4	3.1±4.6	72.2±26.0	122±43.5	28.0±7.6
	364±94.5	239±62.6	0.0±0.0	212±39.4	321±113	69.9±7.3
Reactive switch	−3.8±2.6	1.9±2.2	5775±216	1.7±1.8	2.0±0.9	63.6±59.8
	33.9±18.2	27.4±12.1	0.1±0.3	31.6±11.5	22.4±12.9	15.8±6.9
	173±47.3	80.9±9.9	0.3±0.4	80.8±16.2	87.9±24.3	51.1±17.6

but evaluation values of -15.1 and 33.4. The low Hopper return of TRPO-Lag (417) comes from one strong and four weak seeds.

E Return–Safety Trade-offs and Deployment

The primary Safety-Gymnasium benchmark fixes ten tasks and five seeds. The budget scan below adds budgets and seeds on those tasks and does not replace the primary benchmark.

E.1 Operating Points at Four Budget Levels

Run selection and policy identity. The budget scan contains 1450 evaluated runs: SafeOT, TRPO-Lag, CPO and CPPO-PID at budgets $5, 12.5, 25, 50$ and five additional comparators at 25. Each run contains 100 training epochs (2M steps) and 30 deterministic final-policy episodes. Seeds are 0–4, expanded to 0–9 at 25 where available. In this scan, the TRPO-PID control has five seeds at $B = 25$, and Table 18 adds its seeds 5–9. For the cost-matched comparisons of Appendix E.2, PCPO and TRPO-PID also run at budgets 5, 12.5 and 50 with seeds 0–2 (180 runs). The earliest complete and evaluated attempt is selected per task, method, budget and seed. Final return and final cost always describe the same policy.

Aggregation and uncertainty. For each task–budget pair, we compare mean CV and final return. The aggregate is the median over pairs of CV(SafeOT)/CV(baseline), the inverse orientation of the CV ratio in Table 2, or of the normalized return difference. Return differences are divided by the larger of the baseline's absolute mean and the task's return range across SafeOT and the seven baselines other than TRPO-PID. This range stays fixed during resampling. A 2000-replicate bootstrap samples tasks and independently resamples seeds within each method and task–budget pair. All budgets of a sampled task remain together. These are unadjusted descriptive 95% intervals. A zero baseline mean CV excludes that task–budget pair only from the CV ratios.

Against TRPO-Lag, SafeOT has the lower mean CV on all ten tasks at each of the four budgets. The median of CV(SafeOT)/CV(TRPO-Lag) is 0.55, 0.46, 0.35 and 0.41 at $B = 5$, 12.5, 25 and 50, and 0.47 ($[0.27, 0.55]$) over the 40 task–budget pairs. Across budgets, CPO occupies a different trade-off: the median of CV(SafeOT)/CV(CPO) is 1.56 ($[1.29, 2.16]$), with normalized final-return difference 9.3% ($[2.8\%, 14.9\%]$). Equal training budgets do not imply equal final costs. The following analysis compares operating costs directly.

Return at equal budgets. Across four budgets, the median SafeOT–TRPO-Lag normalized final-return difference is -0.7% ($[-6.4\%, +10.2\%]$, 40 task–budget cells, 250 policies per method). At $B = 25$ alone, ten seeds per method on ten tasks give a median normalized return difference of -0.9% ($[-20.2\%, +14.1\%]$), with TRPO-PID instead of PCPO in the return range. Figure 7 instead shows five-seed task medians of the return divided by the best method mean. Neither interval establishes equivalence or non-inferiority, and on Bullet-Safety-Gym the adjusted tests of Appendix C.2 establish neither return non-inferiority nor return inferiority against TRPO-Lag.

E.2 Cost-Matched Comparisons

At each of the four budgets, SafeOT has a lower mean CV than TRPO-Lag on all ten tasks (Appendix E.1). Table 18 gives the matched-violation counts: SafeOT reaches the final training return of CPO, CPPO-PID, PCPO and TRPO-PID by their final CV on seven or eight of the ten tasks with seeds 0–4, and on seven to nine with seeds 5–9. The comparisons below match final costs instead.

Operating-point interpolation. For each task, retain method–budget means with at least three evaluated seeds, pairing return and cost from the same policies. Remove dominated baseline points and interpolate the remaining return–cost front linearly. Query at each eligible SafeOT operating cost only within that non-dominated frontier's cost range. No constant extension beyond the last non-dominated point or extrapolation is used. Differences within 3% of the task's range of mean returns across SafeOT, TRPO-Lag, CPO and CPPO-PID are called close. Table 21 reports the 40 candidate points and the outside-range counts. Interpolated values are analytical references, not evaluated policies, and queries within a task are dependent.

Table 20: Published counterparts on Safety-Gymnasium: final return/cost. T: ours (2M steps, five seeds, 30 deterministic evaluation episodes). J: Ji et al. (2023), Table 5(a). O: OmniSafe, Table 2 (OmniSafe Team, 2024). J and O train for 10M steps with the cost limit 25. –: not reported by the source. OmniSafe reports no CarPush1 or PointPush1 results. On Ant, our PPO-Lag and CPPO-PID runs coincide because their multipliers stay at zero.

Task	Source	TRPO-Lag	PPO-Lag	CPPO-PID	TRPO-PID
Ant	T	2908.9/10.2	1785.7/52.3	1785.7/52.3	2920.1/10.4
	J	3157.4/3.6	3221.9/5.4	3070.7/10.2	–
	O	3041.9/19.5	3261.9/12.1	3213.4/14.3	3052.9/15.2
CarButton1	T	−15.1/33.4	3.2/118.9	−0.8/31.3	−6.1/33.4
	J	−0.7/26.9	0.1/11.9	−1.7/10.0	–
	O	−0.4/37.5	0.3/55.5	−1.4/14.6	−0.3/15.2
CarGoal1	T	24.3/38.2	20.3/45.3	2.6/24.6	12.9/21.0
	J	27.5/27.3	14.6/9.8	1.0/61.7	–
	O	27.0/26.8	13.3/21.7	10.6/30.7	25.5/28.9
CarPush1	T	0.6/24.8	0.5/20.6	0.3/23.5	0.4/14.8
	J	1.5/17.6	0.5/19.6	0.0/11.8	–
HalfCheetah	T	2091.3/23.2	2614.5/0.6	2600.8/2.6	2071.6/38.2
	J	2952.1/25.2	3025.4/0.0	3336.8/1.1	–
	O	2884.7/9.0	2946.2/3.4	2837.9/8.1	2796.8/11.2
Hopper	T	417.2/6.0	979.6/21.3	1564.0/0.7	1550.5/3.1
	J	1377.9/17.7	1348.0/22.3	1709.1/11.1	–
	O	1391.8/11.2	961.9/14.0	1713.3/9.0	1178.6/18.8
PointButton1	T	−0.4/41.7	6.0/74.3	−0.7/19.1	0.3/25.8
	J	7.1/32.3	5.8/33.0	−0.1/20.9	–
	O	6.9/31.2	4.6/20.8	1.2/29.4	6.4/27.9
PointGoal1	T	25.3/41.9	16.6/57.7	1.0/19.4	10.4/28.6
	J	19.0/22.9	12.5/37.6	8.2/26.3	–
	O	18.5/23.0	13.0/25.8	8.4/25.7	19.2/21.4
PointPush1	T	0.9/11.5	0.4/25.1	0.3/26.3	0.3/37.0
	J	0.7/24.9	0.8/33.2	0.3/8.9	–
Walker2d	T	2230.6/13.3	1485.9/27.5	1368.1/13.9	1477.3/14.9
	J	3209.8/19.2	2756.6/4.9	1704.1/8.9	–
	O	3207.1/15.0	2982.3/13.5	2183.4/14.1	2707.8/9.6

Task	Source	CPO	PCPO	FOCOPS	CUP
Ant	T	2619.4/4.8	2202.6/61.1	2505.9/262.7	1948.5/42.0
	J	3116.8/14.1	2276.2/10.2	3291.3/15.1	3297.3/23.6
	O	3098.5/14.1	2257.1/10.4	3184.5/14.8	3215.8/18.3
CarButton1	T	−1.0/54.3	−4.7/50.7	−4.8/43.3	1.8/77.1
	J	1.3/43.7	0.3/47.6	0.6/30.2	0.7/137.5
	O	0.8/37.9	0.4/40.5	0.2/31.8	1.5/103.2
CarGoal1	T	14.2/28.9	11.3/32.2	21.7/30.1	20.7/43.8
	J	26.2/40.7	20.6/35.4	17.6/23.2	6.4/15.7
	O	25.5/43.3	21.6/38.4	15.2/31.7	6.1/36.1
CarPush1	T	0.4/20.2	0.1/26.6	0.4/17.0	0.2/21.7
	J	0.9/28.5	0.7/59.0	0.3/9.0	0.3/23.9
HalfCheetah	T	1811.7/19.6	1287.7/58.0	2056.3/3.0	2649.1/0.7
	J	2738.4/5.7	1743.7/15.6	2873.1/2.9	2765.4/4.3
	O	2786.5/4.7	1677.9/19.1	2965.2/2.4	2850.6/4.3
Hopper	T	1117.0/0.0	1030.8/34.2	1069.7/9.3	1342.2/16.6
	J	1713.2/12.1	1519.6/12.8	1538.8/7.4	1716.4/5.4
	O	1713.7/13.4	1551.2/15.5	1437.8/10.1	1716.1/7.5
PointButton1	T	2.6/28.4	1.4/36.7	0.9/34.1	7.3/72.2
	J	3.2/40.2	2.2/54.7	6.6/38.3	4.7/31.4
	O	4.0/40.0	2.7/49.4	5.9/38.2	3.5/39.6
PointGoal1	T	9.6/25.0	11.4/36.0	20.6/39.4	20.8/47.2
	J	20.5/27.4	18.8/20.5	14.8/33.0	12.0/18.2
	O	20.5/28.8	18.6/23.0	15.0/33.7	14.4/19.0
PointPush1	T	0.6/27.8	0.2/19.1	0.2/26.6	0.2/21.6
	J	1.8/19.9	1.4/31.3	0.9/62.6	1.9/20.0
Walker2d	T	1396.5/15.2	829.6/9.5	2558.3/0.6	1010.2/35.3
	J	2440.8/20.2	1698.3/17.7	3116.1/3.9	2739.5/4.4
	O	2074.8/21.9	1802.9/18.8	3117.1/8.8	2467.0/6.6

Threshold majority rule. A separate task-level rule compares the best observed return at cost no greater than each of 200 uniformly spaced thresholds in the common observed cost interval. A task meets the rule when the best return of SafeOT is at least that of the comparator on half or more of these thresholds. This holds on $9/10$ tasks against PCPO, $8/10$ against CPO, $7/10$ against CPPO-PID and TRPO-PID, and $3/10$ against TRPO-Lag. The TRPO-PID count is not stable under seed resampling: over 1000 seed-bootstrap replicates its median is 5 tasks (95% interval 3–8). At interpolated equal cost, SafeOT is higher than TRPO-PID at 9 operating points and lower at 10. One CPPO-PID task has no cost overlap and remains in the ten-task denominator. This rule uses the best observed returns, not the interpolation rule above. Neither count means an everywhere-superior frontier or a statistical equivalence result.

Budget response. The descriptive least-squares slope of final mean cost on nominal budget is computed per task with at least three eligible budgets. Median slopes are $0.98/0.39/0.85/0.62$ for SafeOT/TRPO-Lag/CPO/CPPO-PID over $10/10/10/10$ tasks. A slope near one describes response to the budget. It is not feasibility, monotonicity or a guarantee for individual seeds.

E.3 Final-Policy Feasibility

Table 22 reports the numbers of feasible ($\leq B$) and near-feasible ($\leq 1.1B$) seeds and the median costs of the final policies.

On shared task–seed pairs, SafeOT versus TRPO-Lag has $32/50$ versus $27/50$ feasible policies at $B = 12.5$, $64/100$ versus $59/100$ at 25, and $21/50$ versus $39/50$ at 50. At $B = 50$, the training cost averaged over the last ten epochs lies within 10% of the budget on $8/10$ tasks for SafeOT against $4/10$ for TRPO-Lag. This tracking statistic does not identify the cause of the final-policy difference. Across all 250 available SafeOT scan policies, 10 have final cost above $5B$ and 4 above $10B$. The maximum is $13.28B$. All but one of these ten large overruns occur at $B = 5$. These tails remain part of the reported distributions.

Longer training. Table 23 repeats the comparison with TRPO-Lag and CPO for 10M steps, five times the benchmark length, on seven of the ten tasks with five seeds each. SafeOT keeps a lower CV than TRPO-Lag on all seven tasks (median ratio TRPO-Lag/SafeOT 2.42) and than CPO on five. Its final policies are feasible in 23 of the 35 runs, against 22 for TRPO-Lag and 27 for CPO, and its final return is lower than that of TRPO-Lag on five of the seven tasks, by more than 5% only on PointButton1 (6.5 against 10.5) and Walker2d (1549 against 2630).

Proposition 3 assumes convergent training-cost and feedback sequences. Logged stochastic behavior and deterministic final evaluation are different policy/sampling protocols. The proposition supplies neither a finite-time nor a per-seed deployment guarantee. Appendix F measures sampled flow realization. A signed cost difference does not isolate any of the three nonnegative terms of Lemma 2.

Table 21: Two cost-matched summaries. Majority rule: tasks meeting the threshold majority rule. Overlap: tasks with overlapping observed costs. Higher, lower and close: SafeOT return above, below or within 3% of the task's return range of the interpolated comparator front at equal cost. Outside: SafeOT points whose cost lies outside that front's cost range, out of 40 per comparator. PCPO and TRPO-PID have three seeds at each budget other than 25.

Comparator	Majority rule	Overlap	Higher	Lower	Close	Outside
CPO	8/10	10/10	10	6	2	22
PCPO	9/10	10/10	20	3	2	15
CPPO-PID	7/10	9/10	13	2	1	24
TRPO-PID	7/10	10/10	9	10	3	18
TRPO-Lag	3/10	10/10	2	7	3	28

Table 22: Budget scan: numbers of feasible ($\leq B$) and near-feasible ($\leq 1.1B$) seeds and median [interquartile range] costs divided by the budget. Training cost averages the last ten logged epochs. Final evaluation uses deterministic policy actions over 30 episodes.

Method	B	n	$\leq B$	$\leq 1.1B$	Training C/B	Final C/B
SafeOT	5	50	23/50	23/50	$2.70[1.19, 3.68]$	$1.23[0.21, 3.41]$
SafeOT	12.5	50	32/50	32/50	$1.33[0.98, 1.66]$	$0.67[0.26, 1.42]$
SafeOT	25	100	64/100	72/100	$1.05[0.91, 1.21]$	$0.82[0.24, 1.14]$
SafeOT	50	50	21/50	27/50	$0.96[0.89, 1.05]$	$1.04[0.86, 1.50]$
TRPO-Lag	5	50	23/50	23/50	$2.91[1.57, 7.22]$	$1.97[0.18, 5.90]$
TRPO-Lag	12.5	50	27/50	28/50	$1.60[0.74, 2.71]$	$0.74[0.18, 2.35]$
TRPO-Lag	25	100	59/100	63/100	$1.04[0.70, 1.58]$	$0.72[0.32, 1.47]$
TRPO-Lag	50	50	39/50	43/50	$0.92[0.72, 1.03]$	$0.74[0.33, 0.93]$
PPO-Lag	25	100	36/100	41/100	$1.40[0.82, 2.22]$	$1.72[0.43, 2.74]$
FOCOPS	25	100	44/100	49/100	$1.17[0.73, 1.64]$	$1.15[0.14, 1.62]$
CUP	25	100	36/100	43/100	$1.45[0.93, 2.21]$	$1.47[0.64, 2.32]$
CPO	5	50	30/50	30/50	$1.43[1.00, 2.08]$	$0.54[0.02, 2.37]$
CPO	12.5	50	30/50	31/50	$1.16[0.86, 1.43]$	$0.86[0.20, 1.64]$
CPO	25	100	58/100	66/100	$1.14[0.95, 1.33]$	$0.85[0.22, 1.28]$
CPO	50	50	23/50	31/50	$0.98[0.85, 1.08]$	$1.03[0.65, 1.38]$
PCPO	25	100	42/100	46/100	$0.95[0.75, 1.22]$	$1.21[0.59, 1.80]$
CPPO-PID	5	50	23/50	24/50	$1.92[1.32, 3.08]$	$1.17[0.04, 3.38]$
CPPO-PID	12.5	50	29/50	30/50	$1.37[0.99, 1.80]$	$0.86[0.16, 1.74]$
CPPO-PID	25	100	69/100	76/100	$0.99[0.82, 1.21]$	$0.79[0.12, 1.05]$
CPPO-PID	50	50	33/50	38/50	$0.94[0.72, 1.10]$	$0.89[0.48, 1.08]$
TRPO-PID	25	50	33/50	33/50	$1.00[0.81, 1.10]$	$0.74[0.31, 1.25]$

Table 23: Training for 10M steps ($B = 25$, five seeds): CV, final return and number of feasible final policies for SafeOT / TRPO-Lag / CPO.

Task	CV	Final return	Feasible
PointGoal1	84.3 / 184 / 98.1	18.4 / 15.1 / 20.2	1 / 3 / 3
CarGoal1	58.0 / 151 / 92.6	22.8 / 23.6 / 22.1	2 / 1 / 3
PointButton1	99.0 / 201 / 155	6.5 / 10.5 / 6.0	1 / 1 / 1
Ant	50.7 / 134 / 47.8	3143 / 3160 / 3085	5 / 4 / 5
HalfCheetah	31.6 / 117 / 43.2	2834 / 2884 / 2739	5 / 5 / 5
Hopper	58.5 / 141 / 125	1692 / 809 / 1705	5 / 3 / 5
Walker2d	94.8 / 115 / 38.1	1549 / 2630 / 2536	4 / 5 / 5
All seven			23 / 22 / 27

F Empirical Target-to-Policy Decomposition

This appendix measures the terms of Lemma 2 during training on the ten Safety-Gymnasium tasks at seeds 0–2: 30 complete runs and 3,000 saved graphs. Of 2,940 candidate pairs (epochs 2–99 paired with their successors), excluding the 300 pairs that straddle a refit of the clusters (every 10 epochs, Appendix A.7) leaves 88 pairs per run, 2,640 in total. These are stochastic training batches, separate from deterministic final-policy evaluation.

Let $\hat{F}_t$ denote the observed flow of epoch t (nonterminal transition frequencies), $\widehat{C}_t$ its windowed edge costs, $\widetilde{q}_t$ the target flow $F^\star$ of (6) without its solver self-loops and renormalized, j_{t+1} the per-step cost of the next batch (its total cost divided by its number of steps), C^{batch}_{t+1} the edge costs measured on that batch, and $b_t = B/\operatorname{clip}(L_t, 50, 5000)$ with L_t the mean episode length of epoch t. The signed empirical identity is

$$\begin{aligned} j_{t+1} - b_t = (\langle \widehat{C}_t, \widetilde{q}_t\rangle - b_t) + \langle \widehat{C}_t, \hat{F}_{t+1} - \widetilde{q}_t\rangle \\ + \langle C^{\mathrm{batch}}_{t+1} - \widehat{C}_t, \hat{F}_{t+1}\rangle + (j_{t+1} - \langle C^{\mathrm{batch}}_{t+1}, \hat{F}_{t+1}\rangle). \end{aligned}$$

These four terms are the signed target residual, the signed flow-to-policy gap, the signed cost-estimation error and the aggregation remainder. The first three are signed counterparts of the three

terms of (9). Batch cost divided by batch length gives the sampled per-step cost, including episode fragments. The vectors are not exact discounted policy occupancies. The signed cost-estimation error also differs from the absolute, occupancy-weighted error of Lemma 2. The arithmetic identity closes within 3.20×10^{-15} budget units. The absolute aggregation remainder is at most $0.0356 b_t$. The largest run-median absolute remainder is $0.01714 b_t$. Neither statistic certifies a deployment bound.

The plotted target omits the 10^{-7} solver self-loops and is renormalized, differing in cost from the saved solver target by at most $0.00223 b_t$. Including those loops reproduces saved target costs within $1.12 \times 10^{-14} b_t$, with mass error below 1.49×10^{-13} and flow-conservation residual below 9.97×10^{-8}. Strict budget feasibility holds for 2,733/3,000 saved targets. Relative tolerance $0.001 b_t$ gives 2,735, whereas the implementation's absolute 0.001 tolerance gives 2,881. We retain numerical target violations rather than treating every computed target as feasible.

Direction and cost accounting. Table 24 and Figure 10 summarize the sampled flow realization. The projection coefficient is $\langle \hat{F}_{t+1} - \hat{F}_t, \widetilde{q}_t - \hat{F}_t \rangle / \|\widetilde{q}_t - \hat{F}_t\|_2^2$, not a bounded safety fraction. Across the twelve Goal/Push runs, its run medians span 0.33–0.63, and signed flow-gap medians span 0.07–$0.42 b_t$. The corresponding ranges are 0.01–0.37 and 0.005–$0.25 b_t$ for twelve velocity runs, and 0.05–0.32 and 0.20–$1.79 b_t$ for six Button runs. The alignment, the cosine between $\hat{F}_{t+1} - \hat{F}_t$ and $\widetilde{q}_t - \hat{F}_t$, has a positive median in all thirty runs. The distances are $D_{\text{ask}} = \|\widetilde{q}_t - \hat{F}_t\|_1$, $D_{\text{real}} = \|\hat{F}_{t+1} - \widetilde{q}_t\|_1$ and $D_{\text{move}} = \|\hat{F}_{t+1} - \hat{F}_t\|_1$. The cost-estimation error $a^{\text{emp}} = \langle |C_{t+1}^{\text{batch}} - \widehat{C}_t|, \hat{F}_{t+1} \rangle$ spans $0.126 b_t$ to $2.734 b_t$ in run medians, whereas its signed counterpart spans only $-0.246 b_t$ to $0.132 b_t$ because edge errors of opposite sign cancel. The flow-to-policy gap $\Gamma = \|\widehat{C}_t\|_\infty \|\hat{F}_{t+1} - \widetilde{q}_t\|_1$ of (9) spans 3.35–$47.32 b_t$ in run medians. Its median divided by the median absolute signed flow gap is 10.9–467.3 across runs, documenting a loose bound.

Table 24: Empirical flow realization on the ten Safety-Gymnasium tasks (three seeds each). Entries are task medians over 264 retained epoch pairs. Align. and Proj.: cosine and projection coefficient of the realized step on the requested step. D_{ask}, D_{real} and D_{move}: the ℓ_1 distances defined in the text. Here $\Gamma = \|\widehat{C}_t\|_\infty \|\hat{F}_{t+1} - \widetilde{q}_t\|_1$ and $a^{\text{emp}} = \langle |C_{t+1}^{\text{batch}} - \widehat{C}_t|, \hat{F}_{t+1} \rangle$.

Task	Runs	Align.	Proj.	D_{ask}	D_{real}	D_{move}	Γ / b_t	a^{emp} / b_t
PointGoal1	3	0.406	0.541	0.525	0.708	0.667	28.301	1.474
CarGoal1	3	0.415	0.527	0.520	0.683	0.665	27.327	1.383
PointPush1	3	0.385	0.448	0.664	0.929	0.873	37.145	1.543
CarPush1	3	0.348	0.439	0.680	0.934	0.892	37.376	1.443
PointButton1	3	0.275	0.247	0.743	0.886	0.667	35.435	1.548
CarButton1	3	0.190	0.089	0.915	1.070	0.746	42.782	2.553
Ant	3	0.152	0.108	0.374	0.481	0.350	13.364	0.492
HalfCheetah	3	0.231	0.222	0.277	0.372	0.313	13.268	0.512
Hopper	3	0.084	0.039	0.359	0.446	0.306	15.580	0.663
Walker2d	3	0.072	0.032	0.394	0.483	0.299	4.060	0.240

Target residual tails. The normalized target residual $(\langle \widehat{C}_t, F_t^\star \rangle - b_t)/b_t$ of the saved solver targets has median -6.1×10^{-5}, 95th percentile 0.031, 99th percentile 0.61 and maximum 23.5. All 119 targets that exceed the budget beyond the implementation tolerance come from solves whose safety price reaches the bisection bound $\lambda_{\max} = 20$ (130 solves), so every solve below this bound meets its budget within tolerance. Numeric arrays, reconstructed mass and flow conservation pass their checks. The solver status was not recorded, so budget misses cannot be attributed to optimizer failures.

The supplementary data give the target residuals, tolerance counts, bisection-bound hits and tail quantiles of all thirty runs.

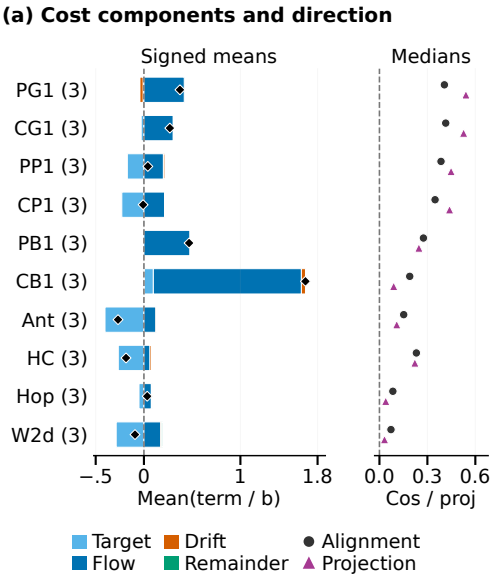

Figure 10: **Sampled flow realization.** Left: means of the four signed terms divided by b_t (Target: target residual, Flow: flow-to-policy gap, Drift: cost-estimation error, Remainder: aggregation remainder). Diamonds give their sum. Right: median alignment/projection. Parentheses show run counts (30 total, 88 pairs/run). Sampled frequencies differ from policy occupancies. The signed cost-estimation error differs from its absolute counterpart a^{emp}. Tasks are abbreviated as in Table 9, with HC for HalfCheetah, Hop for Hopper and W2d for Walker2d.

G Multiple Budgets and Controlled Studies

This appendix reports the controlled studies behind Section 5.3: tabular CMDPs with known dynamics (Appendices G.1 and G.2), learned grid worlds (Appendices G.3 and G.4), MuJoCo robots with per-joint budgets (Appendices G.4–G.6) and further controls. Table 31 summarizes the multiple-budget settings of Appendices G.2 and G.4–G.6. Return kept (at least 90% of the comparator's training return) is a descriptive count, not a non-inferiority test.

G.1 Known-Model Policy Updates

Twenty randomly generated 9×9 grid CMDPs at two budgets give forty instance–budget cells, not forty independent environments. A hazard band creates a risky short path and a safe detour. Fragile cells define hard constraints. Slip is 0.1, discount 0.95, and actions are four cardinal moves. Budgets are $0.3/0.6$ times the unconstrained LP optimum's hazard cost after hard-action masking. Known-model evaluation supplies the constrained LP optimum and its multiplier scale. Every controller shares the permitted actions, initial policy and update

$$\pi_{t+1}(a \mid s) \propto \pi_t(a \mid s) \exp\left(\eta[A_{r,t}(s,a) - \sum_k \lambda_{k,t} A_{c_k,t}(s,a)]\right).$$

Probabilities are floored at 10^{-300} before logs. Excluded actions are then reset to zero. SafeOT uses $\varepsilon = 1/\eta$ and smoothing weight $\xi = 0.5$, initialized from the first solve. Safety prices are unclipped apart from a numerical bound of 10^4. Solves use 300 flow-conservation iterations, tolerance 10^{-9}, 40 bisections and eight coordinate sweeps, without virtual flow. The reactive switch uses $\kappa \mathbf{1}[J_c > b] + \beta$. All rules use the previous feedback term for the actor and update it afterward, unlike SafePO's ordering. Tabular PID is unfiltered and differs from SafePO's PID class.

Tuning and metrics. Gains are $\alpha = c_\alpha \lambda_{\mathrm{LP}}/b$, with LP multipliers floored at 10^{-6} and $c_\alpha \in \{0.03, 0.3, 3\}$. Warm-start levels use $c_\kappa \in \{1, 2\}$ and switch amplitudes $c_\kappa \in \{0.5, 1, 2, 4\}$. SafeOT, Lagrangian and PID have three candidates, warm starts six and switches twelve. Selection minimizes median positive tail return gap plus relative excess on the same forty reporting cells. No held-out tuning set is used. This oracle scale is unavailable in deep RL, and search sizes differ.

Runs record 300 pre-update policies. $V = \sum_{t=0}^{299} \sum_k [(J_{k,t} - b_k)/b_k]_+$ and the last 100 entries define the tail. Return gap is $(J_r^{\mathrm{LP}} - J_r)/J_r^{\mathrm{LP}}$. Negative gaps can accompany violations. The last executed update is not separately evaluated, and normalized tabular V is not numerically comparable with benchmark V. Table 25 summarizes selected settings. Against the reactive switch, SafeOT has

lower V in $25/40$, $40/40$, $40/40$ cells at $\eta = 2, 10, 50$. At $\eta = 10$, its median final return gap is 1.24×10^{-4} and maximum relative excess 0.003. The latter is 0.0202 at $\eta = 50$. Residual violations remain.

Synthetic estimate noise. At $\eta = 10$, nominal noise $r \in \{0.05, 0.15, 0.45\}$ uses $\lceil 1/(br^2) \rceil$ transitions sampled from current exact occupancy, and Gaussian perturbations of exact advantages. Expected one-step model costs remain available. This is not a learned-critic noise model. All methods share random seeds, but their samples diverge with the policies. Gains are selected again at each noise level. At $r = 0.15$, the criterion requires lower V in 70% of cells and tail score within 0.01 in 60% against each comparator. It holds: lower-V counts versus Lagrangian, warm start, PID and switch are $40/39/40/40$ of forty. Score-tolerance counts are $39/39/26/26$. Median ratios to the better of PID/switch increase with noise as $0.159, 0.277, 0.818, 0.838$ (including zero noise), meeting the monotonicity criterion. High-noise tail scores nevertheless deteriorate. Equal final return and feasibility are not established.

Target realization and hard support. The realization diagnostic has 160 runs and 299 target–next-policy comparisons each. Maximum cumulative positive target excess is 6.27×10^{-10} in relative-budget units. This verifies target cost to numerical precision, not full flow feasibility: flow-conservation residuals and solver statuses were not saved. Let q_t and $\widehat{C}_t$ be the target flow and the estimated edge costs of iteration t (Lemma 2), and $\omega_t = J_c(\pi_{t+1}) - \langle \widehat{C}_t, F^{\pi_{t+1}} \rangle$. The diagnostic checks the loose triangle inequality

$$|J_c(\pi_{t+1}) - \langle \widehat{C}_t, q_t \rangle| \le \|\widehat{C}_t\|_\infty \|F^{\pi_{t+1}} - q_t\|_1 + |\omega_t|.$$

Maximum saved left-minus-right value is -3.93×10^{-7}. This is not a signed-identity residual or a policy-safety guarantee. Initial-policy excess 0.047–1.14 lies outside these comparisons. The noisy diagnostic uses $c_\alpha = 0.3$, versus 0.03 in the 15%-noise performance comparison.

Removing actions entering fragile cells gives zero tested fragile occupancy, although actions are restored when none remain. Pricing those events with budget 10^{-3} instead gives cumulative normalized hard masses 0.374 for SafeOT and 0.533 for dual ascent. These are not episode contact rates. Masks change initial support and the feasible set. On aggregated graphs, removing feedback changes median tail relative excess from approximately 5.0×10^{-5} to 0.661 and cumulative excess from 7.74 to 193.96. Unaggregated graphs still have tail excess 0.170 without feedback. Convergence remains conditional.

G.2 Multiple Budgets with Known Dynamics

A separate study splits a 9×9 hazard band into $K = 1, 4, 8$ regions, with alternating budgets $0.2/0.6$ times the cost of the initial uniform policy, in both alternation orders. Twenty reporting instances

Table 25: Selected known-model configurations: medians [interquartile range] across forty cells. Oracle-scale tuning and reporting share cells. Positive return gaps are shortfalls.

Step η	Rule	(c_α, c_κ)	Violation	Tail return gap	Tail violation
2	SafeOT	0.3, 0	2.55 [2.11, 2.95]	0.002 [0.001, 0.002]	0.000 [0.000, 0.000]
2	Reactive switch	3, 0.5	8.29 [2.11, 24.26]	0.003 [0.002, 0.117]	0.007 [0.006, 0.077]
2	Lagrangian	3, 0	20.16 [18.90, 27.76]	0.176 [0.120, 0.274]	0.069 [0.068, 0.091]
2	PID	3, 0	4.42 [3.39, 4.43]	0.002 [0.001, 0.002]	0.000 [0.000, 0.000]
2	Warm start	0.3, 2	52.81 [51.94, 86.99]	0.078 [0.039, 0.103]	0.188 [0.185, 0.331]
10	SafeOT	0.3, 0	2.57 [1.87, 5.94]	0.000 [0.000, 0.001]	0.000 [0.000, 0.000]
10	Reactive switch	0.3, 0.5	18.73 [13.87, 54.19]	0.000 [−0.001, 0.001]	0.037 [0.027, 0.048]
10	Lagrangian	3, 0	23.52 [23.19, 29.28]	0.210 [0.179, 0.377]	0.076 [0.073, 0.105]
10	PID	0.3, 0	53.17 [34.42, 57.25]	0.000 [0.000, 0.001]	0.000 [0.000, 0.000]
10	Warm start	0.3, 1	80.20 [72.82, 91.88]	0.050 [0.050, 0.092]	0.259 [0.245, 0.275]
50	SafeOT	0.03, 0	28.27 [28.11, 31.65]	0.000 [0.000, 0.000]	0.000 [0.000, 0.001]
50	Reactive switch	0.03, 0.5	66.13 [55.58, 77.81]	0.003 [−0.001, 0.009]	0.112 [0.094, 0.132]
50	Lagrangian	0.3, 0	86.87 [84.96, 92.29]	0.118 [0.031, 0.190]	0.289 [0.274, 0.308]
50	PID	0.3, 0	53.20 [34.93, 56.88]	0.000 [0.000, 0.000]	0.000 [0.000, 0.000]
50	Warm start	0.3, 1	80.33 [73.13, 86.51]	0.130 [0.027, 0.214]	0.242 [0.210, 0.297]

give forty cells per K. Every controller uses natural-policy step $\eta = 10$. SafeOT uses $\varepsilon = 0.1$. Configurations are selected on instances 100–109 and reported on disjoint instances 200–219. V sums the positive budget-normalized cost excess across iterations and budgets.

The 600 reporting summaries yield 480 paired ratios on 120 cells. SafeOT lowers V in every cell for each rule and K (median gains 2.3–14.6×). The median gain does not increase from $K = 1$ to $K = 8$ for three of the four rules (Figure 11a). Residual terminal violations preclude exact feasibility or LP optimality.

G.3 Learned Grid Worlds with One Budget

Two grid layouts give three settings: 9×9 with budgets $B = 7.36$ and $B = 14.72$, and 15×15 with $B = 8.65$. Seven rules and five seeds give 105 runs of 2M steps. The smaller-grid budgets share a layout. One-hot states enter a 64×64 Gaussian policy. The largest of four scores selects an action. Episodes last 200 steps. Epochs have 20,000 interactions and logs average the last 50 completed episodes.

The shared TRPO trainer runs SafeOT, unclipped SafeOT, its coarse-graph variant, the reactive switch, TRPO-PID, and TRPO-Lag with cold and warm starts, without per-rule search. The first two use $N = S = 81/225$ k-means clusters, the coarse variant $20/56$. Even $N = S$ is sampled rather than an exact model. Unclipped SafeOT uses the clip level $c = 10^9$ and keeps the bisection bound $\lambda_{\max} = 20$ and the cap $\Lambda = 100$ on the combined price. The reactive switch replaces the raw safety price by $10^6\mathbf{1}[J_{\log} > B]$ and keeps $c = 1$, the smoothing and the feedback term. It still calls the solver, so timing is not solver-free. Feedback starts at zero for SafeOT and $0.001/1$ for cold/warm TRPO-Lag.

Here the unnormalized violation V sums the positive excess of the logged epoch cost, and the return R and cost C average the last ten epochs. Overlapping windows make V an epoch diagnostic, not accumulated episode harm. The advantage criterion requires, against every comparator, a lower V in 11 of the 15 runs (three settings, five seeds) and $R \geq 0.9R_c$ in 9 of 15, where R_c is the comparator's return. Partial support requires switch and one other comparator. Table 26 gives partial support. Coarsening requires a median V increase above 1.2 and fewer switch wins. The ratio 0.92 and the unchanged $11/15$ wins do not support it. With $c = 1$, the ratio 1.00 lies within $[0.8, 1.25]$, a descriptive tolerance rather than equivalence or an explanation of another suite. Actor-mean evaluation uses 30 episodes per checkpoint. Sampled-action evaluation uses different reset seeds and does not rescore training criteria. On the larger grid, unclipped SafeOT gives $R/C = 4.05/8.75$ at $B = 8.65$ with $2/5$ feasible seed means. SafeOT with $c = 1$ gives $4.51/9.79$, and warm-start TRPO-Lag $2.27/7.61$ (also $2/5$ feasible). Neither uniform feasibility nor return non-inferiority follows.

G.4 Multiple Budgets with Learned Policies

We test two sampled neural-policy settings using a shared TRPO trainer, 64×64 networks, 2M interactions, 20,000 steps per epoch and 30 deterministic final-evaluation episodes. The grid study splits the hazard band of the 9×9 and 15×15 layouts of Appendix G.3 into $K \in \{4, 8\}$ regions with alternating tight and loose budgets. Each K has ten runs per rule (two layouts, seeds 0–4). The $K = 1$ reference reuses the runs of Appendix G.3. The robot studies use MuJoCo 3.3.6, Gymnasium 0.28.1, vendored Safety-Gymnasium 1.2.0 and PyTorch 2.14.0 (CPU). The main benchmark uses

Table 26: Learned grid worlds with one budget: unclipped SafeOT against each comparator (15 runs each). Lower violation, return kept (at least 90% of the comparator's return) and return not lower count runs. The 90% threshold is not a statistical non-inferiority test.

Comparator	Lower violation	Return kept	Return not lower
Reactive switch	11/15	14/15	1/15
TRPO-PID	13/15	13/15	2/15
TRPO-Lag (cold start)	15/15	14/15	7/15
TRPO-Lag (warm start)	6/15	14/15	5/15

MuJoCo 2.3.3. All methods within each robot study share one software stack. The robot study replaces the native velocity constraint by six (HalfCheetah) or eight (Ant) joint-speed indicators, $c_k = \mathbf{1}[|\dot{q}_k| > \theta_k]$, with alternating episode budgets $100/250$. Thresholds are the 70th percentiles from five deterministic episodes of a re-trained TRPO-Lag reference. A fixed floor at one quarter of the median threshold avoids near-static Ant joints measuring numerical jitter. All methods share these thresholds and budgets. Unclipped SafeOT is as in Appendix G.3.

For both studies, $V = \sum_{t,k}[C_{k,t} - B_k]_+$ uses the logged 50-episode-smoothed costs, without normalization by the budgets. R averages the last ten logged training returns. Return kept counts runs with $R \geq 0.9R_c$, where R_c is the comparator's return. Final evaluation measures the same saved policy's return and costs separately.

Robot comparisons. The primary comparison uses seeds 0–2 and requires, against TRPO-Lag and TRPO-PID, a lower V in at least four of the six runs (two robots, seeds 0–2) and a kept return in at least four. Both comparisons meet this requirement: V is lower in six of six cells and the return threshold holds in five of six. The secondary five-seed check, fixed before outcomes, uses seven of ten for each count and confirms the result (Table 27). SafeOT + PID feedback, which replaces the observed-cost feedback of SafeOT by the PID Lagrangian update (Appendix G.8), also meets it on both seed sets. The criterion of improving on the reactive switch, a median SafeOT/switch V ratio below 0.9, is not met: the ratios are 1.09 (primary) and 0.98 (secondary). The latter return ratio is 1.17, descriptively. With $c = 1$, the median SafeOT/switch ratios are 1.05 in V and 1.10 in R on five seeds. No equivalence test is claimed. Final policies that meet every joint budget number $4/5$ on HalfCheetah and $1/5$ on Ant for unclipped SafeOT, against $0/5$ and $1/5$ for TRPO-Lag and $2/5$ and $0/5$ for TRPO-PID.

Grid comparisons. Figure 11 summarizes the multiple-budget comparisons, and Table 28 gives the grid results. Unclipped SafeOT has a lower V than TRPO-Lag and TRPO-PID in all ten runs at $K = 4$ and $K = 8$, but than the reactive switch with $c = 1$ in only $4/10$ and $7/10$ runs, and than the switch with $c = 4$ in $0/10$ at either K. The median SafeOT/TRPO-PID V ratio falls from 0.60 to 0.25 to 0.12 as K goes from 1 to 4 to 8, whereas the SafeOT/switch ratios 0.96, 1.08 and 0.92 do not decrease monotonically. With $c = 1$ for SafeOT as well, the median SafeOT/switch ratio at $K = 8$ is 1.37. Thus the advantage over TRPO-Lag and TRPO-PID does not imply an advantage over a fixed saturated switch, nor establish coupling as its causal explanation.

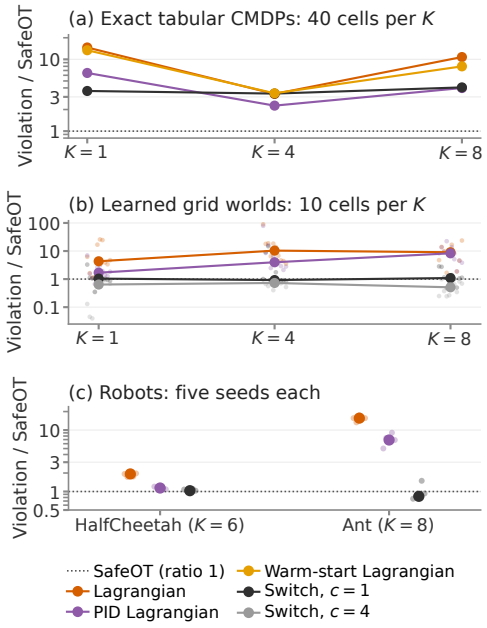

Figure 11: **Multiple-budget comparisons.** Violation ratio: the comparator's violation divided by that of SafeOT, so values above one favor SafeOT, as in Table 31. (a) Exact tabular CMDPs (Appendix G.2). (b) Learned grid worlds (Appendix G.4). (c) Robots with six and eight budgets. Panels (b) and (c) use unclipped SafeOT. Switch: the reactive switch of Appendix G.1 in panel (a) and of Appendix G.3 in panels (b) and (c), at clip level c. Large markers show medians, and faint dots show paired cells. The ratio does not grow with K for every comparator.

Descriptive pricing timing. In the primary robot runs, the combined price exceeds 0.05 in about 94% of over-budget Ant channel-epochs, versus 33–36% for feedback alone. Prices are logged after the rollout and used in the ensuing policy update. This diagnostic does not isolate the coupling of budgets through the residual covariance $\tilde{G}$ of Proposition 2, which raw correlations of the joint indicators do not estimate.

Runs. The grid and robot studies use 150 and 60 runs, and the single-budget reference reuses the 70 runs of Appendix G.3.

G.5 Extension to Hopper and Walker2d

The same joint-speed protocol adds three Hopper and six Walker2d joint-speed constraints (50 runs). The threshold floor applies to Hopper's near-static thigh. The global-coordinate models were converted with MuJoCo 2.3.3's model writer. Compiled numeric arrays agree within 10^{-10}, except joint limits rounded by up to 4×10^{-6} rad. Across three fixed-action sequences per robot, conversion and cross-version checks agree within 3×10^{-6} in state coordinates for the first ten control steps. Later trajectories diverge. These checks do not establish long-horizon equivalence.

Table 27: Multiple budgets on HalfCheetah and Ant. Violation and return ratios are medians of per-run ratios of the first method to the second (a violation ratio below one favors the first), the reverse of Table 31. Lower violation and return kept count runs. The reactive switch uses $c = 1$. Seeds 0–2 decide, and the five-seed check agrees. Return kept uses a 90% threshold and is not a non-inferiority test.

Seeds	Comparison	Violation ratio ↓	Return ratio ↑	Lower violation	Return kept
0–2	Unclipped SafeOT / TRPO-Lag	0.29	1.06	6/6	5/6
0–2	Unclipped SafeOT / TRPO-PID	0.52	1.56	6/6	5/6
0–2	SafeOT + PID feedback / TRPO-PID	0.53	1.28	6/6	5/6
0–2	Unclipped SafeOT / reactive switch	1.09	1.15	3/6	5/6
0–2	SafeOT ($c = 1$) / reactive switch	1.06	1.07	0/6	5/6
0–4	Unclipped SafeOT / TRPO-Lag	0.29	1.06	10/10	7/10
0–4	Unclipped SafeOT / TRPO-PID	0.52	1.47	10/10	9/10
0–4	SafeOT + PID feedback / TRPO-PID	0.53	1.19	10/10	8/10
0–4	Unclipped SafeOT / reactive switch	0.98	1.17	6/10	9/10
0–4	SafeOT ($c = 1$) / reactive switch	1.05	1.10	0/10	9/10

Table 28: Multiple budgets on the learned grids: unclipped SafeOT against each comparator (ten runs per K, two layouts and seeds 0–4). Violation ratio: median over runs of the violation of SafeOT divided by that of the comparator (below one favors SafeOT), the reverse of Table 31. Lower violation and return kept count runs.

K	Comparator	Violation ratio ↓	Lower violation	Return kept
1	TRPO-Lag	0.23	10/10	9/10
1	TRPO-PID	0.60	9/10	8/10
1	Reactive switch ($c = 1$)	0.96	7/10	9/10
1	Reactive switch ($c = 4$)	1.57	0/10	10/10
4	TRPO-Lag	0.11	10/10	9/10
4	TRPO-PID	0.25	10/10	9/10
4	Reactive switch ($c = 1$)	1.08	4/10	10/10
4	Reactive switch ($c = 4$)	1.38	0/10	10/10
8	TRPO-Lag	0.11	10/10	10/10
8	TRPO-PID	0.12	10/10	9/10
8	Reactive switch ($c = 1$)	0.92	7/10	10/10
8	Reactive switch ($c = 4$)	2.11	0/10	10/10

Table 29 distinguishes the new robots from the pooled four-robot set. With five seeds, SafeOT has lower violation than TRPO-Lag in all ten Hopper and Walker2d runs but keeps at least 90% of the comparator's return in only three, and it has lower violation than TRPO-PID in seven of ten runs and keeps 90% of the return in six. Lower violation thus extends to Walker2d against both comparators and to Hopper against TRPO-Lag, but not to Hopper against TRPO-PID. Return counts refer to last-ten-epoch training return, not final evaluation or a statistical non-inferiority test. Across the twenty runs, the median V ratios of SafeOT to TRPO-Lag and to TRPO-PID are 0.15 and 0.58. On Walker2d, unclipped SafeOT has training $V/R = 105/619$ versus $1725/1096$ for TRPO-Lag. Final returns are $708/1276$. On Hopper the corresponding training values are $1489/926$ versus $6269/736$, and final returns $881/458$. Thus the return trade-off differs by robot. Final policies meeting every joint budget number $1/5, 4/5, 2/5$ on Hopper and $2/5, 1/5, 2/5$ on Walker2d for unclipped SafeOT, TRPO-Lag and TRPO-PID, respectively (30 deterministic episodes per policy). The supplementary data contain all fifty progress logs and the 1,500 final episodes.

Final policies of Table 3. Table 30 gives, for the four robots of Table 3, the mean and standard deviation over the five seeds of the excess cost, the satisfied constraints and the final return of the final policy, with Welch's t-test of each method against the best mean of each measure. SafeOT has a significantly lower excess cost than the Lagrangian method on all four robots and than the PID Lagrangian method on HalfCheetah and Ant, and neither the Lagrangian nor the PID Lagrangian method has a significantly higher satisfied-constraint count or final return than SafeOT on any robot. The ratio of the mean excess costs of the Lagrangian method and SafeOT reaches 16 on Walker2d ($1725/105$), whereas Table 31 reports medians of per-run ratios. The return counts of Table 29 use the last ten training epochs, not these final returns.

Table 29: Robot extension: paired counts of lower violation and of return kept (at least 90% of the comparator's return). The first count is violation. The second is return. Each must reach the threshold for unclipped SafeOT to pass against that comparator. Partial: passes against one of the two comparators.

Tasks	Seeds	Cells	Required	vs Lagrangian	vs PID Lagrangian	Joint result
Hopper/Walker2d	0–2	6	4	6/1	4/4	Partial
All four	0–2	12	8	12/6	10/9	Partial
Hopper/Walker2d	0–4	10	7	10/3	7/6	Not supported
All four	0–4	20	14	20/10	17/15	Partial

Table 30: The four robots of Table 3 (unclipped SafeOT, five seeds per method): mean $\pm$ standard deviation over seeds of the excess cost ($\times 10^3$), of the number of constraints the final policy satisfies and of its final return, with the p-value of Welch's t-test (Appendix D.3) against the best mean of each measure (–: the best mean). The means are those of Table 3.

		Excess cost ($\times 10^3$)		Satisfied constraints		Final return	
Robot	Method	Mean $\pm$ SD	p	Mean $\pm$ SD	p	Mean $\pm$ SD	p
HalfCheetah ($K = 6$)	Lagrangian	26.77 ± 2.00	<0.001	4.0 ± 0.0	<0.001	1842 ± 51	–
	PID Lagrangian	16.36 ± 0.90	0.007	5.0 ± 1.0	0.158	1236 ± 450	0.039
	SafeOT	14.33 ± 0.87	–	5.8 ± 0.4	–	1715 ± 623	0.674
Ant ($K = 8$)	Lagrangian	11.79 ± 2.25	<0.001	6.2 ± 1.3	0.416	2639 ± 319	–
	PID Lagrangian	5.25 ± 0.58	<0.001	6.0 ± 0.7	0.142	2032 ± 260	0.011
	SafeOT	0.77 ± 0.14	–	6.8 ± 0.8	–	2415 ± 330	0.306
Hopper ($K = 3$)	Lagrangian	6.27 ± 2.07	0.006	2.6 ± 0.9	–	458 ± 261	0.031
	PID Lagrangian	1.29 ± 0.23	–	2.2 ± 0.8	0.486	839 ± 196	0.775
	SafeOT	1.49 ± 0.28	0.261	2.2 ± 0.4	0.406	881 ± 250	–
Walker2d ($K = 6$)	Lagrangian	1.73 ± 0.92	0.017	5.0 ± 0.7	0.694	1276 ± 594	–
	PID Lagrangian	1.14 ± 0.88	0.058	5.2 ± 0.8	–	800 ± 185	0.151
	SafeOT	0.10 ± 0.04	–	5.2 ± 0.8	1.000	708 ± 177	0.099

Table 31: Multiple budgets (K: number of budgets). Cells are instance–order pairs of exact tabular CMDPs (two alternation orders of the budgets) or seeded learned-policy runs. Violation ratio: cell-median baseline/SafeOT of the violation summed over epochs and budgets, unnormalized for learned policies and budget-normalized for the tabular CMDPs (above $1\times$ favors SafeOT). Parentheses count cells with lower SafeOT violation. Return kept: cells retaining at least 90% of baseline training return. † favors the baseline. Tabular rows: Appendix G.2. Grid and robot rows use unclipped SafeOT (Appendices G.4–G.6). Merged controls combine all budgets ($K = 1$). –: return is not reported for the tabular CMDPs.

			vs Lagrangian		vs PID Lagrangian	
Setting	K	Cells	Violation ratio (SafeOT lower)	Return kept	Violation ratio (SafeOT lower)	Return kept
Exact tabular	1	40	$15\times$ (40/40)	–	$6.5\times$ (40/40)	–
Exact tabular	4	40	$3.3\times$ (40/40)	–	$2.3\times$ (40/40)	–
Exact tabular	8	40	$11\times$ (40/40)	–	$4.0\times$ (40/40)	–
Learned grid	4	10	$10\times$ (10/10)	9/10	$4.0\times$ (10/10)	9/10
Learned grid	8	10	$9.0\times$ (10/10)	10/10	$8.3\times$ (10/10)	9/10
HalfCheetah	6	5	$1.9\times$ (5/5)	4/5	$1.1\times$ (5/5)	4/5
Ant	8	5	$16\times$ (5/5)	3/5	$6.9\times$ (5/5)	5/5
Hopper	3	5	$3.8\times$ (5/5)	3/5	$0.93\times$ (2/5)†	5/5
Walker2d	6	5	$19\times$ (5/5)	0/5†	$9.0\times$ (5/5)	1/5†
HalfCheetah, merged	1	3	$1.3\times$ (3/3)	3/3	$1.4\times$ (3/3)	3/3
Ant, merged	1	3	$19\times$ (3/3)	1/3†	$12\times$ (3/3)	2/3

Seeds 0–2. Table 32 repeats the Hopper and Walker2d rows of Table 31 for seeds 0–2, which were run before seeds 3–4.

G.6 Merging the Robot Budgets

The merged-budget control retains the robot costs, thresholds, TRPO trainer and five rules but uses $c_\Sigma = \sum_k c_k$ and $B_\Sigma = 1050/1400$ for HalfCheetah/Ant. Seeds 0–2 give 30 completed 2M-step runs with 30 deterministic final episodes each. The advantage over TRPO-PID would count as shrunk if the median paired V ratio of SafeOT to TRPO-PID rose to at least 0.80, from 0.516 in the six-run multiple-budget comparison. Other comparisons are descriptive.

The advantage does not shrink: the ratio is 0.325, with a lower V in all six pairs and a training-return ratio of 1.57. Per robot, the ratios are 0.69 on HalfCheetah and 0.087 on Ant, the inverses of the merged rows of Table 31. The V and return ratios are $0.38/1.09$ against TRPO-Lag and $1.00/1.10$ against the reactive switch with $c = 1$. These paired medians establish neither equivalence nor return non-inferiority. Return uses the last ten training epochs. Final return and cost are evaluated separately on the same policy.

Merging the budgets does not isolate coupling alone. It enlarges the feasible set, allowing one joint to exceed its original budget if another uses less, and changes the excess statistic. For a fixed sequence

Table 32: Primary three-seed multi-budget study. Violation: unnormalized violation summed over epochs and budgets. The ratio and the parenthesized counts of runs with lower SafeOT violation follow Table 31. †: the ratio favors the baseline. Return kept counts cells with at least 90% of baseline training return.

			vs Lagrangian		vs PID Lagrangian	
Setting	K	Cells	Violation ratio (SafeOT lower)	Return kept	Violation ratio (SafeOT lower)	Return kept
Hopper	3	3	$4.0\times$ (3/3)	1/3†	$0.65\times$ (1/3)†	3/3
Walker2d	6	3	$7.4\times$ (3/3)	0/3†	$5.9\times$ (3/3)	1/3†

of constraint costs,

$$\sum_k [J_{k,t} - B_k]_+ \;\geq\; \left[\sum_k J_{k,t} - \sum_k B_k\right]_+ .$$

The policies in the two studies are trained separately, so this inequality is not a comparison of their realized trajectories. The result shows that the observed advantage over TRPO-PID persists under the merged budget. It neither proves coupling is necessary nor identifies an alternative causal mechanism.

The supplementary data contain all 100 logged epochs and 30 final episodes per run.

G.7 Additional Controls

Two soft budgets. PointGoal2 and CarGoal2 use vase and hazard budgets $(5, 25)$, five seeds per task, in-house TRPO and 100 epochs. Here the violation is $\sum_k V_k/B_k$, where V_k is the unnormalized violation of budget k. Its mean is $103/130$ on Point/Car for SafeOT versus $282/299$ for TRPO-Lag, descriptively. Every method still exceeds the vase budget on average on both tasks.

Timing of the TRPO-Lag multiplier. A descriptive TRPO-Lag check assigns 75.4% of V to epochs before its first logged multiplier reaches one (task means 45–100%). In fourteen of the fifty runs the multiplier never reaches one, so all of their V counts as accrued before the crossing. Pooling V over runs gives 62.4%. Averaging crossing runs gives 65.9%. The value one is a reference threshold, not a necessary safety price. Lemma 1 assumes a fixed step, whereas the TRPO-Lag multiplier uses Adam (Table 35), so the lemma does not describe these traces exactly.

G.8 Single-Budget Controller Comparisons

Two follow-ups use the six Bullet tasks, five seeds per task, budget 10, and one million training steps. SafeOT + PID feedback replaces only the feedback term of the default SafeOT trainer. Unclipped SafeOT instead raises the clip level c from 1 to 10^9 and keeps the default feedback and the cap $\Lambda = 100$ on the combined price. Each variant has all 30 runs and 30 deterministic final evaluation episodes per run. Return and cost are measured on the same final policy.

Primary estimates. Table 33 reports both follow-ups. The statistic is the median of six task-level ratios of seed-mean CV, variant over comparator. We use 2,000 paired hierarchical bootstrap draws over tasks and then seeds. Return differences are task-normalized by the greater of the comparator's absolute mean return and the range of the nine original benchmark method means, then aggregated by the task median. A favorable decision requires the ratio interval's upper endpoint below one and normalized return difference at least -0.05. None of the three primary comparisons with TRPO-PID and CPPO-PID (the first three rows of Table 33) meets this requirement. Final policies meet the budget in 21 of 30 runs for SafeOT + PID feedback, 19 for unclipped SafeOT, 17 for the default SafeOT and 21 for each PID Lagrangian comparator.

Comparator and source limits. TRPO-PID is the closest benchmark comparator, not an exact safety-price ablation: cost estimation, first-epoch input, conjugate-gradient iterations and critic learning rates differ.

Table 33: Single-budget follow-ups on Bullet-Safety-Gym. Violation ratio: median over tasks of the CV of the variant divided by that of the comparator (below one favors the variant). Intervals are 95% hierarchical bootstrap intervals. Return differences are descriptive normalized percentage points, not raw relative return gains.

Variant	Comparator	Violation ratio ↓	Return difference ↑
SafeOT + PID feedback	TRPO-PID	1.02 [0.85, 1.42]	$+1.3\%$
SafeOT + PID feedback	CPPO-PID	0.92 [0.51, 1.81]	-4.7%
Unclipped SafeOT	TRPO-PID	1.12 [0.86, 1.43]	$+8.3\%$
SafeOT + PID feedback	SafeOT	0.70 [0.50, 1.06]	-8.7%
Unclipped SafeOT	SafeOT	0.81 [0.52, 1.00]	-0.5%

H Hyperparameter Sensitivity and Compute

H.1 Single-Parameter Sensitivity on Ten Tasks

We vary one parameter around the default $(\varepsilon, N, \xi) = (1, 128, 0.5)$: $N \in \{64, 256\}$, the smoothing weight $\xi \in \{0.25, 1\}$, or $\varepsilon \in \{0.3, 3\}$. Each variant covers all ten primary tasks with five seeds per setting, $c = 1$, $B = 25$, a TRPO actor and 2M steps. The study comprises 300 variant runs and 50 shared default runs. Let bars denote seed means. Training return is the last-ten-epoch logged mean. V sums positive logged-cost excess over all 100 epochs. We report

$$\delta_V = \bar{V}_{\text{variant}}/\bar{V}_{\text{default}} - 1, \qquad \delta_R = \frac{\bar{R}_{\text{variant}} - \bar{R}_{\text{default}}}{|\bar{R}_{\text{default}}|}.$$

Pointwise 95% bootstrap intervals independently resample seeds within each setting (2000 draws). The pre-registered tolerance concerns point estimates: $1 + \delta_V \in [0.67, 1.5]$ and $|\delta_R| \leq 0.2$. The tolerance holds for a variant if all ten tasks are covered and at least eight lie within both bands.

Changing the graph size or the smoothing weight keeps V within its band on all ten tasks, whereas training return leaves its band on three to five tasks. Table 34 counts the tasks within each band, and no variant has eight tasks within both. These are descriptive tolerance counts, not equivalence or parameter-independence results.

Table 34: Sensitivity of the default configuration on the ten Safety-Gymnasium tasks, five seeds per setting: tasks within each band, from seed-mean point estimates. Bands: $1 + \delta_V \in [0.67, 1.5]$ and $|\delta_R| \leq 0.2$.

Variant	V within band	Return within band	Both
$N = 64$	10/10	5/10	5/10
$N = 256$	10/10	7/10	7/10
$\xi = 0.25$	10/10	6/10	6/10
$\xi = 1$	10/10	6/10	6/10
$\varepsilon = 0.3$	6/10	5/10	2/10
$\varepsilon = 3$	9/10	7/10	7/10

Return is more sensitive than V to the tested node-count and smoothing-weight changes. On PointPush1, the two smoothing-weight variants give $\delta_R = -0.571$ $[-0.730, -0.337]$ and -0.577 $[-0.723, -0.346]$, respectively. Relative changes can be large when the default mean return is close to zero, and several intervals are wide. The supplementary data give every default denominator and every absolute return change and interval. Because the bands are two-sided, improvements can fall outside tolerance: $N = 256$ gives Walker2d $\delta_R = 0.824$ $[0.099, 3.414]$, while $\varepsilon = 3$ lowers its V below the lower tolerance boundary. Lowering ε to 0.3 moves the regularization parameter toward the LP limit in Proposition 2. It does not identify a finite-ε update with that limit. Its mean V rises on all ten tasks (relative changes 0.17–1.64).

The default runs are those of Appendix H.2. For $\varepsilon \in \{0.3, 3\}$, this study also requires a final evaluation, so 30 of its runs differ from those of Appendix H.2, and the two studies are reported separately.

These experiments change one parameter at a time and do not establish robustness to joint changes. $N = 32$, unstandardized features and random-projection features were not tested.

H.2 Entropy Sweep on All Ten Tasks

We sweep the entropic regularization on all ten Safety-Gymnasium tasks at $\varepsilon \in \{0, 0.3, 1, 3, 10\}$, $c = 1$, $B = 25$ and 2M steps. Each of the fifty task–setting cells contains seeds 0–4, giving 250 runs and 25,000 logged epochs. We use the same earliest-complete-attempt rule as the benchmark: among 319 available attempts, 58 incomplete attempts and 11 duplicate complete retries are excluded. A run is complete when it has all 100 training epochs. Because 64 of the 250 runs have no final evaluation, this sweep and Table 37 report training metrics only.

For each seed, z is the fraction of epochs with logged cost above 25 whose raw safety price is at most 10^{-6}. We average these conditional fractions across seeds, then report the median and range across

equally weighted tasks. All selected seeds have nonzero denominators (22–100 violating epochs). A future run with no violating epoch would have undefined z, not zero. The logged episode-cost window need not match the current-batch switch input, and raw prices precede clipping and smoothing.

The task-median z at $\varepsilon = 0, 0.3, 1, 3, 10$ is $75.9, 43.4, 20.3, 17.1, 9.8\%$, respectively. The endpoint $V(10)/V(0)$ ratio has median 0.54 and range 0.15–0.95, with lower means on all ten tasks. Relative to $\varepsilon = 1$, the median V ratio is 1.83 at $\varepsilon = 0$, higher on all ten tasks, and 1.05 at $\varepsilon = 10$, higher on seven (Table 37). Nine tasks have monotonically decreasing z. Only CarButton1 and Walker2d have monotonically decreasing V. Hopper's intermediate zero-price fraction increases. The mechanism trend therefore does not imply monotone performance, reward preservation, or a statistical superiority claim. The supplementary data report all settings with uncertainty and the same-run reward/cost trade-offs.

H.3 Controller and Solver Settings

Table 35 lists the safety-controller settings of the Lagrangian and PID Lagrangian methods, the reactive switch and SafeOT in the Safety-Gymnasium benchmark (CPO, PCPO, FOCOPS and CUP use the SafePO defaults), and the paragraphs below give the shared actor settings and the numerical settings of the flow pricer.

The PID controller initializes its integral state from the shared command-line setting, overriding the class default 0.005. Its logged multiplier learning rate is unused. Both PID variants normalize the penalty sum but not its difference, so the optional penalty cap of the PID class is inactive. EMA retention 0.95 assigns weight 0.05 to the current input. The safety-price smoothing assigns weight $\xi = 0.5$. Its first update is initialized directly from the clipped price. The main TRPO implementation returns the sum of the feedback term and the smoothed safety price without a cap on the combined price ($\Lambda = \infty$). The in-house implementation caps the combined price at $\Lambda = 100$ and initializes the feedback term at zero.

Shared actor settings. The matched TRPO family uses trust-region KL 0.01, ten conjugate-gradient steps, damping 0.1 and up to fifteen backtracking steps with factor 0.8. Its critics use Adam rate 10^{-3}, ten passes and minibatch 128. The SafePO PPO family uses actor rate 3×10^{-4} with linear decay, ratio clip 0.2, forty passes, minibatch 64 and KL stopping threshold 0.02. Critic rates are 3×10^{-4}. Both use $[64, 64]$ networks, $\gamma = 0.99$, 20,000 rollout steps per epoch and one environment. Thus benchmark CPPO-PID and matched TRPO-PID isolate different comparisons.

Flow-specific numerical choices. The main flow pricer uses two coordinate sweeps, twelve bisections and at most sixty Newton iterations per potential solve, residual tolerance 10^{-7}, bisection bound $\lambda_{\max} = 20$ in standardized advantage units and budget tolerance 10^{-3}. Multipliers are restored to input-advantage units before clipping. Reference self-loops have mass 10^{-7} on visited nodes. The graph budget is $B/\mathrm{clip}(L, 50, 5000)$. The acceptance rule checks a flow-conservation residual

Table 35: Safety-controller settings on Safety-Gymnasium. Each method keeps the actor settings of the named actor, which the text lists.

Method	Actor	Safety-controller settings
TRPO-Lag	TRPO	Adam multiplier: initial value 0.001, learning rate 0.035, no upper bound.
PPO-Lag	PPO	Same multiplier settings as TRPO-Lag.
CPPO-PID (benchmark)	PPO	$(K_P, K_I, K_D) = (0.1, 0.01, 0.01)$, P/D EMA retention 0.95, derivative delay 10, integral initial value 0.001, initial output 0.
TRPO-PID (matched)	TRPO	Same PID class, gains, filtering and initialization. Same actor as SafeOT.
Reactive switch	TRPO	Raw price $10^6 \mathbf{1}[J_{\mathrm{batch}} > B]$, clip level $c = 1$, smoothing weight $\xi = 0.5$, and the Adam feedback of TRPO-Lag.
SafeOT	TRPO	Adam feedback of TRPO-Lag, clip level $c = 1$, smoothing weight $\xi = 0.5$ with the first output equal to the first clipped price, $\varepsilon = 1$, $N = 128$, edge-statistics decay 0.5 per epoch, clusters refit every 10 epochs.

below 10^{-5}, finite flow and total-mass error below 10^{-2}, separately from budget feasibility. One configuration is reused across the ten tasks after selection on three development tasks. This does not remove graph, solver or controller design choices.

H.4 Solver Timing

Table 36 pools the logged solve times of the in-house implementation at $\varepsilon = 1$ for each node count. The timer covers the solve and immediate result handling, excluding graph construction, rollout collection and policy optimization. These are implementation-specific timings, not total method overhead or a controlled hardware/node-scaling experiment. The pooled medians round to 0.02, 0.06 and 0.24 s for $N = 64, 128, 256$. At $N = 128$, the 95th percentile is 0.16 s and the maximum 2.1 s. Appendix A.8 gives the per-iteration pricing time of the benchmark implementation, which includes graph construction. Pooling all configuration variants would describe a different timing population.

Offline solver benchmark. A separate benchmark re-solves 100 saved training graphs ($N = 128$, one budget) with the training solver settings. The median solve takes 0.028 s on one CPU core and 0.026 s on one A100 GPU, whose float64 port of the solver returns identical prices. The in-training medians above also include immediate result handling and run under training load. With four or eight budgets at $N = 128$, a solve takes 0.6–1.2 s on either device. On synthetic graphs, the GPU is 16–17 times faster than one CPU core at $N = 512$ and 218–238 times faster at $N = 2048$, for one, four or eight budgets. With one budget, a GPU solve stays below 0.3 s up to $N = 2048$. Training used the CPU solver.

H.5 Feedback and Safety-Price Ablations

Ablations of the price rule. Table 37 removes or changes one part of the price rule on the ten Safety-Gymnasium tasks. Removing either part of the combined price raises CV two- to threefold: without the safety price, SafeOT is TRPO-Lag, and without the observed-cost feedback, CV doubles. Within the safety price, what matters is that it is positive when a budget is active. LP prices ($\varepsilon = 0$), which are zero in most over-budget epochs, raise CV, whereas prices near the saturated end, with a large ε or the reactive switch, keep it low. A Lagrangian multiplier that starts at zero cannot provide such a price before violation accumulates (Lemma 1). The last four settings approach one end of the range in Figure 4b. Against the baseline at that end, each has a lower CV on 3 to 10 of the ten tasks and keeps at least 90% of its training return on 5 to 9.

Multi-budget ablations. Table 38 lists the multi-budget ablations: the safety price added to the PID Lagrangian multipliers, the reactive switch at clip level $c = 1$, and the safety price clipped at $c = 1$, each against its reference on HalfCheetah ($K = 6$) and Ant ($K = 8$).

No feedback on four tasks. Table 39 lists the four tasks of the no-feedback row of Table 37 with default SafeOT and TRPO-Lag.

Table 36: In-house solve timings of the default configuration. All runs contribute 100 epochs. At $N = 64, 256$, the sample is PointGoal1, seeds 0–2. At $N = 128$, it also contains PointGoal1 seeds 3–4 and CarGoal1, PointButton1 ($K = 1, 3$) and PointPush1 seeds 0–2. The task mix differs across N.

N	Runs	Epochs	Median (s)	95th percentile (s)	Maximum (s)
64	3	300	0.019	0.036	0.090
128	17	1700	0.062	0.159	2.123
256	3	300	0.238	0.457	1.474

Table 37: Ablations and settings of the price rule (ten Safety-Gymnasium tasks, five seeds). CV ratio: median over tasks of the setting's CV divided by that of default SafeOT, with the tasks on which default SafeOT has the lower CV. Endpoint method: the baseline at the end of the range that the setting approaches (Figure 4b), with the tasks on which the setting has a lower CV or keeps at least 90% of the method's training return. Feedback: observed-cost feedback. §Four tasks, seeds 0–2, compared with the same seeds of default SafeOT. ‡: unclipped, with a halved feedback step. –: reference row or no endpoint method.

	vs default SafeOT	vs endpoint method		
Setting	CV ratio	Method	Lower CV	Return kept
Default ($\varepsilon = 1$, $c = 1$)	$1\times$	–	–	–
No safety price ($c = 0$)	$2.8\times$ (10/10)	identical to TRPO-Lag		
No feedback§	$2.0\times$ (4/4)	–	–	–
LP prices ($\varepsilon = 0$)	$1.8\times$ (10/10)	TRPO-Lag	10/10	5/10
Saturating ($\varepsilon = 10$)	$1.05\times$ (7/10)	CPO	3/10	9/10
Reactive switch	$0.91\times$ (3/10)	CPO	4/10	9/10
Conservative‡	$0.72\times$ (3/10)	CPO	9/10	8/10

Table 38: Multi-budget ablations. Ratio: median over runs of the variant's violation, summed over all budgets, divided by that of the reference (above 1 favors the reference). Two values separated by a slash belong to the two budget counts K listed in the setting, in the same order. Last column: runs in which the reference has the lower violation.

Variant / reference	Setting	Ratio	Reference lower
TRPO-PID / SafeOT + PID feedback	Robots $K = 6/8$	1.16 / 4.91	5/5 / 5/5
Reactive switch ($c = 1$) / unclipped SafeOT	Robots $K = 6/8$	1.03 / 0.84	5/5 / 1/5
SafeOT ($c = 1$) / unclipped SafeOT	Robots $K = 6/8$	1.09 / 1.31	5/5 / 3/5

Table 39: Ablations of SafeOT with the TRPO actor ($\varepsilon = 1$, $c = 1$, $B = 25$, 2M steps): training return / cost and unnormalized violation $V = B \cdot \mathrm{CV}$ in cost units. *No feedback* sets $\beta \equiv 0$. §: seeds 0–2. The other rows average five seeds. Table 37 pairs the no-feedback runs with the default runs of the same three seeds, which gives its ratio of 2.03.

	PointGoal1		CarGoal1		Ant		Hopper	
Variant	R/C	V	R/C	V	R/C	V	R/C	V
Default	14.9 / 26.6	1065	19.8 / 26.0	678	2791 / 31.1	297	1588 / 23.6	585
No feedback§	24.7 / 46.5	2338	28.6 / 40.4	1834	2767 / 36.0	396	1597 / 39.6	921
TRPO-Lag	24.2 / 41.0	2369	26.9 / 43.9	2746	2766 / 27.5	2055	383 / 8.6	2253

I Hard and Soft Constraints in the Executor

On-policy exclusion. With margin 0.8, 2M steps and three seeds, on-policy PPO-based controllers that exclude the edges in H are unsafe or earn little return (supplementary data). We therefore use a separate off-policy learner with critic-guided selection (Appendix A.9). Excluding edges from the graph does not by itself ensure policy safety.

I.1 Fixed Margins and Margin Schedules

PointGoal2 fixed-margin controls use $d = 0.85$, budgets $5/25$ and five seeds.

Table 41 uses the 312-episode, seed-level interval protocol of Appendix C. These comparisons do not isolate pricing. The Car/Point endpoints are $0.78/0.85$. These policies differ from the uniform-schedule diagnostic.

Table 40: **Hard contact and soft hazard constraints on Goal2 tasks** (CG2: CarGoal2, PG2: PointGoal2, RG2: RacecarGoal2). Five seeds and 312 evaluation episodes per seed, scored on native contacts. Return and hazard cost: means $\pm$ 95% t half-width over seeds. Contact: seed-bootstrap 95% interval. SafeOT, SDAC and SRCPO use critic-guided selection with their own critics. SafeOT anneals the margin from 0.95 to 0.85, and SDAC and SRCPO train with the fixed margin 0.8. Sauté RL (remaining contact budget 0), the Safety Layer and ET-MDP (training episodes end at the first contact) train a SAC learner with a Lagrange multiplier for the hazard budget 25 at the fixed margin 0.8, without selection. Feasible seeds: mean hazard cost at most 25.

Task	Method	Return ↑	Contact % ↓ [95% CI]	Hazard ↓	Feasible seeds
CG2	SafeOT	23.3 ± 1.4	$1.67\,[0.90, 2.44]$	22.8 ± 5.0	3/5
CG2	SDAC	-0.7 ± 0.4	$3.46\,[2.37, 4.62]$	17.9 ± 13.4	4/5
CG2	SRCPO	0.2 ± 0.2	$3.72\,[2.76, 4.81]$	33.0 ± 12.9	1/5
CG2	Sauté RL	-2.9 ± 5.0	$12.56\,[5.13, 20.51]$	11.8 ± 9.0	5/5
CG2	Safety Layer	-0.7 ± 1.0	$22.12\,[13.40, 32.05]$	21.7 ± 16.8	3/5
CG2	ET-MDP	-0.9 ± 0.3	$2.95\,[1.86, 4.04]$	10.7 ± 8.0	5/5
PG2	SafeOT	14.7 ± 0.6	$0.77\,[0.26, 1.28]$	22.6 ± 5.7	4/5
PG2	SDAC	-1.0 ± 0.5	$5.83\,[3.65, 8.78]$	7.6 ± 6.3	5/5
PG2	SRCPO	0.1 ± 0.2	$4.62\,[3.21, 6.54]$	37.3 ± 10.6	1/5
PG2	Sauté RL	-4.2 ± 4.5	$22.95\,[13.59, 32.31]$	34.0 ± 14.4	1/5
PG2	Safety Layer	-0.3 ± 0.4	$13.33\,[8.78, 17.88]$	27.6 ± 13.5	3/5
PG2	ET-MDP	-3.6 ± 3.0	$1.92\,[1.03, 2.82]$	15.1 ± 5.7	5/5
RG2	SafeOT	0.5 ± 0.4	$3.14\,[1.03, 5.38]$	4.1 ± 5.9	5/5
RG2	SDAC	-1.2 ± 0.3	$11.15\,[6.86, 15.71]$	16.1 ± 8.5	5/5
RG2	SRCPO	0.0 ± 0.1	$7.24\,[6.41, 8.14]$	31.3 ± 21.3	3/5
RG2	Sauté RL	-0.5 ± 0.4	$13.33\,[9.10, 17.56]$	19.8 ± 5.1	4/5
RG2	Safety Layer	-0.1 ± 0.1	$12.50\,[11.03, 14.49]$	23.6 ± 13.7	2/5
RG2	ET-MDP	-0.1 ± 0.0	$1.60\,[1.22, 1.99]$	15.9 ± 8.2	5/5

Table 41: Hard-constraint systems without critic-guided selection. Return and cost are reported as means $\pm$ 95% t half-width. Contact is reported with a seed-bootstrap 95% interval. Fixed margins CG2 0.8/PG2 0.85. Selected schedules CG2 $0.95 \to 0.78$ (1M), PG2 $0.95 \to 0.85$ (1.5M). Feasible: seeds with mean hazard cost at most 25. On CG2, SDAC and SRCPO have three seeds.

Task	Method	n	Return	Contact (%) [95% CI]	Hazard cost	Feasible
CG2	SDAC	3	-5.7 ± 15.0	1.92 [1.28, 2.24]	10.0 ± 23.5	3/3
CG2	SRCPO	3	-0.1 ± 0.1	3.74 [2.24, 5.45]	25.7 ± 34.0	2/3
CG2	SafeOT, fixed d	5	19.0 ± 2.2	3.72 [2.05, 5.00]	24.2 ± 8.5	2/5
CG2	SafeOT, scheduled d	5	14.2 ± 2.6	1.41 [1.15, 1.60]	21.4 ± 5.4	4/5
PG2	SDAC	5	-1.2 ± 1.3	5.45 [3.08, 7.95]	17.2 ± 8.5	4/5
PG2	SRCPO	5	0.3 ± 0.5	5.45 [3.33, 7.76]	23.7 ± 13.6	2/5
PG2	SafeOT, fixed d	5	13.1 ± 1.7	2.69 [1.60, 3.46]	18.0 ± 3.3	5/5
PG2	SafeOT, scheduled d	5	15.5 ± 1.1	1.60 [0.96, 2.24]	24.4 ± 6.8	4/5

I.2 Uniform-Schedule Diagnostic

All three robots use margin $0.95 \to 0.85$ over 1.5M steps, hazard budget 25 and sixteen sampled candidates plus the deterministic action. Five seeds each are evaluated in three 104-episode blocks at the final checkpoint (about 2M steps). Table 40 compares these policies with SDAC and SRCPO under their own shields, Sauté RL, the Safety Layer and ET-MDP on all three robots. On Car, the common-schedule policy without selection gives return 25.83 and $150/1560$ contact episodes (9.62%), more than ET-MDP (2.95% in Table 40). Critic-guided selection gives 23.33 and $26/1560$ (1.67%). On Point the corresponding values are 15.46, $25/1560$ (1.60%), and 14.72, $12/1560$ (0.77%). With selection, the hazard means are $22.83/22.60$, with $3/5$ and $4/5$ feasible seeds. The pooled CP95 upper bounds are $2.31\%/1.24\%$. These nominal bounds ignore shared policies/reset

instances. Table intervals resample five seed policies. Paired differences are in the supplementary data.

Car's random-candidate control uses the 0.78 schedule, so its attribution does not transfer to 0.85. Point's records with selection are unchanged. Selection often falls back to minimum predicted risk, not exact support exclusion. Different critic calibrations and margins prevent a pricing-only comparison.

Training differences. CarGoal2 and PointGoal2 do not use the episode-start repair. RacecarGoal2 uses it for the typed variants but not for the untyped control. These training differences preclude a clean cross-robot or typed–untyped attribution.

I.3 Own-Critic Shield Controls

On Car/Point, each SDAC/SRCPO actor's critic selects among sixteen samples and the deterministic action. Five seeds, fixed margin 0.8, budgets $5/25$ and three 104-episode blocks give 6240 episodes. The score is the maximum mean cost quantile across critics. Threshold 0.02 is uncalibrated discounted cost, not contact probability.

Table 40 contains these four shield rows and the two RacecarGoal2 shields, which train with the same fixed margin. Without selection, the task-selected schedules of SafeOT (Table 41) have a higher return and a lower mean contact rate than these shields, with different margins and critic calibration. Their hazard costs are higher than those of SDAC, so neither dominates in all costs.

I.4 Random Policy-Sample Control

Table 42 compares three execution rules on the same policies. Five fixed checkpoints per task compare deterministic execution, critic-guided selection, and the first stochastic policy sample with critics ignored. The last rule is not a uniform draw from the full candidate set. All three rules use the same 16 candidates and policy standard deviation.

Table 42: Three execution rules on the task-selected schedules of Table 41 (CarGoal2 $0.95 \to 0.78$, unlike Table 40): five seeds, three blocks of 104 episodes per seed. Return uncertainty is the s.e. across seeds. Contact bounds are pooled one-sided CP95 upper bounds. All three rules have four of five feasible seeds (mean hazard cost at most 25).

Task	Execution	Return $\pm$ s.e.	Contacts / episodes	Contact % (CP95)	Hazard cost
CarGoal2	Deterministic	14.24 ± 0.92	22/1560	1.41 (2.01)	21.40
CarGoal2	Critic-guided	12.33 ± 0.76	4/1560	0.26 (0.59)	21.43
CarGoal2	First policy sample	14.61 ± 0.80	36/1560	2.31 (3.04)	23.12
PointGoal2	Deterministic	15.46 ± 0.40	25/1560	1.60 (2.23)	24.43
PointGoal2	Critic-guided	14.72 ± 0.21	12/1560	0.77 (1.24)	22.60
PointGoal2	First policy sample	16.13 ± 0.32	28/1560	1.79 (2.45)	24.34

Compared with random execution, critic-guided selection reduces contacts in all five Car and four Point seed pairs, with a lower return in all ten. Paired contact changes are -2.05 ± 0.47 and -1.03 ± 0.42 percentage points. Random execution has a higher aggregate return and slightly higher contact than deterministic execution. This supports selection over sampling at a reward cost, not a graph-pricing attribution.

Pairing. Pairing uses seed, block and serial index.

I.5 RacecarGoal2: Native-Contact Evaluation

Critic-guided selection gives a return of 0.46 at $49/1560$ contacts (3.14%), against 0.51 and $51/1560$ (3.27%) without selection. With selection, all five seeds are feasible. Returns and contact improvements remain small. Table 43 reports all nine completed execution settings. The SDAC/SRCPO

records here use deterministic execution without shields. Their RacecarGoal2 runs with shields are in Table 40.

Table 43: RacecarGoal2. Each row evaluates five final checkpoints on 312 native episodes per seed. Return: mean ± seed s.e. Contact: pooled rate and nominal one-sided CP95 upper bound. Hazard: mean cost. Only rows marked selection use critic-guided selection. Fixed 0.8 and $0.95 \to 0.85$: training margin of SafeOT. Native cost and margin 0.8: SDAC and SRCPO trained on the native contact cost or with margin 0.8. Untyped FDPI: FDPI with the native aggregate cost, without margin or pricing (Appendix C).

Method / execution	Return ↑	Contact % ↓	CP95 upper %	Hazard ↓
SafeOT, fixed 0.8	0.27 ± 0.23	1.22	1.78	1.91
SafeOT, $0.95 \to 0.85$	0.51 ± 0.16	3.27	4.11	3.41
SafeOT, $0.95 \to 0.85$, selection	0.46 ± 0.15	3.14	3.97	4.06
SafeOT, fixed 0.8, selection	0.21 ± 0.19	1.22	1.78	3.00
Untyped FDPI (aggregate cost)	0.51 ± 0.19	3.53	4.39	0.26
SDAC, native cost	0.23 ± 0.08	7.76	8.96	27.91
SRCPO, native cost	1.43 ± 0.18	27.50	29.42	34.28
SDAC, margin 0.8	-2.68 ± 0.96	10.96	12.35	11.83
SRCPO, margin 0.8	-0.06 ± 0.03	6.15	7.25	38.71

The ten typed runs use repaired episode starts. The five untyped and earlier Car/Point runs do not, precluding a typing-only comparison.

I.6 Exploratory Deployment on a Vision-Language-Action Policy

As an exploration beyond the benchmark suites, we apply an entropic projection of the same form at deployment to the action probabilities of a fixed SafeVLA policy (Zhang et al., 2025a) on the Safety-CHORES tasks ObjectNav, Pickup and Fetch, without retraining. The projection uses static per-action cost priors for the five cost types of the benchmark (danger, corner, blind, fragile and critical) with budgets $(5, 5, 3, 3, 2)$. Table 44 compares the projected policy with the unchanged one on paired episodes of four evaluation seeds. The mean episode cost is lower on all three tasks and in 10 of the 12 task–seed blocks, and the success rate changes by 0.0, -1.5 and $+2.3$ percentage points. The 95% intervals of the cost differences include zero, so this exploratory result does not establish a reliable improvement.

Table 44: Exploratory deployment on a fixed SafeVLA policy: unchanged policy (A) against the projected policy (B), with paired episodes pooled over four evaluation seeds. Cost: mean episode cost. Δ: B minus A. The Δcost interval is a 95% paired-bootstrap interval ($10{,}000$ resamples) that ignores clustering by scene and seed.

Task	Episodes	Cost A / B	Δcost [95% CI]	Success A / B (%)	Δsuccess (points)
ObjectNav	800	1.26 / 0.98	-0.29 [-1.40, 0.53]	84.8 / 84.8	0.0
Pickup	684	0.47 / 0.42	-0.05 [-0.13, 0.02]	93.3 / 91.8	-1.5
Fetch	688	9.90 / 9.18	-0.72 [-2.90, 1.56]	63.8 / 66.1	$+2.3$

J Related Work

Safe RL methods. Safe RL methods scalarize costs into the return with fixed weights (Zhang et al., 2022), adapt one multiplier per budget by integral, PID or other feedback on observed costs (Tessler et al., 2019; Stooke et al., 2020; Spoor et al., 2026; Zhang et al., 2026a), or solve a constrained local surrogate within a trust region (Achiam et al., 2017; Yang et al., 2020; Zhang et al., 2020; Yang et al., 2022). State-augmented and state-wise methods target almost-sure or high-probability safety (Sootla et al., 2022; Zhao et al., 2024). Constrained variational policy optimization (CVPO) fits the policy to a constrained variational distribution (Liu et al., 2022). The safe distributional actor-critic (SDAC) (Kim et al., 2023) and spectral-risk-constrained policy optimization (SRCPO) (Kim et al.,

2024) constrain risk measures of the cost with distributional critics, and we adapt both as baselines for the hard-constraint experiments (Appendix I.3), together with Sauté RL, the Safety Layer and ET-MDP (Sun et al., 2021), which ends each training episode at the first violation (Table 40). SafeOT keeps the penalized-advantage update of Lagrangian methods but changes where its multipliers come from: one flow projection over all budgets supplies the prices, and observed-cost feedback, integral or PID, is kept and added to these prices.

Optimal transport and reinforcement learning. Existing OT–RL work uses transport as a distributional loss, a policy-update geometry, a robustness perturbation or a safety guide for temporal-difference learning (Sun et al., 2024; Pfau et al., 2025; Queeney et al., 2024; Shahrooei & Baheri, 2025), and occupancy methods optimize stationary distributions or their corrections (Givchi et al., 2021; Lee et al., 2021; 2022). SafeOT builds on the occupancy linear programs of MDPs and CMDPs (Manne, 1960; d'Epenoux, 1963; Puterman, 1994; Altman, 1999) and on the entropic optimal-flow template of OFT (Shi et al., 2025; Cuturi, 2013). The new ingredient is a discounted flow projection on the empirical transition graph whose multipliers for the budget constraints act as safety prices in an online policy update. The policy is never asked to match the projected flow.

Entropic regularization and embodied safety. Entropy-regularized MDPs relate policy optimization to convex regularization and mirror descent (Neu et al., 2017; Geist et al., 2019), and the same deterministic-to-entropic interpolation underlies Schrödinger bridges, rectified flows and OT flow matching (De Bortoli et al., 2024; Liu et al., 2023; Shi et al., 2023; Tong et al., 2024). In SafeOT, ε sets the entropic regularization of a constrained transition flow, and its limits give LP and active-set prices (Proposition 2). Safety layers, recovery policies, shields and barrier filters intervene at action time (Dalal et al., 2018; Thananjeyan et al., 2021; Alshiekh et al., 2018; Cheng et al., 2019), while SafeVLA, SafeDreamer and constrained diffusion planners bring constraints to vision-language-action policies, world models and generated plans (Zhang et al., 2025a; Huang et al., 2024; Zhang et al., 2025b). SafeOT acts on the learning update instead and needs only rollout data. For hard constraints at action time, we pair it with an executor based on feasible dual policy iteration (FDPI) (Yang et al., 2026).

K Numerical Checks of the Flow Projection

Random-chain checks test the closed form, dual limits and local expansion using tighter tolerances than training. Training runs did not store solver status flags, so the checks below re-solve saved graphs offline. Undiscounted length-1000 counts have a discounted flow-conservation mismatch of about 1.6% of flow. Complete settings, numerical tables and code are in the supplement.

Four seed-0 training trajectories (PointGoal1, CarGoal1, HalfCheetah, Hopper, 100 graphs each) are re-solved offline. Conditional on $\theta^I > 10^{-9}$, a tenfold increase in ε reduces the median error $|\lambda^\star_\varepsilon/\varepsilon - \theta^I|$ by approximately tenfold. Solving the linear program ($\varepsilon = 0$) with the HiGHS solver on every second epoch returns an optimal status for all 120 graphs whose observed flow violates the budget. Zero prices occur in 62.5–87.5%, always with slack budgets. Removing the added self-loops changes neither outcome, and their flow is zero. Thus budget inactivity explains these selected zero prices, not solver failure or loop routing. These checks concern the graph problem, not policy feasibility.

At 1% budget perturbation, local-expansion median errors are 0.9–2.2% on balanced references versus 4.4–35.7% on raw estimates, conditional on positive multipliers. This discrepancy limits use of the local approximation on implemented inputs. The supplementary data give all denominators, residual tails and additional perturbations.