Multi-Constraint Safe RL with Objective Suppression for Safety-Critical Applications

Zihan Zhou
University of Toronto
Vector Institute
footoredo@gmail.com

Jonathan Booher
Nuro Inc.
jbooher@nuro.ai

Wei Liu
Nuro Inc.
w@nuro.ai

Aleksandr Petiushko
Nuro Inc.
apetiushko@nuro.ai

Animesh Garg
University of Toronto
Vector Institute
Nvidia
garg@cs.toronto.edu

Abstract

Safe reinforcement learning tasks with multiple constraints are a challenging domain despite being very common in the real world. To address this challenge, we propose Objective Suppression, a novel method that adaptively suppresses the task reward maximizing objectives according to a safety critic. We benchmark Objective Suppression in two multi-constraint safety domains, including an autonomous driving domain where any incorrect behavior can lead to disastrous consequences. Empirically, we demonstrate that our proposed method, when combined with existing safe RL algorithms, can match the task reward achieved by our baselines with significantly fewer constraint violations.

1 Introduction

Reinforcement learning (RL) is a general approach to solving challenging tasks in many domains such as robotics, navigation, and even generative modeling. Policies learned with RL seek to maximize a "task reward" which can be specified either via manually defined functions or through learned models. However, without careful tuning of this reward function, it can be difficult for policies to learn to perform well in safety-critical situations like those in the autonomous driving domain. In order to reliably use RL policies in safety-critical situations, we consider the use of constrained RL in order to prevent the policy from exhibiting dangerous behavior. Most prior works on constrained or safe RL have only considered a single constraint violation; however, in most real-world settings there are often multiple constraints that can even be conflicting, which is under-explored in prior work. Take for instance the case of autonomous driving, two very simple constraints are (1) avoiding collision and (2) maintaining a buffer distance from static objects. In some cases, it is impossible to satisfy both constraints while also simultaneously making progress along the ego route.

Existing safe RL works rarely address the multi-constraint issue. Methods that rely on linearly combining the task reward and constraints can struggle to assign a set of weights for all the constraints without some of the constraints being overshadowed by others; hierarchical methods, on the other hand, can face difficulties in building multiple hierarchies. In light of this, we present a method called Objective Suppression that makes adaptive choices of suppressing and balancing the task reward objective and constraint-satisfying objectives. Objective Suppression can be easily combined with existing safe RL approaches like Recovery RL as a new regime of constraint-enforcing methods, which is shown to help policies handle multiple conflicting constraints.

Symposium on Machine Learning for Autonomous Driving (ML4AD 2023).

We empirically test our method in two challenging domains featuring multiple constraints: a *Mujoco-Ant* [17, 4] domain with dynamic obstacles, where our method lowers the number of collisions by 33%; and the *Safe Bench* [19] domain, where our method reduces the constraint violations by at least half. In both domains, our method achieves the results without significant sacrifice of task reward.

2 Related Works

Lagrangian Methods Lagrangian relaxation [2, 3] uses a primal-dual method to turn the constrained optimization problem of safe RL into an unconstrained one. [6] demonstrates Lagrangian relaxation can be adopted to safe RL problems and achieve satisfying empirical performances. RCPO [15] turns Lagrangian multipliers into reward penalty weights to reach constrained goals with both theoretical and empirical evidence.

Hierarchical Methods Another way of solving safe RL problems is to apply a safety layer [7] on top of the task reward-maximizing policy. [7] derives a closed-form solution for action correction by learning a linearized model. Recovery RL [16] learns a parameterized recovery policy that leads the agent away from dangerous areas.

Learning in Autonomous Vehicles Most prior work in learning-based methods in AV relies on behavior cloning approaches [11, 14]. Some methods add layers of hierarchy [5] in order to learn improved BC policies but can still struggle with safety. Pure cloning-based approaches have been demonstrated to be insufficient to handle long-tail cases or ensure safety [12, 10] which motivates the exploration of RL-based approaches.

3 Preliminaries

Constrained MDPs We describe a safe RL problem under the assumption of a Constrained Markov Decision Process (CMDP, [2, 9, 1]), where the agent is required to maximize its expected return while ensuring all the safety constraints are met. Formally, a CMDP is defined as a 7-tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma, \mathcal{C}, \epsilon, \mu_0)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the stochastic state transition function, $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the task reward function, γ is the discount factor, $\mathcal{C} = \{(C_i : \mathcal{S} \to \{0, 1\}, \gamma_{C_i})\}$ is the set of constraints and their corresponding discount factors, $\epsilon \in \mathbb{R}^+$ is the constraint violation limit, and $\mu_0 : \mathcal{S} \to [0, 1]$ is the initial state distribution. Let $n = |\mathcal{C}|$ denote the number of constraints. In the context of safe RL, these constraints are also *risks*.

For a stochastic policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, the task state value function, state-action value function, and advantage function are:

$$V_R^\pi(s) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})|s_0 = s\right] \tag{1}$$

$$Q_R^\pi(s, a) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})|s_0 = s, a_0 = a\right] \tag{2}$$

$$A_R^\pi(s, a) := Q^\pi(s, a) - V^\pi(s) \tag{3}$$

$$\tag{4}$$

Similarly, the state value function, state-action value function, and advantage function for risk constraints $i = 1, \dots n$ are:

$$V_{C_i}^{\pi}(s) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma_{C_i}^{t}C_i(s_t)|s_0=s\right] \tag{5}$$

$$Q_{C_i}^{\pi}(s,a) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma_{C_i}^{t}C_i(s_t)|s_0=s,a_0=a\right] \tag{6}$$

$$A_{C_i}^{\pi}(s,a) := Q_{C_i}^{\pi}(s,a) - V_{C_i}^{\pi}(s) \tag{7}$$

Let $\mathcal{J}_R^{\pi} = \mathbb{E}_{s\sim\mu_0}\left[V_R^{\pi}(s)\right]$ and $\mathcal{J}_{C_i}^{\pi} = \mathbb{E}_{s\sim\mu_0}\left[V_{C_i}^{\pi}(s)\right]$ for $i = 1,\ldots,n$. The objective of a safe RL problem is to solve the following constrained optimization problem:

$$\pi^* := \arg\max_{\pi}\mathcal{J}_R^{\pi} \quad \text{where} \quad \mathcal{J}_{C_i}^{\pi} \le \epsilon \quad \text{for all } i = 1,\ldots,n \tag{8}$$

Hierarchical methods One regime of solving CMDPs is to apply a safety layer to adjust the unsafe actions [7, 16, 20]. The safety layer overwrites the proposed actions that are deemed unsafe according to the safety critic $Q_{C_i}^{\pi}$:

$$a_t = \begin{cases} a\sim\pi(s) & \text{if } Q_{C_i}^{\pi}(s_t,a)\le\epsilon \text{ for all } i\in\{1,\ldots,n\},\\ a\sim\sigma(s) & \text{otherwise.}\end{cases} \tag{9}$$

$\sigma : \mathcal{S}\times\mathcal{A}\to[0,1]$ is a safety-ensuring policy. One choice of σ is to project the actions into a feasible space [13, 7], e.g., $\sigma(s) = \arg\min_a \|a-\tilde{a}\|$ s.t. $Q_{C_i}^{\pi}(s_t,a)\le\epsilon$ for all $i\in\{1,\ldots,n\}$, where $\tilde{a}\sim\pi(s)$. Another choice is to use a separate parameterized policy [16, 20]. Recovery RL [16] trains a recovery policy to minimize the constraint violations, i.e., $\sigma^* = \arg\min_\sigma \sum_{i=1}^{n} w_i\mathcal{J}_{C_i}^{(\pi,\sigma)}$.

Policy parameterization In this work we consider parameterized policies denoted as π_θ. The derived gradients [18] of $\mathcal{J}_R^{\pi_\theta}$ and $\mathcal{J}_{C_i}^{\pi_\theta}$ are:

$$\nabla_\theta\mathcal{J}_R^{\pi_\theta} = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{\infty}A_R(s_t,a_t)\nabla_\theta\log\pi(s_t,a_t;\theta)\right], \tag{10}$$

and

$$\nabla_\theta\mathcal{J}_{C_i}^{\pi_\theta} = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{\infty}A_{C_i}(s_t,a_t)\nabla_\theta\log\pi(s_t,a_t;\theta)\right]. \tag{11}$$

4 Objective Suppression

We observe that previous safe RL algorithms struggle in multi-constraint scenarios. We propose a new method to enforce safety constraints on policy optimization based on adaptively suppressing the task reward objectives that combines well with other safe RL algorithms such as Recovery RL in multiple constraint scenarios.

Objective suppression We propose a new method to enforce safety constraints on policy optimization. To solve (8), we want to train a policy that automatically switches between optimizing for task reward objective $\mathcal{J}_R^{\pi}$ and risk minimization objective $\mathcal{J}_{C_i}^{\pi}$. One way to accomplish this is to switch the optimization objective in hindsight. Specifically, at step t of a trajectory τ, if any risk is encountered after t, we then switch to the risk minimization objective; otherwise, we remain using the original task reward objective. Let $z_{t,i}(\tau) = \mathbb{1}\left[\tau \text{ encounters risk } i \text{ after step } t\right]$ and $z_{t,-}(\tau) = \mathbb{1}\left[\tau \text{ encounters no risk after step } t\right] = \prod_i 1 - z_{t,i}(\tau)$. The hard-switching objective is

$$\nabla_\theta\mathcal{J}_{switch}^{\pi_\theta} = \mathbb{E}_\tau\left[\sum_{t=0}^{\infty}\left(z_{t,-}(\tau)A_R(s_t,a_t) - \sum_{i=1}^{n}w_i z_{t,i}(\tau)A_{C_i}(s_t,a_t)\right)\nabla_\theta\log\pi(s_t,a_t;\theta)\right] \tag{12}$$

$w_1, \ldots, w_n$ are the weights to balance the different risk minimization objectives. The hard-switching objective (12) is an interpolation of $\nabla_\theta \mathcal{J}_R^{\pi_\theta}$ and $\nabla_\theta \mathcal{J}_{C_i}^{\pi_\theta}$. However, this objective faces problems in practice. For one, the hard-switching between multiple objectives raises the variance of the gradient, which is widely acknowledged to have a negative impact on training; for another, balancing the different objectives with weights introduces a new set of hyperparameters. If handled improperly, some of the objectives will become dominant of others and can result in disastrous outcomes.

To deal with the aforementioned two problems, we rewrite (12) into:

$$\nabla_\theta \mathcal{J}_{switch}^{\pi_\theta} = \sum_{t=0}^{\infty} \mathbb{E}_\tau \left[\left(z_{t,-}(\tau) A_R(s_t, a_t) - \sum_{i=1}^{n} w_i z_{t,i}(\tau) A_{C_i}(s_t, a_t) \right) \nabla_\theta \log \pi(s_t, a_t; \theta) \right] \tag{13}$$

$$= \sum_{t=0}^{\infty} \mathbb{E}_{s_t, a_t} \left[\left(\mathbb{E}_\tau \left[z_{t,-}(\tau) \right] A_R(s_t, a_t) - \sum_{i=1}^{n} w_i \, \mathbb{E}_\tau \left[z_{t,i}(\tau) \right] A_{C_i}(s_t, a_t) \right) \nabla_\theta \log \pi(s_t, a_t; \theta) \right] \tag{14}$$

The first part $\mathbb{E}_\tau \left[z_{t,-}(\tau) \right]$ is the probability of not encountering any risks after taking (s_t, a_t). We provide an estimation of this with a transformed summation of the risk critics, denoted $\hat{z}_-(s_t, a_t) = e^{-\kappa \sum_i Q_{C_i}(s_t, a_t)}$. For the second part $\mathbb{E}_\tau \left[z_{t,i}(\tau) \right]$, similar to the first part, we use $\hat{z}_i(s_t, a_t) = Q_{C_i}(s_t, a_t)$ as an estimation. We conclude our objective suppression gradient:

$$\nabla_\theta \mathcal{J}_{supp}^{\pi_\theta} = \mathbb{E}_\tau \left[\sum_{t=0}^{\infty} \left(\hat{z}_-(s_t, a_t) A_R(s_t, a_t) - \sum_{i=1}^{n} w_i \hat{z}_i(s_t, a_t) A_{C_i}(s_t, a_t) \right) \nabla_\theta \log \pi(s_t, a_t; \theta) \right] \tag{15}$$

Combining with existing safe RL algorithms We empirically find that Objective Suppression works better when combined with other safe RL algorithms. We conjecture that this is because applying multiple regimes of constraint enforcement increases the coverage of different constraints, preventing certain constraints from being dominated by others in one regime. In our experiments, we build our method on top of Recovery RL by enforcing our objective suppression objective (15) on both the task and recovery policy.

The original Recovery RL formulation relied on a pre-training stage in order to train the risk critic from demonstrations. This ensures that the method is safe *during exploration*; however, we use a purely online version of Recovery RL where the risk critic is trained jointly with the policy.

5 Experiments

5.1 Environments and baselines

Safe Mujoco-Ant This environment is adapted from the 8-DOF *ant-v4* environment from gym [4] Mujoco [17]. The main task of the agent is to control the ant to reach a target point. There are two constraints in this environment. The first constraint is to avoid randomly spawned obstacles and the second is to avoid collapsing.

Safe Bench This environment is a CARLA [8] based benchmark for safety in various different driving scenarios [19]. The task reward comes from making progress along the designated route while the constraints come from (1) collisions with obstacles (e.g., pedestrians, curbs, etc) which will terminate the episode, and (2) leaving the lane, i.e. lateral deviation from the lane of travel. The policy is given access to a bird-eve-view rendering of the scene in addition to a 4-D observation space covering lane placement, speed, and the distances to objects. The action space is a continuous 2-D space consisting of control parameters: acceleration and steering angle.

Baselines We test our objective suppression method in the two aforementioned environments, *Safe Mujoco-Ant* and *Safe Bench*. We also implement two baselines, a naive *Reward Penalty* baseline, where the optimization objective is a fine-tuned weighted sum of the task reward objective and risk minimization objective; and *Recovery RL* [16].

Table 1: Results in *Safe Mujoco-Ant* collected from 5 seeds with 20m environment steps each. Standard deviations are shown in parentheses.

Name	Task Reward ↑	Collisions ↓	Collapse ↓
Reward Penalty	1133.63 (321.17)	69.70 (32.68)	0.39 (0.10)
Recovery RL	1279.62 (143.40)	6.47 (1.88)	0.42 (0.20)
Ours ($\kappa = 3$)	1215.49 (143.40)	4.36 (0.75)	0.43 (0.08)
Ours ($\kappa = 1$)	1175.79 (172.13)	2.93 (0.67)	0.54 (0.12)

Table 2: Results in *Safe Bench*. All rows given 200k environment steps.

Name	Task Reward ↑	Collisions ↓	Out-of-Lane ↓
Recovery RL	135.98	0.22	44.28
Ours	109.87	0.07	22.85

5.2 Results

Safe Mujoco-Ant We compare our method with *Reward Penalty* and *Recovery RL*. Every experiment is run with 5 seeds. The results are shown in Table 1. For *Reward Penalty*, even with a hyperparameter sweep, we could not find a suitable set of weights to reduce the number of collisions without sacrificing too much task reward. Compared with *Recovery RL*, our method with $\kappa = 3$ achieves 33% less collisions at a mere expense of 5% less task reward.

We notice from our experimentation that although Recovery RL is able to effectively lower the collisions compared with the Reward Penalty baseline thanks to its policy switching mechanism, it struggles in training a good recovery policy. Even though policy switching eliminates the need to optimize for both task reward and constraint avoidance, in our setting, the two constraint-minimization objectives still constitute a conflicting training objective for the recovery policy, which is further exacerbated by the lack of on-policy examples. In fact, one extremely sensitive hyperparameter to tune for Recovery RL is the weight of the collapse-avoiding objective for the recovery policy. With a high weight, the recovery policy ignores the collision-avoiding objective, resulting in a surge in collisions; with a low weight, the recovery policy tends to collapse in front of obstacles, resulting in a surge in collapsed finishes. The introduction of Objective Suppression effectively alleviates the problem, striking a balance between the two constraints for the recovery policy. This demonstrates how our method can shine in multi-constraint scenarios.

Safe Bench We compare our method with *Recovery RL*. Each method is evaluated using 50 rollouts from the policy in different environments. The results are shown in Table 2. Compared with the baseline *Recoevery RL*, our method outperforms on constraint satisfaction while only incurring a small decrease in task reward.

6 Conclusion

We propose Objective Suppression, a novel algorithm that adaptively suppresses the task reward objective to enforce safety constraints. Combined with existing safe RL algorithms, we demonstrate that Objective Suppression can maintain the task reward of the base algorithm while significantly lowering the constraint violations in multi-constraint scenarios, including an autonomous driving domain where incorrect behaviors can be disastrous.

References

[1] J. Achiam, D. Held, A. Tamar, and P. Abbeel. Constrained policy optimization. *ArXiv*, abs/1705.10528, 2017. URL https://api.semanticscholar.org/CorpusID:10647707.

[2] E. Altman. Constrained markov decision processes. volume 7, 1999. URL https://api.semanticscholar.org/CorpusID:14906227.

[3] D. Bertsekas. *Nonlinear Programming*. Athena scientific optimization and computation series. Athena Scientific, 2016. ISBN 9781886529052. URL https://books.google.ca/books?id=TwOujgEACAAJ.

[4] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym, 2016.

[5] E. Bronstein, M. Palatucci, D. Notz, B. White, A. Kuefler, Y. Lu, S. Paul, P. Nikdel, P. Mougin, H. Chen, J. Fu, A. Abrams, P. Shah, E. Racah, B. Frenkel, S. Whiteson, and D. Anguelov. Hierarchical model-based imitation learning for planning in autonomous driving, 2022. URL https://arxiv.org/abs/2210.09539.

[6] Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18(167):1–51, 2018. URL http://jmlr.org/papers/v18/15-636.html.

[7] G. Dalal, K. Dvijotham, M. Vecerík, T. Hester, C. Paduraru, and Y. Tassa. Safe exploration in continuous action spaces. *ArXiv*, abs/1801.08757, 2018. URL https://api.semanticscholar.org/CorpusID:711218.

[8] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017.

[9] J. García and F. Fernández. A comprehensive survey on safe reinforcement learning. *J. Mach. Learn. Res.*, 16:1437–1480, 2015. URL https://api.semanticscholar.org/CorpusID:2497153.

[10] C. Gulino, J. Fu, W. Luo, G. Tucker, E. Bronstein, Y. Lu, J. Harb, X. Pan, Y. Wang, X. Chen, J. D. Co-Reyes, R. Agarwal, R. Roelofs, Y. Lu, N. Montali, P. Mougin, Z. Yang, B. White, A. Faust, R. McAllister, D. Anguelov, and B. Sapp. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2023.

[11] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li. Planning-oriented autonomous driving, 2023.

[12] Y. Lu, J. Fu, G. Tucker, X. Pan, E. Bronstein, B. Roelofs, B. Sapp, B. White, A. Faust, S. Whiteson, et al. Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios. *arXiv preprint arXiv:2212.11419*, 2022.

[13] T.-H. Pham, G. D. Magistris, and R. Tachibana. Optlayer - practical constrained optimization for deep reinforcement learning in the real world. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6236–6243, 2017. URL https://api.semanticscholar.org/CorpusID:37266866.

[14] H. Shao, L. Wang, R. Chen, S. L. Waslander, H. Li, and Y. Liu. Reasonnet: End-to-end driving with temporal and global reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13723–13733, June 2023.

[15] C. Tessler, D. J. Mankowitz, and S. Mannor. Reward constrained policy optimization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=SkfrvsA9FX.

[16] B. Thananjeyan, A. Balakrishna, S. Nair, M. Luo, K. P. Srinivasan, M. Hwang, J. E. Gonzalez, J. Ibarz, C. Finn, and K. Goldberg. Recovery rl: Safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters*, 6:4915–4922, 2020. URL `https://api.semanticscholar.org/CorpusID:226221775`.

[17] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

[18] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992. URL `https://api.semanticscholar.org/CorpusID:2332513`.

[19] C. Xu, W. Ding, W. Lyu, Z. Liu, S. Wang, Y. He, H. Hu, D. Zhao, and B. Li. Safebench: A benchmarking platform for safety evaluation of autonomous vehicles, 2022.

[20] H. Yu, W. Xu, and H. Zhang. Towards safe reinforcement learning with a safety editor policy. *arXiv*, 2022.