# A Note on First Order Constrained Optimization in Policy Space

高嘉豪
Institute of Multimedia Engineering
National Yang Ming Chiao Tung University
chiahaok.cs10@nycu.edu.tw

## 1 Introduction

In this report, I would present the paper First Order Constrained Optimization in Policy Space ( Zhang et al. [2020]) from NeurIPS 2020. The paper takes on the problem of constrained reinforcement problem, which extends the regular RL setting by introducing additional constraints. The objective of constrained RL is to maximize the accumulated reward while keeping the constraints. This could be applied on more realistic situations like avoiding certain harmful actions or restricting resources. Some previous work, CPO ( Achiam et al. [2017]), tackles this problem by via local policy search and approximate the constrained evaluation function, which induces approximation error that may leads to infeasible updates. This paper ( Zhang et al. [2020]) tackles this problem by proposing a novel method called First Order Constrained Optimization in Policy Space (FOCOPS). The method tries to solve the issue by two steps, first it finds the optimal policy update in non-parametric policy space, then project this policy back into the parametric policy space. This method resolves the issues of previous methods, and it is not only easy to implement but also achieve impressive performance on different tasks. On a personal note, I think this paper presents a pretty simple solution to a complex problem which is not very commonly seen in RL field even though constraints (and safety) are often a practical concern.

## 2 Problem Formulation

### 2.1 Constrained Markov Decision Process (CMDP)

Here, Constrained Markov Decision Process (CMDP) ( Altman [1999]) is utilized to frame the problem, which is denoted by $(\mathcal{S}, \mathcal{A}, R, P, \mu, \mathcal{C})$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P$ is the transition probability, $R$ is the reward function, $\mu$ is the initial state distribution, and $\mathcal{C}$ is a set of cost functions, $C_i$, $i = 1, \ldots, m$. Let $\Pi$ denotes the *non-parameterized policy space*, while $\Pi_\theta \in \Pi$ is a set of parameterized policies. The value function is expressed as $V^\pi(s) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t, a_t)\middle| s_0 = s\right]$ and action-value function as $Q^\pi(s,a) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t, a_t)\middle| s_0 = s, a_0 = a\right]$, where $\tau$ is a sampled trajectory and $\gamma \in (0, 1)$ is the discount factor. The advantage function is defined as $A^\pi(s,a) := Q^\pi(s,a) - V^\pi(s)$, and the discounted future state visitation distribution as $d^\pi(s) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t P(s_t = s|\pi)$. In the similar vein as $V^\pi, Q^\pi, and A^\pi$, the cost value function $V^\pi_{C_i}$, cost action-value function $Q^\pi_{C_i}$, and cost advantage function $A^\pi_{C_i}$ are defined with $C_i$ replacing $R$. The expected discount return $J(\pi) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t, a_t)\right]$. Define the $C_i$-*return* as $J_{C_i}(\pi) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t C_i(s,a)\right]$.The feasible set of policies is $\Pi_\mathcal{C} := \{\pi \in \Pi : J_{C_i}(\pi) \leq b_i, i = 1, \ldots, m\}$. The object is to find the optimal policy which $\pi^* = \arg\max_{\pi\in\Pi_\mathcal{C}} J(\pi)$.

### 2.2 Preliminary (CPO)

In order to avoid collecting samples from updated policy for checking constraint satisfactory, CPO (Achiam et al. [2017]) is proposed to use a surrogate cost function to replace constraint. The surrogate function evaluates cost return $J_{C_i}(\pi_\theta)$ using samples collected from current policy $\pi_{\theta_k}$, which is a good approximation of true constraint when the KL-divergence of $\pi_\theta$ and $\pi_{\theta_k}$ is small enough. For CPO, the policy update can be achieved by solving this optimization problem,

$$\underset{\pi_\theta \in \Pi_\theta}{\text{maximize}} \quad \mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi_\theta}} \left[A^{\pi_{\theta_k}}(s,a)\right] \tag{1}$$

$$\text{subject to} \quad J_C(\pi_{\theta_k}) + \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi_\theta}} \left[A_C^{\pi_{\theta_k}}(s,a)\right] \leq b \tag{2}$$

$$\mathbb{E}_{s\sim d^{\pi_{\theta_k}}} \left[D_{\text{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s]\right] \leq \delta. \tag{3}$$

However, to solve this problem, Achiam et al. [2017] uses first and second-order Taylor approximation on the objective and constraints, which introduce additional approximation error.

## 3 Theoretical Analysis

The proposed method solves the optimization problem in two steps: (1) find the optimal policy update in ***non-parameterized*** policy space, (2) project the policy found back into parameterized policy space.

### 3.1 Finding Optimal Update Policy

The optimal update policy $\pi*$ can be obtained by solving the optimization problem,

$$\underset{\pi \in \Pi}{\text{maximize}} \quad \mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi}} \left[A^{\pi_{\theta_k}}(s,a)\right] \tag{4}$$

$$\text{subject to} \quad J_C(\pi_{\theta_k}) + \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi}} \left[A_C^{\pi_{\theta_k}}(s,a)\right] \leq b \tag{5}$$

$$\mathbb{E}_{s\sim d^{\pi_{\theta_k}}} \left[D_{\text{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s]\right] \leq \delta. \tag{6}$$

The problem is very similar to Eqs. (1)-(3), except that now the policy $\pi$ doesn't have to be in parametric space $\Pi_\theta$. The solution to this problem takes the form of

**Theorem 1.** *Let $\tilde{b} = (1-\gamma)(b - \tilde{J}_C(\pi_{\theta_k}))$. If $\pi_{\theta_k}$ is a feasible solution, the optimal policy for (4-6) takes the form*

$$\pi^*(a|s) = \frac{\pi_{\theta_k}(a|s)}{Z_{\lambda,\nu}(s)} \exp\left(\frac{1}{\lambda}\left(A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a)\right)\right) \tag{7}$$

*where $Z_{\lambda,\nu}(s)$ is the partition function which ensures* (7) *is a valid probability distribution, $\lambda$ and $\nu$ are solutions to the optimization problem:*

$$\min_{\lambda,\nu\geq 0} \; \lambda\delta + \nu\tilde{b} + \lambda\mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi^*}} \left[\log Z_{\lambda,\nu}(s)\right] \tag{8}$$

*Proof.* We can prove this by first showing the optimization problem has strong duality. Constraints 5 and 6 are both convex, and since there exist an $\pi_\theta$ that is strictly feasible (all constraints satisfied), such like $\pi_{\theta_k}$. By Slater's Condition, the strong duality holds. The optimal value $p^*$ of the problem can be solved by solving the dual problem,

$$p^* = \max_{\pi\in\Pi} \min_{\lambda,\nu\geq 0} L(\pi,\lambda,\nu) = \min_{\lambda,\nu\geq 0} \max_{\pi\in\Pi} L(\pi,\lambda,\nu) \tag{9}$$

, where

$$L(\pi,\lambda,\nu) = \lambda\delta + \nu\tilde{b} + \mathbb{E}_{s\sim d^{\pi_{\theta_k}}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}[A^{\pi_{\theta_k}}(s,a)] - \nu\mathbb{E}_{a\sim\pi(\cdot|s)}[A_C^{\pi_{\theta_k}}(s,a)] - \lambda D_{\text{KL}}\left(\pi\|\pi_{\theta_k}\right)[s]\right] \tag{10}$$

Let $\pi^*, \lambda^*$, and $\nu^*$ be the optimal for Eq. 9, then $\pi^*$ is also the optimal policy for original problem 4- 6. Eq. 9 can be decomposed as separate problem for each $s$ in the form,

$$\begin{aligned} \underset{\pi}{\text{maximize}} \quad & \mathbb{E}_{a\sim\pi(\cdot|s)}\Big[A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) - \lambda(\log\pi(a|s) - \log\pi_{\theta_k}(a|s))\Big] \\ \text{subject to} \quad & \sum_a \pi(a|s) = 1, \\ & \pi(a|s) \geq 0, \quad \text{for all } a \in \mathcal{A} \end{aligned} \tag{11}$$

, which can be solved using Lagrangian that is written as,

$$G(\pi) = \sum_a \pi(a|s)\Big[A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) - \lambda(\log\pi(a|s) - \log\pi_{\theta_k}(a|s)) + \zeta\Big] - 1. \tag{12}$$

$\zeta > 0$ is the Lagrange multiplier. By differentiate Eq. 12 and set it to zero, we can get,

$$\pi(a|s) = \pi_{\theta_k}(a|s)\exp\left(\frac{1}{\lambda}\left(A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a)\right) + \frac{\zeta}{\lambda} + 1\right) \tag{13}$$

Let $Z_{\lambda\nu}(s)$ be $\zeta/\lambda + 1$ and choose $\zeta$ so $\sum_a \pi(a|s) = 1$, then optimal policy $\pi^*$ takes the form

$$\pi^*(a|s) = \frac{\pi_{\theta_k}(a|s)}{Z_{\lambda,\nu}(s)}\exp\left(\frac{1}{\lambda}\left(A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a)\right)\right). \tag{14}$$

Plugging back to Eq. 9, we can get

$$\begin{aligned} p^* &= \min_{\lambda,\nu\geq 0} \lambda\delta + \nu\tilde{b} + \mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi^*}}\left[A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) - \lambda(\log\pi^*(a|s) - \log\pi_{\theta_k}(a|s))\right] \\ &= \min_{\lambda,\nu\geq 0} \lambda\delta + \nu\tilde{b} + \mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi^*}}\left[A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) - \lambda(\log\pi_{\theta_k}(a|s) - \log Z_{\lambda,\nu}(s)\right. \\ &\left. + \frac{1}{\lambda}(A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a)) - \log\pi_{\theta_k}(a|s))\right] \\ &= \min_{\lambda,\nu\geq 0} \lambda\delta + \nu\tilde{b} + \lambda\mathbb{E}_{\substack{s\sim d^{\pi_{\theta_k}} \\ a\sim\pi^*}}\left[\log Z_{\lambda,\nu}(s)\right] \end{aligned}$$

□

### 3.2 Approximating the Policy

After solving the problem (4-6), we want to project $\pi^*$ back into parameterized policy space by minimizing the following function,

$$\mathcal{L}(\theta) = \mathbb{E}_{s\sim d^{\pi_{\theta_k}}}\left[D_{\text{KL}}\left(\pi_\theta\|\pi^*\right)[s]\right], \tag{15}$$

where $\pi_\theta \in \Pi_\theta$ is a projected policy. The idea is to use first-order method to minimize this function. The following result shows,

**Corollary 1.** *The gradient of $\mathcal{L}(\theta)$ takes the form*

$$\nabla_\theta\mathcal{L}(\theta) = \mathbb{E}_{s\sim d^{\pi_{\theta_k}}}\left[\nabla_\theta D_{KL}\left(\pi_\theta\|\pi^*\right)[s]\right], \tag{16}$$

*where*

$$\nabla_\theta D_{KL}\left(\pi_\theta\|\pi^*\right)[s] = \nabla_\theta D_{KL}\left(\pi_\theta\|\pi_{\theta_k}\right)[s] - \frac{1}{\lambda}\mathbb{E}_{a\sim\pi_{\theta_k}}\left[\frac{\nabla_\theta\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}\left(A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a)\right)\right]. \tag{17}$$

*Proof.*

$$
\begin{aligned}
D_{\mathrm{KL}}\left(\pi_\theta \| \pi^*\right)[s] =& \sum_a \pi_\theta(a|s) \log \pi_\theta(a|s) - \sum_a \pi_\theta(a|s) \log \pi^*(a|s) \\
=& \sum_a \pi_\theta(a|s) \log \pi_\theta(a|s) \\
& - \sum_a \pi_\theta(a|s) \log \left( \frac{\pi_{\theta_k}(a|s)}{Z_{\lambda,\nu}(s)} \exp\left[ \frac{1}{\lambda} \left( A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) \right) \right] \right) \\
=& \sum_a \pi_\theta(a|s) \log \pi_\theta(a|s) - \sum_a \pi_\theta(a|s) \log \pi_{\theta_k}(a|s) + \log Z_{\lambda,\nu}(s) \\
& - \frac{1}{\lambda} \sum_a \pi_\theta(a|s) \left( A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) \right) \\
=& D_{\mathrm{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s] + \log Z_{\lambda,\nu}(s) - \frac{1}{\lambda} \sum_a \pi_\theta(a|s) \left( A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) \right) \\
=& D_{\mathrm{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s] + \log Z_{\lambda,\nu}(s) - \frac{1}{\lambda} \mathbb{E}_{a \sim \pi_{\theta_k}(\cdot|s)} \left[ \frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} \left( A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) \right) \right]
\end{aligned}
$$

The last equality is resulted from applying importance sampling. Next, taking gradient on both side, we would get,

$$
\nabla_\theta D_{\mathrm{KL}}\left(\pi_\theta \| \pi^*\right)[s] = \nabla_\theta D_{\mathrm{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s] - \frac{1}{\lambda} \mathbb{E}_{a \sim \pi_{\theta_k}(\cdot|s)} \left[ \frac{\nabla_\theta \pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} \left( A^{\pi_{\theta_k}}(s,a) - \nu A_C^{\pi_{\theta_k}}(s,a) \right) \right].
$$

□

With Corollary 1, the FOCOPS algorithm goes as follow. We start with an initial policy $\pi_{\theta_k}$ and collect samples by interacting with environment. After, draw a batch of samples and evaluate Eq. 15 then take a gradient step to update the policy. The process would iterated for multiple times.

### 3.3 Practical Implementation

In practice, the cost to solve Eq. 8 every iteration is too high to actually implement for environment with large action and state space. So the authors chose to use another approach on deciding $\lambda$, and $\nu$. As $\lambda$ affects the exploratory of the policy, the smaller the $\lambda$, the greedier the policy. However, through experiments, the authors conclude that it's sufficient to use a fixed $\lambda$. As for $\nu$, it needs to be constantly updated so the constraint would hold. Again from Eq. 7, we can see that the larger $\nu$ is, the less likely it is for (s,a) pairs with high cost advantage to get sampled. As mentioned before, optimal $\mu^*, \lambda^*$ minimize the dual problem $L(\pi^*, \lambda, \mu)$. Therefore, the update rule for $\mu$ proposed by the authors is do gradient descent on the $L$.

**Corollary 2.** *The derivative of $L(\pi^*, \lambda, \nu)$ w.r.t. $\nu$ is*

$$
\frac{\partial L(\pi^*, \lambda, \nu)}{\partial \nu} = \tilde{b} - \mathbb{E}_{\substack{s \sim d^{\pi_{\theta_k}} \\ a \sim \pi^*}} \left[ A^{\pi_{\theta_k}}(s,a) \right]. \tag{18}
$$

However, seeing that $\mathbb{E}_{s \sim d^{\pi_{\theta_k}} a \sim \pi^*}[A^{\pi_{\theta_k}}(s,a)]$ is impossible to evaluate without the access to $\pi^*$. The workaround that was proposed is to leverage the fact that $\pi^*$ and $\pi_{\theta_k}$ are close because of Eq. 3. So $\mathbb{E}_{s \sim d^{\pi_{\theta_k}} a \sim \pi^*}[A^{\pi_{\theta_k}}(s,a)]$ would approximate $\mathbb{E}_{s \sim d^{\pi_{\theta_k}} a \sim \pi_{\theta_k}}[A^{\pi_{\theta_k}}(s,a)]$, which is 0. In the end, the update for $\nu$ is,

$$
\nu \leftarrow \mathrm{proj}_\nu \left[ \nu - \alpha (b - J_C(\pi_{\theta_k})) \right], \tag{19}
$$

where $\alpha$ is the step size, which includes the term $(1-\gamma)$ of $\tilde{b}$. $\mathrm{proj}_\nu$ is the projection operation to project $\nu$ into $[0, \nu_{\max}]$, and $\nu_{\max}$ is set to infinite in the paper. I would say that the approximation of $\mathbb{E}_{s \sim d^{\pi_{\theta_k}} a \sim \pi^*}[A^{\pi_{\theta_k}}(s,a)]$ seems not elegant enough. That being said, the final update rule does seem intuitive, when $J_C(\pi_{\theta_k})$ is larger than $b$ (which means the constraint is not satisfied), then $\nu$ would be increased.

Since the proposed method is a first-order method, an additional condition ($\pi_\theta$, and $\pi_{\theta_k}$ are close) need to be satisfied so the approximation would be accurate. Therefore, a indicator $I(s_j) :=$

$\mathbb{1}_{D_{\mathrm{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s_j] \leq \delta}$ is added to Eq. 17 to ensure that the states where two policy are too apart won't contribute to update gradient. The final actual update term becomes

$$\tilde{\nabla}_\theta \mathcal{L}(\theta) \approx \frac{1}{N} \sum_{j=1}^{N} \left[ \nabla_\theta D_{\mathrm{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s_j] - \frac{1}{\lambda} \frac{\nabla_\theta \pi_\theta(a_j|s_j)}{\pi_{\theta_k}(a_j|s_j)} \left( \hat{A}(s_j, a_J) - \nu \hat{A}_C(s_j, a_j) \right) \right] I(s_j), \tag{20}$$

where $\hat{A}$, and $\hat{A}_C$ are estimations of advantage function of reward and cost obtained using critic networks. There's also an early stopping rule that if $\frac{1}{N} \sum_{j=1}^{N} D_{\mathrm{KL}}\left(\pi_\theta \| \pi_{\theta_k}\right)[s_j] > \delta$. The downside is that these measures might lead to wasted samples and low sample efficiency.

## 4 Conclusion

This paper (Zhang et al. [2020]) proposed a novel approach on solving constrained optimization problem that I think can be seen as an improvement from the previous method CPO (Achiam et al. [2017]). The main idea of the proposed FOCOPS algorithm is quite interesting and elegant, first finding optimal update policy then projecting it into parameterized policy space. The theoretical part all checks out as will. However, in my opinion, the caveat lies in the practical implementation part (Section 3.3). There are some parts of this algorithm that cannot implement as is in practice, such as $\nu, \lambda$, and there's also a need for additional measure ($I(s, a)$) to make sure the approximation is accurate. I would say this is the limitation or crutch to this method. Though the experimental results do show promising performance, so these workarounds seem to be justified.

A more recent paper that tackles constrained RL I studied is NFWPO (Lin et al. [2021]). The paper present a algorithm (NFWPO) leveraging classic Frank-Wolfe to solve the action-constrained RL problem, where the constraint is in the form of state-wise feasible action sets. The main idea of NFWPO is to first finding the reference action that's inside the feasible set via Frank-Wolfe, then take a gradient step updating parameters to minimize the MSE between reference action and action of current policy. The paper also mentioned FOCOPS, saying that the algorithm seems to take longer to produce a policy with less constraint violation comparing to other methods, which I suspect might be resulted from the indicator $I(s, a)$ that prevent some samples to take part in update and thus less efficient.

## References


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

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31. PMLR, 2017.

Eitan Altman. *Constrained Markov decision processes: stochastic modeling*. Routledge, 1999.

Jyun-Li Lin, Wei Hung, Shang-Hsuan Yang, Ping-Chun Hsieh, and Xi Liu. Escaping from zero gradient: Revisiting action-constrained reinforcement learning via frank-wolfe policy optimization. In *Uncertainty in Artificial Intelligence*, pages 397–407. PMLR, 2021.