

Lecture 10: Constrained RL

CS885 Reinforcement Learning

2025-02-06

Complementary readings:

Ray, Achiam, Amodei, Benchmarking Safe Exploration in Deep Reinforcement Learning.

Liu, Alev, Liu, Policy Learning with Constraints in Model-free Reinforcement Learning: A Survey, IJCAI, 2021

Pascal Poupart

David R. Cheriton School of Computer Science



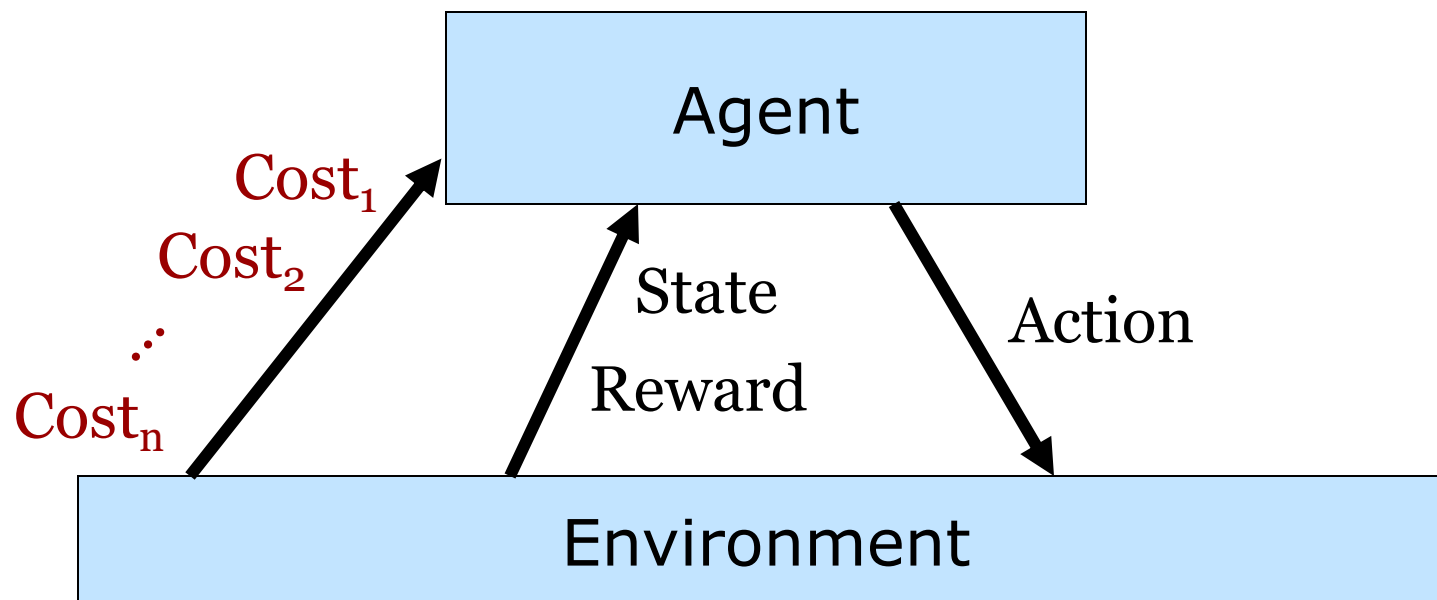
Outline

- Constrained Reinforcement Learning
 - Ensure safety in RL
 - Incorporate constrained objectives
- Algorithms
 - PPO-Penalty
 - PPO-Lagrange

Motivation

- Constraints
 - Prevent catastrophic events
 - Ensure safety
 - Take into multiple objectives with bounds
- Examples:
 - Autonomous driving: minimize arrival time such that $P(\textit{collision}) \leq \beta$
 - Routing: maximize throughput such that $\textit{latency} \leq \beta$
 - Robotics: minimize path length subject to **environment obstacles/boundaries**

Constrained Reinforcement Learning



Goal: Learn to choose actions that maximize rewards
subject to bounds on costs

Constrained RL

- Definition

- States: $s \in S$
- Actions: $a \in A$
- Rewards: $r \in \mathbb{R}$
- Costs: $c^{(i)} \in \mathbb{R}$
- Transition model: $\Pr(s_t | s_{t-1}, a_{t-1})$
- Reward model: $\Pr(r_t | s_t, a_t)$
- Cost model: $\Pr(c_t^{(i)} | a_{t-1}, s_t)$
- Discount factor: $0 \leq \gamma \leq 1$. (discounted: $\gamma < 1$, undiscounted: $\gamma = 1$)
- Horizon (i.e., # of time steps): h (Finite horizon: $h \in \mathbb{N}$, infinite horizon: $h = \infty$)

} unknown model

- Goal: $\pi^* = \operatorname{argmax}_{\pi} E_{\pi} \left[\sum_{t=0}^h \gamma^t r_t \right]$
such that $E_{\pi} \left[\sum_{t=0}^h \gamma^t c_t^{(i)} \right] \leq \beta_i \forall i$

Constraint Types

	Deterministic	Probabilistic	Expected
Instantaneous	$c_t \leq \beta \forall t$	$P_\pi(c_t \leq \beta) \geq 1 - \epsilon \forall t$	$E_\pi[c_t] \leq \beta \forall t$
Cumulative	$\sum_t \gamma^t c_t \leq \beta$	$P_\pi\left(\sum_t \gamma^t c_t \leq \beta\right) \geq 1 - \epsilon$	$E_\pi\left[\sum_t \gamma^t c_t\right] \leq \beta$

Algorithms for Expected Constraints

- Penalty methods
- Lagrangian relaxation
- Lyapunov methods
- State augmentation

Penalty Methods

- Constrained MDP

$$\pi^* = \operatorname{argmax}_{\pi} E_{\pi} \left[\sum_{t=0}^h \gamma^t r_t \right]$$
$$\text{such that } E_{\pi} \left[\sum_{t=0}^h \gamma^t c_t^{(i)} \right] \leq \beta_i \forall i$$

- Penalized MDP

$$\pi^* = \operatorname{argmax}_{\pi} E_{\pi} \left[\sum_{t=0}^h \gamma^t r_t \right] + \sum_i \lambda_i \operatorname{penalty} \left(E_{\pi} \left[\sum_{t=0}^h \gamma^t c_t^{(i)} \right] - \beta_i \right)$$

- Barrier: $\operatorname{penalty}(x) = \log(-x)$
- ReLU: $\operatorname{penalty}(x) = \operatorname{ReLU}(x) = \max(x, 0)$

PPO with ReLU penalty

Alternate between:

$$\theta \leftarrow \operatorname{argmax}_{\theta} E_{s \sim \mu_{\theta_{old}}, a \sim \pi_{\theta_{old}}} \min \left\{ \begin{array}{l} \frac{\pi_{\theta}(a|s)}{\pi_{\theta_{old}}(a|s)} (G_r - V_w(s)), \\ \operatorname{clip} \left(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_{old}}(a|s)}, 1 - \epsilon, 1 + \epsilon \right) (G_r - V_w(s)) \end{array} \right\}$$
$$\theta \leftarrow \theta - \sum_i \alpha_i \nabla_{\theta} \operatorname{ReLU}(E_{\tau \sim \pi_{\theta}} [G_{c,0}^{(i)}] - \beta)$$

Where: $\nabla_{\theta} \operatorname{ReLU}(E_{\pi_{\theta}} [G_{c,0}^{(i)}] - \beta) = I(E_{\pi_{\theta}} [G_{c,0}^{(i)}] - \beta) \nabla_{\theta} E_{\pi_{\theta}} [G_{c,0}^{(i)}]$

$$\begin{aligned} &= I(E_{\pi_{\theta}} [G_{c,0}^{(i)}] - \beta) E_{\pi_{\theta}} [\sum_n G_{c,n}^{(i)} \nabla_{\theta} \log \pi_{\theta}(A_n | S_n)] \\ &\approx I(G_{c,0}^{(i)} - \beta) \sum_n G_{c,n}^{(i)} \nabla_{\theta} \log \pi_{\theta}(a_n | s_n) \end{aligned}$$

PPO with a Replay Buffer

Initialize π_θ and V_w to anything

Loop forever

 Loop for each episode

 Generate episode $s_0, a_0, r_0, s_1, a_1, r_1, \dots, s_{N-1}, a_{N-1}, r_{N-1}$ with π_θ

 Loop for each step $n = 0, 1, \dots, N - 1$ of the episode

$$G_{r,n} \leftarrow \sum_{t=0}^{N-1-n} \gamma^t r_{n+t}$$

 Add $(s_n, a_n, G_{r,n}, \log \pi_\theta(a_n, s_n))$ to replay buffer

 Loop for each training epoch

 Sample a minibatch of experiences $(s_n, a_n, G_{r,n}, \pi_{\theta_{old}}(a_n, s_n))$

$$w \leftarrow w + \alpha_w (G_{r,n} - V_w(s_n)) \nabla_w V_w(s_n)$$

$$L_{PPO} = \min \left\{ \frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)} (G_{r,n} - V_w(s_n)), \text{clip} \left(\frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)}, 1 - \epsilon, 1 + \epsilon \right) (G_{r,n} - V_w(s_n)) \right\}$$

$$\theta \leftarrow \theta + \alpha \nabla_\theta L_{PPO}$$

PPO-Penalty

Initialize π_θ and V_w to anything

Loop forever

 Loop for each episode

 Generate episode $s_0, a_0, r_0, s_1, a_1, r_1, \dots, s_{N-1}, a_{N-1}, r_{N-1}$ with π_θ

 Loop for each step $n = 0, 1, \dots, N - 1$ of the episode

$$G_{r,n} \leftarrow \sum_{t=0}^{N-1-n} \gamma^t r_{n+t}$$

$$G_{c,n}^{(i)} \leftarrow \sum_{t=0}^{N-1-n} \gamma^t c_{n+t}^{(i)} \quad \forall i$$

 Add $(s_n, a_n, G_{r,n}, \log \pi_\theta(a_n, s_n))$ to replay buffer

 Loop for each training epoch

 Sample a minibatch of experiences $(s_n, a_n, G_{r,n}, \pi_{\theta_{old}}(a_n, s_n))$

$$w \leftarrow w + \alpha_w (G_{r,n} - V_w(s_n)) \nabla_w V_w(s_n)$$

$$L_{PPO} = \min \left\{ \frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)} (G_{r,n} - V_w(s_n)), \text{clip} \left(\frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)}, 1 - \epsilon, 1 + \epsilon \right) (G_{r,n} - V_w(s_n)) \right\}$$

$$\theta \leftarrow \theta + \alpha \nabla_\theta L_{PPO}$$

 Loop over collected episodes

$$\theta \leftarrow \theta - \sum_i \alpha_i I(G_{c,0}^{(i)} - \beta_i) \sum_{n=0}^{N-1} G_{c,n}^{(i)} \nabla_\theta \log \pi(a_n|s_n)$$



Lagrangian Relaxation

- Constrained MDP

$$\pi^* = \operatorname{argmax}_{\pi} E_{\pi} \left[\sum_{t=0}^h \gamma^t r_t \right]$$

such that $E_{\pi} \left[\sum_{t=0}^h \gamma^t c_t^{(i)} \right] \leq \beta_i \forall i$

- Lagrangian relaxed MDP

$$\min_{\lambda_i \geq 0 \forall i} \max_{\pi} E_{\pi} \left[\sum_{t=0}^h \gamma^t r_t \right] + \sum_i \lambda_i \left(E_{\pi} \left[\sum_{t=0}^h \gamma^t c_t^{(i)} \right] - \beta_i \right)$$

PP0-Lagrange

Initialize π_θ , V_w and $V_{\phi^{(i)}} \forall i$ to anything; initialize $\lambda_i \leftarrow 1 \forall i$

Loop forever

 Loop for each episode

 Generate episode $s_0, a_0, r_0, s_1, a_1, r_1, \dots, s_{N-1}, a_{N-1}, r_{N-1}$ with π_θ

 Loop for each step of the episode $n = 0, 1, \dots, N-1$

$$G_{r,n} \leftarrow \sum_{t=0}^{N-1-n} \gamma^t r_{n+t}, \quad G_{c,n}^{(i)} \leftarrow \sum_{t=0}^{N-1-n} \gamma^t c_{n+t}^{(i)} \forall i$$

 Add $(s_n, a_n, G_{r,n}, G_{c,n}^{(i)} \forall i, \pi_\theta(a_n, s_n))$ to replay buffer

$$\lambda_i \leftarrow \lambda_i + \alpha_{\lambda_i} (G_{c,0}^{(i)} - \beta_i) \forall i$$

 Loop for each training epoch

 For each $(s_n, a_n, G_{r,n}, G_{c,n}^{(i)} \forall i, \pi_{\theta_{old}}(a_n, s_n))$ in minibatch sampled from replay buffer

$$w \leftarrow w + \alpha_w (G_{r,n} - V_w(s_n)) \nabla_w V_w(s_n), \quad \phi^{(i)} \leftarrow \phi^{(i)} + \alpha_{\phi^{(i)}} (G_{c,n}^{(i)} - V_{\phi^{(i)}}^{(i)}) \nabla_{\phi^{(i)}} V_{\phi^{(i)}}^{(i)}(s_n) \forall i$$

$$L_{PPO} = \min \left\{ \frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)} (G_{r,n} - V_w(s_n)), \text{clip} \left(\frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)}, 1 - \epsilon, 1 + \epsilon \right) (G_{r,n} - V_w(s_n)) \right\}$$

$$L_{Lag}^{(i)} = \frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)} (G_{c,n}^{(i)} - V_{\phi^{(i)}}^{(i)})$$

$$\theta \leftarrow \theta + \alpha_\theta \nabla_\theta (L_{PPO} - \sum_i \text{softplus}(\lambda_i) L_{Lag}^{(i)})$$

Experiments

Figure from “Benchmarking Safe Exploration in Deep Reinforcement Learning”, arxiv

— CPO — PPO — PPO-Lagrangian — TRPO — TRPO-Lagrangian

PointPush1

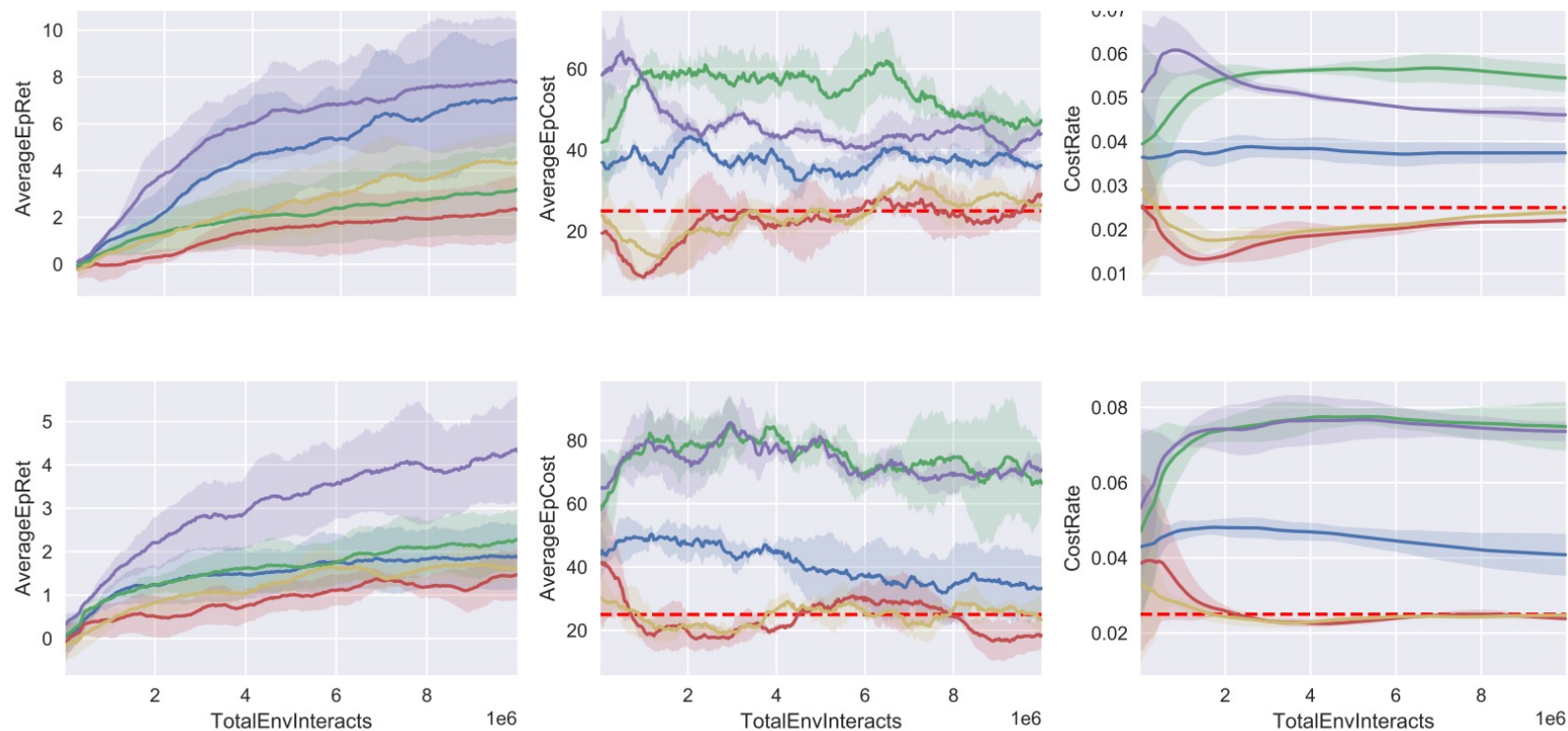


Figure 7: Results from benchmarking unconstrained and constrained RL algorithms on all Point level 1 and 2 environments. Dashed red lines indicate the target value for a constraint-satisfying policy (AverageEpCost curves) or approximately constraint-satisfying training run (CostRate curves).