Lecture 10: Constrained RL
CS885 Reinforcement Learning

2022-10-21

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

UNIVERSITY OF WATERLOO

Outline

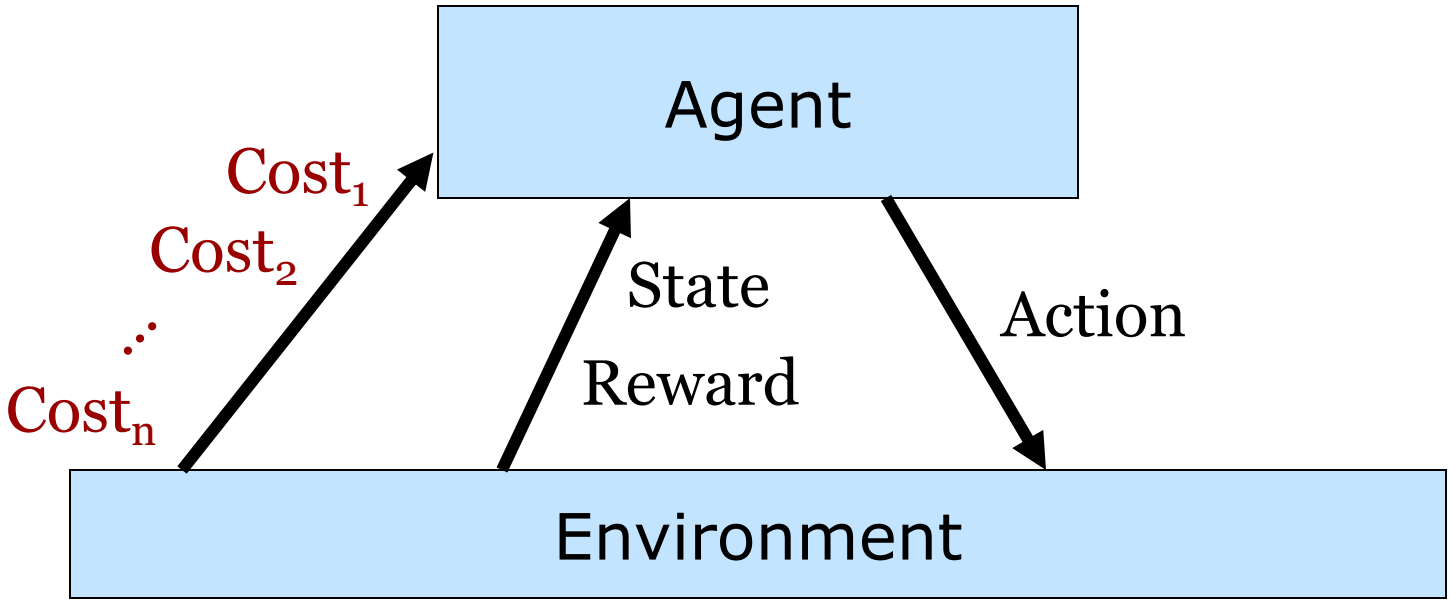
- Constrained Reinforcement Learning
  - Ensure safety in RL
  - Incorporate constrained objectives
- Algorithms
  - PPO-Penalty
  - PPO-Lagrange

Motivation

- Constraints
  - Prevent catastrophic events
  - Ensure safety
  - Take into multiple objectives with bounds

- Examples:
  - Autonomous driving: minimize arrival time such that $P(collision) \leq \beta$
  - Routing: maximize throughput such that $latency \leq \beta$
  - 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: $\Pr(c_t^{(i)}|a_{t-1}, s_t)$

  } unknown model

  - 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: $h = \infty$)

- Goal: $\pi^* = argmax_\pi E_\pi\left[\sum_{t=0}^{h} \gamma^t r_t\right]$

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

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

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

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

- Penalized MDP

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

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

PPO with ReLU penalty

Alternate between:

$$\theta \leftarrow \underset{\theta}{\text{argmax}}\, E_{s\sim\mu_{\theta_{old}},\, a\sim\pi_{\theta_{old}}} \min\left\{ \begin{array}{c} \frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)}\left(G_r - V_w(s)\right), \\ clip\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)}, 1-\epsilon, 1+\epsilon\right)\left(G_r - V_w(s)\right) \end{array} \right\}$$

$$\theta \leftarrow \theta - \sum_i \alpha_i \nabla_\theta ReLU(E_{\tau\sim\pi_\theta}\left[G_{c,0}^{(i)}\right] - \beta)$$

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

$$= I\left(E_{\pi_\theta}\left[G_{c,0}^{(i)}\right] - \beta\right) E_{\pi_\theta}\left[\sum_n G_{c,n}^{(i)} \nabla_\theta \log \pi_\theta(A_n|S_n)\right]$$

$$\approx I\left(G_{c,0}^{(i)} - \beta\right)\sum_n G_{c,n}^{(i)} \nabla_\theta \log \pi_\theta(a_n|s_n)$$

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

Loop for each training epoch

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

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

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

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

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

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

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

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

Loop for each training epoch

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

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

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

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

Loop over collected episodes

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

Lagrangian Relaxation

- Constrained MDP

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

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

- Lagrangian relaxed MDP

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

PPO-Lagrange

Initialize π_θ, V_w and $V_{\phi^{(i)}}^{(i)}$ $\forall i$ 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 $\left(s_n, a_n, G_{r,n}, G_{c,n}^{(i)} \forall i, \pi_\theta(a_n, s_n)\right)$ to replay buffer

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

Loop for each training epoch

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

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

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

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

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

Experiments

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

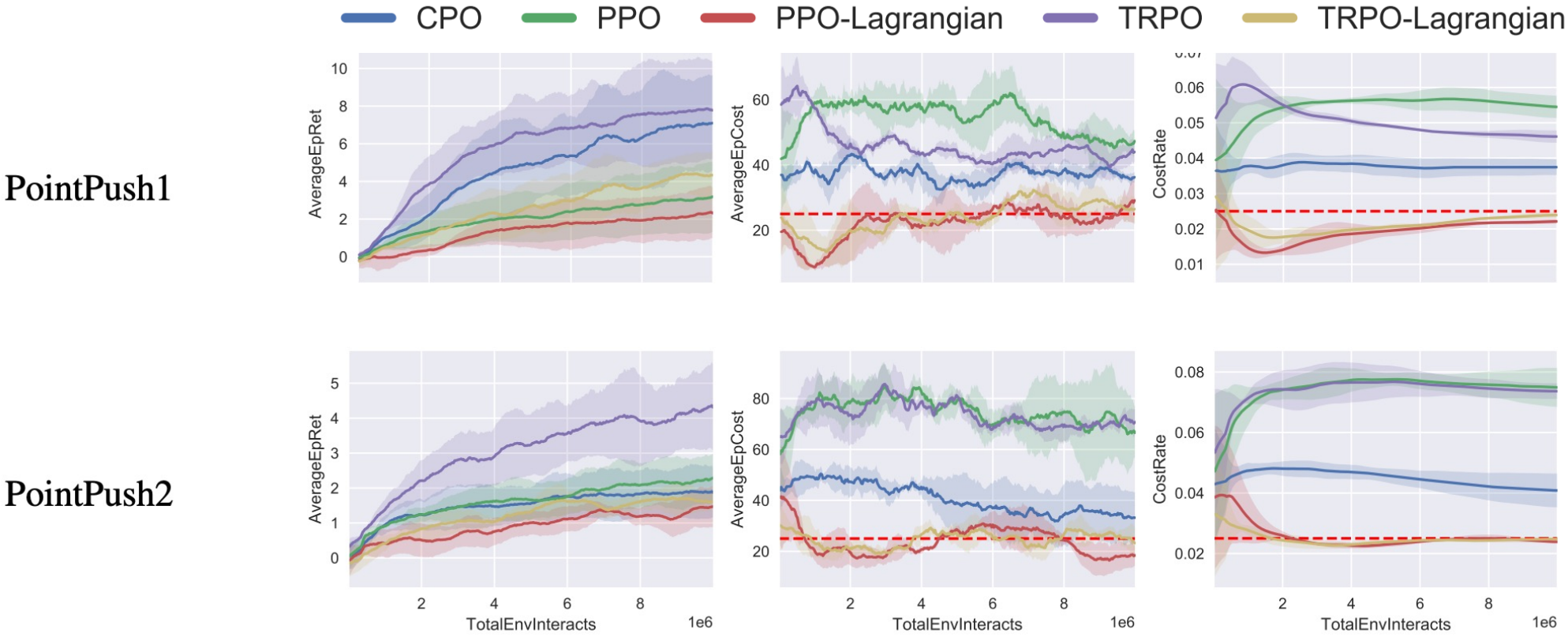


Figure 7: Results from benchmarking unconstrained and constrained RL algorithms on all Point level 1 and 2 environments. Dashed red lines indicate the target value for a constraint-satisfying policy (AverageEpCost curves) or approximately constraint-satisfying training run (CostRate curves).