Constrained Reinforcement Learning

2025 MIC Symposium

Minseok Seo[1],

[1]Mobility Intelligence and Control Laboratory (MIC Lab)
CCS Graduate School of Mobility
Korea Advanced Institute of Science and Technology (KAIST)

September 5, 2025

Outline

Figure: Waymo and Tesla

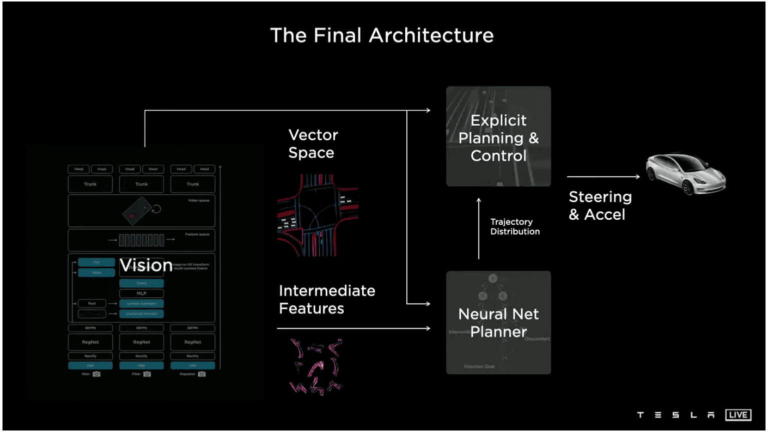

Figure: Tesla's Architecture in 2021 (source: AI Day 2021)

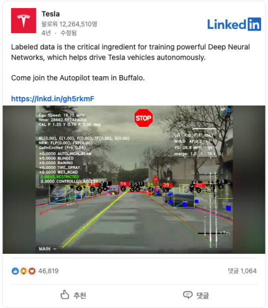

Figure: Tesla's recruitment post for data labeling positions

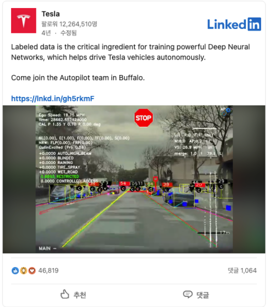

Figure: Tesla's recruitment post for data labeling positions

- Supervised learning requires a large amount of labeled data.

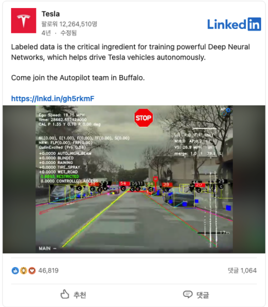

Figure: Tesla's recruitment post for data labeling positions

- Supervised learning requires a large amount of labeled data.
- Since it is created by humans, it is expensive.

Motivation Supervised Learning

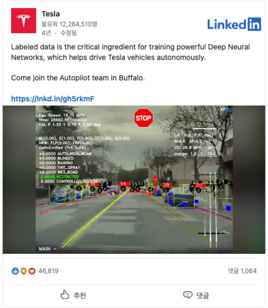

Figure: Tesla's recruitment post for data labeling positions

- Supervised learning requires a large amount of labeled data.
- Since it is created by humans, it is expensive.
- The performance of supervised learning is depends on human-labeled data.

Alpha Go [1]

Deep Q-Network [2]

Proximal Policy Optimization [3]

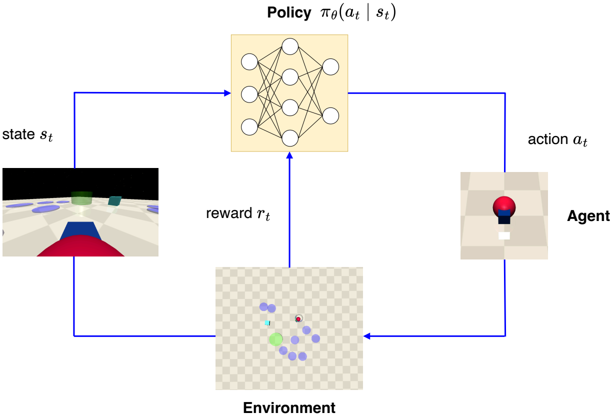

Figure: Overview of the reinforcement learning framework

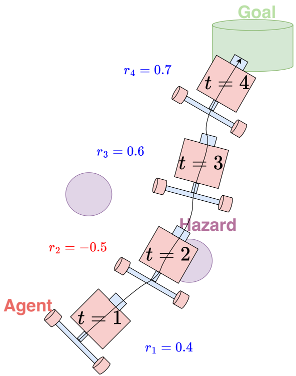

Figure: Illustration of sequential decision making: the agent receives a reward after each action.

- Policy parameterized by θ, denoted as $\pi_\theta(a|s)$
- Goal: find the optimal policy π_θ^* that maximizes the expected cumulative reward

$$\begin{aligned} \theta^* &= \arg\max_\theta J(\theta) \\ J(\theta) &= \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \end{aligned} \tag{1}$$

1. Reward engineering can induce desired behaviors (multi-objective), but it is time-consuming and difficult

1. Reward engineering can induce desired behaviors (multi-objective), but it is time-consuming and difficult
2. Even with reward engineering, the agent may still take unsafe actions
   - Reward hacking [4]
   - Out-of-distribution cases (unseen during training)

1. Reward engineering can induce desired behaviors (multi-objective), but it is time-consuming and difficult
2. Even with reward engineering, the agent may still take unsafe actions
   - Reward hacking [4]
   - Out-of-distribution cases (unseen during training)

My research

Challenges 2 & 3: learning safe policies without relying on reward engineering

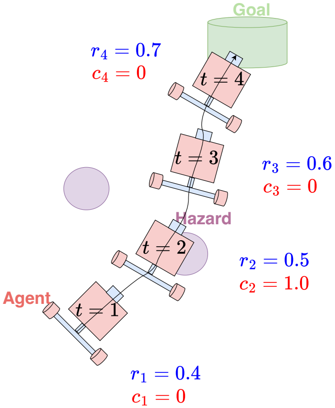

Figure: Constrained RL example — agent trajectory with rewards (blue) and costs (red)

Constrained Policy Optimization Problem

$$
\begin{aligned}
\theta^* &= \arg\max_{\theta} J(\theta) \\
J(\theta) &= \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \text{ subject to } \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} c_t\right] \leq d
\end{aligned} \tag{2}
$$

$$\theta^* = \arg\max_{\theta} J(\theta)$$
$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \text{ subject to } \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} c_t\right] \leq d \tag{2}$$

- Find a policy that maximizes rewards while satisfying constraints.

Constrained Policy Optimization Problem

$$\theta^* = \arg\max_{\theta} J(\theta)$$
$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \text{ subject to } \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} c_t\right] \leq d \tag{2}$$

- Find a policy that maximizes rewards while satisfying constraints.
- Directly solving constrained optimization is difficult.

Constrained Policy Optimization Problem

$$\theta^* = \arg\max_{\theta} J(\theta)$$
$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \text{ subject to } \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} c_t\right] \leq d \tag{2}$$

- Find a policy that maximizes rewards while satisfying constraints.
- Directly solving constrained optimization is difficult.
- By applying Lagrangian relaxation, we can convert it to an unconstrained problem.

Constrained Policy Optimization Problem

$$\theta^* = \arg\max_{\theta} \mathcal{L}(\theta, \lambda)$$
$$\mathcal{L}(\theta, \lambda) = \mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T} r_t\right] - \lambda\left(\mathbb{E}_{\tau \sim \pi_\theta}\left[\sum_{t=0}^{T} c_t\right] - d\right) \tag{3}$$

$$\lambda \leftarrow \left[\lambda + \beta\left(\hat{J}_c - d\right)\right]_+ \tag{4}$$

Penalty (λ) increases when constraints are violated $\rightarrow$ policy is encouraged to satisfy them.

MICLab
Mobility Intelligence and Control Laboratory

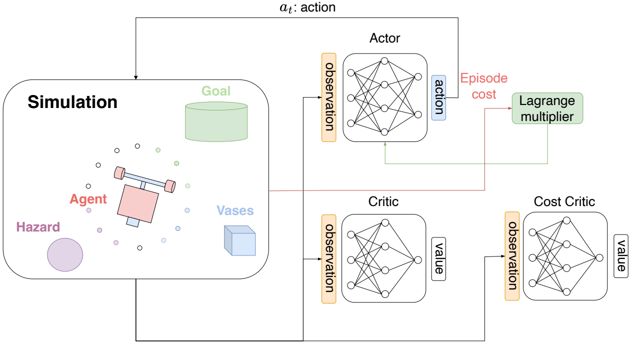

Figure: Overview of the Constrained RL with Lagrangian relaxation

In Constrained RL, constraints are imposed on the trajectory level.

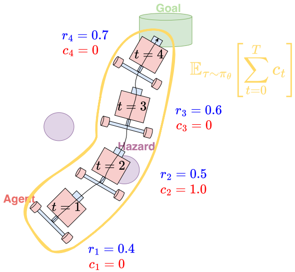

Figure: Constrained RL example — agent trajectory with rewards (blue) and costs (red)

In Constrained RL, constraints are imposed on the trajectory level.

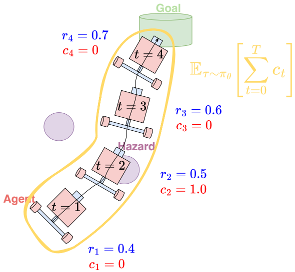

Figure: Constrained RL example — agent trajectory with rewards (blue) and costs (red)

Wouldn't imposing constraints at the state level allow for more precise constraint enforcement?

In state-wise constrained MDPs, constraints are imposed on each state.

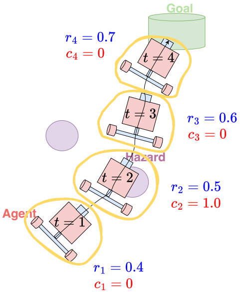

Figure: State-wise constrained RL example — agent trajectory with rewards (blue) and costs (red)

$$\pi^* = \arg\max_{\pi_\theta} J(\theta)$$

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \text{ subject to } \mathbb{E}_{\tau\sim\pi_\theta}[c(s,a)] \leq w, \quad \forall s \in S \tag{5}$$

$$\lambda(s) \leftarrow \lambda(s) + \beta(\hat{J}_c - w) \tag{6}$$

$$\pi^* = \arg\max_{\pi_\theta} J(\theta)$$
$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} r_t\right] \text{ subject to } \mathbb{E}_{\tau\sim\pi_\theta}[c(s,a)] \leq w, \quad \forall s \in S \tag{5}$$

$$\lambda(s) \leftarrow \lambda(s) + \beta(\hat{J}_c - w) \tag{6}$$

The Lagrange multiplier is replaced by a neural network's output instead of a scalar.

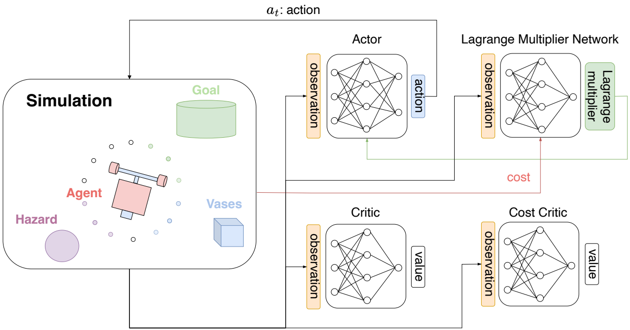

Figure: Overview of the proposed method

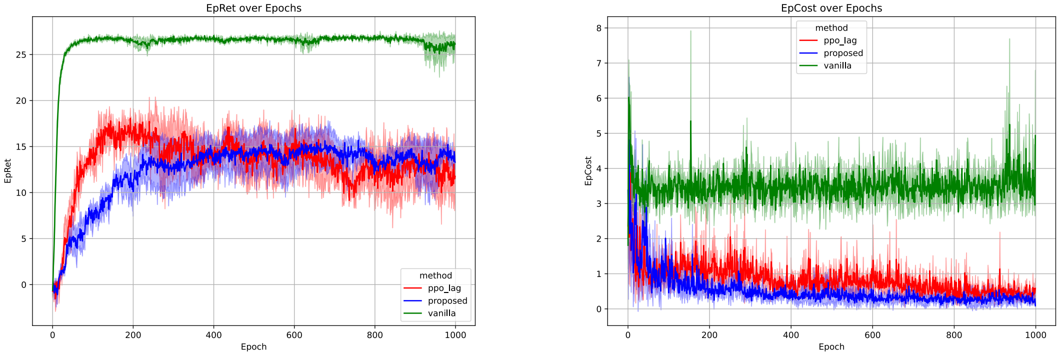

Figure: Performance comparison on Safety Gym Point Goal tasks

PPO	PPO Lagrangian	Proposed Method

Limitations of the Proposed Approach

- Sensitivity to Lagrange multiplier (init., learning rate)

Limitations of the Proposed Approach

- Sensitivity to Lagrange multiplier (init., learning rate)
- Constraint threshold setting is difficult
  - Too strict $\rightarrow$ overly conservative policy
  - Too lenient $\rightarrow$ constraints not enforced

Limitations of the Proposed Approach

- Sensitivity to Lagrange multiplier (init., learning rate)
- Constraint threshold setting is difficult
  - Too strict $\rightarrow$ overly conservative policy
  - Too lenient $\rightarrow$ constraints not enforced
- Needs sufficient violations during training

Outline

Summary

Summary

- Challenges in applying RL to the real world

Summary

- Challenges in applying RL to the real world
- Reward engineering is needed for diverse behaviors

Summary

- Challenges in applying RL to the real world
- Reward engineering is needed for diverse behaviors
- Constrained RL enables learning without reward engineering

Summary

- Challenges in applying RL to the real world
- Reward engineering is needed for diverse behaviors
- Constrained RL enables learning without reward engineering
- Constrained RL imposes constraints at the trajectory level

Summary

- Challenges in applying RL to the real world
- Reward engineering is needed for diverse behaviors
- Constrained RL enables learning without reward engineering
- Constrained RL imposes constraints at the trajectory level
- Extension to state-wise constrained RL

Summary

- Challenges in applying RL to the real world
- Reward engineering is needed for diverse behaviors
- Constrained RL enables learning without reward engineering
- Constrained RL imposes constraints at the trajectory level
- Extension to state-wise constrained RL

But there are still challenges to solve..

Future Work

- Hard to set appropriate constraint threshold → Curriculum learning [5, 6]

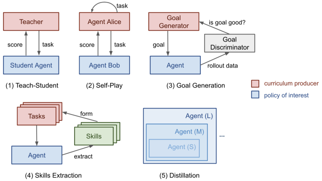

Figure: Curriculum learning overview

Future Work

- Hard to set appropriate constraint threshold → Curriculum learning [5, 6]

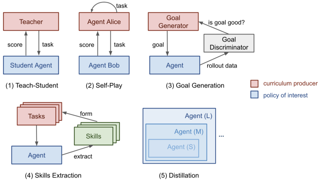

Figure: Curriculum learning overview

- Constraint violations during training → Combine with Model-Based methods [7, 8, 9, 10]

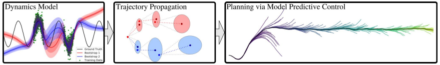

Figure: Model-Based RL overview

Thank you for your attention!

[1] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[2] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[3] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[4] L. Weng, "Reward hacking in reinforcement learning." *lilianweng.github.io*, Nov 2024. [Online]. Available: https://lilianweng.github.io/posts/2024-11-28-reward-hacking/

[5] ——, "Curriculum for reinforcement learning," *lilianweng.github.io*, Jan 2020. [Online]. Available: https://lilianweng.github.io/posts/2020-01-29-curriculum-rl/

[6] Y. Kim, H. Oh, J. Lee, J. Choi, G. Ji, M. Jung, D. Youm, and J. Hwangbo, "Not only rewards but also constraints: Applications on legged robot locomotion," *IEEE Transactions on Robotics*, vol. 40, pp. 2984–3003, 2024.

[7] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," *Advances in neural information processing systems*, vol. 31, 2018.

[8] Z. Liu, H. Zhou, B. Chen, S. Zhong, M. Hebert, and D. Zhao, “Safe model-based reinforcement learning with robust cross-entropy method,” *arXiv preprint arXiv:2010.07968*, vol. 3, 2020.

[9] A. K. Jayant and S. Bhatnagar, “Model-based safe deep reinforcement learning via a constrained proximal policy optimization algorithm,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 432–24 445, 2022.

[10] G. Paolo, J. Gonzalez-Billandon, A. Thomas, and B. Kégl, “Guided safe shooting: model based reinforcement learning with safety constraints,” *arXiv preprint arXiv:2206.09743*, 2022.