Practical Introduction to Safe Reinforcement Learning

Kryspin Varys

University of Southampton

4 February 2024
Outline

Introduction to Safe RL
   When to Use RL?
   What is RL?
   The Role of Open-Source in RL
   When is RL safe?

 Practical Scenarios
   Scenario 1: Modification of the Optimality Criterion
   Scenario 2: Modification of the Agent’s Actions
Introduction to Safe RL

When to Use RL?

To solve control problems:
Introduction to Safe RL

What is RL?

The environment:

\[
\text{Markov decision process} = \langle S, A, R : S \times A \rightarrow R, T : S \times A \rightarrow S \rangle
\]

discrete env.
continuous env.
Introduction to Safe RL

What is RL?

The environment:

Markov decision process = \( \langle S, A, \\
R : S \times A \to \mathbb{R}, \\
T : S \times A \to S \rangle \)
Introduction to Safe RL

What is RL?

The agent $\pi$ is either:

1. table in the case of small discrete spaces, or
2. neural network in the case of large spaces.

![Diagram of environment and agent](image)
Introduction to Safe RL

What is RL?

The agent $\pi$ aims to maximize an optimality criterion:

$$\max E_\pi \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \right]$$

- discrete env.
- continuous env.
The agent’s lifecycle has two phases:

1. the training phase and
2. the deployment phase.
Training

1. Exploration
   1.1 Random actions

2. Exploitation
   2.1 Best actions according to the optimality criterion
Introduction to Safe RL

What is RL?

Training

1. Exploration
   1.1 Random actions

2. Exploitation
   2.1 Best actions according to the optimality criterion

Deployment

1. Exploitation

The agent $\pi$ aims to maximize the optimality criterion:

$$\max \mathbb{E}_{\pi} \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)$$
Introduction to Safe RL

What is RL?

Training

1. Exploration
   1.1 Random actions

2. Exploitation
   2.1 Best actions according to the optimality criterion

Deployment

1. Exploitation

The agent $\pi$ aims to maximize the optimality criterion:

$$\max \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \right]$$
Reinforcement learning is enabled by many great projects such as:

1. OpenAI's Gymnasium

https://gymnasium.farama.org/
Reinforcement learning is enabled by many great projects such as:

1. OpenAI’s Gymnasium \(^1\),

\(^1\)https://gymnasium.farama.org/
Reinforcement learning is enabled by many great projects such as:

1. OpenAI’s Gymnasium,
2. DeepMind’s MuJoCo \(^2\),

\(^2\)https://mujoco.org/
Reinforcement learning is enabled by many great projects such as:

1. OpenAI’s Gymnasium,
2. DeepMind’s MuJoCo,
3. SUMO and Carla \(^3\).

\(^3\)https://eclipse.dev/sumo/ and http://carla.org/
Reinforcement learning is enabled by many great projects such as:

1. OpenAI’s Gymnasium,
2. DeepMind’s MuJoCo,
3. SUMO and Carla,
4. PettingZoo and Melting Pot

---

4pettingzoo.farama.org/ github.com/google-deepmind/meltingpot
import gymnasium as gym

class YourEnv(gym.Env):

...  

    def step(action) -> reward, state:
        # environment dynamics
        ...

        return reward, state
import gymnasium as gym

class YourEnv(gym.env):
    :
    
def step(action) -> reward, state:
        # environment dynamics
        :
        return reward, state

class Agent:
    :
    
def act(reward, state) -> action:
        # agent dynamics
        :
        return action
Agent trying to maximize an optimality criterion must be creative.

This creativity has the potential to endanger the agent or its environment.\(^5\)

\(^1\)https://www.reuters.com/article/factcheck-ai-drone-kills-idUSL1N38023R/
Introduction to Safe RL

What makes RL safe?\(^6\)

Modifying the optimality criterion to include safety.

---

Introduction to Safe RL

What makes RL safe?\(^6\)

Modifying the optimality criterion to include safety.

Modifying the agent's actions to ensure safety.

---

Scenario 1: Modification of the Optimality Criterion
Practical Scenarios
Scenario 1: Modification of the Optimality Criterion

Markov decision process $= \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle$

To make the agent consider safety, we modify the original reward function $R$:

$\hat{R} = R + H$

where $H : S \times A \rightarrow \mathbb{R}$ is our safety modification.

How to obtain $H$:
1. self-engineer it,
2. infer from some data.

---

Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = $\langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle$

To make the agent consider safety, we modify the original reward function $R$:

$$\hat{R} = R + H$$

where $H : S \times A \rightarrow \mathbb{R}$ is our safety modification.

---

Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = $\langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle$

To make the agent consider safety, we modify the original reward function $R$:

$$\hat{R} = R + H$$

where $H : S \times A \to \mathbb{R}$ is our safety modification.

How to obtain $H$:

1. self-engineer it,
2. infer from some data\(^7\).

Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle \)

\[
\max_{\pi} \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t (R(s_t, a_t) + H(s_t, a_t)) \right] \\
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases}

H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases}
\]
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[ R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases} \]

\[ H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases} \]

Diagram:

- Environment
- Agent
- State, reward
- State, safe reward
- Action

Trajectory 1:
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise} 
\end{cases}
\]

Trajectory 1:
1. reward is -1 + 0
Practical Scenarios
Scenario 1: Modification of the Optimality Criterion

Markov decision process = \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle

\begin{align*}
R &= \begin{cases}
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases} \\
H &= \begin{cases}
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases}
\end{align*}

Trajectory 1:
1. reward is -1 + 0
2. reward is -1 + 0
Practical Scenarios
Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise} 
\end{cases}
\]

Trajectory 1:
1. reward is \(-1 + 0\)
2. reward is \(-1 + 0\)
3. reward is \(-1 + 0\)
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle

\[ R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases} \]

\[ H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases} \]

Trajectory 1:
1. reward is -1 + 0
2. reward is -1 + 0
3. reward is -1 + 0
4. reward is -1 + 0
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process $= \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle$

$$R = \begin{cases} 100 & \text{reaching goal} \\ -1 & \text{otherwise} \end{cases}$$

$$H = \begin{cases} -100 & \text{reaching water} \\ 0 & \text{otherwise} \end{cases}$$

Trajectory 1:
1. reward is $-1 + 0$
2. reward is $-1 + 0$
3. reward is $-1 + 0$
4. reward is $-1 + 0$
5. reward is $-1 + 0$
**Practical Scenarios**

**Scenario 1: Modification of the Optimality Criterion**

Markov decision process = $\langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle$

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases}
\]

Trajectory 1:
1. reward is -1 + 0
2. reward is -1 + 0
3. reward is -1 + 0
4. reward is -1 + 0
5. reward is -1 + 0
6. reward is 100 + 0
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \langle S, \mathcal{A}, R : S \times \mathcal{A} \rightarrow \mathbb{R}, T : S \times \mathcal{A} \rightarrow S \rangle

\[ R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases} \]

\[ H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise} 
\end{cases} \]
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process $= \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle$

$R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases}$

$H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases}$

Trajectory 2:

1. reward is $-1 + 0$
Practical Scenarios
Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise}
\end{cases}
\]

Trajectory 2:
1. reward is -1 + 0
2. reward is -1 - 100
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise} 
\end{cases}
\]

Trajectory 2:
1. reward is \(-1 + 0\)
2. reward is \(-1 - 100\)
3. reward is \(-1 + 0\)
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise} 
\end{cases}
\]

Trajectory 2:
1. reward is -1 + 0
2. reward is -1 - 100
3. reward is -1 + 0
4. reward is -1 + 0

Environment \rightarrow R + H \rightarrow Agent 

state, reward \rightarrow R + H \rightarrow state, safe reward
Practical Scenarios
Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

\[
H = \begin{cases} 
-100 & \text{reaching water} \\
0 & \text{otherwise} 
\end{cases}
\]

Trajectory 2:
1. reward is -1 + 0
2. reward is -1 - 100
3. reward is -1 + 0
4. reward is -1 + 0
5. reward is -1 + 0
Practical Scenarios

Scenario 1: Modification of the Optimality Criterion

Markov decision process = \( \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle \)

\[ R = \begin{cases} 
100 & \text{reaching goal} \\ 
-1 & \text{otherwise} 
\end{cases} \]

\[ H = \begin{cases} 
-100 & \text{reaching water} \\ 
0 & \text{otherwise} 
\end{cases} \]

Trajectory 2:
1. reward is \(-1 + 0\)
2. reward is \(-1 - 100\)
3. reward is \(-1 + 0\)
4. reward is \(-1 + 0\)
5. reward is \(-1 + 0\)
6. reward is \(100 + 0\)
Properties:

1. Safety only during the deployment phase.
2. Requires the dataset of safe behaviours.
3. We don’t need to define what “safety” means.
Scenario 2: Modification of the Agent’s Actions
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Formal methods:

- Transition System
- Model Checker - Does the system satisfy the specification?
- Yes
- No

software or hardware

Requirements

Specification
Formal methods for reinforcement learning\(^8\):

Formal methods for reinforcement learning

- Environment
- Agent
- Shield

State, reward: \( \rightarrow \)

Safe action: \( \leftarrow \)

Action: \( \rightarrow \)
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Markov decision process $= \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle$

$$R = \begin{cases} 100 & \text{reaching goal} \\ -1 & \text{otherwise} \end{cases}$$

Trajectory 1:
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Markov decision process \(= \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle\)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

Trajectory 1:
1. reward is -1
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases}
\]

Trajectory 1:
1. reward is -1
2. reward is -1
Scenario 2: Modification of the Agent’s Actions

Markov decision process $= \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle$

$R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases}$

Trajectory 1:
1. reward is -1
2. reward is -1
3. reward is -1
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Markov decision process = \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle

\[ R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases} \]

Trajectory 1:
1. reward is -1
2. reward is -1
3. reward is -1
4. reward is -1
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[ R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases} \]

Trajectory 1:
1. reward is -1
2. reward is -1
3. reward is -1
4. reward is -1
5. reward is -1
Practical Scenarios

Scenario 2: Modification of the Agent’s Actions

Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[ R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise} 
\end{cases} \]

Trajectory 1:
1. reward is -1
2. reward is -1
3. reward is -1
4. reward is -1
5. reward is -1
6. reward is 100
Markov decision process = \( \langle S, A, R : S \times A \rightarrow \mathbb{R}, T : S \times A \rightarrow S \rangle \)

\[
R = \begin{cases} 
100 & \text{reaching goal} \\
-1 & \text{otherwise}
\end{cases}
\]
Markov decision process = \( \langle S, A, R : S \times A \to \mathbb{R}, T : S \times A \to S \rangle \)

\[ R = \begin{cases} 100 & \text{reaching goal} \\ -1 & \text{otherwise} \end{cases} \]

Trajectory 2:
1. reward is -1
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Properties:

1. Keeps the agent provably safe during training and deployment.
Practical Scenarios
Scenario 2: Modification of the Agent’s Actions

Properties:
1. Keeps the agent provably safe during training and deployment.
2. The guarantee is only with respect to the transition system!
3. We must be able to come up with the transition system.
4. We must know the safety specifications.
Thank You!
