Reinforcement Learning with Almost Sure Constraints

Enrique Mallada

A. Castellano  H. Min  J. Bazerque

Workshop on Control and Machine Learning: Challenges and Progress
Oct 12, 2023
A World of Success Stories

2017 Google DeepMind’s DQN

2017 AlphaZero – Chess, Shogi, Go

2019 AlphaStar – Starcraft II

OpenAI – Rubik’s Cube

Boston Dynamics

Waymo
Can we adapt reinforcement learning algorithms to address physical systems challenges?
Challenges of RL for Physical Systems

- Physical systems must meet **multiple objectives**
  - Need to **trade off** between the different goals
  - Constrained RL allows to explore the **Pareto Front** \cite{zheng2022constrained, you2021saddle}

\[
\begin{align*}
\max_{\pi} \quad & \mathbb{E}_{\pi,S_0 \sim q} \left[ \sum_{t=0}^{+\infty} \gamma^t R_{t+1}^{(0)} \right] \\
\text{s.t.} \quad & \mathbb{E}_{\pi,S_0 \sim q} \left[ \sum_{t=0}^{+\infty} \gamma^t R_{t+1}^{(i)} \right] \geq c_i, \quad \forall i \in [n]
\end{align*}
\]

- **Failures** have a qualitatively different impact
  - Expectation constraints cannot meet safety requirements
  - **Hard** (almost sure) constraints can guarantee safety \cite{castellano2021reinforcement, castellano2023learning, castellano2022correct}

\[
\begin{align*}
\max_{\pi} \quad & \mathbb{E}_{\pi,S_0 \sim q} \left[ \sum_{t=0}^{+\infty} \gamma^t R_{t+1} \right] \\
\text{s.t.} \quad & \mathbb{P}_{\pi,S_0 \sim q} \left[ S_t \notin G \right] = 1, \quad \forall t \geq 0
\end{align*}
\]

[1] Zheng, You, and M, Constrained reinforcement learning via dissipative saddle flow dynamics, Asilomar 2022
[2] You, and M, Saddle flow dynamics: Observable certificates and separable regularization, ACC 2021
[3] Castellano, Min, Bazerque, M, Reinforcement Learning with Almost Sure Constraints, L4DC 2022
[4] Castellano, Min, Bazerque, M, Learning to Act Safely with Limited Exposure and Almost Sure Certainty, IEEE TAC, 2023
[5] Castellano, Min, Bazerque M, Correct-by-design Safety Critics Using Non-contractive Bellman Operators, submitted
Safety-critical Constraints in Dynamical Systems

Reachability Theory\textsuperscript{[1-2]}

• **Model-based**: Via Hamilton Jacobi Issacs Equations (cont. time), or iterative set updates (discrete time).
• **Constraints**: Provides hard/almost sure guarantees
• **Output**: Finds the maximum control invariant set (M-CIS) outside $\mathcal{G}$

Control Barrier Functions (CBF)\textsuperscript{[3-4]}

• **Model-based**: Requires knowledge of dynamics and finding such CBF!
• **Constraints**: Provides hard/almost sure guarantees
• **Output**: Possibly conservative CIS

Safety Critics (SC)\textsuperscript{[5-7]}

• **Model-free**: Q-Learning-like algorithms, computes function such that $Q_{safe}(s,a) \geq \eta_{thresh} \Rightarrow \text{"safety"}$
• **Constraints**: Provides soft/approximate guarantees, depending on discounting factor $\gamma \in (0,1)$
• **Output**: Converges to maximum CIS as $\gamma \to 1$

\textsuperscript{[1]} I Mitchell, A Bayen, and C Tomlin. “A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games.” IEEE TAC, 2005
\textsuperscript{[2]} D Bertsekas. “Infinite time reachability of state-space regions by using feedback control.” IEEE TAC, 1972
\textsuperscript{[3]} A Ames, X Xu, J Grizzle, and P Tabuada, “Control barrier function based quadratic programs for safety critical systems,” IEEE TAC, 2017.
\textsuperscript{[4]} A Ames, S Coogan, M Egerstedt, G Notomista, K Sreenath, and P Tabuada. “Control barrier functions: Theory and applications” ECC, 2019
\textsuperscript{[5]} J Fisac, N Lugovoy, V Rubies-Royo, S Ghosh, and C Tomlin, “Bridging Hamilton-Jacobi safety analysis and reinforcement learning,” ICRA, 2019.
\textsuperscript{[6]} K Srinivasan, B Eysenbach, S Ha, J Tan, and C Finn. "Learning to be safe: Deep RL with a safety critic." arXiv preprint arXiv:2010.14603 (2020).
\textsuperscript{[7]} B Thananjeyan, A Balakrishna, S Nair, M Luo, K Srinivasan, M Hwang, J E Gonzalez, J Ibarz, C Finn, and K Goldberg. Recovery RL: Safe reinforcement learning with learned recovery zones. IEEE Robotics and Automation Letters, 2021
# Safety-critical Constraints in Dynamical Systems

**Reachability Theory**\(^{[1-2]}\)
- **Model-based:** Via Hamilton Jacobi Issacs Equations (cont. time), or iterative set updates (discrete time).
- **Constraints:** Provides hard/almost sure guarantees
- **Output:** Finds the maximum control invariant set (M-CIS) outside \(\mathcal{G}\)

**Control Barrier Functions (CBF)**\(^{[3-4]}\)
- **Model-based:** Requires knowledge of dynamics and finding such CBF!
- **Constraints:** Provides hard/almost sure guarantees
- **Output:** Possibly conservative CIS

**Safety Critics (SC)**\(^{[5-7]}\)
- **Model-free:** Q-Learning-like algorithms, computes function such that \(Q_{safe}(s,a) \geq \eta_{thresh} \Rightarrow “safety”\)
- **Constraints:** Provides soft/approximate guarantees, depending on discounting factor \(\gamma \in (0,1)\)
- **Output:** Converges to maximum CIS as \(\gamma \rightarrow 1\)

| Method                        | Model-free | Constraint Type | Size                   | Control Invariant? |
|-------------------------------|------------|-----------------|------------------------|--------------------|
| Reachability Theory\(^{[1-2]}\) | No         | Hard            | Maximum CIS            | Yes                |
| Control Barrier Functions\(^{[3-4]}\) | No         | Hard            | Subset of M-CIS        | Yes                |
| Safety Critics\(^{[5-7]}\)    | Yes        | Soft/Approx.    | Maximum CIS as \(\gamma \rightarrow 1\) | No                |
Safety-critical Constraints in Dynamical Systems

Reachability Theory\(^{[1-2]}\)
- **Model-based:** Via Hamilton Jacobi Issacs Equations (cont. time), or iterative set updates (discrete time).
- **Constraints:** Provides hard/almost sure guarantees
- **Output:** Finds the *maximum control invariant set (M-CIS) outside* \(\mathcal{G}\)

Control Barrier Functions (CBF)\(^{[3-4]}\)
- **Model-based:** Requires knowledge of dynamics and *finding such CBF!*
- **Constraints:** Provides hard/almost sure guarantees
- **Output:** Possibly conservative CIS

Safety Critics (SC)\(^{[5-7]}\)
- **Model-free:** Q-Learning-like algorithms, computes function such that \(Q_{safe}(s,a) \geq \eta_{thresh} \Rightarrow "safety"\)
- **Constraints:** Provides soft/approximate guarantees, depending on discounting factor \(\gamma \in (0,1)\)
- **Output:** *Converges to maximum CIS as* \(\gamma \to 1\)

| Method                  | Model-free | Constraint Type     | Size                     | Control Invariant? |
|-------------------------|------------|---------------------|--------------------------|--------------------|
| Reachability Theory\(^{[1-2]}\) | No         | Hard                | Maximum CIS             | Yes                |
| Control Barrier Functions\(^{[3-4]}\) | No         | Hard                | Subset of M-CIS          | Yes                |
| Safety Critics\(^{[5-7]}\) | Yes        | Soft/Approx.        | Maximum CIS as \(\gamma \to 1\) | No                 |
| Ours                    | Yes        | Hard                | M-CIS and Subsets        | Yes                |
Reinforcement Learning with Almost Sure Constraints
Agustin Castellano, Hancheng Min, Juan Bazerque, Enrique Mallada

Learning to Act Safely with Limited Exposure and Almost Sure Certainty
Agustin Castellano, Hancheng Min, Juan Bazerque, Enrique Mallada
Learning for Safety-critical Sequential Decision Making

Methodology:

• Enhance RL with **logical** feedback naturally arising from constraint violations
  \[ S_t \in \mathcal{G} \iff D_t = 1 \]

• Decouple **feasibility** from optimality: Separation Principle

• Develop algorithms for learning fixed points of **non-contractive operators**

Requirements:

**High Priority -> Safety**
- Limited Failures/Mistakes
- Hard Constraints/ A.S. Guarantees

**Lower Priority -> Accuracy**
- Optimality of the policy
Recap: RL with Almost Sure Constraints

$$\max_{\pi} \mathbb{E}_{\pi,S_0 \sim q} \left[ \sum_{t=0}^{+\infty} \gamma^t R_{t+1} \right]$$

s.t.  $\mathbb{P}_{\pi,S_0 \sim q} \left[ S_t \notin \mathcal{G} \right] = 1, \ \forall t \geq 0 \iff D_{t+1} = 0$ almost surely $\forall t$

- Damage indicator $D_t \in \{0,1\}$ turns on ($D_t = 1$) when constraints are violated
- Constraints not given a priori: Need to learn from experience!
- **Notice**: Model free $\Rightarrow$ Constraint violations are inevitable
Outline

• Separation Principle for Joint Safety & Optimality

• Learning Safety with Limited Failures

• One-sided Bellman Equations for Continuous States
Formulation via hard barrier indicator

Safe RL problem:

\[ V^*(s) := \max_{\pi} \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t R_{t+1} \mid S_0 = s \right] \]

s.t.: \( D_{t+1} = 0 \) almost surely \( \forall t \)

Equivalent unconstrained formulation:

\[ \max_{\pi} \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t R_{t+1} + \log(1 - D_{t+1}) \mid S_0 = s \right] \]

\[ \begin{array}{ll}
0 & \text{if } D_{t+1} = 0 \\
-\infty & \text{if } D_{t+1} = 1
\end{array} \]

Questions/Comments:
- Is this just a standard RL problem with \( \tilde{R}_{t+1} = R_{t+1} + \log(1 - D_{t+1}) \)?
- Standard MDP assumptions for Value Iteration, Bellman’s Eq., Optimality Principle, etc., do not hold!
- Not to mention convergence of stochastic approximations.

Key idea: Separate the problem of safety from optimality
Hard Barrier Action-Value Functions

Consider the Q-function for a given policy $\pi$,

$$Q^\pi(s, a) = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} (\gamma^t R_{t+1} + \log(1 - D_{t+1})) \mid S_0 = s, A_0 = a \right]$$

and define the hard-barrier function

$$B^\pi(s, a) = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \log(1 - D_{t+1}) \mid S_0 = s, A_0 = a \right]$$

**Notes on $B^\pi(s, a)$:**

- $B^\pi(s, a) \in \{0, -\infty\}$
- Summarizes safety information
  - $B^\pi(s, a) = 0$ iff $\pi$ is safe after choosing $A_t = a$ when $S_t = s$
- It is independent of the reward process
**Separation Principle**

**Theorem** (Separation principle)
Assume rewards $R_{t+1}$ are bounded almost surely for all $t$. Then for every policy $\pi$:

$$Q^\pi(s, a) = Q^\pi(s, a) + B^\pi(s, a)$$

In particular, for optimal $\pi^*$

$$Q^*(s, a) = Q^*(s, a) + B^*(s, a)$$

**Approach:** Learn feasibility (encoded in $B^*$) independently from optimality.
Optimal Hard Barrier Action-Value Function

**Theorem** (Safety Bellman Equation for $B^*$)

Let $B^*(s, a) := \max_{\pi} B^\pi(s, a)$, then the following holds:

$$B^*(s, a) = \mathbb{E} \left[ -\log(1 - D_{t+1}) + \max_{a'} B^*(S_{t+1}, a') \mid S_0 = s, A_0 = a \right]$$

Understanding $B^*(s, a)$:

- $B^*(s, a) \in \{0, -\infty\}$ summarizes safety information of the entire MDP
  - $B^*(s, a) = 0$ if $\exists$ safe $\pi$ after choosing $A_t = a$ when $S_t = s$ **Control Invariant**
  - $B^*(s, a) = -\infty$ if no safe policy exists after choosing $A_t = a$ when $S_t = s$ **Unsafe**

### Discrete States

- $V^*(s) = \max_a B^*(s, a) = 0$
- $V^*(s) = \max_a B^*(s, a) = -\infty$

### Continuous States

**Remark**

- $\mathbb{R}(\mathcal{G})$ represents the reachable set of the continuous states
- $\mathcal{G}$ represents the set of constraints or safe regions

---

$D_t = 1$ controlled safe trajectory
Properties of Safety Bellman Equation

**Theorem (Safety Bellman Equation for $B^*$)**

Let $B^* (s, a) := \max_\pi B^\pi (s, a)$, then the following holds:

$$B^* (s, a) = \mathbb{E} \left[ -\log(1 - D_{t+1}) + \max_{a'} B^* (S_{t+1}, a') \mid S_0 = s, A_0 = a \right]$$

Understanding the Solutions to the Safety Bellman Equation (SBE):

- SBE can have **multiple solutions**, including $\tilde{B} (s, a) = -\infty$, for all pairs $(s, a)$
- If the function $\tilde{B}$ is a solution to the SBE, then:
  - The set $\mathcal{C} := \{ s : \max_a \tilde{B} (s, a) = 0 \}$ is a **control invariant safe set**
  - $\mathcal{C}$ is **maximal**: If $S_0 \notin \mathcal{C}$, then $S_t$ never reaches $\mathcal{C}$ for all policies $\pi$
Outline

- Separation Principle for Joint Safety & Optimality
- Learning Safety with Limited Failures
- One-sided Bellman Equations for Continuous States
Learning the barrier in finite MDPs...

Pros:
- Wraps around learning algorithms (Q-learning, SARSA)
- Use the B to trim the exploration set and avoid repeating unsafe actions

...with a generative model:
- Sample a transition \((s, a, s', d)\) according to the MDP. Update barrier function.

Algorithm 3: barrier_update

\[ B\text{-function (initialized as all-zeros)}; \]

Input: \((s, a, s', d)\)

Output: Barrier-function \(B(s, a)\)

\[
B(s, a) \leftarrow B(s, a) + \log(1 - d) + \max_{a'} B(s', a')
\]

Algorithm 5: Barrier Learner Algorithm

Data: Constrained Markov Decision Process \(\mathcal{M}\)

Result: Optimal action-value function \(B^*\)

Initialize \(B^{(0)}(s, a) = 0, \forall (s, a) \in \mathcal{S} \times \mathcal{A}\)

for \(t = 0, 1, \ldots\) do

- Draw \((s_t, a_t) \sim \text{Unif}\{\{(s, a) : B^{(t)}(s, a) \neq -\infty\}\}\)
- Sample transition \((s_t, a_t, s'_t, d_t)\) according to
  \[
P(S_1 = s'_t, D_1 = d_t | S_0 = s_t, A_0 = a_t)
\]
- \(B^{(t+1)} \leftarrow \text{barrier_update}(B^{(t)}, s_t, a_t, s'_t, d_t)\)

end

Initially, all \((s, a)\)-pairs are “safe”

Draw \((s, a)\)-pair uniformly among those considered to be “safe” at time \(t\)

Update barrier function
Theorem (Safety Guarantee): Let $T = \min_t \{B(t) = B^*\}$, then

$$\mathbb{E}T \leq (L + 1) \frac{|S||A|}{\mu} \left( \sum_{k=1}^{\frac{|S||A|}{1}} \frac{1}{k} \right)$$

- After $T = \min_t \{B(t) = B^*\}$, all “unsafe” $(s, a)$-pairs are detected
- $\mu$: Lower bound on the non-zero transition probability
  \[ \mu = \min \{ p(s', d|s, a): p(s', d|s, a) \neq 0 \} \]
- $L$: Lag of the MDP

\[ L = \max_{(s, a)} \begin{cases} \text{Minimum number of transitions} \\ \text{needed to observe damage,} \\ \text{starting from unsafe } (s, a) \end{cases} \]
Lag of the MDP: L

\[
L = \max_{(s,a)} \left\{ \right. \quad \text{Minimum number of transitions needed to observe damage, starting from unsafe (s, a)} \quad \left. \right\}
\]

\[
B^*(s,a) = -\infty
\]

\[L = 3\]
Sample Complexity of Safety

Theorem (Sample Complexity): With at least $1 - \delta$ probability, the algorithm learns optimal barrier function $B^*$ after

$$ (L + 1) \frac{|S||A|}{\mu} \left( \sum_{k=1}^{\frac{|S||A|}{\mu}} \frac{1}{k} \right) \log \frac{1}{\delta} $$

iterations

- Concentration of sum of exponential random variables

- **Much more sample-efficient** than “learning an $\epsilon$-optimal policy with $1 - \delta$ probability” (Li et al. 2020)

$$ N = \frac{|S||A|}{(1 - \gamma)^4 \epsilon^2} \log^2 \left( \frac{|S||A|}{(1 - \gamma)\epsilon\delta} \right) $$
Sample Complexity of Safety

Theorem (Sample Complexity): With at least $1 - \delta$ probability, the algorithm learns optimal barrier function $B^*$ after

$$(L + 1) \frac{|S||A|}{\mu} \left( \sum_{k=1}^{\frac{|S||A|}{\mu}} \frac{1}{k} \right) \log \frac{1}{\delta}$$

iterations

- Concentration of sum of exponential random variables

- If the Barrier Function is learnt first, then learning an $\epsilon$-optimal policy takes

$$N' = \frac{|S_{safe}||A_{safe}|}{(1 - \gamma)^4 \epsilon^2} \log^2 \left( \frac{|S_{safe}||A_{safe}|}{(1 - \gamma)\epsilon\delta} \right)$$

samples (Trimming the MDP by learning the barrier)
**Numerical Experiments**

**Goal:** Reach the end of the aisle ($R_{t+1} = 10$)

Touching the wall gives $D_{t+1} = 1$, resets the episode.

**Results**

**Why does Assured Q-learning perform much better?**

$$\text{If } D_{t+1} = 1 \implies B_{\pi}(s, a) = -\infty \implies \text{Never take action } a \text{ at } s \text{ again!}$$

**Takeaways:**

- Adding constraints to the problem can accelerate learning
- Barrier function avoids actions that lead to further wall bumps
**Numerical Experiments II**

**Setup:** Rectangular grid, stepping into **holes** gives damage $D_t = 1$.

Actions $A = \{up, down, left, right\}$.

With every action, small probability to move to a random adjacent state.

**Result:** Barrier-learner identifies **all** the state space as unsafe.

![Image of a grid with unsafe states indicated](image)

![Graph showing proportion of unsafe states detected over iterations](graph)
Numerical Experiments II

Setup: Rectangular grid, stepping into holes gives damage $D_t = 1$. Actions $A = \{up, down, left, right\}$. With every action, small probability to move to a random adjacent state.

Result: Barrier-learner identifies all the state space as unsafe. Immediately unsafe states (near damage) are identified first.
Outline

• Separation Principle for Joint Safety & Optimality

• Learning Safety with Limited Failures

• One-sided Bellman Equations for Continuous States
Recall: Properties of Safety Bellman Equation

**Theorem** (Safety Bellman Equation for $B^*$)

Let $B^*(s, a) := \max_{\pi} B^\pi(s, a)$, then the following holds:

\[
B^*(s, a) = \mathbb{E} \left[ -\log(1 - D_{t+1}) + \max_{a'} B^*(S_{t+1}, a') \mid S_0 = s, A_0 = a \right]
\]

Understanding the Solutions to the Safety Bellman Equation (SBE):

- SBE can have multiple solutions, including $\tilde{B}(s, a) = -\infty$, for all pairs $(s, a)$
- If the function $\tilde{B}$ is a solution to the SBE, then:
  - The set $\mathcal{C} := \{ s : \max_a \tilde{B}(s, a) = 0 \}$ is a control invariant safe set
  - $\mathcal{C}$ is maximal: If $S_0 \notin \mathcal{C}$, then $S_t$ never reaches $\mathcal{C}$ for all policies $\pi$

**Problem:** Maximal solutions can be very close to unsafe region $\mathcal{R}(\mathcal{G})$
Theorem (One-Sided Safety Bellman Equation)

Let $\bar{B}(s, a)$ be a solution of the following set of inequalities:

$$\bar{B}(s, a) \leq \mathbb{E} \left[ -\log(1 - D_{t+1}) + \max_{a'} \bar{B} (S_{t+1}, a') \mid S_0 = s, A_0 = a \right]$$

The set $\mathcal{C} := \left\{ s : \max_a \bar{B}(s, a) = 0 \right\}$ is a control invariant safe set, not necessarily maximal.
Learning CIS Using Deep Neural Nets

Algorithm Summary

• Uses axiomatic data \((s, a, d, s') \in \mathcal{D}_{safe}\) known to be safe

• Initialize \(\hat{b}^\theta(s, a) = 0\), where \(\hat{b}(s, a) = 1 - e^{B(s,a)}\) (all presumed safe)

• At each iteration, take \(N\) episodes starting from \(\mathcal{D}_{safe}\)
  • Behavioral policy: uniform safe policy
    \[
    \pi^\theta(a|s) = \begin{cases} 
    0 & \text{if } \hat{b}^\theta(s, a) = 1 \\
    \frac{1}{\sum_{a' \in \mathcal{A}} \mathbb{1}\{\hat{b}^\theta(s, a') = 0\}} & \text{if } \hat{b}^\theta(s, a) = 0
    \end{cases}
    \]

• Train NN using SGD until fully fitting the data

• Start a new iteration, and repeat
Numerical Illustration

Control Engineer Favorite’s: Inverted Pendulum

SBE = Fisac’s ‘19 Safety Critic
Summary and future work

- Reinforcement Learning for Safety-Critical Systems
- Treat constraints separately or in parallel (Barrier Learner)
- Can characterize all feasible policies ($D_t \equiv 0$) with finite mistakes
- Requires learning using Bellman equations with non-unique solutions

Takeaways:
- Learning feasible policies is simpler than learning the optimal ones
- Adding constraints makes optimal policies, easier to find
- One-sided Safe Bellman can be used to find CISs that are not maximal
Thanks!

Related Publications:
[1] Castellano, Min, Bazerque, M, Reinforcement Learning with Almost Sure Constraints, L4DC, 2022
[2] Castellano, Min, Bazerque, M, Learning to Act Safely with Limited Exposure and Almost Sure Certainty, IEEE TAC, 2023
[3] Castellano, Min, Bazerque M, Correct-by-design Safety Critics Using Non-contractive Bellman Operators, submitted

Enrique Mallada
mallada@jhu.edu
http://mallada.ece.jhu.edu