Aprendizaje por Refuerzo con Restricciones de Probabilidad Uno

(Reinforcement Learning with Almost Sure Constraints)

Enrique Mallada

Topología y Probabilidad en análisis de datos

Universidad de la República

[Submitted on 9 Dec 2021 (v1), last revised 7 Apr 2022 (this version, v2)]

Reinforcement Learning with Almost Sure Constraints

Agustin Castellano, Hancheng Min, Juan Bazerque, Enrique Mallada

arXiv > cs > arXiv:2112.05198

[Submitted on 18 May 2021 (v1), last revised 25 May 2021 (this version, v2)]

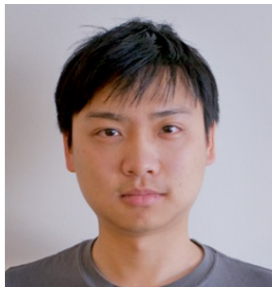
Learning to Act Safely with Limited Exposure and Almost Sure Certainty

Agustin Castellano, Hancheng Min, Juan Bazerque, Enrique Mallada

arXiv > eess > arXiv:2105.08748

Agustin Castellano

JOHNS HOPKINS UNIVERSITY



Hancheng Min

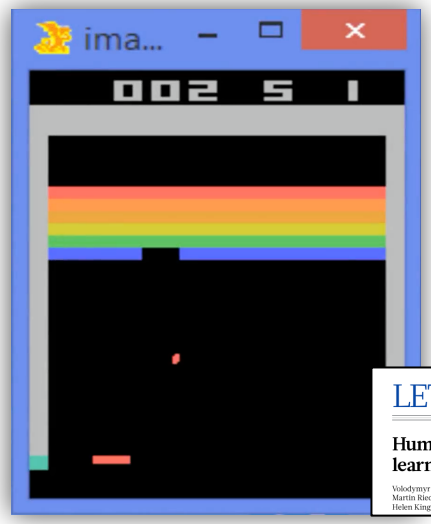
JOHNS HOPKINS UNIVERSITY

Juan Bazerque

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

A World of Success Stories

2017 Google DeepMind's DQN



LETTER

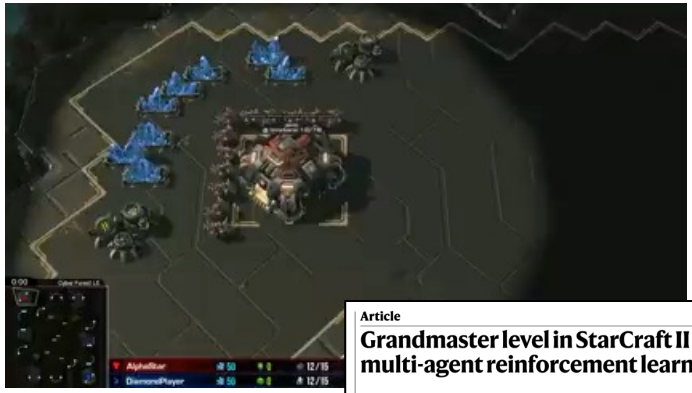
doi:10.1038/nature14236

Human-level control through deep reinforcement learning

Volodymyr Mnih[1*], Koray Kavukcuoglu[1*], David Silver[1*], Andrei A. Rusu[1], Joel Veness[1], Marc G. Bellemare[1], Alex Graves[1], Martin Riedmiller[1], Andreas K. Fidjeland[1], Georg Ostrovski[1], Stig Petersen[1], Charles Beattie[1], Amir Sadik[1], Ioannis Antonoglou[1], Helen King[1], Dharshan Kumaran[1], Daan Wierstra[1], Shane Legg[1] & Demis Hassabis[1]

2017 AlphaZero – Chess, Shogi, Go

2019 AlphaStar – Starcraft II



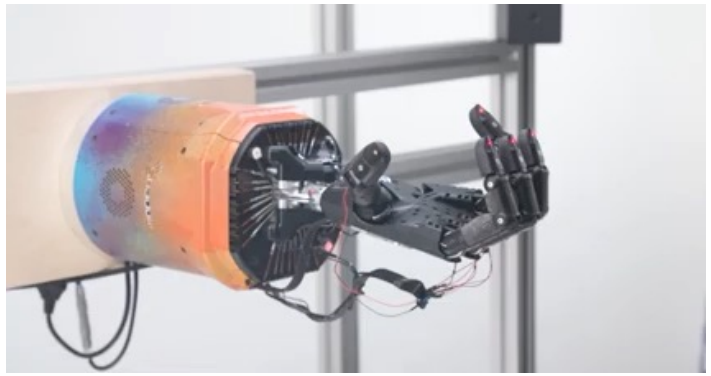
Article

Grandmaster level in StarCraft II using multi-agent reinforcement learning

https://doi.org/10.1038/s41586-019-1724-z
Received: 30 August 2019
Accepted: 10 October 2019
Published online: 30 October 2019

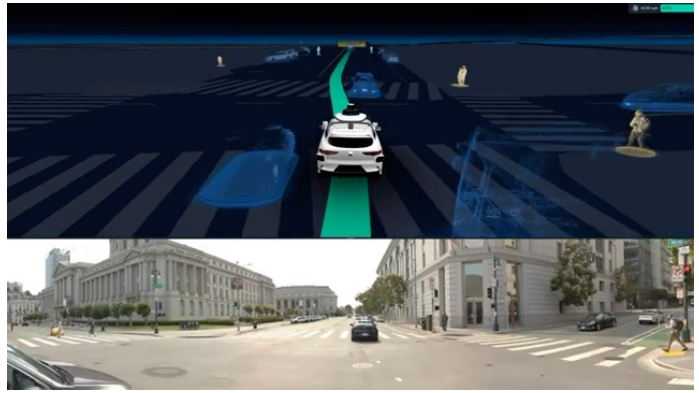
Oriol Vinyals[1,2*], Igor Babuschkin[1,2], Wojciech M. Czarnecki[1,2], Michaël Mathieu[1,2], Andrew Dudzik[1,2], Junyoung Chung[1,2], David H. Choi[1,2], Richard Powell[1,2], Timo Ewalds[1,2], Petko Georgiev[1,2], Junhyuk Oh[1,2], Dan Horgan[1,2], Manuel Kroiss[1,2], Ivo Danihelka[1,2], Aja Huang[1,2], Laurent Sifre[1,2], Trevor Cai[1,2], John P. Agapiou[1,2], Max Jaderberg[1], Alexander S. Vezhnevets[1], Rémi Leblond[1], Tobias Pohlen[1], Valentin Dalibard[1], David Budden[1], Yury Sulsky[1], James Molloy[1], Tom L. Paine[1], Caglar Gulcehre[1], Ziyu Wang[1], Tobias Pfaff[1], Yuhuai Wu[1], Roman Ring[1], Dani Yogatama[1], Dario Wünsch[1], Katrina McKinney[1], Oliver Smith[1], Tom Schaul[1], Timothy Lillicrap[1], Koray Kavukcuoglu[1], Demis Hassabis[1], Chris Apps[1,2] & David Silver[1,2*]

OpenAI – Rubik's Cube



Boston Dynamics

Waymo



Reality Kicks In

Angry Residents, Abrupt Stops: Waymo Vehicles Are Still Causing Problems in Arizona

RAY STERN | MARCH 31, 2021 | 8:26AM

GARY MARCUS BUSINESS 08.14.2019 09:00 AM

DeepMind's Losses and the Future of Artificial Intelligence

Alphabet's DeepMind unit, conqueror of Go and other games, is losing lots of money. Continued deficits could imperil investments in AI.

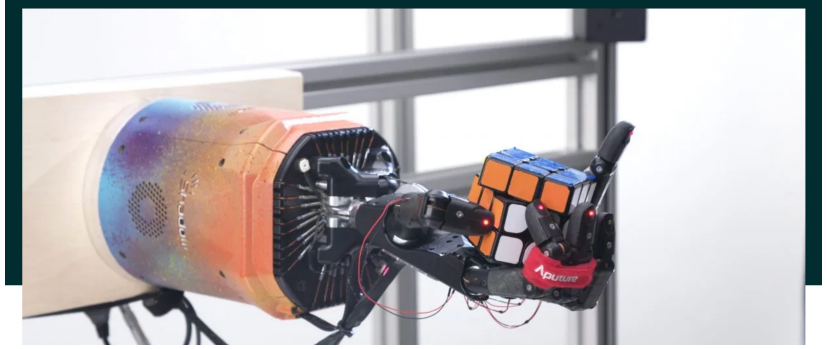
AARIAN MARSHALL BUSINESS 12.07.2020 04:06 PM

Uber Gives Up on the Self-Driving Dream

The ride-hail giant invested more than $1 billion in autonomous vehicles. Now it's selling the unit to Aurora, which makes self-driving tech.

OpenAI disbands its robotics research team

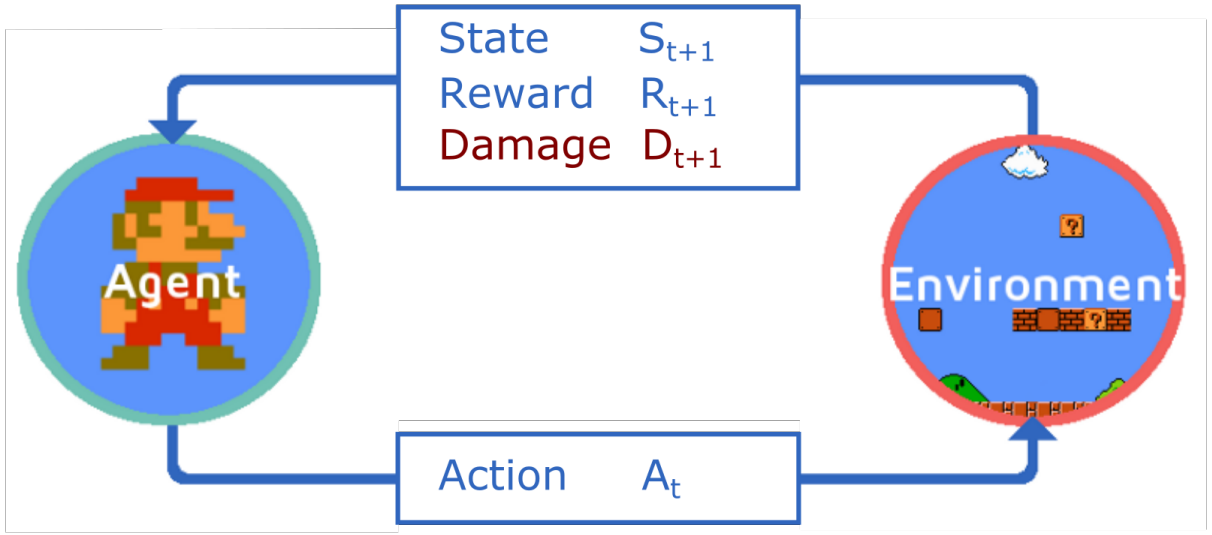
Kyle Wiggers @Kyle_L_Wiggers July 16, 2021 11:24 AM



Self-driving Uber car that hit and killed woman did not recognize that pedestrians jaywalk

The automated car lacked "the capability to classify an object as a pedestrian unless that object was near a crosswalk," an NTSB report said.

Learning for Safety-critical Sequential Decision Making



Requirements:

High Priority -> Safety

- Limited Failures/Mistakes
- Hard Constraints/ A.S. Guarantees

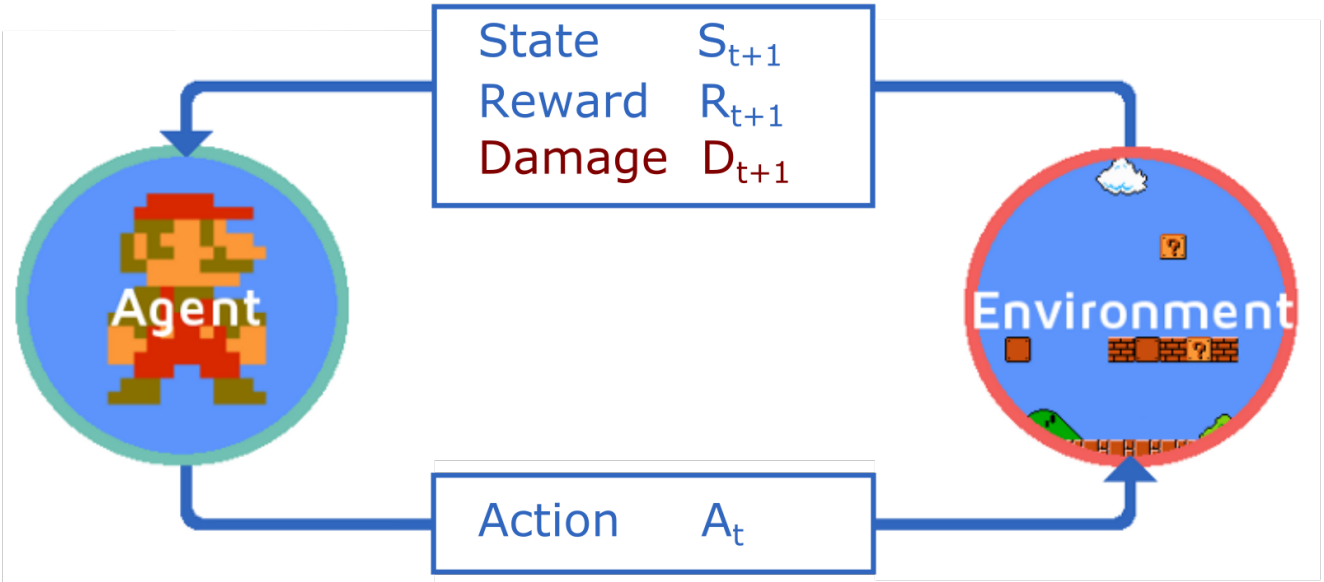
Lower Priority -> Accuracy

- Optimality of the policy

Key ideas:

- Focus on almost sure **feasibility**, not optimality (Egerstedt et al.,2018)
- Enhanced with **logical** feedback ($D_{t+1} = 0$ or 1), naturally arising from constraint violations

Learning for Safety-critical Sequential Decision Making



Talk Punchline:

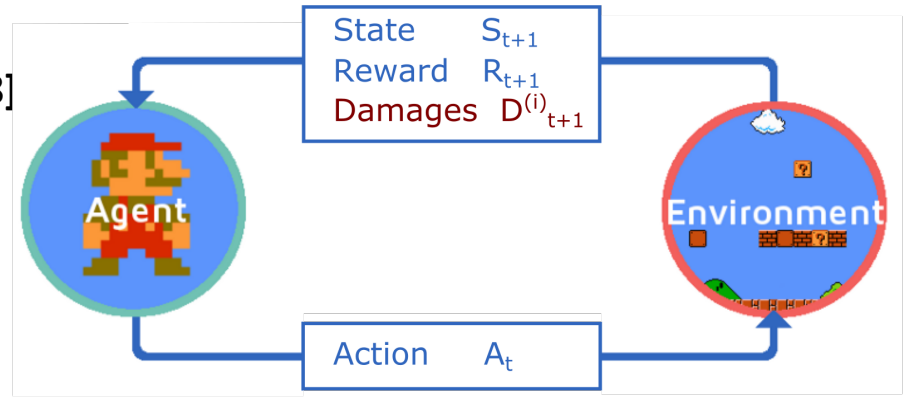
- Can **characterize** all feasible policies with **finite unsafe events** ($D_t = 1$)
- **Learning feasible policies is simpler** than learning the optimal ones
- Adding **almost sure constraints** makes **optimal policies easier to find**

Related work

- **Constrained Markov Decision Processes (CMDPs)** **[Altman'98]**

$$\max_{\pi \in \Pi} \quad V^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R_{t+1} | S_0 = s\right]$$

$$\text{s.t.:} \quad C_i^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t D_{t+1}^{(i)} | S_0 = s\right] \leq c_i \quad i = 1, \ldots, m$$



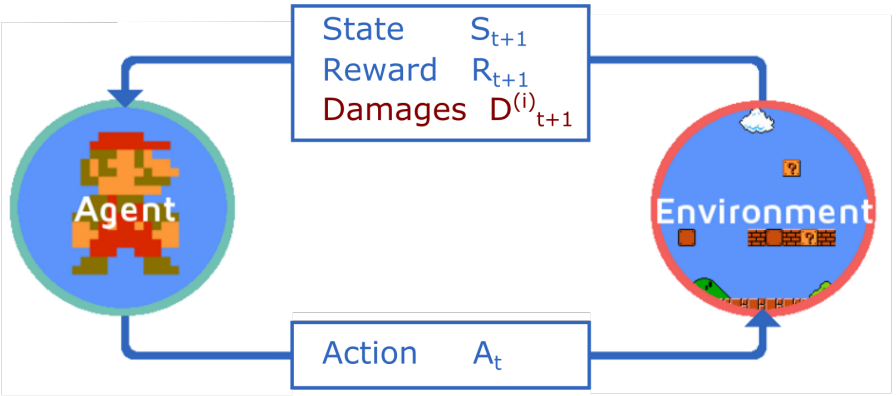
  - Solvable if MDP is "known" and state-space finite (Linear Program).
  - ∃ stationary optimal solution $\pi^*(a|s)$

- **What to do if MDP is "unknown"? Examples of Model-based and Model-free methods**
- (MB) Learn transitions and reward/constraint signals, solve for a (near) optimal policy.
  [Aria HZ et al'20], [Bai et al'20], [Wang et al 20], [Chen et al'21]
- (MF) Primal or Primal-dual methods.
  [Chow et al'17], [Tessler et al'19], [Paternain et al'19], [Ding et al'20], [Stooke et al. '20], [Xu et al'21]

Related work: Solving CMDPs

$$\max_{\pi\in\Pi} \quad V^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}R_{t+1}|S_0 = s\right]$$

$$\text{s.t.:} \quad C_i^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}D_{t+1}^{(i)}|S_0 = s\right] \leq c_i \quad i = 1,\dots,m$$



Generative model (MB):

1. $\forall(s,a)$ sample n transitions $(s,a) \to s', r$
2. Build kernel $\hat{P}(s'|s,a) = \frac{count(s,a,s')}{n}$ and rewards $\hat{r}(s,a) = avg(r)$
3. Find an optimal policy in the modified MDP $\widehat{\mathcal{M}}$ (Linear Program)
4. How good is this policy in the true MDP?

Can get ϵ optimal solution w.h.p. [Aria HZ et al'20]:

$$\mathbb{P}\left(V^{\hat{\pi}^*}(s) \geq V^{*}(s) - \epsilon \text{ and } C_i^{\hat{\pi}^*}(s) \leq c_i - \epsilon,\ \forall i \in \{1,\dots,m\}\right) \geq 1-\delta$$

provided $\quad n \geq C_1 \frac{\gamma^2}{\epsilon^2(1-\gamma)^3}|\mathcal{S}|^2|\mathcal{A}| \log \frac{C_2(m+2)|\mathcal{S}|^3|\mathcal{A}|}{\delta}$

Primal-dual (MF):

1. Lagrangian:

$$\mathcal{L}(\pi,\lambda) := V_{\pi}(s) - \sum_{i=1}^{m}\lambda_i\left(C_i^{\pi}(s) - c_i\right)$$

2. max-min problem:

$$\max_{\pi\in\Pi} \quad \mathcal{L}(\pi,\lambda) = \max_{\pi\in\Pi} \quad \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}\underbrace{\left(R_{t+1} - \sum_{i=1}^{m}\lambda_i D_{t+1}^{(i)}\right)}_{=\tilde{R}_{t+1}}|S_0 = s\right]$$

$$\min_{\lambda\geq 0}\mathcal{L}(\pi^*,\lambda)$$

3. Gradient descent + dual ascent
(e.g.: policy gradient methods [Ding et al'20])

Problem is non-concave. Guarantees?
"Constrained RL has zero-duality gap'' [Paternain et al'19]

Outline of the presentation

- Motivation & Background
- RL with almost sure constraints
  1. $\Delta = 0$: “Live or die” approach
  2. $\Delta \geq 0$: General case
- Summary and future work

Reinforcement Learning with Almost Sure constraints

Almost surely constrained RL

$$\max_{\pi \in \Pi_H} \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} R_{t+1} \mid S_0 = s \right]$$

$$\text{s.t: } P_\pi \left(\sum_{t=0}^{\infty} D_{t+1} \leq \Delta \mid S_0 = s \right) = 1$$

Total *budget* ($\Delta \in \mathbb{N}$)

Standard Constrained RL

$$\max_{\pi \in \Pi} \quad V^\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t R_{t+1} | S_0 = s \right]$$

$$\text{s.t.:} \quad C^\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t D_{t+1} | S_0 = s \right] \leq c$$

- Recall: $D_{t+1} = 1$ indicates an unsafe event.
- Proposed constraint: never allow more than Δ unsafe events in an episode.
- May be better suited for safety-critical applications.
  - Average constraints perform poorly in some trajectories.

Reinforcement Learning with almost sure constraints

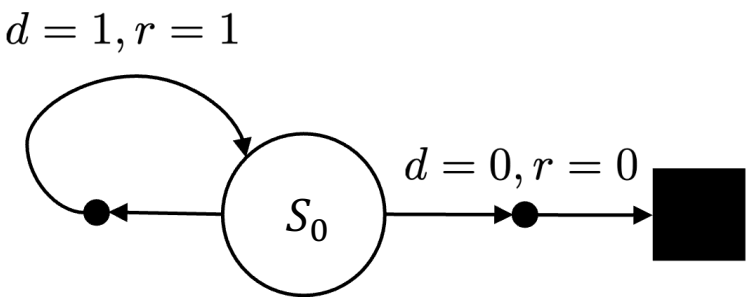
$$\max_{\pi \in \Pi_H} \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} R_{t+1} \mid S_0 = s \right]$$

$$\text{s.t: } P_\pi \left(\sum_{t=0}^{\infty} D_{t+1} \leq \Delta \mid S_0 = s \right) = 1$$

Outside the usual realm of CMDPs

Π_H : history-dependent policies $\quad h_t = (S_0, A_0, R_1, D_1, \dots, S_t); \quad \pi(a|h_t)$

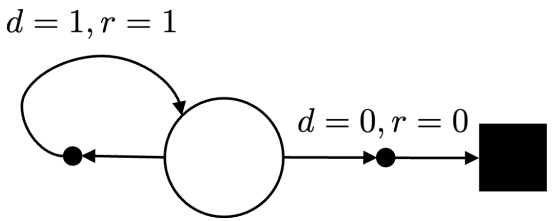
- Can we find (as for standard CMDPs) an optimal **stationary** policy?
- In general, **NO!**



Optimal policy: $V^{\pi_H^*} = \Delta$

The only feasible stationary policy has $V^{\pi_S} = 0$

Experiment: comparing constraints



Goal

$$\max_{\pi} \quad \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} R_{t+1}\right]$$

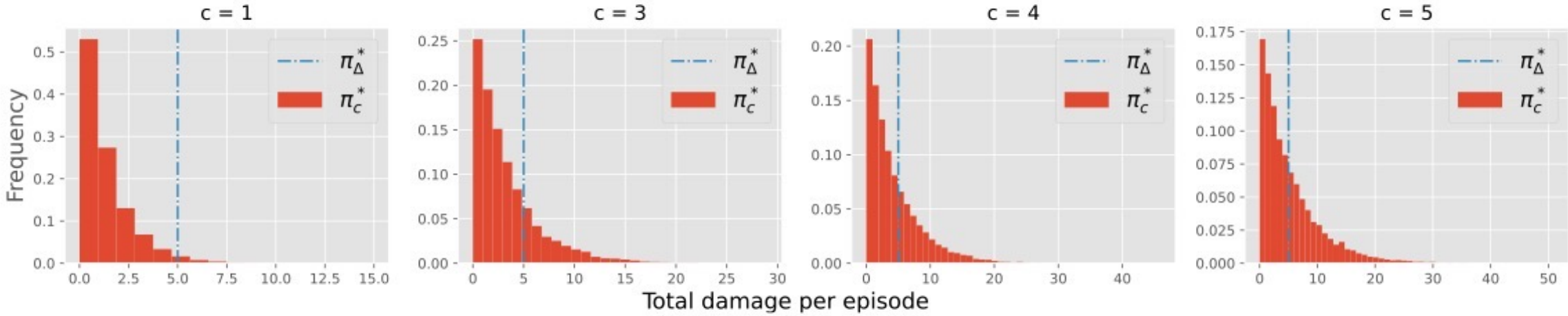
1) Proposed constraint

$$\mathbb{P}_{\pi_\Delta}\left(\sum_{t=0}^{\infty} D_{t+1} \leq \Delta \;\middle|\; S_0 = s\right) = 1$$

2) Classic CMDP constraint

$$\mathbb{E}_{\pi_c}\left[\sum_{t=0}^{\infty} D_{t+1}\right] \leq c$$

Safety of assured π_Δ^* with $\Delta = 5$ vs expectation-based constraint π_c^*; $P(d = 1) = 1$



- Takeaway: average constraints lead to poor performance in some trajectories

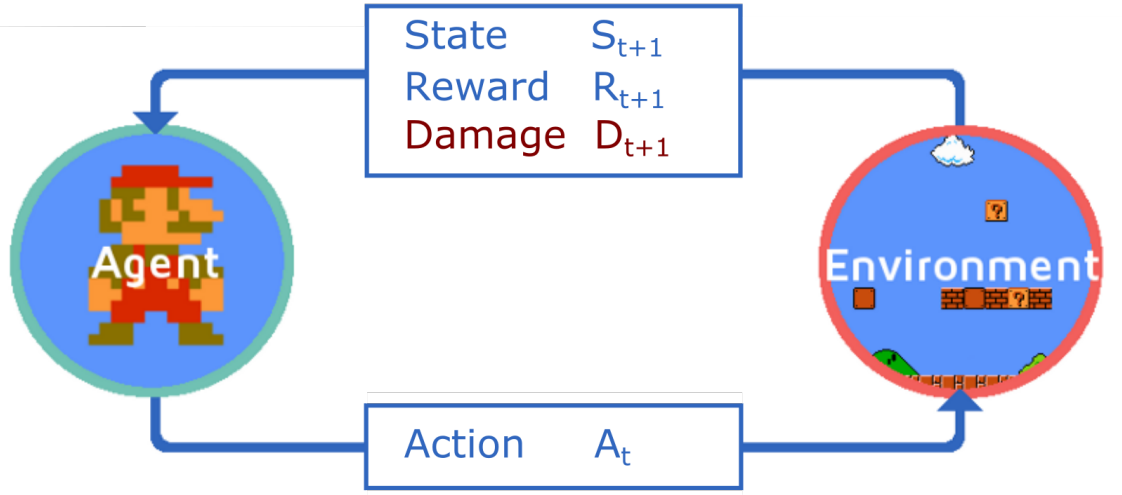
Outline of the presentation

- Motivation & Background
- RL with almost sure constraints
  1. $\Delta = 0$: “Live or die” approach
  2. $\Delta \geq 0$: General case
- Summary and future work

RL with almost sure constraints: $\Delta = 0$ case

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

$$\text{s.t.:} \quad \mathbb{P}_{\pi} \left(\sum_{t=0}^{\infty} D_{t+1} \leq 0 \mid S_0 = s \right) = 1 \iff D_{t+1} = 0 \text{ almost surely } \forall t$$



- Damage indicator $D_t \in \{0,1\}$ turns on ($D_t = 1$) when constraints are violated
- Particular case $\Delta = 0 \Rightarrow$ constraint can be put in average value $\Rightarrow$
  - Stationary policies are optimal for this problem

Formulation via hard barrier indicator

Safe RL problem:

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

$\sim$

Equivalent unconstrained formulation:

$$\max_{\pi} \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^t R_{t+1} + \underbrace{\log[1 - D_{t+1}]}_{\begin{matrix} 0 & if\ D_{t+1} = 0 \\ -\infty & if\ D_{t+1} = 1 \end{matrix}} \mid S_0 = s\right]$$

Questions/Comments:

- Is this just a standard RL problem with $\tilde{R}_{t+1} = R_{t+1} + \log(1 - D_{t+1})$?
- Special care: notice $\tilde{R}_{t+1}$ unbounded on the left.
- Convergence of stochastic approximations not readily guaranteed.

Key idea: Separate the problem of safety from optimality

Hard Barrier Action-Value Functions

Consider the Q-function for a given policy π,

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

and define the hard-barrier function

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

Notes on $\boldsymbol{B^{\pi}(s,a)}$:

- $\boldsymbol{B^{\pi}(s,a) \in \{0, -\infty\}}$
- Summarizes safety information
  - $\boldsymbol{B^{\pi}(s,a) = 0}$ iff π is safe after choosing $A_t = a$ when $S_t = s$

Optimal Hard Barrier Action-Value Function

Theorem (Separation principle)

Assume rewards R_{t+1} are bounded almost surely for all t. Then for optimal π_* we have

$$Q^*(s,a) = \underbrace{Q^*(s,a)}_{\text{Reward optimization}} + \underbrace{B^*(s,a)}_{\text{Constraint satisfaction}}$$

- $\boldsymbol{B^*(s,a) \in \{0, -\infty\}}$ summarizes safety information of the entire MDP

$$B^*(s,a) = \max_{\pi} \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \log(1 - D_{t+1}) \mid S_0 = s, A_0 = a\right] = \begin{cases} -\infty & \iff \text{no } \pi \text{ is safe from } (s,a) \\ 0 & \iff \text{otherwise.} \end{cases}$$

- **B^* is a descriptor of all feasible policies:**

$$\Pi_{\text{Safe}} = \{\pi : \pi(a|s) = 0 \text{ whenever } B^*(s,a) = -\infty\}$$

- **Idea:** Learn B^* (coming up next)

Optimal Hard Barrier Action-Value Function

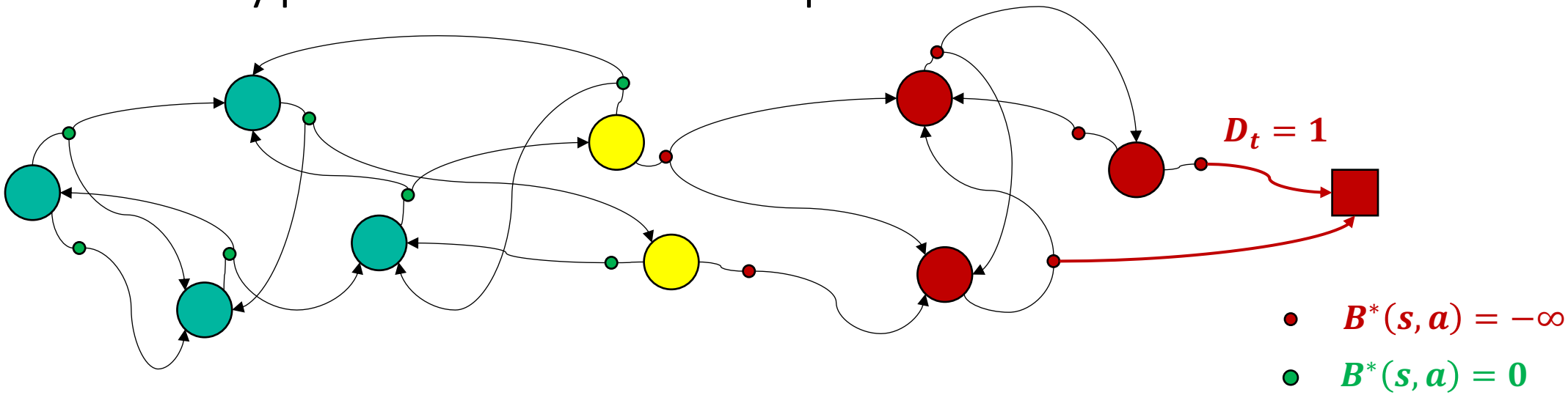
- How to learn B^*:

Theorem (Bellman Equation for B^*)

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

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

- B^* naturally partitions the state-action space

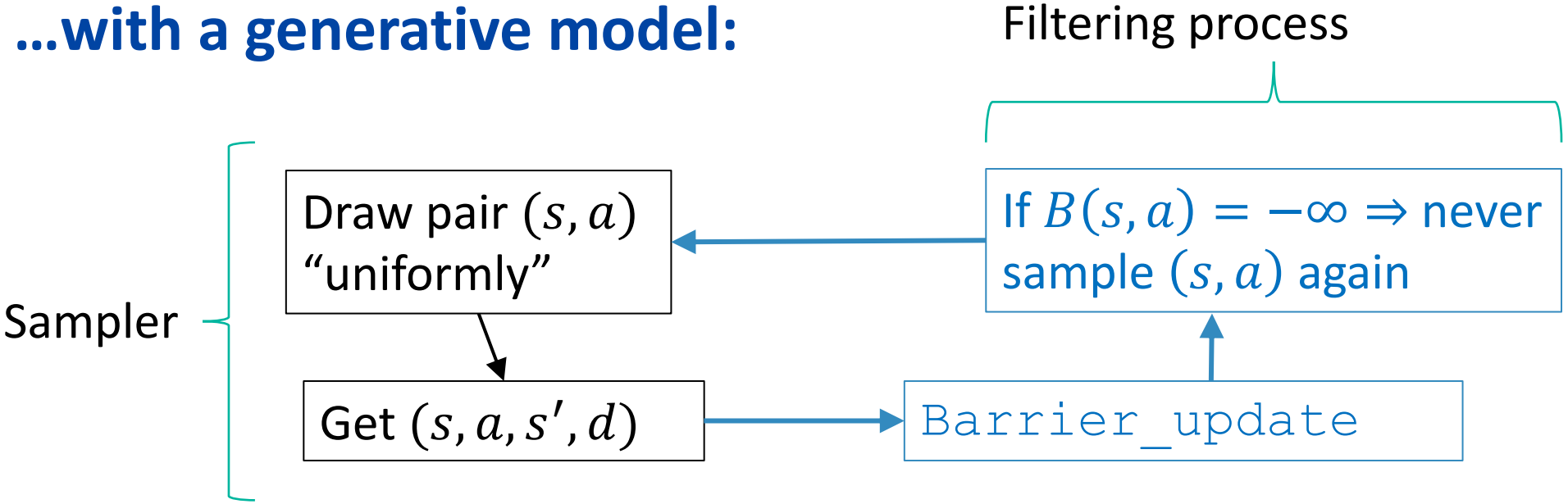


Learning the barrier…

Algorithm 3: `barrier_update`

B-function (initialized as all-zeroes);
Input: (s, a, s', d)
Output: Barrier-function $B(s, a)$
$B(s, a) \leftarrow B(s, a) + \log(1 - d) + \max_{a'} B(s', a')$

…with a generative model:



Convergence in Expected Finite Time

Theorem (Safety Guarantee): Let N be the number of steps needed to learn B^*, using the generative model. Then

$$\mathbb{E}[N] \leq (L+1)\frac{|S||A|}{\mu}\log(|S||A|)$$

- After N all "unsafe" (s,a)-pairs are detected
- μ: Lower bound on the non-zero transition probability

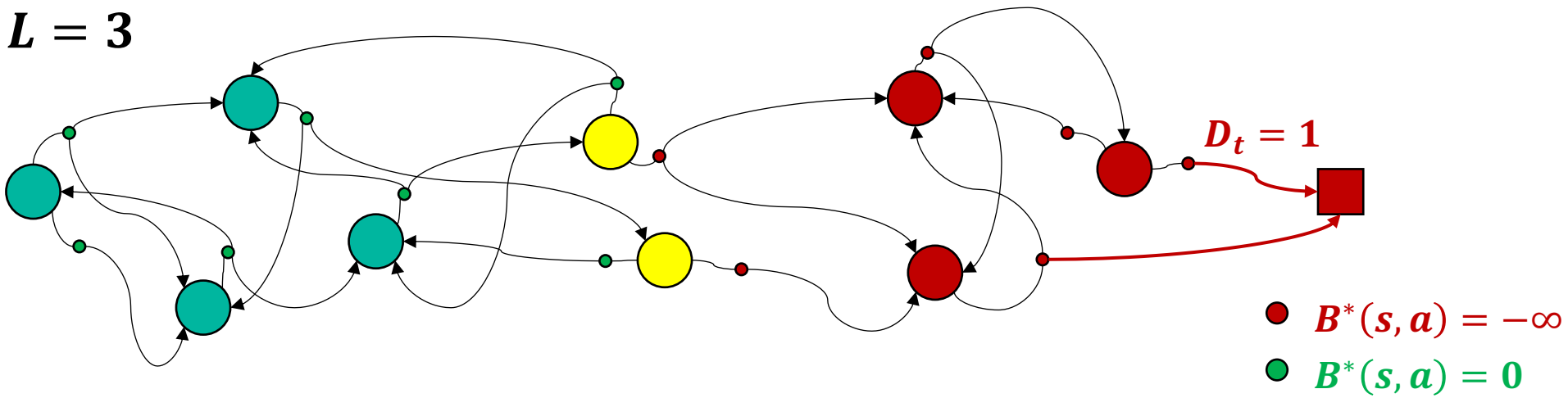
$$\mu = min\{p(s',d|s,a): p(s',d|s,a) \neq 0\}$$

- L: **Lag of the MDP**

$$L = \max_{\substack{(s,a) \\ B^*(s,a)=-\infty}} \left\{ \begin{array}{l} \underline{\text{Minimum}} \text{ number of transitions} \\ \text{needed to observe damage,} \\ \text{starting from unsafe } (s,a) \end{array} \right\}$$

Lag of the MDP: L

$$L = \max_{\substack{(s,a) \\ B^*(s,a) = -\infty}} \left\{ \begin{array}{l} \underline{\text{Minimum}} \text{ number of transitions needed to} \\ \text{observe damage, starting from unsafe } (s,a) \end{array} \right\}$$



Sample Complexity for Safety

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

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

iterations

- Concentration of sum of sub-exponential random variables [Janson'17]
- **Much more sample-efficient** than "learning an ϵ-optimal policy with $1-\delta$ probability" (Li et al. 2020)

$$N \geq \frac{|S||A|}{(1-\gamma)^3\varepsilon^2}\log\left(\frac{|S||A|}{(1-\gamma)\varepsilon\delta}\right)$$

Sample Complexity for Safety

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

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

iterations

- Concentration of sum of sub-exponential random variables [Janson'17]
- If the Barrier Function is learnt first, then learning an ϵ-optimal policy takes

$$N' \geq \frac{|S_{safe}||A_{safe}|}{(1-\gamma)^3\varepsilon^2}\log\left(\frac{|S_{safe}||A_{safe}|}{(1-\gamma)\varepsilon\delta}\right)$$

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

Experiments: barrier-learning with generative model

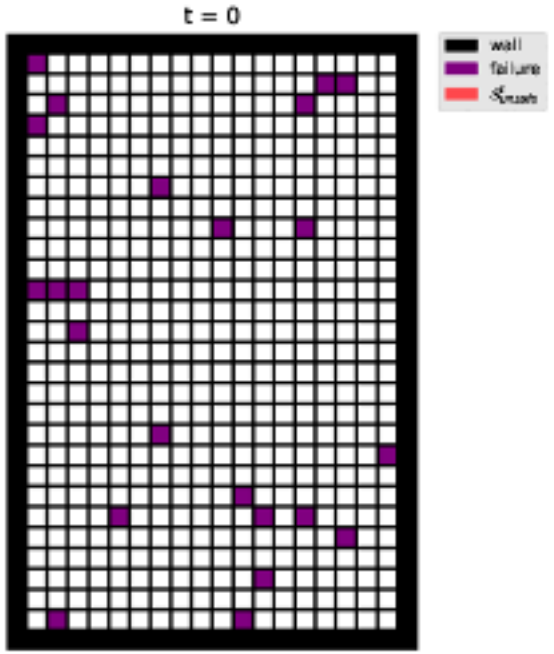
Setup: Rectangular grid, stepping into **holes** gives damage $D_t = 1$.

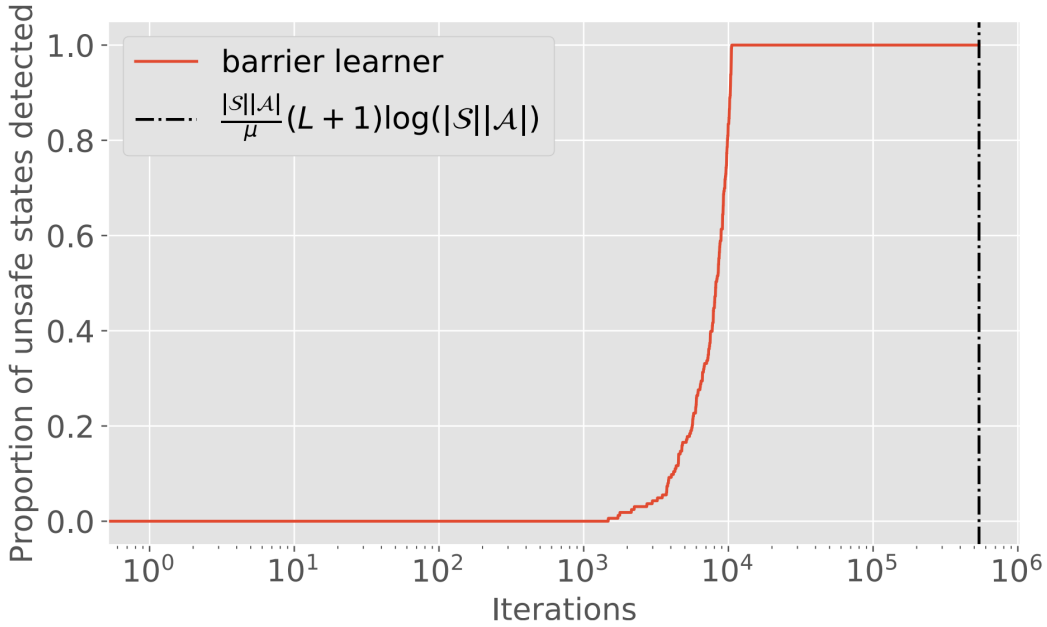
Actions $A = \{up, down, left, right\}$.

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.





Experiments: barrier-learning with generative model

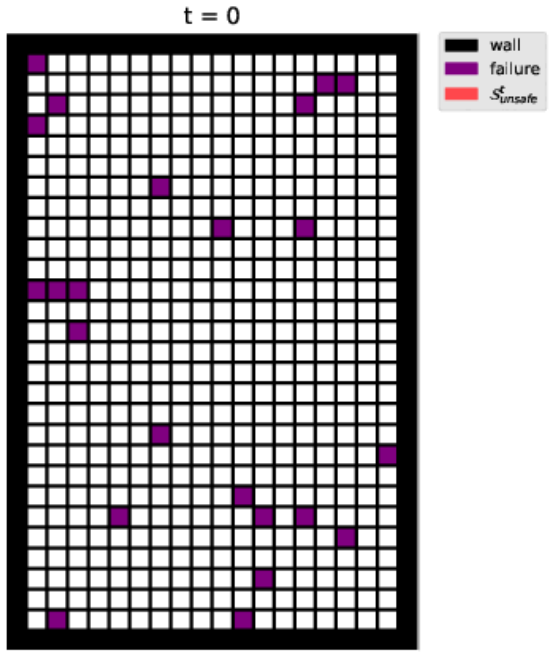
Setup: Rectangular grid, stepping into **holes** gives damage $D_t = 1$.

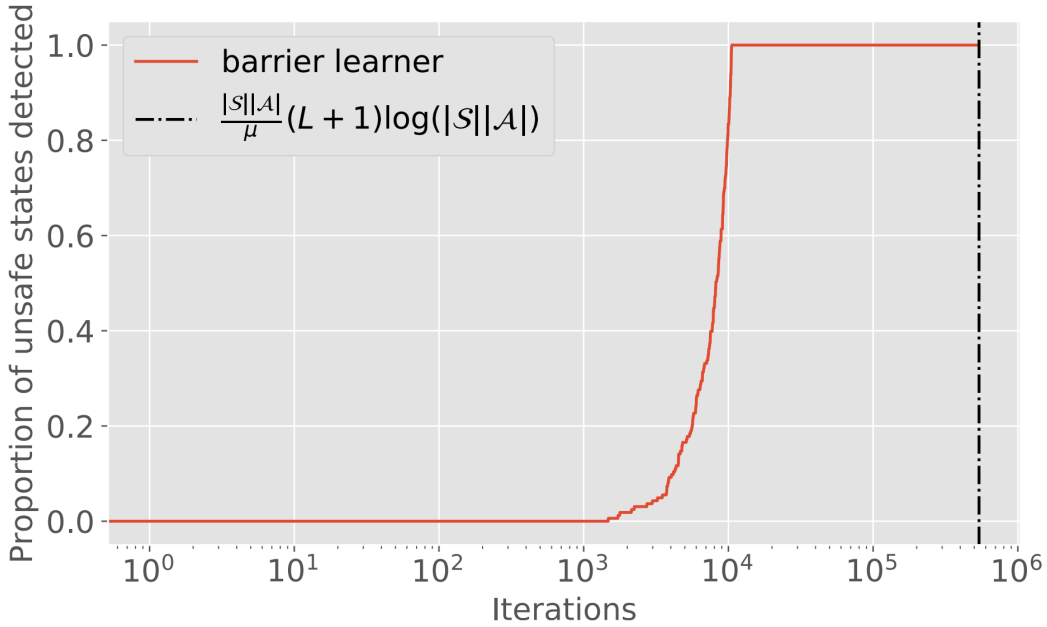
Actions $A = \{up, down, left, right\}$.

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

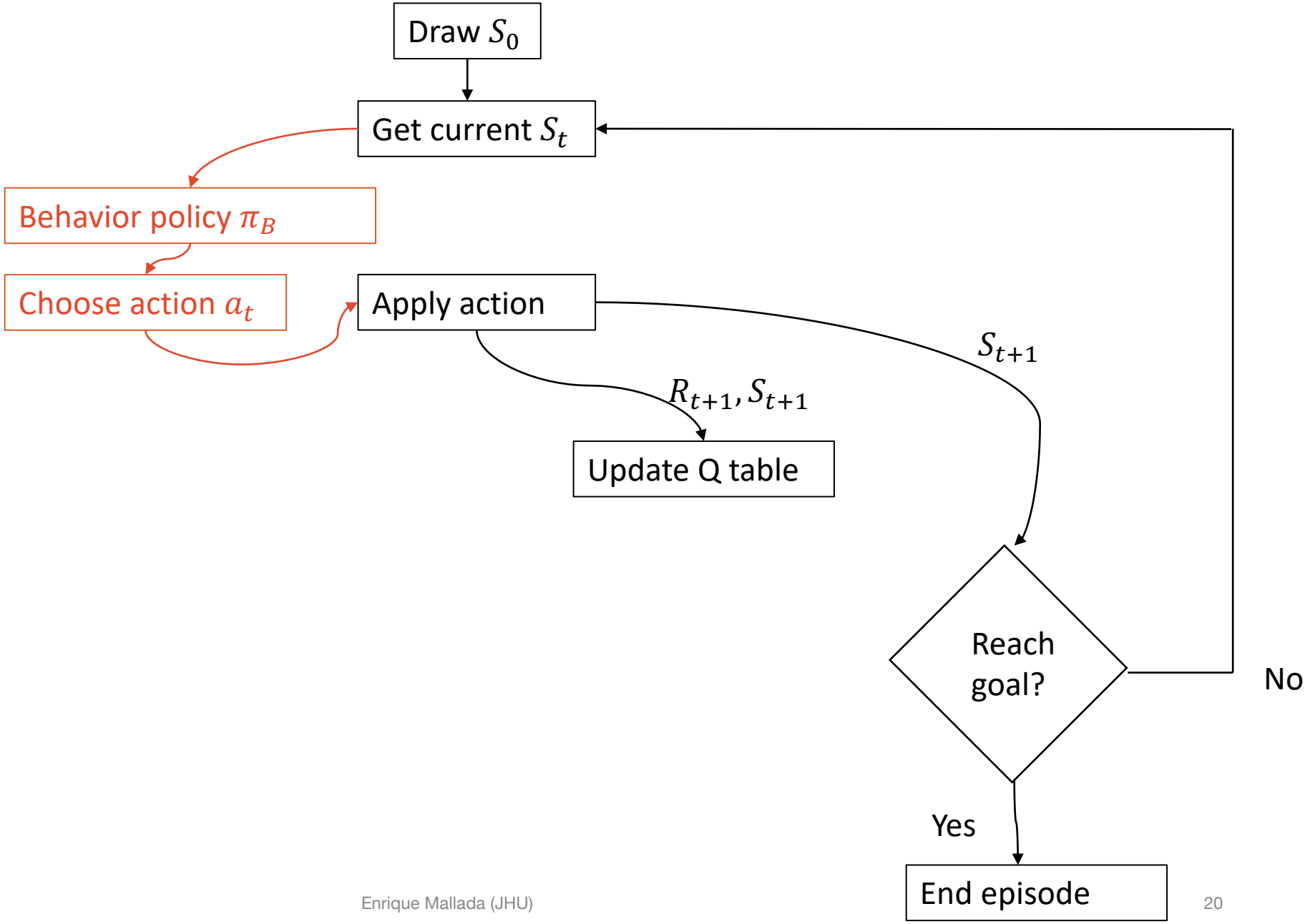
Result: Barrier-learner identifies **all** the state space as unsafe.

Immediately unsafe states (near **damage**) are identified first.

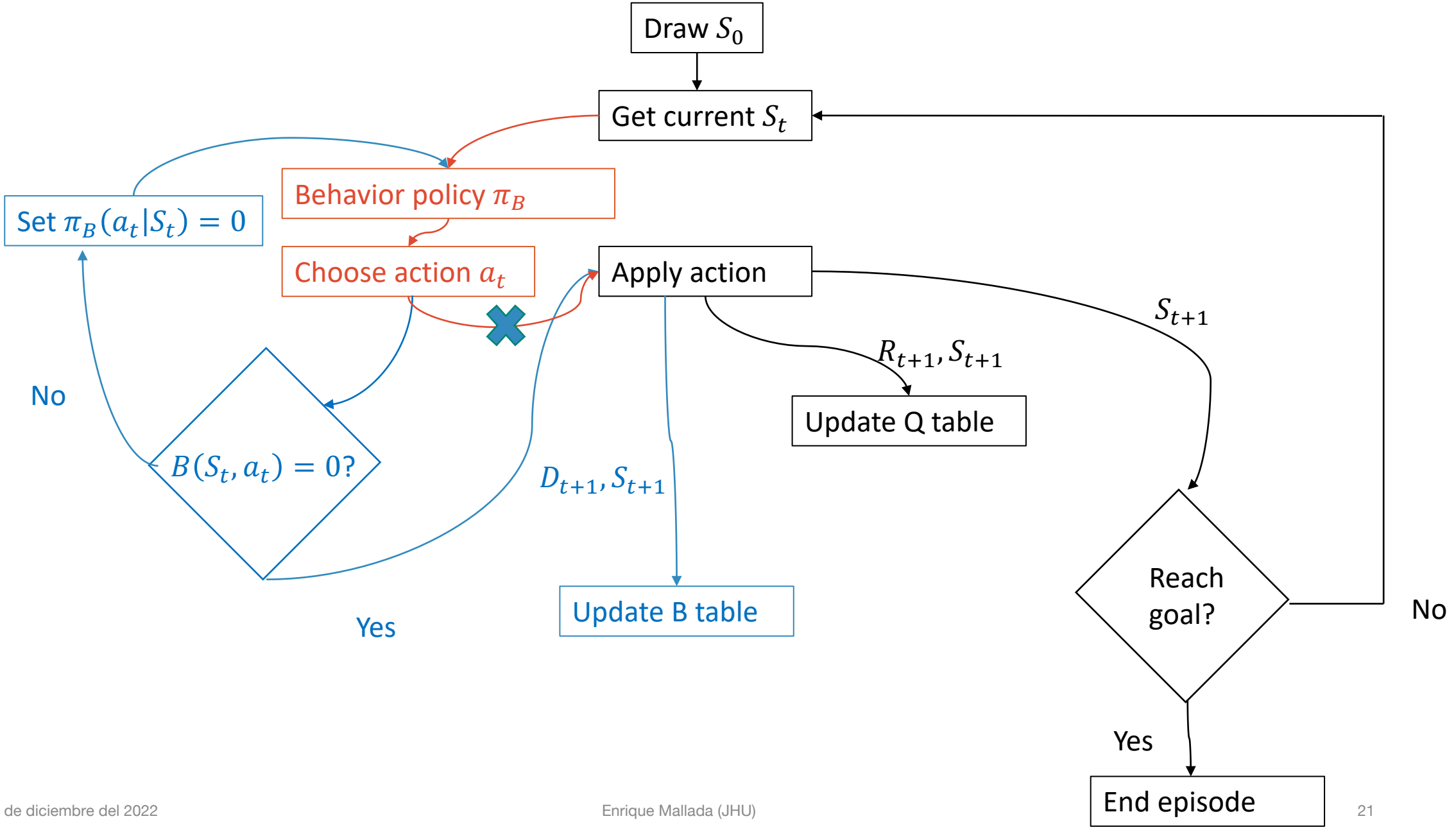




Example ”wrap-around”: Episodic Q-learning



Example ”wrap-around”: Episodic assured Q-learning

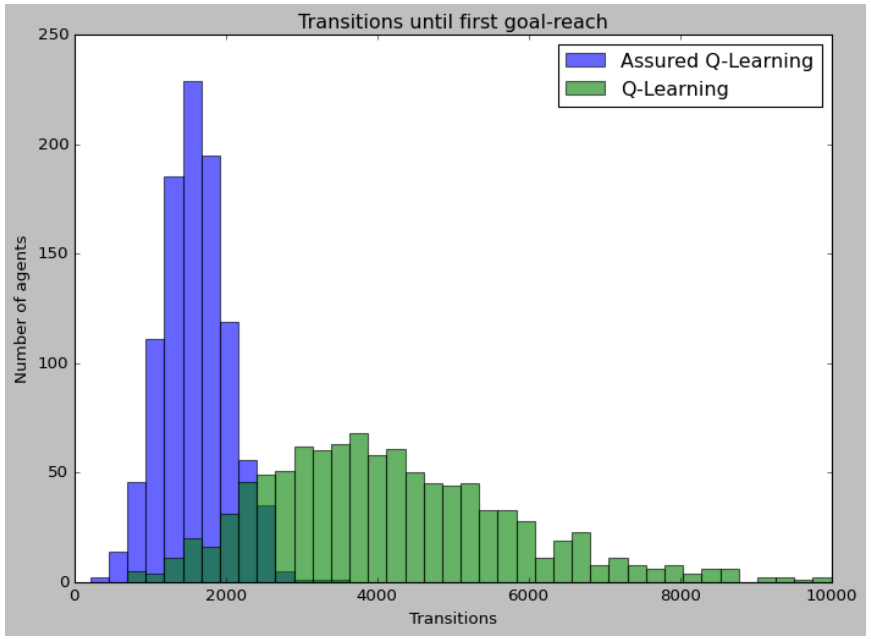


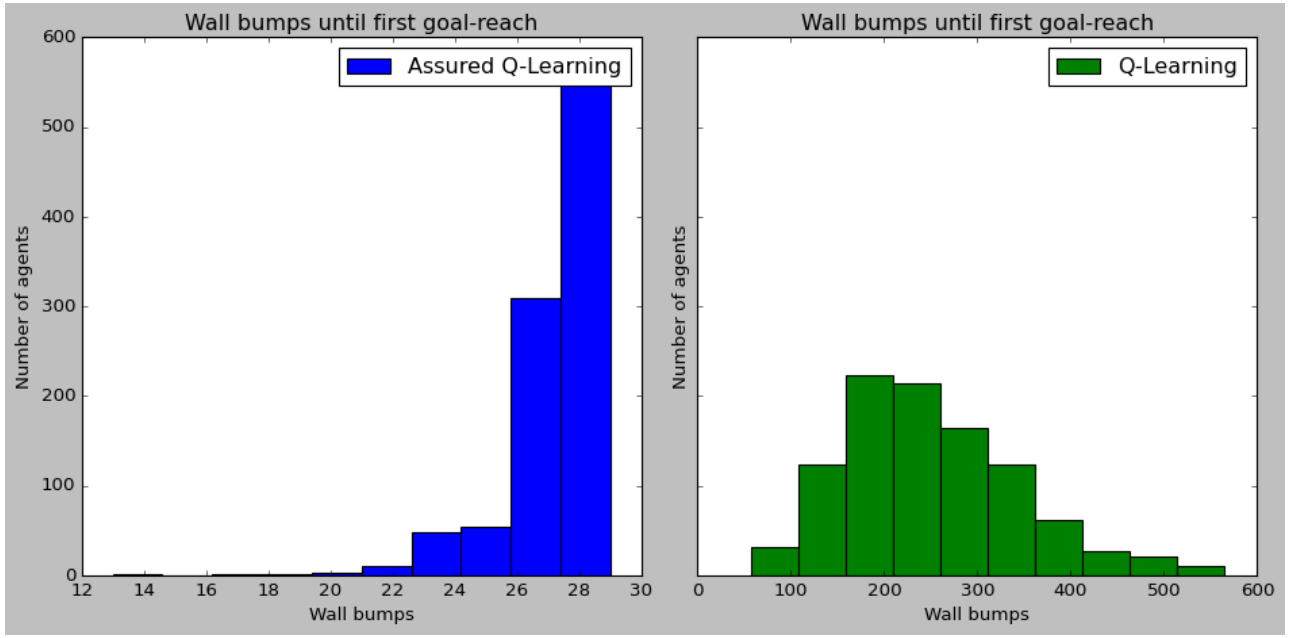
Experiments: assured Q-learning

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?

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

Takeaways:

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

Outline of the presentation

- Motivation & Background
- RL with almost sure constraints
  1. $\Delta = 0$: "Live or die" approach
  2. $\Delta \geq 0$: General case
- Summary and future work

Almost sure RL with positive budget (Δ)

- Almost Sure RL with positive budget

$$\max_{\pi \in \Pi_H} \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} R_{t+1} \mid S_0 = s \right]$$

$$\text{s.t: } P_\pi \left(\sum_{t=0}^{\infty} D_{t+1} \leq \Delta \mid S_0 = s \right) = 1$$

Π_H: history-dependent policies

$h_t = (S_0, A_0, R_1, D_1, \ldots, S_t); \qquad \pi(a|h_t)$

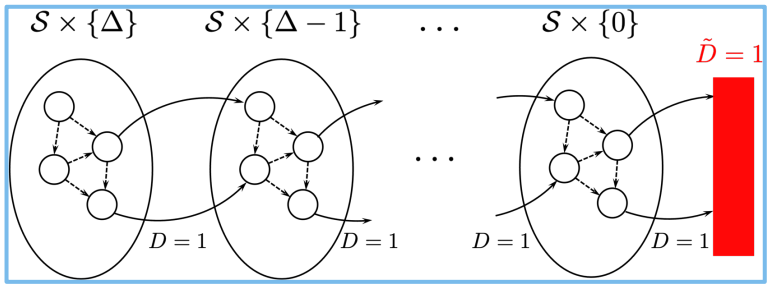
- Current budget at time t:

$$K_t = \Delta - \sum_{\ell=0}^{t-1} D_{\ell+1} \quad \forall t \geq 1$$

"How much more damage I can sustain and still be feasible"

- Augmented MDP $\tilde{\mathcal{M}}$

$$\tilde{S}_t = (S_t, K_t), \qquad \tilde{D}_{t+1} = \mathbf{1}\{K_t - D_{t+1} < 0\}.$$



- Equivalent problem:

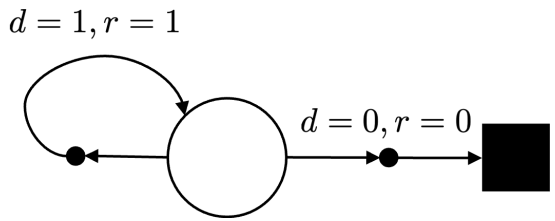
$$\max_{\tilde{\pi} \in \tilde{\Pi}_H} \mathbb{E}_{\tilde{\pi}, \tilde{\mathcal{M}}} \left[\sum_{t=0}^{\infty} R_{t+1} \,\middle|\, (S_0, K_0) = (s, \Delta) \right]$$

$$\text{s.t: } P_{\tilde{\pi}} \left(\tilde{D}_{t+1} = 0 \right) = 1 \qquad \forall t \geq 0$$

Fits previous formulation! →

- Could learn $B^*(s, k, a)$
- Separation & Feasibility Principles
- Potential drawback: working in higher dimensions? No!

Experiment: comparing constraints



Goal

$$\max_{\pi} \quad \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} R_{t+1}\right]$$

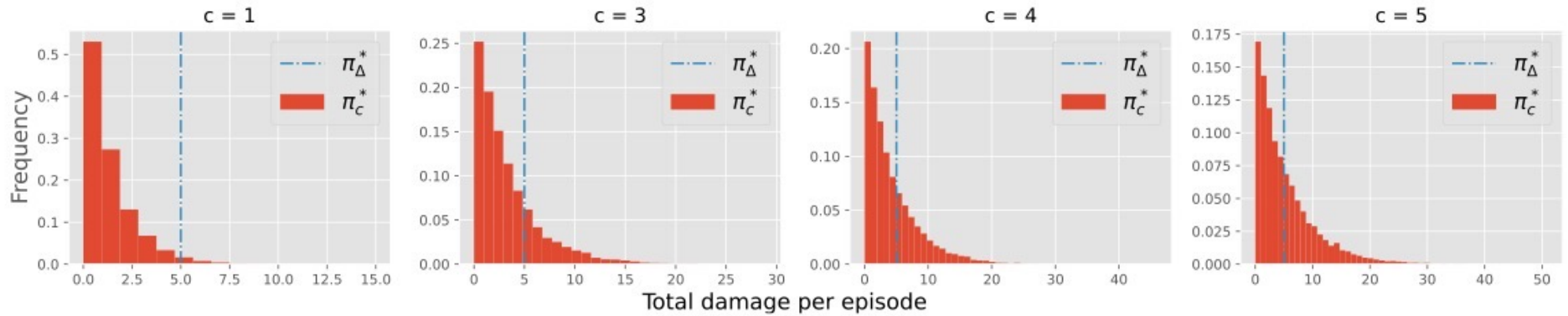
1) Proposed constraint

$$\mathbb{P}_{\pi_\Delta}\left(\sum_{t=0}^{\infty} D_{t+1} \leq \Delta \;\middle|\; S_0 = s\right) = 1$$

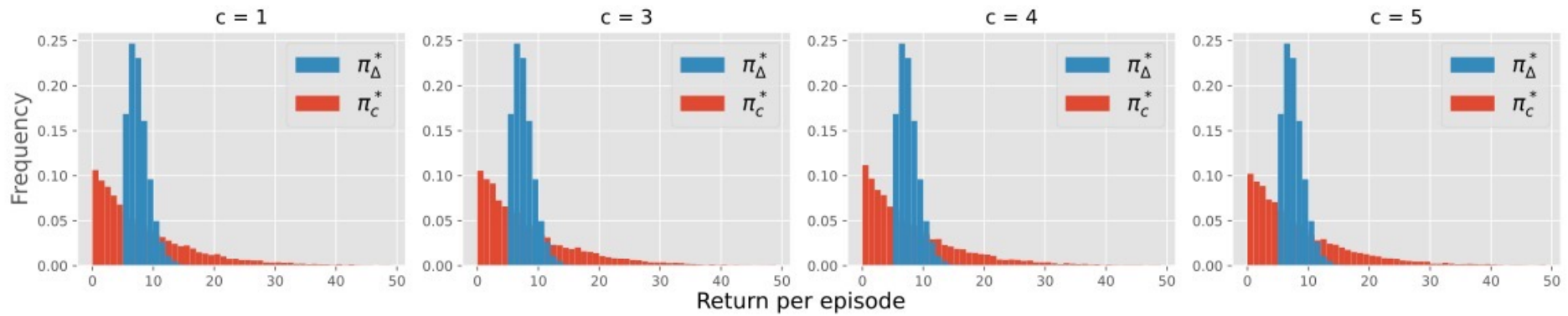
2) Classic CMDP constraint

$$\mathbb{E}_{\pi_c}\left[\sum_{t=0}^{\infty} D_{t+1}\right] \leq c$$

Safety of assured π^*_Δ with $\Delta = 5$ vs expectation-based constraint π^*_c; $P(d = 1) = 1$



Return of assured π^*_Δ with $\Delta = 5$ vs. expectation-based constraint π^*_c; $P(d = 1) = 0.6$



Summary and future work

Summary

- Reinforcement Learning for safety critical systems
- Treat constraints separately, or in parallel (Barrier Learner)
- Can **characterize** all feasible policies with **finite mistakes**
- **Take aways:**
  - **Learning feasible policies** is simpler **than learning** the optimal ones
  - Adding **constraints** makes **optimal policies easier to find**

Future work:

- Theory: Bounds for trajectory-based learning; Extensions to continuous spaces
- Application: Deep RL with almost sure constraints

Gracias!

Related Publications:

[L4DC 22] Castellano, Min, Bazerque, Mallada, *Reinforcement Learning with Almost Sure Constraints,* **Learning for Dynamics and Control (L4DC) Conference, 2022**

[arXiv 21] Castellano, Min, Bazerque, Mallada, *Learning to Act Safely with Limited Exposure and Almost Sure Certainty,* **submitted to IEEE TAC, 2021, under review**, preprint arXiv:2105.08748

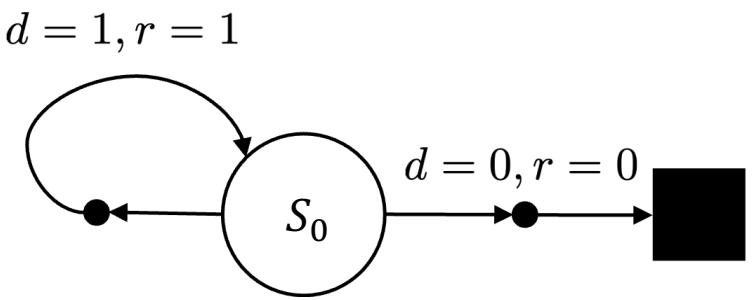
RL with almost sure constraints: $\Delta \geq 0$ case

$$\max_{\pi \in \Pi_H} \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} R_{t+1} \mid S_0 = s \right]$$

$$\text{s.t: } P_\pi \left(\sum_{t=0}^{\infty} D_{t+1} \leq \Delta \mid S_0 = s \right) = 1$$

Π_H : history-dependent policies $\quad h_t = (S_0, A_0, R_1, D_1, \ldots, S_t); \quad \pi(a|h_t)$

- Can we find (as in Part I) an optimal **stationary** policy?
- In general, **NO!**



Optimal policy: $V^{\pi_H^*} = \Delta$

The only feasible stationary policy has $V^{\pi_S} = 0$

What if we track the total damage encountered so far?

Current budget & the augmented MDP

- Current budget at time t:

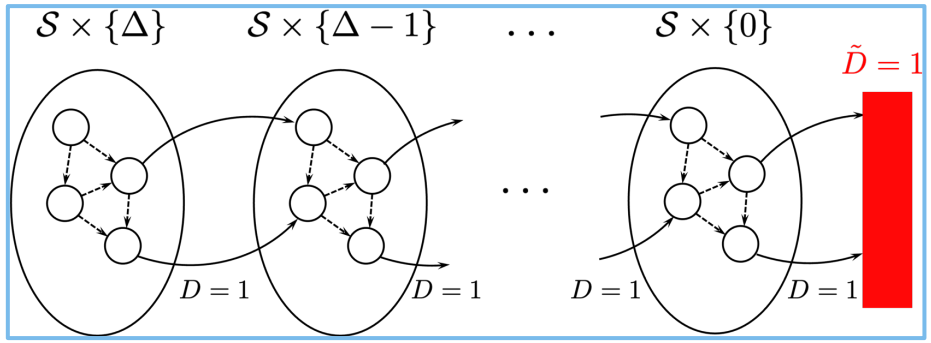
$$K_t = \Delta - \sum_{\ell=0}^{t-1} D_{\ell+1} \quad \forall t \geq 1$$

"How much more damage I can sustain and still be feasible"

Claim: ∃ optimal policy $\pi^*(a \mid (s,k))$

- Augmented MDP $\tilde{\mathcal{M}}$

$$\tilde{S}_t = (S_t, K_t)\,, \qquad \tilde{D}_{t+1} = \mathbf{1}\{K_t - D_{t+1} < 0\}\,.$$



- Equivalent problem:

$$\max_{\tilde{\pi} \in \tilde{\Pi}_H} \mathbb{E}_{\tilde{\pi}, \tilde{\mathcal{M}}} \left[\sum_{t=0}^{\infty} R_{t+1} \;\middle|\; (S_0, K_0) = (s, \Delta) \right]$$
$$\text{s.t.: } \mathbb{P}_{\tilde{\pi}} \left(\tilde{D}_{t+1} = 0 \mid (S_0, K_0) = (s, \Delta) \right) = 1 \quad \forall t \geq 0$$

Fits previous formulation! →

- Could learn $B^*(s,k,a)$
- Separation & Feasibility Principles
- Potential drawback: working in higher dimensions

Minimal required budget (k_*)

$$\mathbb{I}\{x\} = \log(1-x) = \begin{cases} 0 & \text{if } x \text{ holds} \\ -\infty & \text{otherwise} \end{cases}$$

$$Q_{\tilde{\pi}}(s,k,a) := \mathbb{E}_{\tilde{\pi}}\left[\sum_{\ell=t}^{\infty} R_{\ell+1} + \mathbb{I}\left\{\sum_{\ell=t}^{\infty} D_{\ell+1} \leq K_t\right\} \mid S_t = s, K_t = k, A_t = a\right]$$

Return

Constraint satisfaction

$$B_{\tilde{\pi}}(s,k,a) := \mathbb{E}_{\tilde{\pi}}\left[\mathbb{I}\left\{\sum_{l=t}^{\infty} D_{l+1} \leq K_t\right\} \,\middle|\, S_t = s, K_t = k, A_t = a\right].$$

- From each (s,a), what is the smallest budget needed that guarantees safety?

$$k_*(s,a) = \min_{0 \leq k \leq \infty} k \quad \text{s.t.: } B_*(s,k,a) = 0$$

- k_* is intrinsic to the MDP
- **k_* is a descriptor of all feasible policies:**

$$\Pi_{Safe} = \{\pi : \pi(a|(s,k)) = 0 \text{ whenever } k < k_*(s,a)\}$$

- Idea: learn $k_* \ \forall(s,a)$ instead of $B_* \ \forall(s,k,a)$

Easier!!

How can we learn k_*?

$$\mathbf{1}_d^p(s,a,s') := \mathbf{1}\{p(d=1 \mid s,a,s') > 0\}$$

Theorem 10 (Fixed point for k_*) *For each (s,a), the minimal budget satisfies the recursion:*

$$k_*(s,a) = \max_{s':p(s'|s,a)>0} \left[\mathbf{1}_d^p(s,a,s') + \min_{a'} k_*(s',a') \right],$$

Budget I need now — =YES! As long as $p(s'|s,a) > 0 \iff \hat{p}(s'|s,a) > 0$ — min budget I need later

=1 if transition w/ damage

And $\mathbf{1}_d^p(s,a,s') = 1 \iff \mathbf{1}_d^{\hat{p}}(s,a,s')$

k_* can be computed with:

Before learning k_*, sample and learn **consistent kernel $\hat{p}$**

Algorithm 1 Fixed point budget iteration

Input: Transition kernel p from $\mathcal{M}$
Result: k_* for $\mathcal{M}$

$k_0(s,a) \leftarrow 0 \quad \forall (s,a) \in \mathcal{S} \times \mathcal{A}$

for $n = 0, 1, \ldots$ **do**
  for $(s,a) \in \mathcal{S} \times \mathcal{A}$ **do**
    $k_{n+1}(s,a) \leftarrow \max_{s':p(s'|s,a)>0} [\mathbf{1}\{p(d=1 \mid s,a,s') > 0\} + \min_{a'} k_n(s',a')]$
  end
end

What if we don't know $\mathcal{M}$? Can we use approximate kernel $\hat{p} \sim p$ instead?

Samples needed to get a consistent kernel

Algorithm 2 Kernel builder

Input: Transition kernel p from $\mathcal{M}$, number of sample queries N'

Result: Empirical kernel $\hat{p}$.

for $(s, a) \in \mathcal{S} \times \mathcal{A}$ **do**

| Sample N' transitions $(s', d) \sim p(\cdot|s, a)$

end

Build estimate kernel $\hat{p}(s', d|s, a) = \frac{\text{count}(s', d; s, a)}{N'} \quad \forall s' \in \mathcal{S}, d \in \{0, 1\}, s \in \mathcal{S}, a \in \mathcal{A}$

Lemma 13 (Sample complexity for Algorithm 2) *Assume that* $p(s', d|s, a) = 0$ *or* $p(s', d|s, a) \geq \mu > 0$, *for every* $(s, a, s', d) \in \mathcal{S} \times \mathcal{A} \times \mathcal{S} \times \{0, 1\}$. *Then with probability at least* $1 - \delta$, `Kernel builder` *produces a consistent kernel* $\hat{p}$ *of* p, *provided that*

$$\mathrm{N} = \mathrm{N}'.|\mathrm{S}||\mathrm{A}| \geq \frac{|S||A|}{\mu} \log\left(\frac{2|S|^2|A|}{\delta}\right) \tag{17}$$

Compare with “barrier-learner” (Part I): $\mathrm{N} \geq (L+1)\frac{|S||A|}{\mu} \log\left(\frac{|S||A|}{\delta}\right)$