Learning Optimal Strategies for Temporal Tasks in Stochastic Games

Miroslav Pajic

Department of Electrical and Computer Engineering

Duke University

April 30, 2021

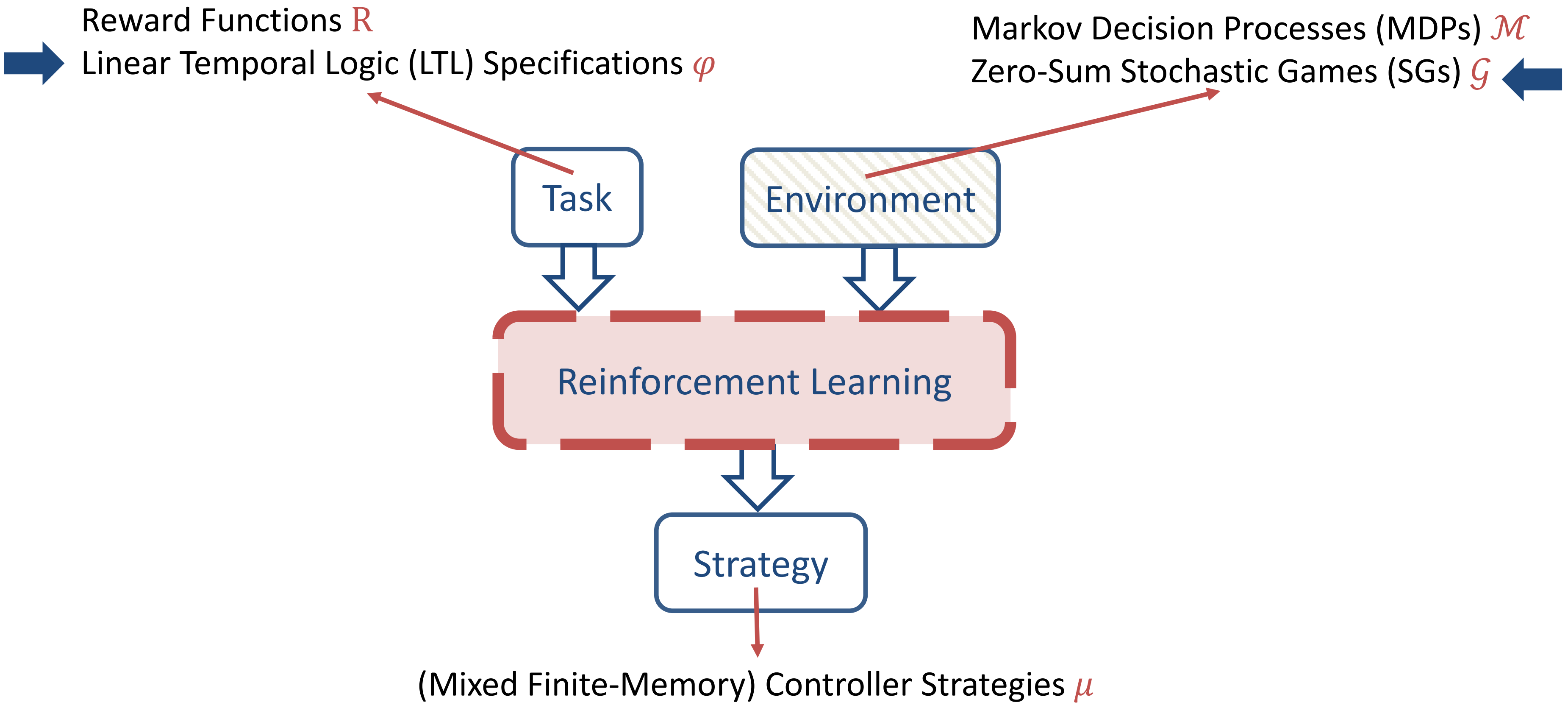
Duke PRATT SCHOOL *of* ENGINEERING

CPSL Duke

DUKE ROBOTICS

Problem Formulation

- **Framework**



- **Problem**

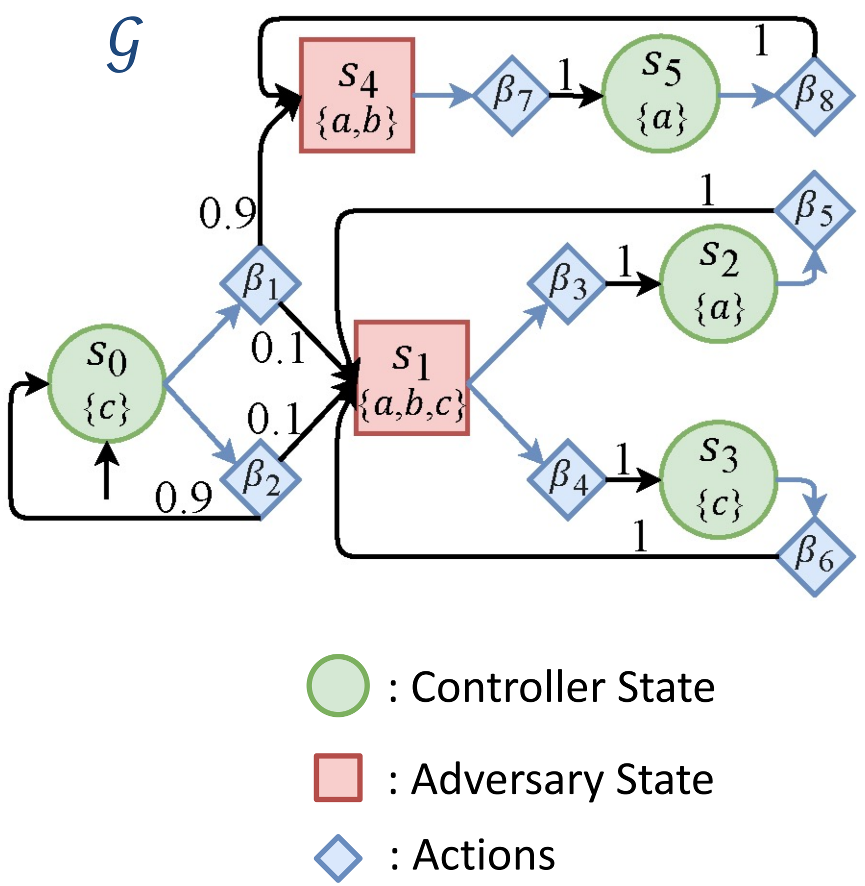
  Given an unknown $\mathcal{G}$ and a specification φ,
  learn a strategy in $argmax_{\mu}\, min_{\nu}\, Pr_{\mu,\nu}(\mathcal{G} \vDash \varphi)$
  where μ and ν are controller and adversary strategies.

Stochastic Games and Linear Temporal Logic

- **Labeled Turn-Based Zero-Sum Stochastic Games**

$$\mathcal{G} = \left(S, (S_\mu, S_\nu), s_0, A, P, AP, L\right)$$

  - $S = S_\mu \cup S_\nu$ is a finite set of states; s_0 is an initial state
  - S_μ, S_ν are the controller and the adversary states
  - A is a finite set of actions
  - P is the transition probability function (unknown)
  - AP is a set of labels/atomic propositions
  - $L: S \to 2^{AP}$ is a labeling function



- **LTL Grammar**

$$\varphi \coloneqq \text{true} \mid a \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 \,\mathrm{U}\, \varphi_2, \quad a \in AP$$

  - $\varphi_1 \vee \varphi_2 \coloneqq \neg(\neg\varphi_1 \wedge \neg\varphi_2)$;
  - $\varphi_1 \rightarrow \varphi_2 \coloneqq \neg\varphi_1 \vee \varphi_2$
  - $\Diamond\varphi \coloneqq \text{true}\ \mathrm{U}\ \varphi$
  - $\Box\varphi \coloneqq \neg(\Diamond\neg\varphi)$

RL Framework for LTL

- **Key Idea: Reduction**
  - From the LTL objective

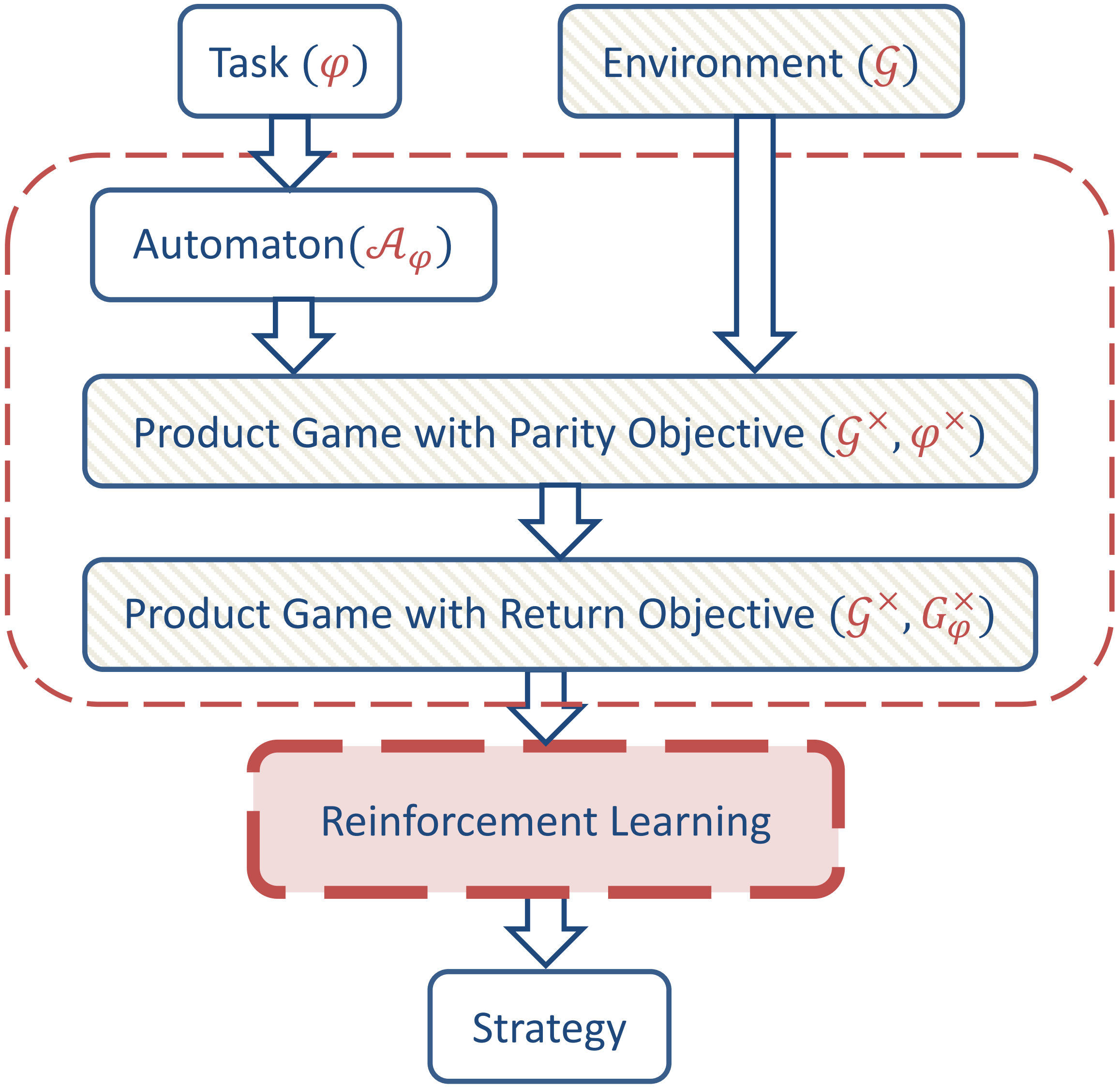
$$argmax_{\mu}\ min_{\nu}\ Pr_{\mu,\nu}(\mathcal{G} \vDash \varphi)$$

  - To a return objective

$$argmax_{\mu}\ min_{\nu}\ \mathbb{E}_{\mu,\nu}[G_{\varphi}^{\times}]$$

$$argmax_{\mu}\ min_{\nu}\ \mathbb{E}_{\mu,\nu}\left[\sum_{i=0}^{\infty} \gamma^{i} r_{(i)}\right]$$

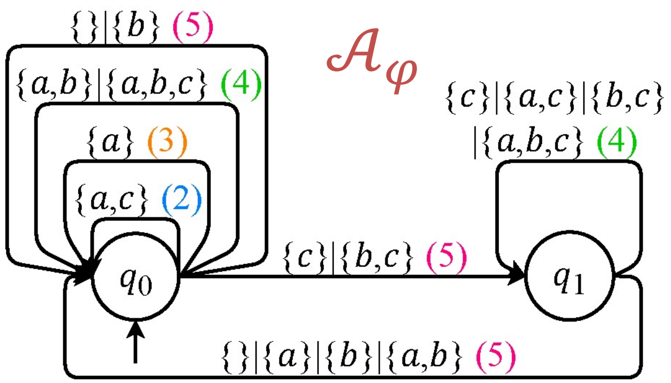
- **Reduction Steps:**
  - LTL -> Automaton
  - Product Game Construction
  - Reduction from Parity to Return
  - Model-free Learning



Product Game Construction

- **LTL to Deterministic Parity Automata (DPA) Translation**
  - The set of traces satisfying φ is an ω-regular language
  - A DPA $\mathcal{A}_\varphi$ recognizing the language can be automatically constructed
  - Example: $\varphi = (\Diamond\Box a \wedge \Box\Diamond b) \vee \Diamond\Box c$

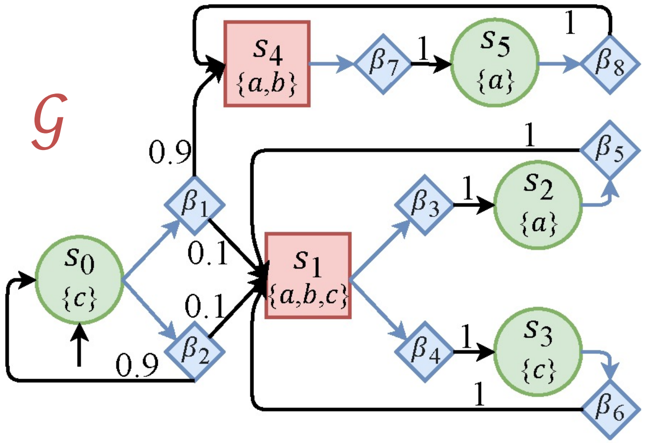
$\mathcal{A}_\varphi$



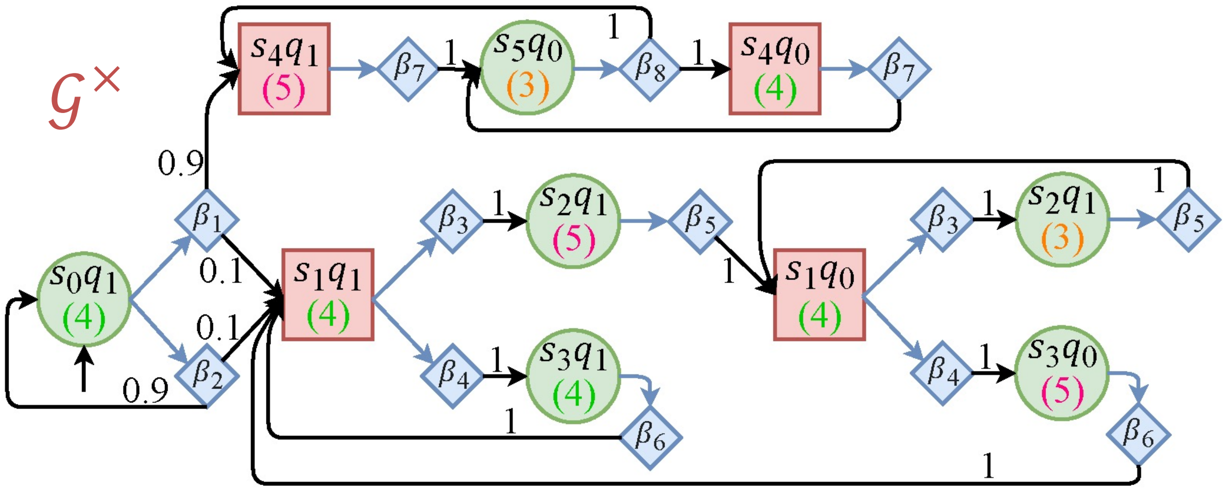
- **Product Game**
  - Simultaneous execution of the SG $\mathcal{G}$ and the DPA $\mathcal{A}_\varphi$
  - Does not have to be constructed explicitly
  - Winning Condition: Parity Objective

$$\varphi^\times := max\{Color(s^\times) \mid s^\times \in Inf(\pi^\times)\}$$

$\mathcal{G}$

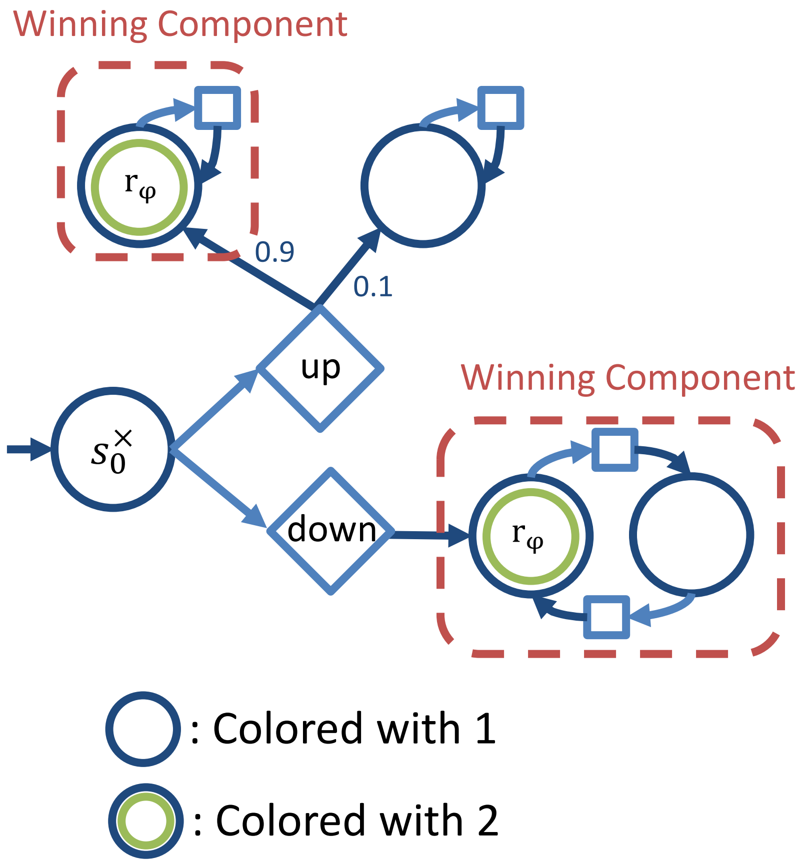


$\mathcal{G}^\times$



Reduction I: Distinct Discount Factors - Büchi Conditions

- **Büchi Conditions**
  - Two colors: Color 1 and Color 2
  - Suffices for MDPs (with some additional structured nondeterminism)
  - $\varphi^{\times}$: Repeatedly visit states colored with 2
  - Example: $Pr\big((s_0^{\times}, \mathrm{up}) \models \varphi^{\times}\big) = 0.9$ and $Pr\big((s_0^{\times}, \mathrm{down}) \models \varphi^{\times}\big) = 1$



- **Reduction to Return Objectives**
  - Reward Function: $R_{\varphi}(s^{\times}) \coloneqq \begin{cases} r_{\varphi} & \text{if } Color(s^{\times}) = 2 \\ 0 & \text{if } Color(s^{\times}) = 1 \end{cases}$
  - Discount Function: $\Gamma_{\varphi}(s^{\times}) \coloneqq \begin{cases} 1 - r_{\varphi} & \text{if } Color(s^{\times}) = 2 \\ 1 - r_{\varphi}^2 & \text{if } Color(s^{\times}) = 1 \end{cases}$
  - Example: $q(s^{\times}, \mathrm{up}) = 0.9$ and $q(s^{\times}, \mathrm{down}) = \frac{1}{1 + r_{\varphi}(1 - r_{\varphi})}$

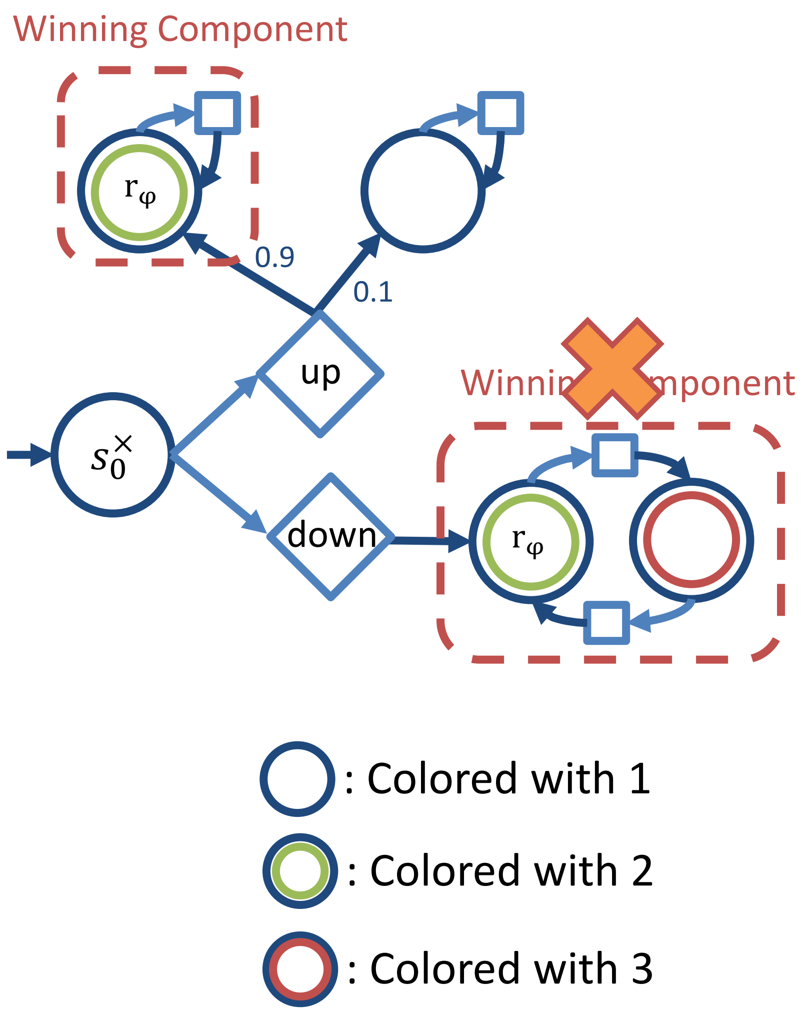
- **Theorem I [1,2]:**
  - For a Büchi condition $\varphi^{\times}$ and any strategy pair (μ,ν),

$$Pr_{\mu,\nu}(\mathcal{G}^{\times} \models \varphi^{\times}) = \lim_{r_{\varphi} \to 0^+} \mathbb{E}_{\mu,\nu}\left[\sum_{i=0}^{\infty}\left(\prod_{j=1}^{i} \Gamma_{\varphi}(s_{(j)}^{\times})\right) R_{\varphi}(s_{(i)}^{\times})\right]$$

[1] A. K. Bozkurt, Y. Wang, M. M. Zavlanos, and M. Pajic. "Control Synthesis from Linear Temporal Logic Specifications using Model-Free Reinforcement Learning". ICRA, 2020.
[2] A. K. Bozkurt, Y. Wang, M. M. Zavlanos, and M. Pajic. "Model-Free Reinforcement Learning for Stochastic Games with Linear Temporal Logic Objectives". ICRA, 2021, accepted.

Reduction I: Distinct Discount Factors - Generalization

- **Parity Conditions with $k > 2$**
  - Example: $Pr\big((s_0^\times, \mathrm{up}) \vDash \varphi^\times\big) = 0.9$ and $Pr\big((s_0^\times, \mathrm{down}) \vDash \varphi^\times\big) = 0$
- **A distinct power for each color**
  - Reward Function: $R_\varphi(s^\times) := \begin{cases} r_\varphi^{k-Color(s^\times)} & \text{if } Color(s^\times) \text{ is } even \\ 0 & \text{if } Color(s^\times) \text{ is } odd \end{cases}$
  - Discount Function: $\Gamma_\varphi(s^\times) := 1 - r_\varphi^{k-Color(s^\times)}$
  - Example: $q(s^\times, \mathrm{up}) = 0.9$ and $q(s^\times, \mathrm{down}) = \frac{1}{1+\frac{1-r_\varphi^2}{r_\varphi}}$
  - Distinct powers of rewards and discount factors captures the order
  - Not Scalable
- **An approximation is provided in [2]**



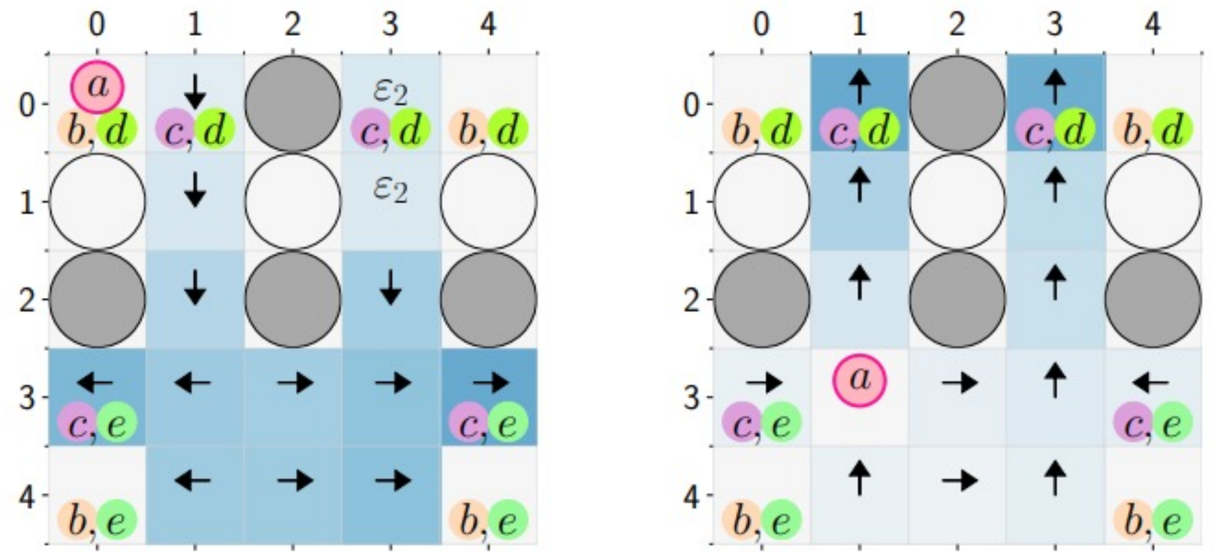
[2] A. K. Bozkurt, Y. Wang, M. M. Zavlanos, and M. Pajic. "Model-Free Reinforcement Learning for Stochastic Games with Linear Temporal Logic Objectives". ICRA, 2021, accepted

Case Study: Avoiding an Adversary

- **Grid World**
  - The agent and the adversary can take four actions: *North*, *South*, *East*, *West*
  - The probability of moving in the intended direction: 0.8
  - The probability of moving in a direction orthogonal to the intended direction: 0.2

- **Objective**
  - Repeatedly visit a b and a c cell
  - Reach a safe region labeled with d or e and do not leave
  - Avoid the adversary (a) at all costs.

$$\varphi = \Box\Diamond b \wedge \Box\Diamond c \wedge (\Diamond\Box d \vee \Diamond\Box e) \wedge \Box\neg a$$



(a) Adversary is at $(0, 0)$ and i=1

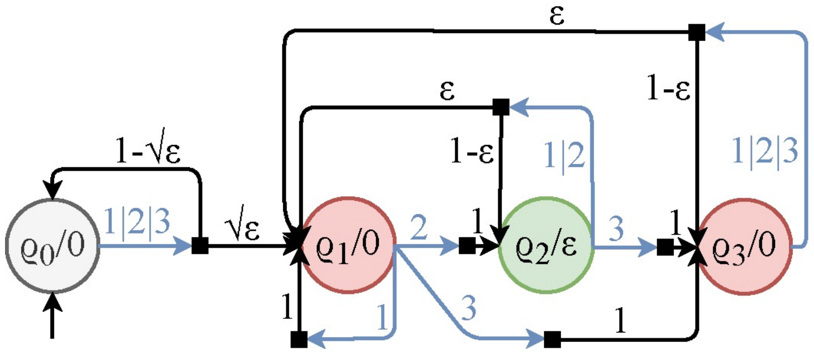
(b) Adversary is at $(3, 1)$ and i=2

The darker blue, the higher estimated satisfaction probability

Reduction II: Priority Reward Machines

- **Objective:**
  - Design a reduction where the discount factors, rewards, transition probabilities do not depend on the number of colors

- **Priority Reward Machines (PRMs)**
  - The Moore machines consisting of priority modes ϱ_i
  - Output: $R^{\star}_{\varphi}(s^{\times}, \varrho) := \begin{cases} \varepsilon_{\varphi} & \text{if } \varrho > 0 \text{ and } \varrho \text{ is } \mathit{even} \\ 0, & \mathit{otherwise} \end{cases}$
  - A priority mode ϱ_i is overruled by ϱ_j when Color j is consumed
  - PRMs reset to ϱ_1 w.p. ε_{φ}
  - PRMs move from ϱ_0 to ϱ_1 w.p. $\sqrt{\varepsilon_{\varphi}}$



- **Theorem II [3]:**
  - For a parity condition $\varphi^{\times}$ and any strategy pair (μ,ν),

$$Pr_{\mu,\nu}(\mathcal{G}^{\times} \vDash \varphi^{\times}) = \lim_{\varepsilon_{\varphi}\to 0^{+}} \mathbb{E}_{\mu,\nu}\left[\sum_{i=0}^{\infty} \left(1-\varepsilon_{\varphi}\right)^{i} R^{\star}_{\varphi}(s^{\times}_{(i)}, \varrho_{(\mathrm{i})})\right]$$

[3] A. K. Bozkurt, Y. Wang, and M. Pajic. “Learning Optimal Strategies for Temporal Tasks in Stochastic Games”. 2021, submitted.

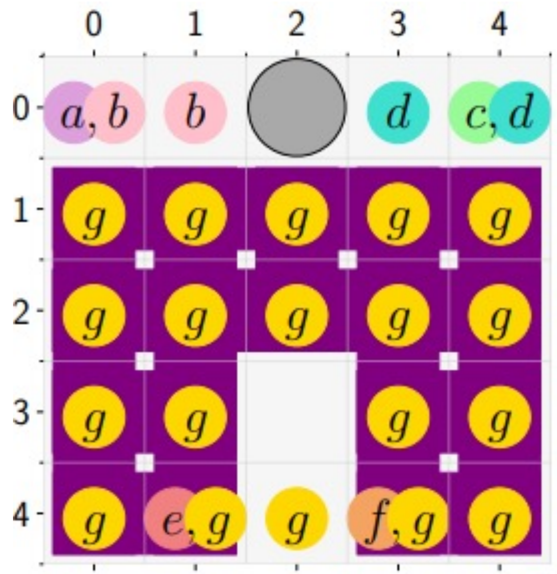
Case Study: Surveillance

Duke
PRATT SCHOOL of ENGINEERING

- **Grid World**
  - The agent can take four actions: *North*, *South*, *East*, *West*
  - The adversary can disrupt the movement so that the agent might move in a perpendicular direction

- **Objective**
  - Eventually perform one of the following surveillance tasks:
    - Repeatedly visit a without leaving the region b
    - Repeatedly visit c without leaving the region d
    - Repeatedly visit e and f without leaving the region g

$$\varphi = (\Box\Diamond a \wedge \Diamond\Box b) \vee (\Box\Diamond c \wedge \Diamond\Box d) \vee (\Box\Diamond e \wedge \Box\Diamond f \wedge \Diamond\Box g)$$

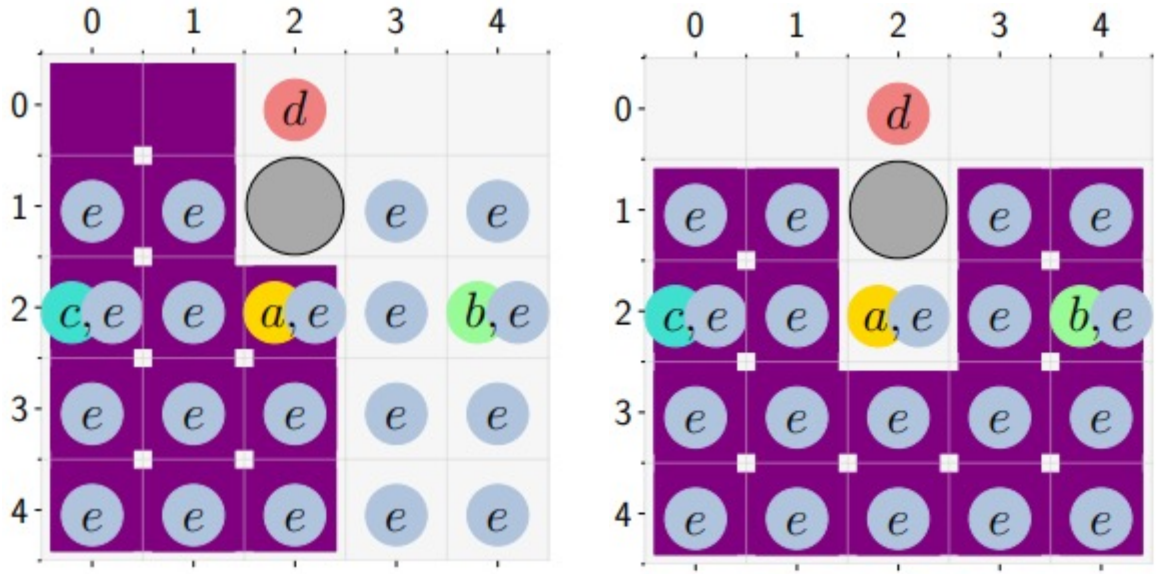


The cells visited under the optimal strategies are highlighted in purple.

Case Study: Nursery Scenario

- **Objective**
  - Start at $(0,0)$
  - Enter the region labeled with e and stay there
  - Inform the adult a exactly once
  - Repeatedly visit the baby b and the charger station c
  - Avoid the danger zone d

  $\varphi = \Diamond\Box e \wedge \Diamond a \wedge \Box(a \rightarrow \bigcirc\Box\neg a) \wedge \Box\Diamond b \wedge \Box\neg d$



Before a After a

The cells visited under the optimal strategies are highlighted in purple.

Quality of Control Optimization under LTL Specifications

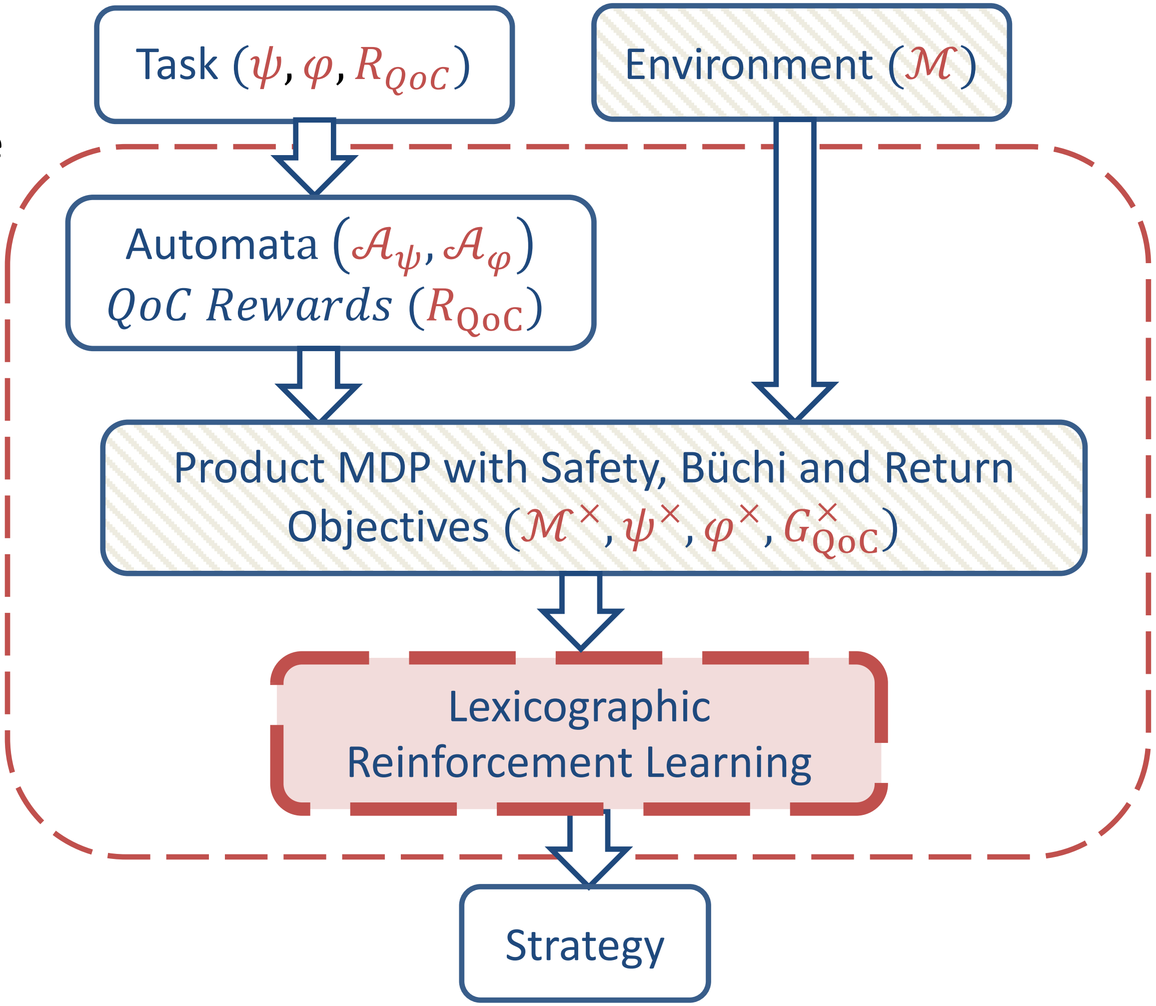
- **Multiple Objectives with Lexicographic Order:**
  - Priority 1: Safety ψ
    - Safety LTL formula
    - Ensuring the safety is usually of utmost importance
  - Priority 2: LTL φ
    - Important system specifications as an LTL formula other than safety
  - Priority 3: QoC R_{QoC}
    - External Rewards
    - LTL cannot specify objectives including cost, yield or quality optimization
    - Example: Minimization of Energy Consumption

- **Problem:**
  - For a given MDP $\mathcal{M}$ learn a strategy $\mu \in \Sigma_{\psi,\varphi}^{\mathrm{QoC}}$ where

$$\Sigma_{\psi} \coloneqq argmax_{\mu}\, Pr_{\mu}(\mathcal{M} \vDash \psi)$$

$$\Sigma_{\psi,\varphi} \coloneqq argmax_{\mu \in \Sigma_{\psi}}\, Pr_{\mu}(\mathcal{M} \vDash \varphi)$$

$$\Sigma_{\psi,\varphi}^{\mathrm{QoC}} \coloneqq argmax_{\mu \in \Sigma_{\psi,\varphi}}\, \mathbb{E}_{\mu}\left[\sum_{i=0}^{\infty} \gamma^{i} r_{(i)}^{\mathrm{QoC}}\right]$$



QoC Optimization via Lexicographic RL

- **Algorithm I: QoC Optimization Under Safety and LTL Specifications**
  - **Learning Actions Sets**
    - $V_\psi(s^\times) \leftarrow max_{a^\times} Q_\psi(s^\times, a^\times)$
    - $\hat{A}_\psi^\times \leftarrow \{ a^\times \mid V_\psi(s^\times) - Q_\psi(s^\times, a^\times) \leq \tau_\psi \}$
    - $V_\psi(s^\times) \leftarrow max_{a^\times \in \hat{A}_\psi^\times} Q_{\psi,\varphi}(s^\times, a^\times)$
    - $\hat{A}_{\psi,\varphi}^\times(s^\times) \leftarrow \{ a^\times \in A_\psi^\times(s^\times) \mid V_{\psi,\varphi}(s^\times) - Q_{\psi,\varphi}(s^\times, a^\times) \leq \tau_\varphi \}$
    - $\hat{A}_{\psi,\varphi}^{\times *} \leftarrow \underset{a^\times \in \hat{A}_{\psi,\varphi}^\times(s^\times)}{argmax} Q_{\psi,\varphi}^{QoC}(s^\times, a^\times)$
  - **Action Selection**
    - Choose a random action w.p. e (during exploration)
    - Choose a random action from $\hat{A}_{\psi,\varphi}^\times(s^\times)$ w.p. υ (for LTL)
    - Choose an action in $\hat{A}_{\psi,\varphi}^{\times *}$
  - **Q-Value Updates**
    - Use Q-learning to update Q_ψ and $Q_{\psi,\varphi}$
    - Use SARSA to update $Q_{\psi,\varphi}^{QoC}$

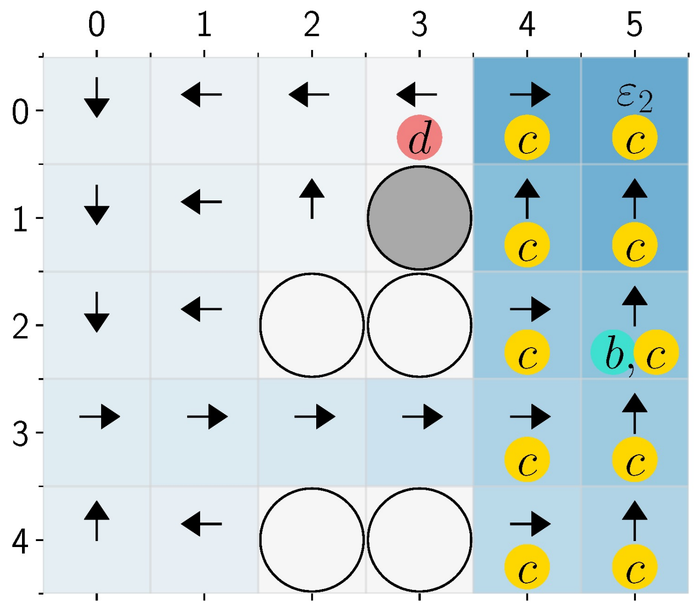
- **Theorem III [4]:**
  - Algorithm I learns a lexicographically near-optimal strategy $\mu \in \Sigma_{\psi,\varphi}^{\text{QoC}-\nu}$ for sufficiently small $\tau_\psi, \tau_\varphi > 0$, for the rewards provided in Reduction I.

[4] A. K. Bozkurt, Y. Wang, and M. Pajic. "Model-Free Learning of Safe yet Effective Controllers". 2021, submitted.

Case Study: QoC Optimization

- **Grid World**
  - The agent can take four actions: *North*, *South*, *East*, *West*
  - The agent moves in the intended direction w.p. 0.8
  - The agent moves in a direction orthogonal to the intended direction w.p. 0.2

- **QoC Optimization Example**
  - Safety:
    - *Avoid visiting a danger state d consecutively*
    - $\psi = \square\neg(d \wedge \bigcirc d)$
  - LTL:
    - Occasionally visit a checkpoint b
    - $\varphi = \square\lozenge b$
  - QoC:
    - Stay at the top-right corner as long as possible
    - $R^{\mathrm{QoC}}(\langle 0,5\rangle) = 1$



Thank you