PADERBORN UNIVERSITY

Department of Advanced Control

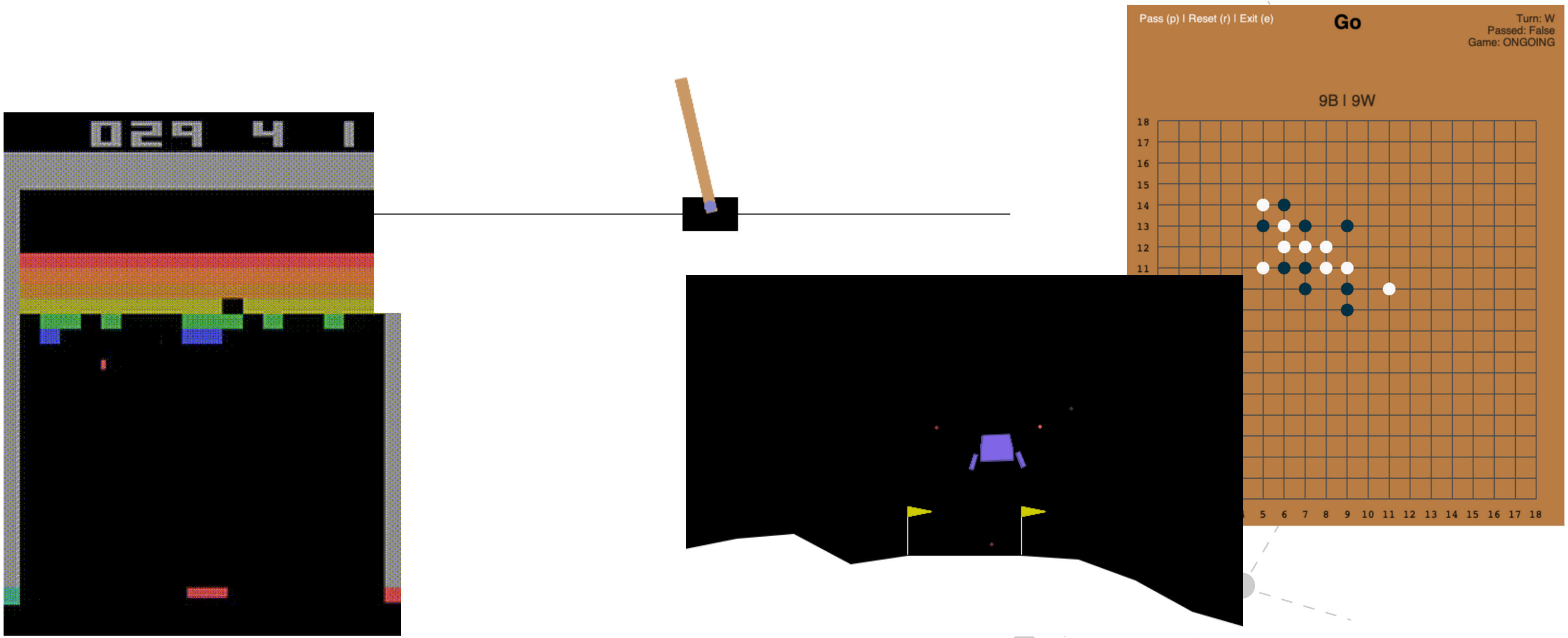
COMBINING MODEL-BASED AND MODEL-FREE OPTIMAL CONTROL FOR DYNAMIC SYSTEMS

MASTER'S THESIS PRESENTATION - HANDOUT

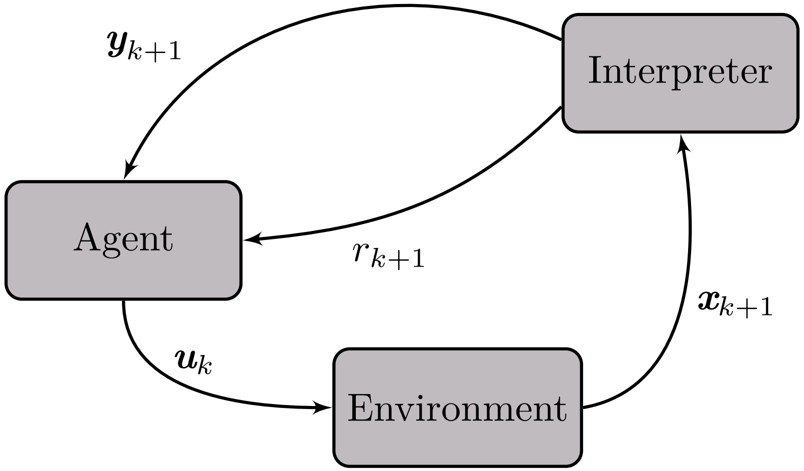
Pascal Peters

Paderborn, 04.05.2022

Motivation


Pass (p) I Reset (r) I Exit (e)
Go
Turn: W
Passed: False
Game: ONGOING
9B I 9W

Reinforcement Learning



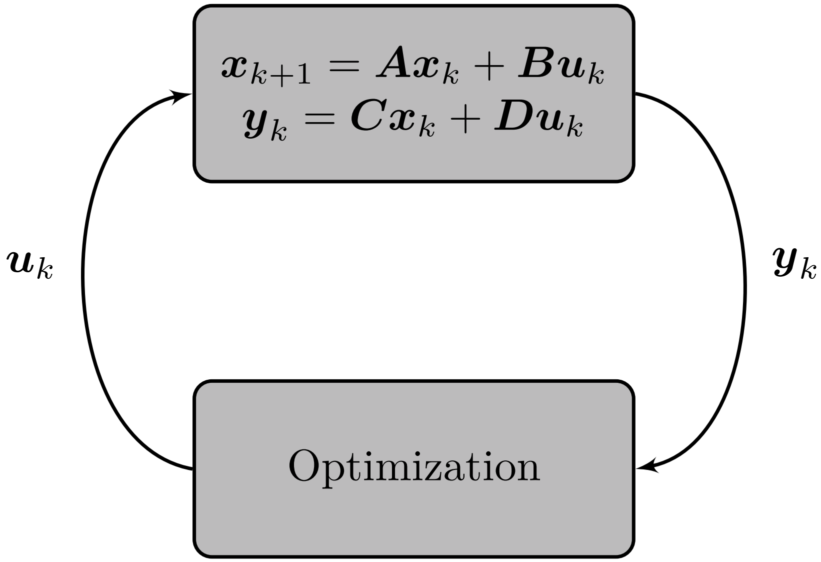
- Model-free
- Maximize the return: $g_k = \sum_{i=0}^{N} \gamma^i r_{k+i+1}$
- Solves Bellman equation for a horizon $N \to \infty$:

$$q_\pi(\boldsymbol{x}_k, \boldsymbol{u}_k) = \mathbb{E}_\pi\left[R_{k+1} + \gamma q_\pi(\boldsymbol{X}_{k+1}, \boldsymbol{U}_{k+1}) \middle| \boldsymbol{X}_k, \boldsymbol{U}_k\right]$$

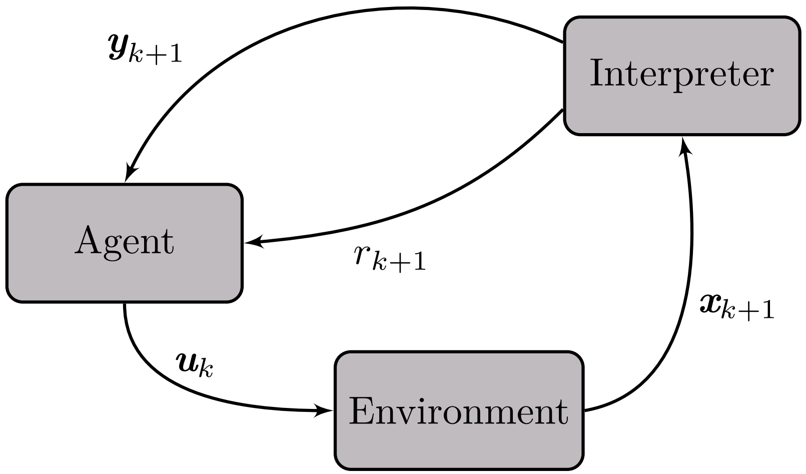
- Explicit control law to determine action $\boldsymbol{u}_k$
- Learn through experience and feedback

Model Predictive Control

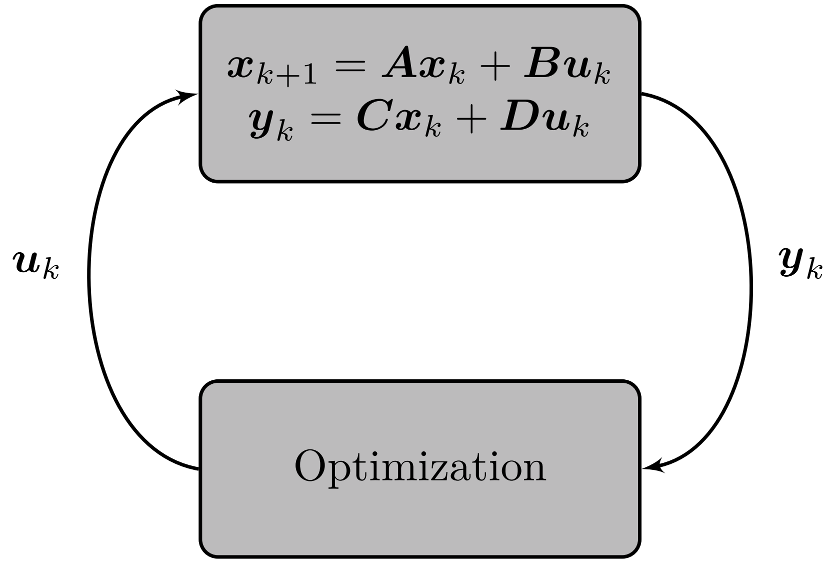
- Model-based
- Minimize costs
- Solve optimization problem for finite N-step prediction horizon
- Implicit control law with complexity growing with horizon length N
- Constrained optimization for safe control



Motivation


$\boldsymbol{y}_{k+1}$
Interpreter
Agent
r_{k+1}
$\boldsymbol{x}_{k+1}$
$\boldsymbol{u}_k$
Environment

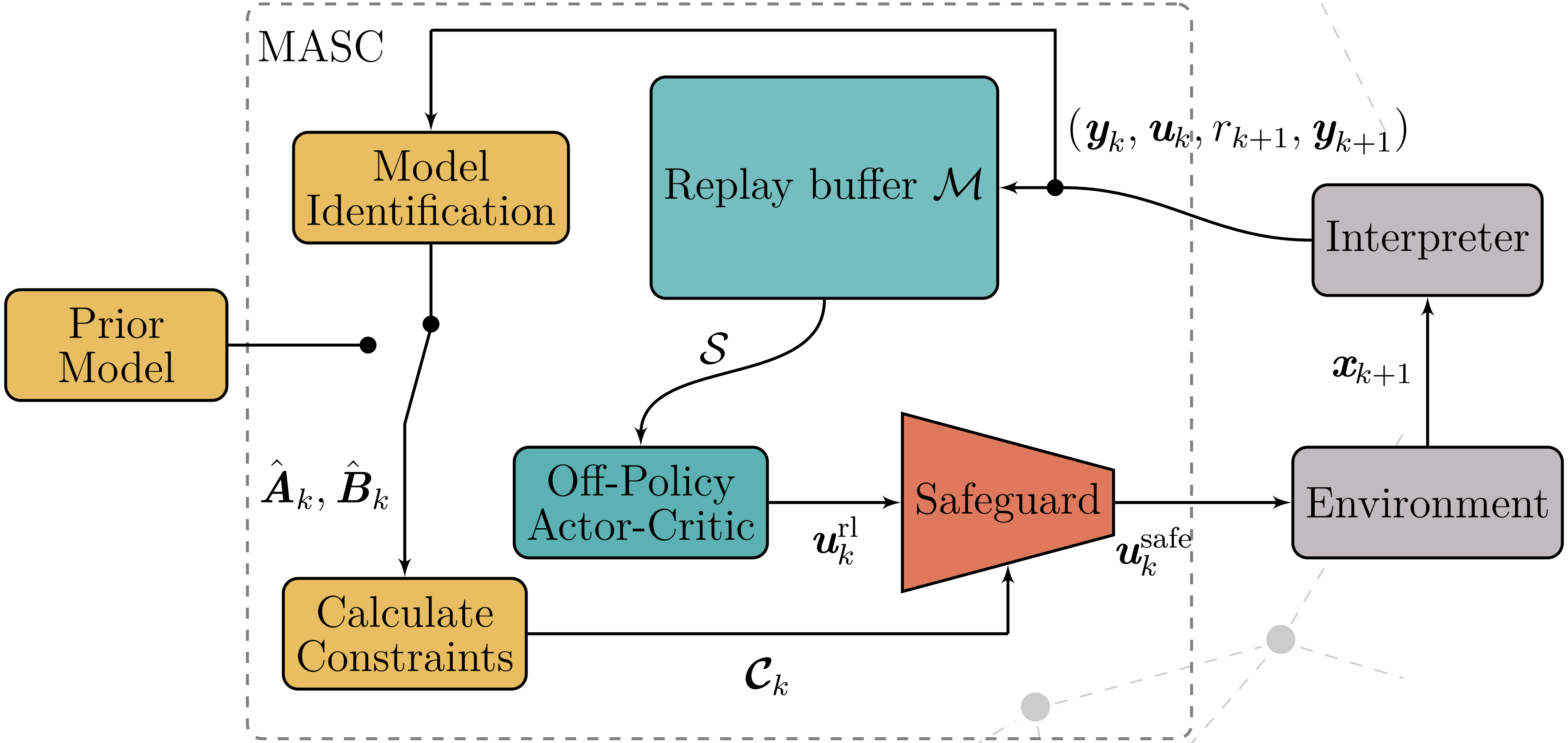
+


$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k$
$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k$
$\boldsymbol{u}_k$
$\boldsymbol{y}_k$
Optimization

Developed System

- Using a model to extend an off-policy actor-critic controller to achieve safe behavior
  - By monitoring actions for constraint violation
  - Modifying unsafe actions such that the system stays within the constraints
- Eliminate the dependency on prior available model knowledge by identifying model parameters online

Model Adaptive Safeguard Controller



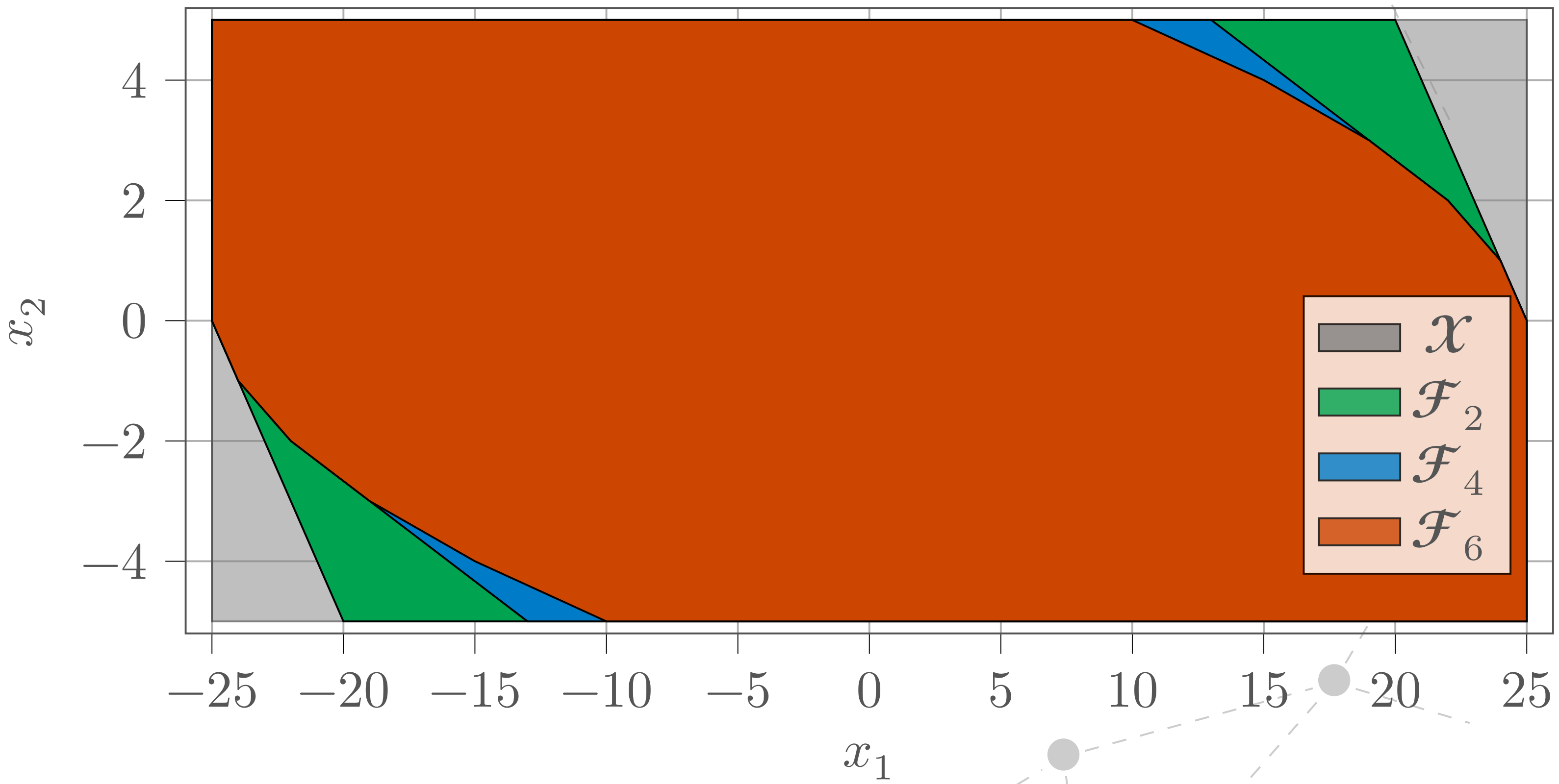
Safeguard

- Check for possible constraint violation
  - $\boldsymbol{u}_k^{\text{RL}}$ safe if: $\begin{bmatrix}\boldsymbol{x}_k^{\mathsf{T}} & \boldsymbol{u}_k^{\text{RL}\,\mathsf{T}}\end{bmatrix}^{\mathsf{T}} \in \mathcal{C}$
  - $\boldsymbol{u}_k^{\text{RL}}$ unsafe if: $\begin{bmatrix}\boldsymbol{x}_k^{\mathsf{T}} & \boldsymbol{u}_k^{\text{RL}\,\mathsf{T}}\end{bmatrix}^{\mathsf{T}} \notin \mathcal{C}$
- Solving optimization problem for unsafe actions:

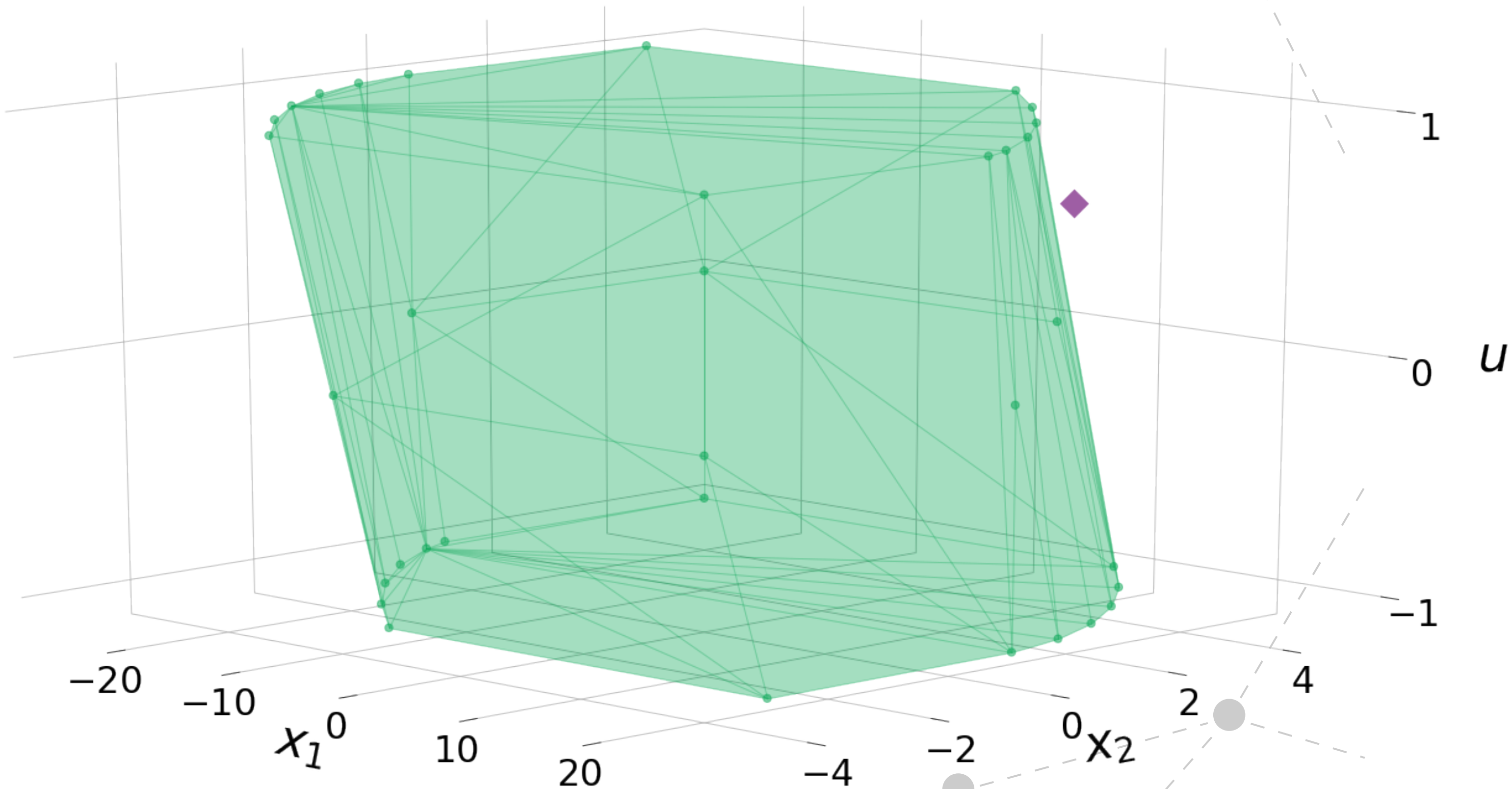
$$\boldsymbol{u}_k^{\text{safe}} = \boldsymbol{u}_k^* = \underset{\boldsymbol{u}_k}{\operatorname{argmin}} \,||\, \boldsymbol{u}_k - \boldsymbol{u}_k^{\text{RL}} \,||^2,$$

$$\text{s.t. } \boldsymbol{u}_k \in \mathcal{C}_u$$

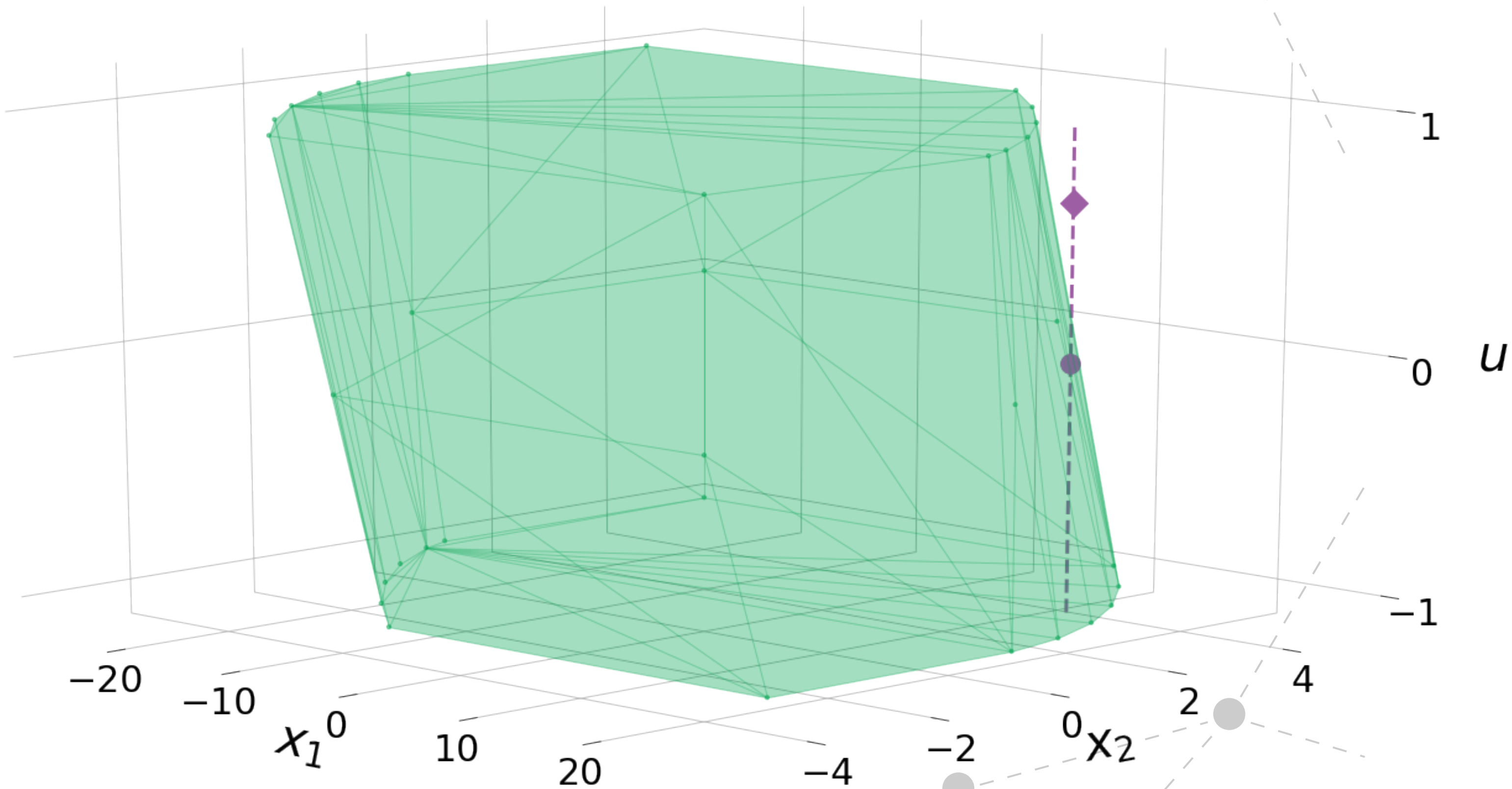
Safeguard Constraints



Safeguard Constraints



Safeguard Constraints



Model Identification

- Online model identification with the recursive least squares algorithm

$$\kappa_k = \frac{\boldsymbol{P}_k \xi_{k+1}}{\lambda_{k+1} + \xi_{k+1}^\mathsf{T} \boldsymbol{P}_k \xi_{k+1}}$$

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \kappa_k \left(\psi_{k+1} - \xi_{k+1}^\mathsf{T} \hat{\theta}_k \right)$$

$$\boldsymbol{P}_{k+1} = \left(\boldsymbol{I} - \kappa_k \xi_{k+1}^\mathsf{T} \right) \boldsymbol{P}_k \frac{1}{\lambda_{k+1}}$$

- Update feasible set only if change of new estimates is significant:

$$\Delta \boldsymbol{A}_k = \frac{||\hat{\boldsymbol{A}}_k - \hat{\boldsymbol{A}}_{k-1}||_F}{||\hat{\boldsymbol{A}}_k||_F}, \Delta \boldsymbol{B}_k = \frac{||\hat{\boldsymbol{B}}_k - \hat{\boldsymbol{B}}_{k-1}||_F}{||\hat{\boldsymbol{B}}_k||_F}$$

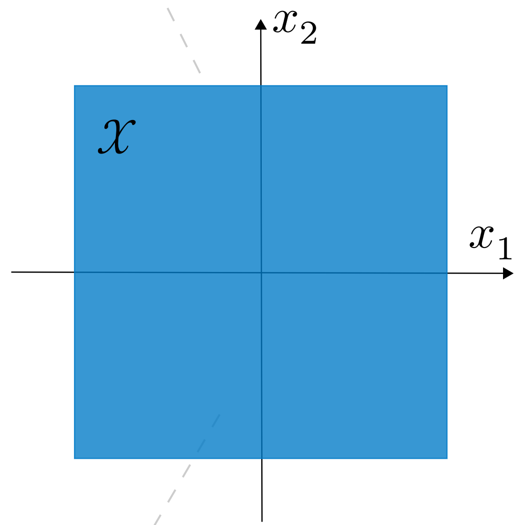
Evaluation

- Double Integrator as test environment

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \boldsymbol{u}(t),$$

$$\mathrm{y}(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{x}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0$$

s.t. $\boldsymbol{M}_x \boldsymbol{x} \leq \boldsymbol{m}_x,$



$\boldsymbol{M}_u \boldsymbol{u} \leq \boldsymbol{m}_u$

Evaluation

- Scaled weighted sum of errors (SWSE):

$$r_{k+1}^{\mathrm{WSE}} = \begin{cases} -\frac{1}{f}\left(\sum_n w_n \left|\frac{x_{k,n} - x_{k,n}^{\mathrm{ref}}}{2x_n^{\mathrm{lim}}}\right|\right) & , \boldsymbol{x}_{k+1} \in \mathcal{X} \\ r^{\mathrm{violation}} & , \boldsymbol{x}_{k+1} \notin \mathcal{X} \end{cases}$$

- Incorporate safeguard-penalty r^{SG} for actuated safeguard

Metrics

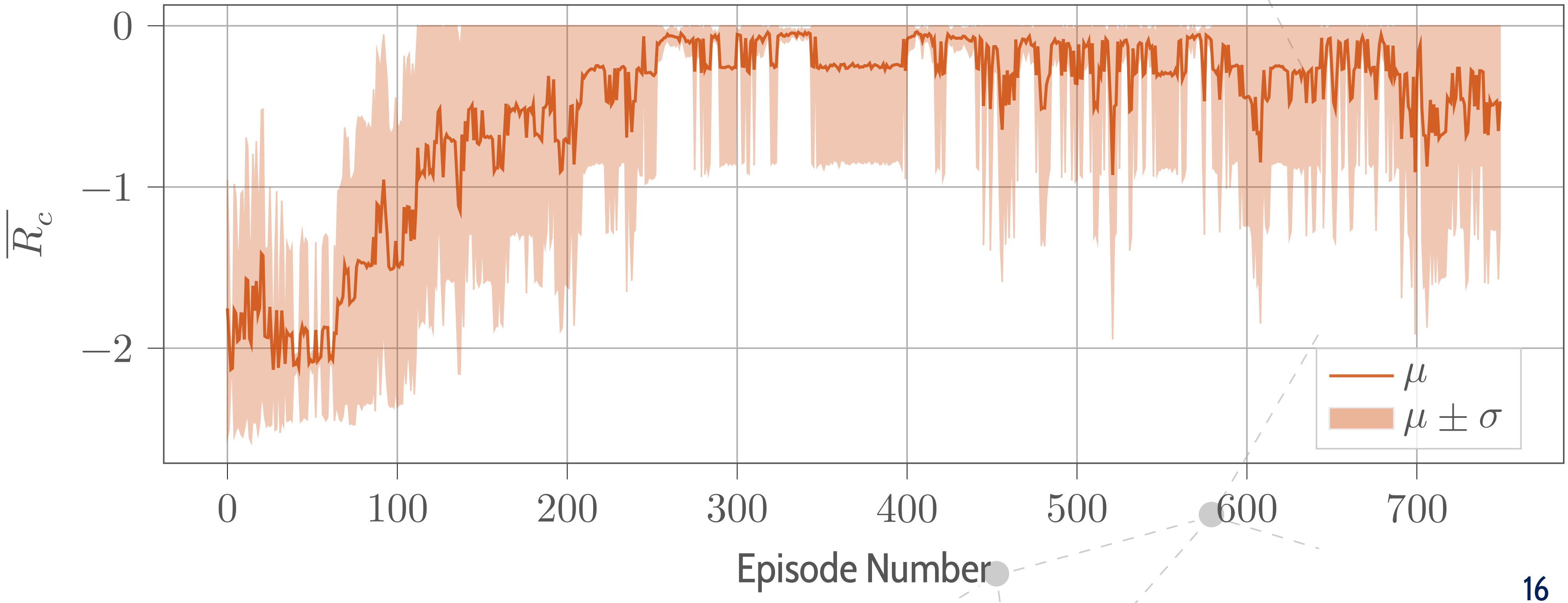
- Averaged over multiple seeds
- Cumulated reward per episode:

$$R_c = \sum_{k=1}^{N} r_k$$

- Cumulated constraint violations (CCV):

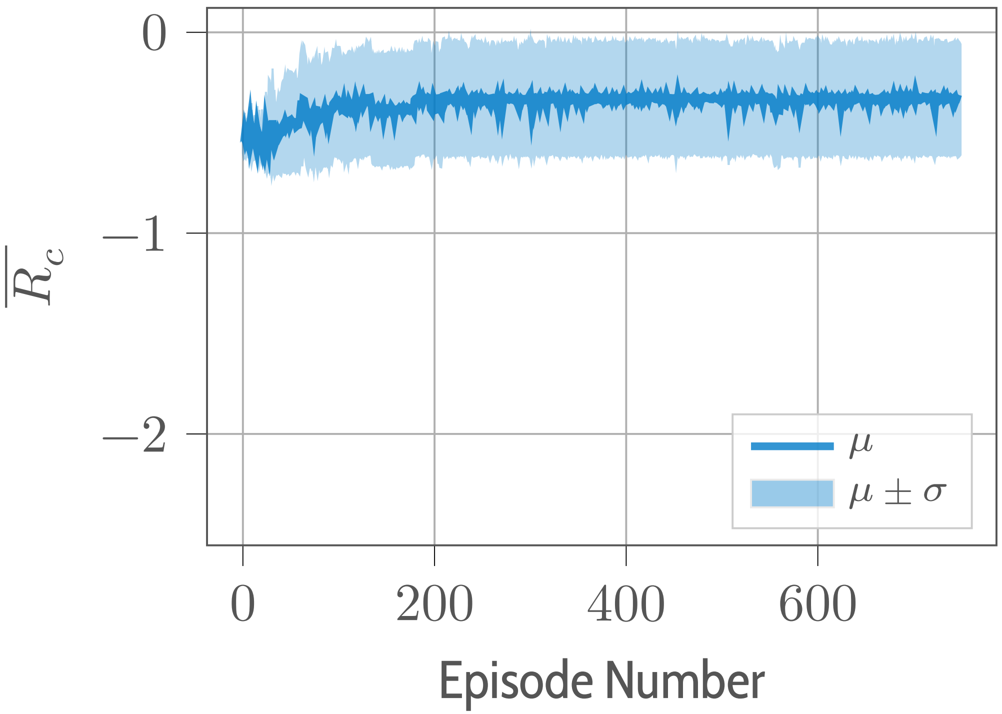
$$\text{CCV} = \sum_{e=1}^{E} c_e(\boldsymbol{x}), \quad \text{with } c_e(\boldsymbol{x}) = \begin{cases} 0 & \text{, if } \boldsymbol{x} \in \mathcal{X} \\ 1 & \text{, otherwise} \end{cases}$$

Reinforcement Learning Controller

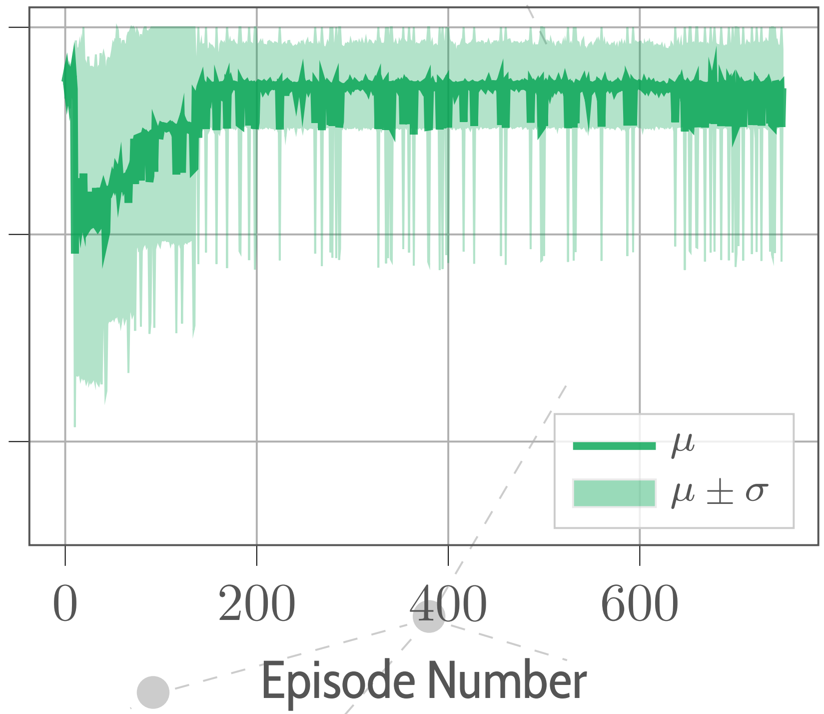


Safeguard Controller

- Prior available model:

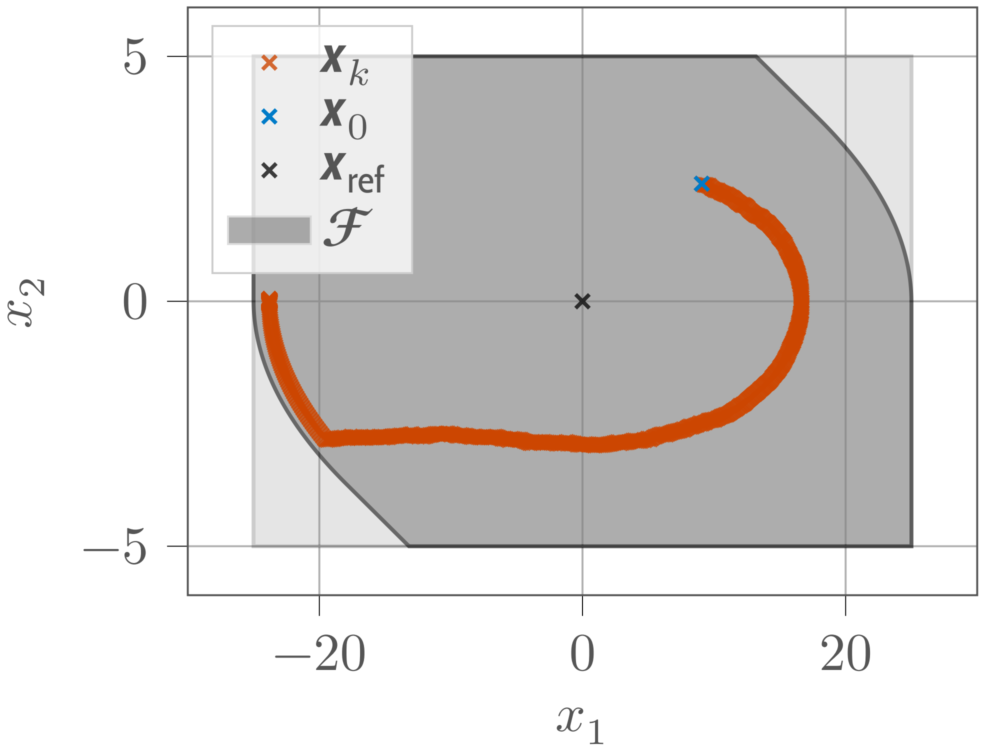


- Estimated model:

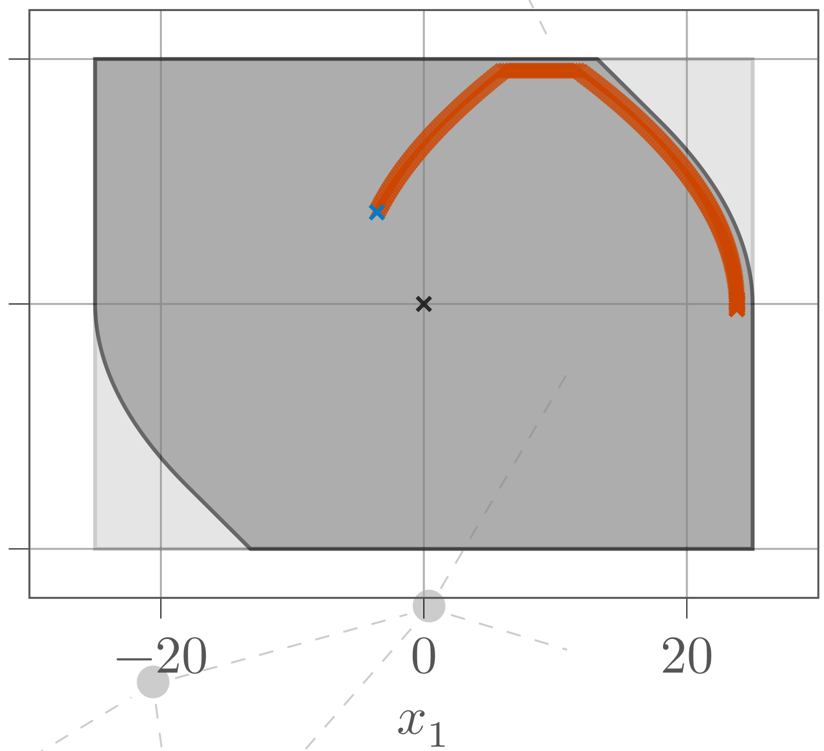


Exemplary Safeguard Controller Trajectories

- Training start:

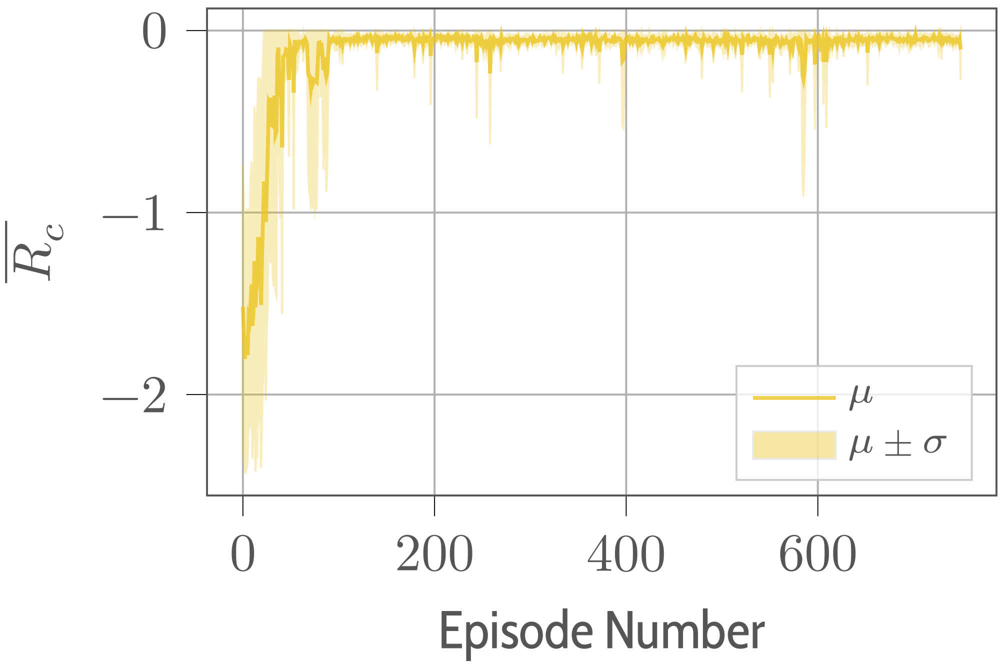


- Training end:

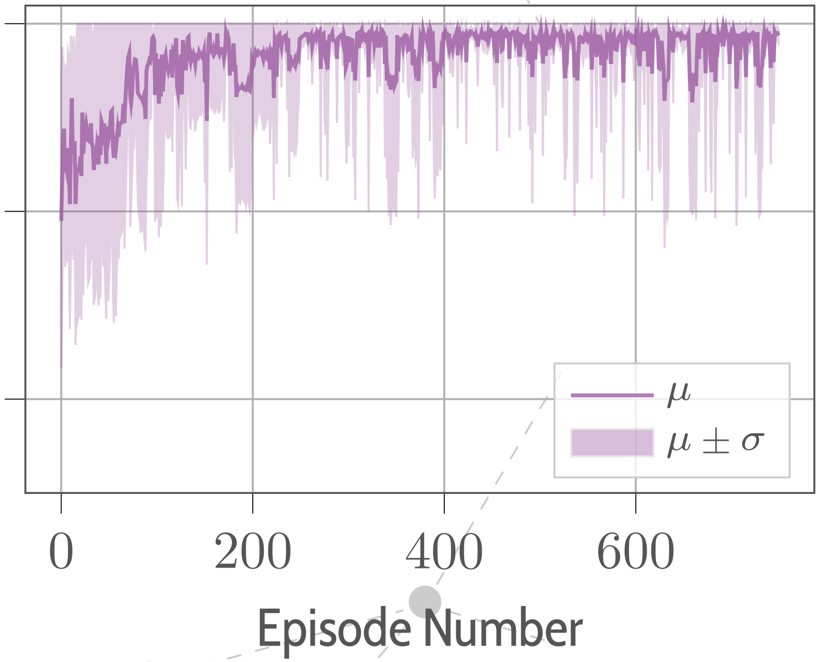


Safeguard Penalty

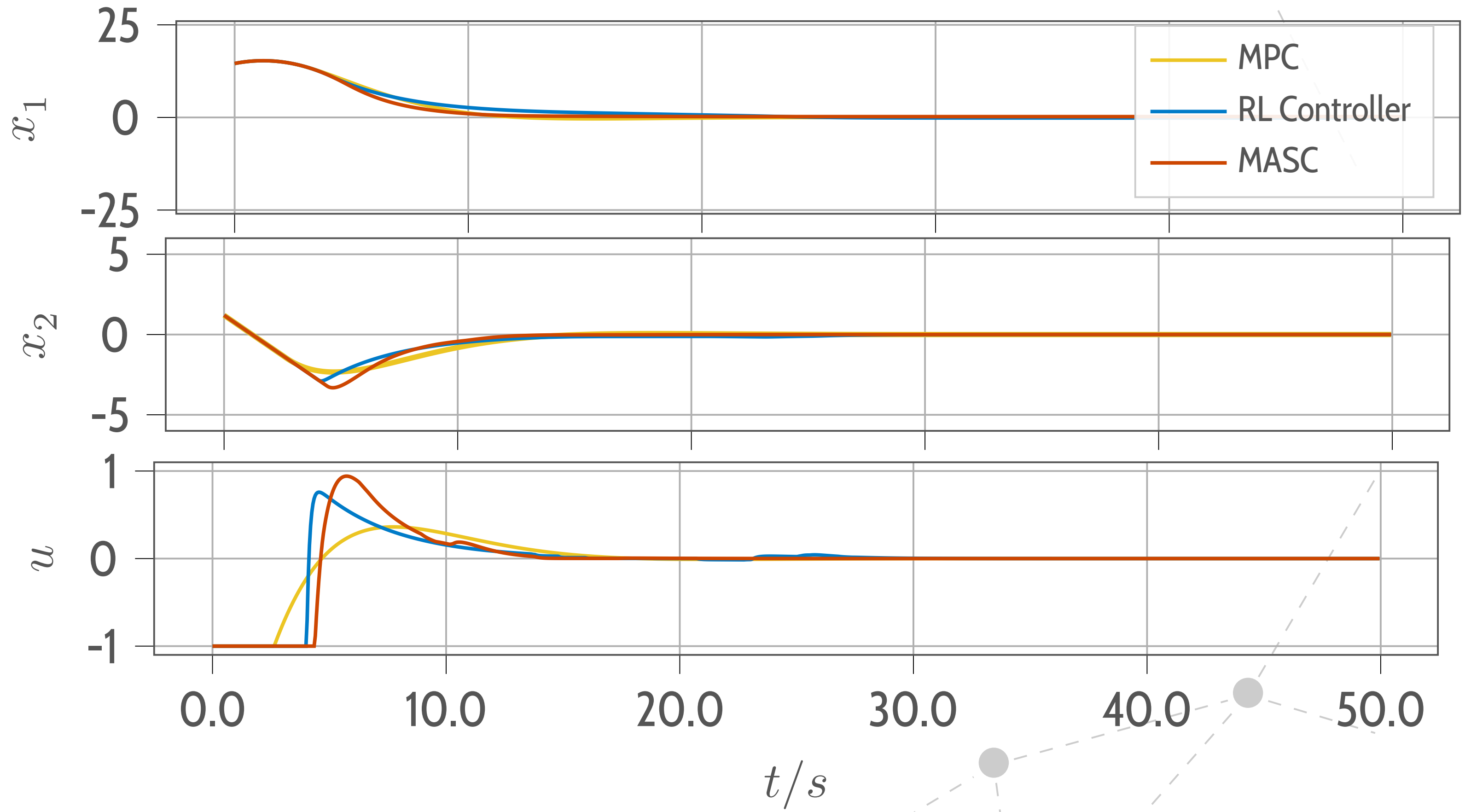
- Prior available model with safeguard penalty:



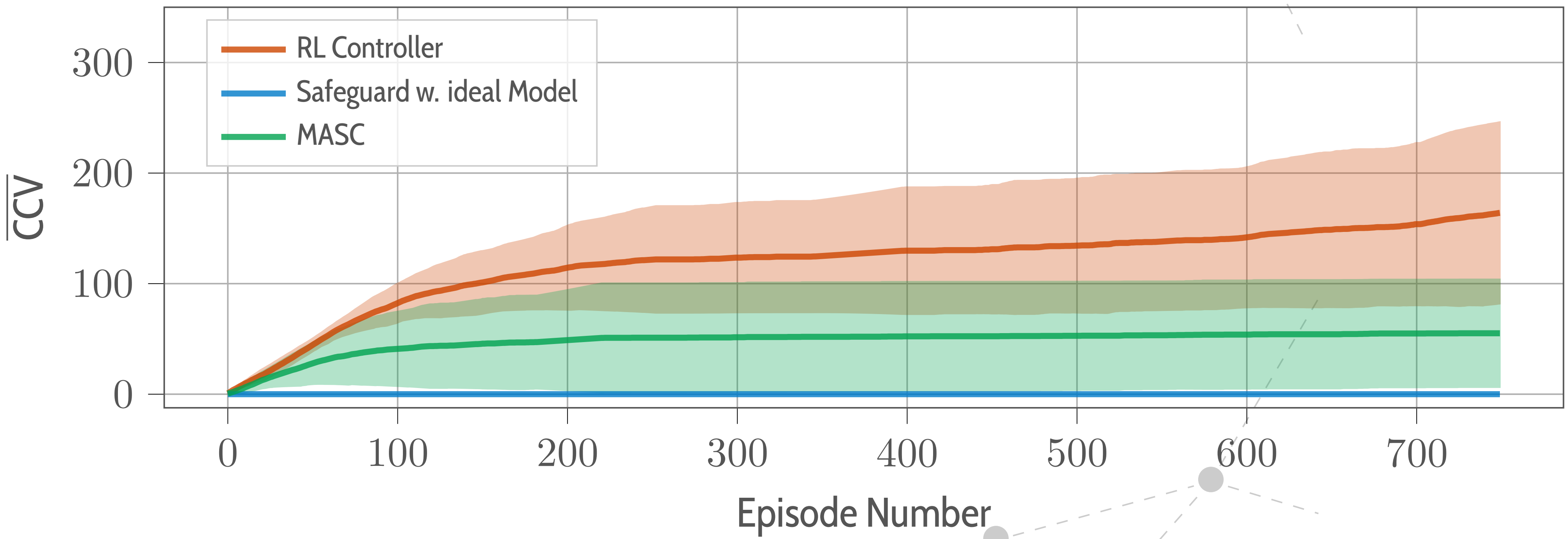
- Estimated model with safeguard penalty:



Exemplary Control Trajectories



Cumulated Constraint Violations



Summary

- Extension of an off-policy actor-critic controller with the so called safeguard to enforce constraint satisfaction
- Utilize the feasible-set as safeguard constraints to restrain the RL controller to move in the feasible state-action space
- Use the RLS algorithm for online model estimation

Future Work

- Use computational less expensive methods to determine the feasible set
- Extend exploration mechanism of the RL controller:
  - Utilize estimated model to employ MPC for directional, controlled exploration on basis of risk and curiosity criteria
  - providing safer exploration during the starting model identification phase
- Extensive hyperparameter optimization of full setup MASC

Thank you for your attention.