Safe and Efficient robot control

Combining learning and trajectory optimization

Andrea Del Prete

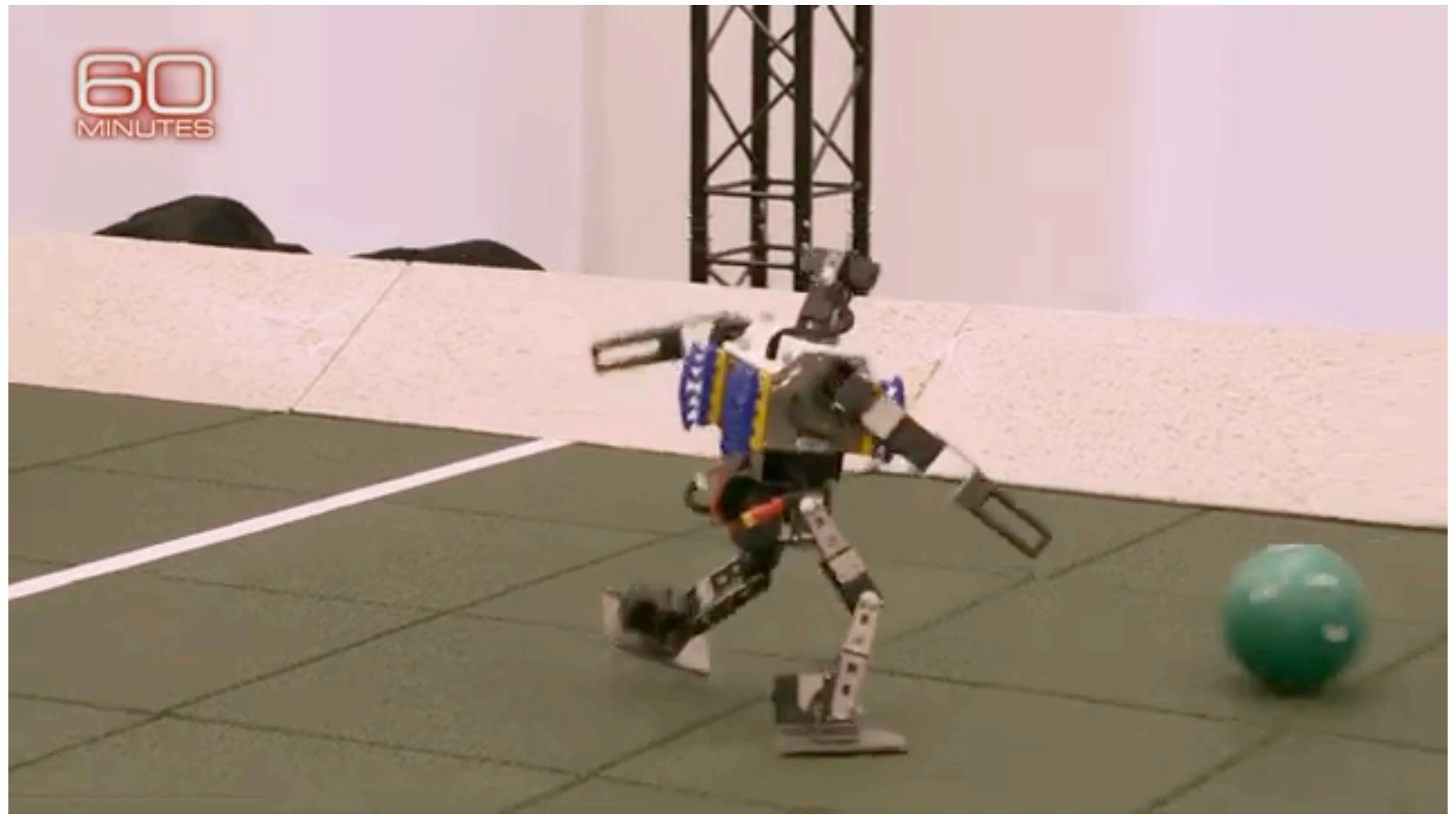
UNIVERSITY OF TRENTO

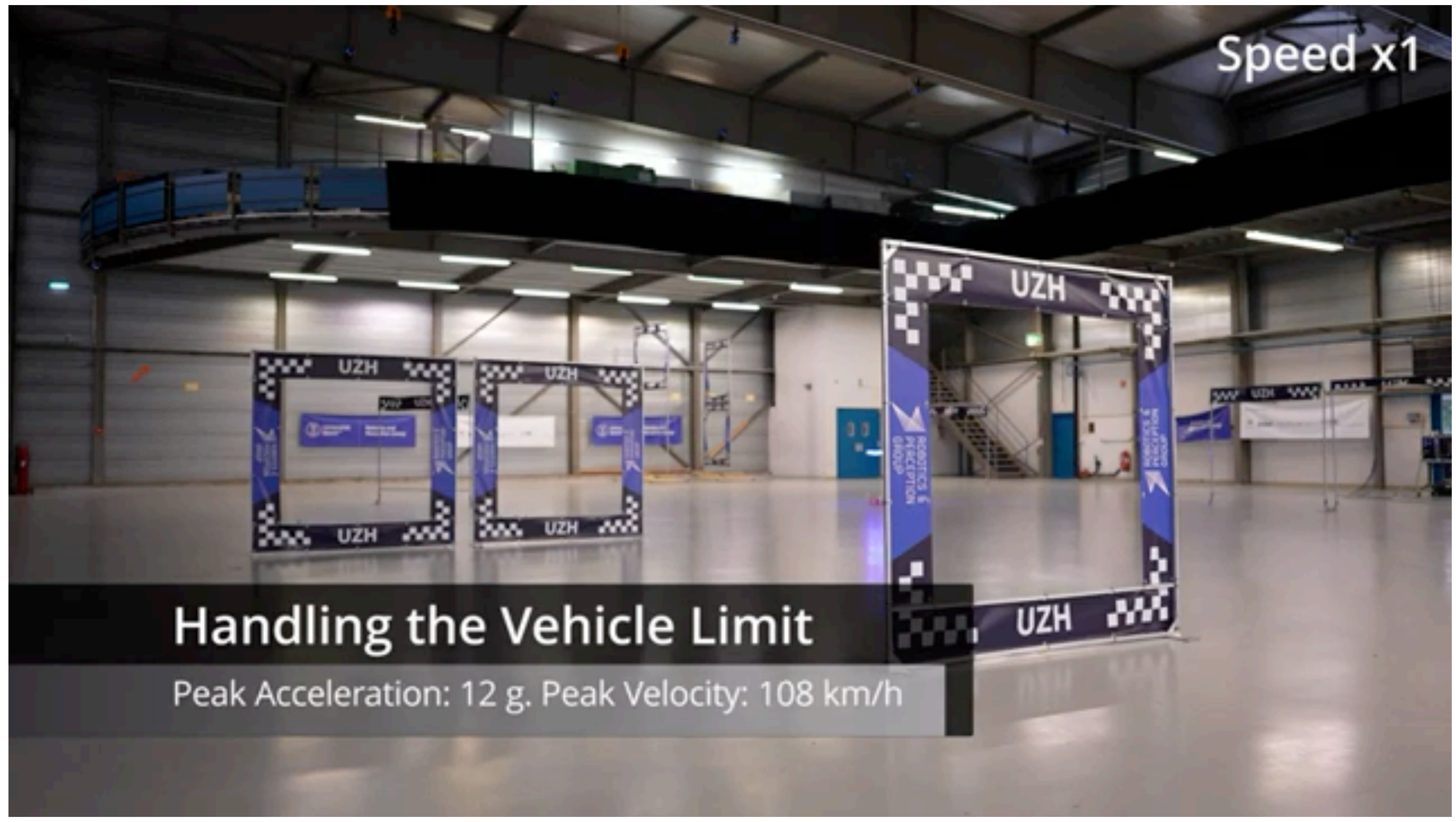
Is there anything RL cannot do?

Is Trajectory Optimization bound to die?

Lee, Hwangbo, Wellhausen, Koltun, Hutter (2020). Learning quadrupedal locomotion over challenging terrain. Science Robotics

Haarnoja, T., Moran, B., Lever, G., Huang, S. H., Tirumala, D., Wulfmeier, M., … Heess, N. (2023). Learning Agile Soccer Skills for a Bipedal Robot with Deep Reinforcement Learning





Song, Romero, Müller, Koltun, Scaramuzza, (2023). Reaching the limit in autonomous racing: Optimal control versus reinforcement learning. Science Robotics

The issues with RL

My two cents

Poor efficiency

- Data efficiency
- Energy efficiency
- Time efficiency

Poor safety

- No explicit constraints
- No guarantees
- Safety-critical applications

Can we use ideas from Trajectory Optimization to make RL safe and efficient?

Safe and Efficient robot control

Combining learning and trajectory optimization

Andrea Del Prete

UNIVERSITY OF TRENTO

* UNIVERSITY OF TRENTO ** UNIVERSITY OF NOTRE DAME *** ENS | PSL

CACTO
Continuous Actor-Critic with Trajectory Optimization

Gianluigi Grandesso*, Elisa Alboni*, Gastone Rosati Papini*, Patrick Wensing, Justin Carpentier***, Andrea Del Prete***

[1] Grandesso, Alboni, Rosati Papini, Wensing, Del Prete (2023). CACTO: Continuous Actor-Critic With Trajectory Optimization - Towards Global Optimality. IEEE Robotics and Automation Letters
[2] Alboni, Grandesso, Rosati Papini, Carpentier, Del Prete (2024). CACTO-SL: Using Sobolev Learning to improve Continuous Actor-Critic with Trajectory Optimization. In Learning for Dynamics and Control Conference (L4DC)

Reinforcement Learning ~~VS~~ Trajectory Optimization

WITH?

$$
\begin{aligned}
\min_{x(t),u(t)} \; & \int_0^T l\left(x(t), u(t)\right)dt + l_f\left(x(T)\right) \\
\text{s. t.} \quad & \dot{x}(t) = f(x(t), u(t), t) \quad \forall t \in [0, T] \\
& x(0) = x_0 \\
& u_{min} < u(t) < u_{max} \quad \forall t \in [0, T]
\end{aligned}
$$

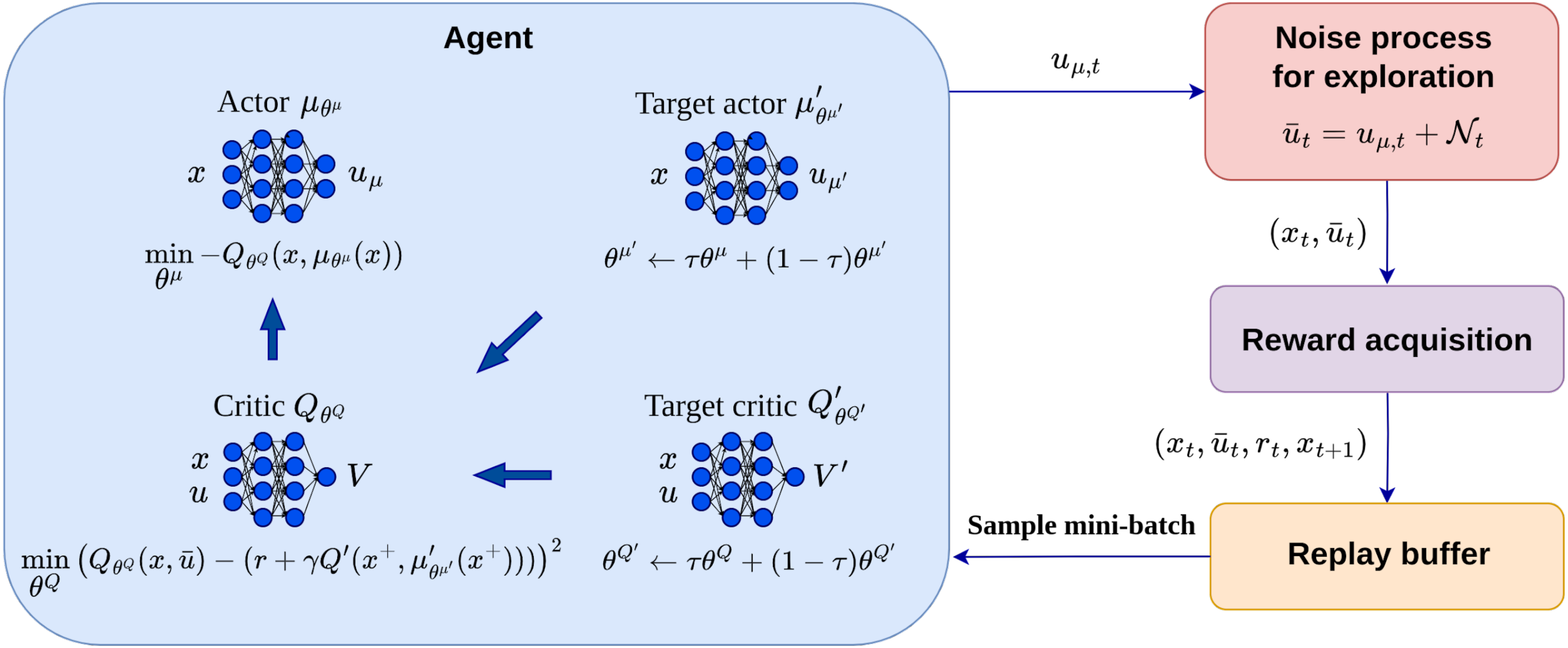
Reinforcement Learning

- + Less prone to poor local minima
- + Derivative free
- + Policy as output
- − Poor data efficiency (slow)

Trajectory Optimization

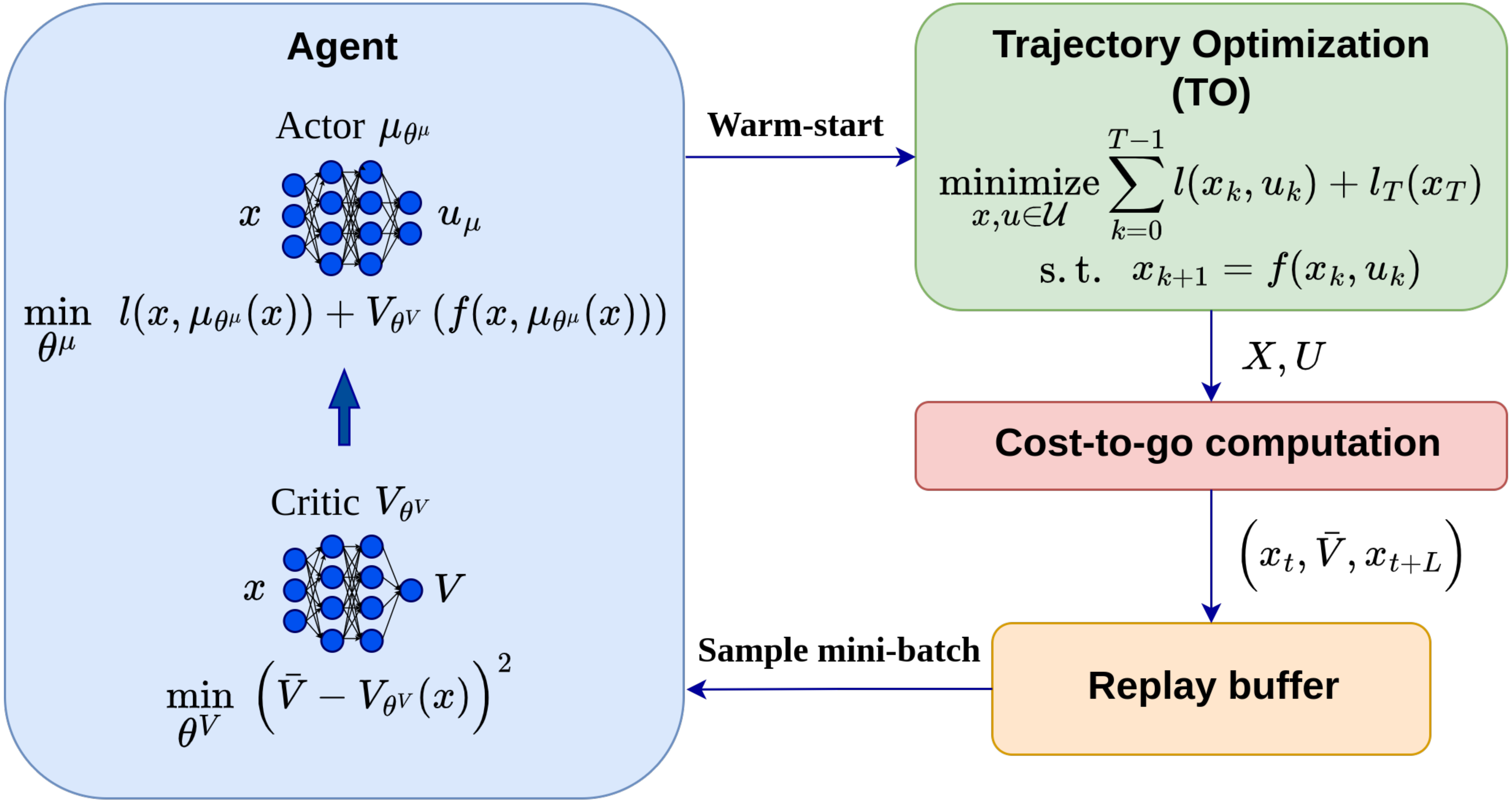
- + Data efficient (fast)
- + Exploits knowledge of dynamics derivatives
- − Can get stuck in poor local minima
- − Trajectory as output

Deep Deterministic Policy Gradient (DDPG)



Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., … Wierstra, D. (2015). Continuous control with deep reinforcement learning. In Foundations and Trends in Machine Learning

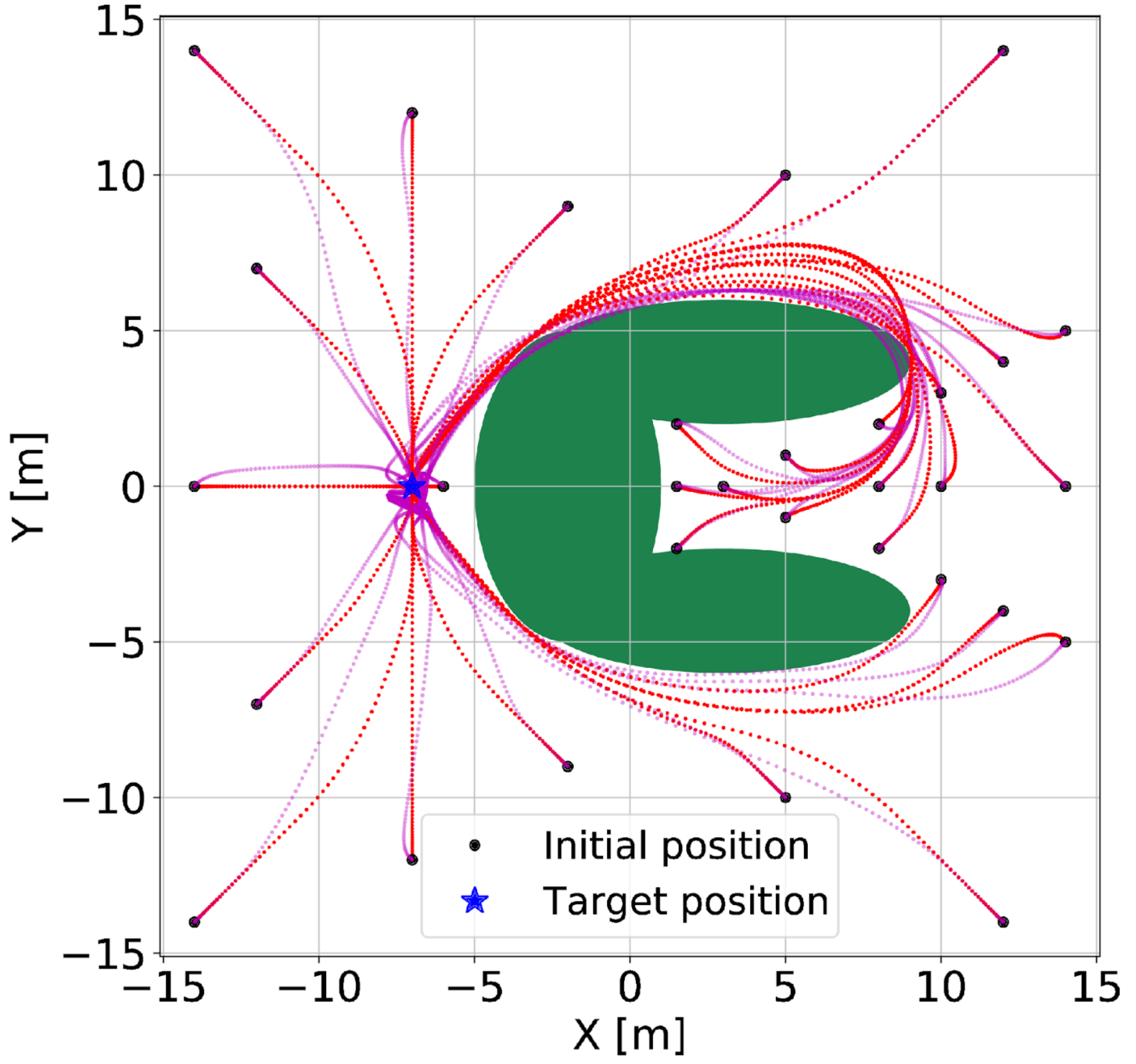
CACTO



[1] Grandesso, Alboni, Rosati Papini, Wensing, Del Prete (2023). CACTO: Continuous Actor-Critic With Trajectory Optimization - Towards Global Optimality. IEEE Robotics and Automation Letters

Results

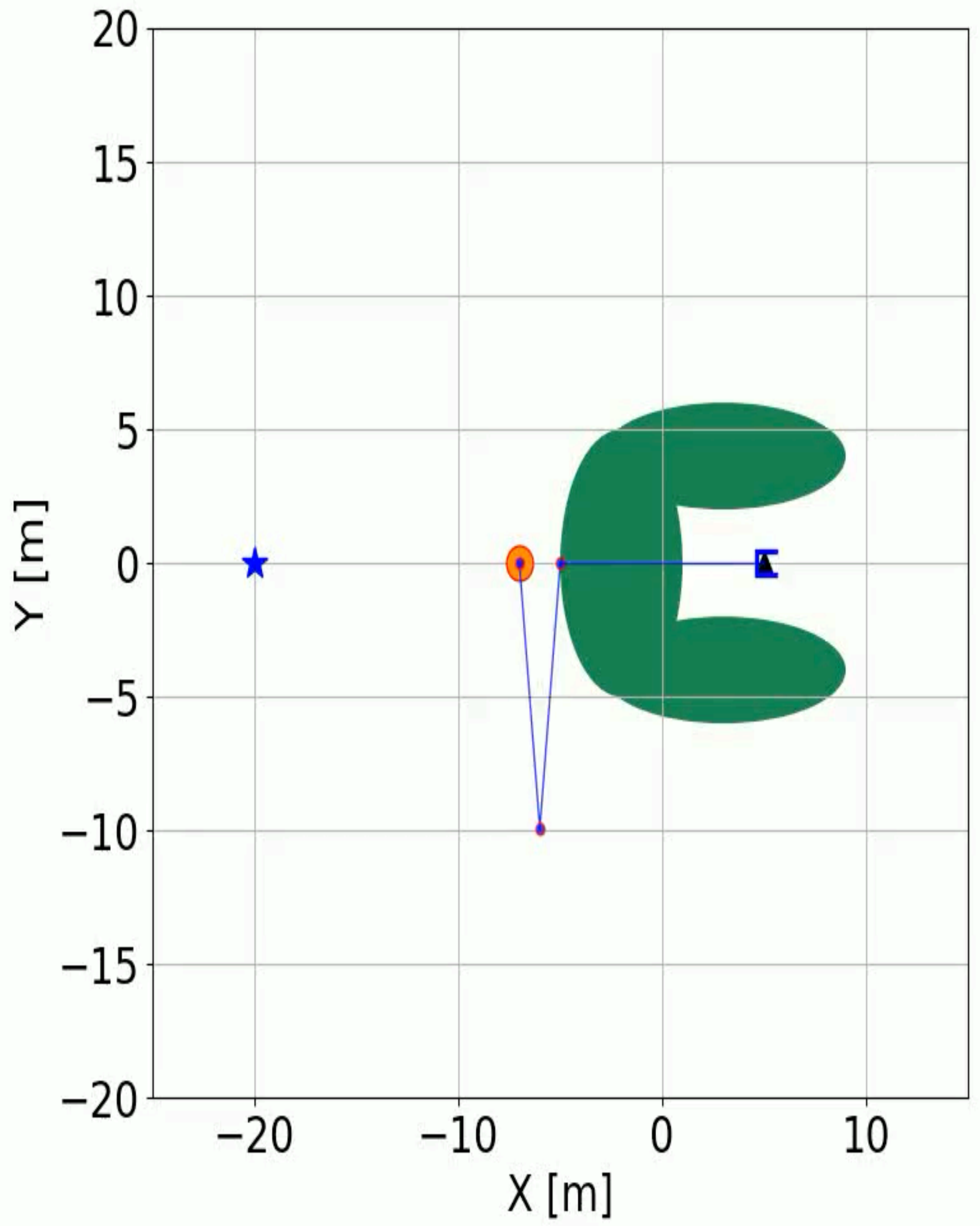
Task: find shortest path to target using low control effort and avoiding obstacles



Systems: 2D single/double integrator, 6D car model, 3-joint manipulator

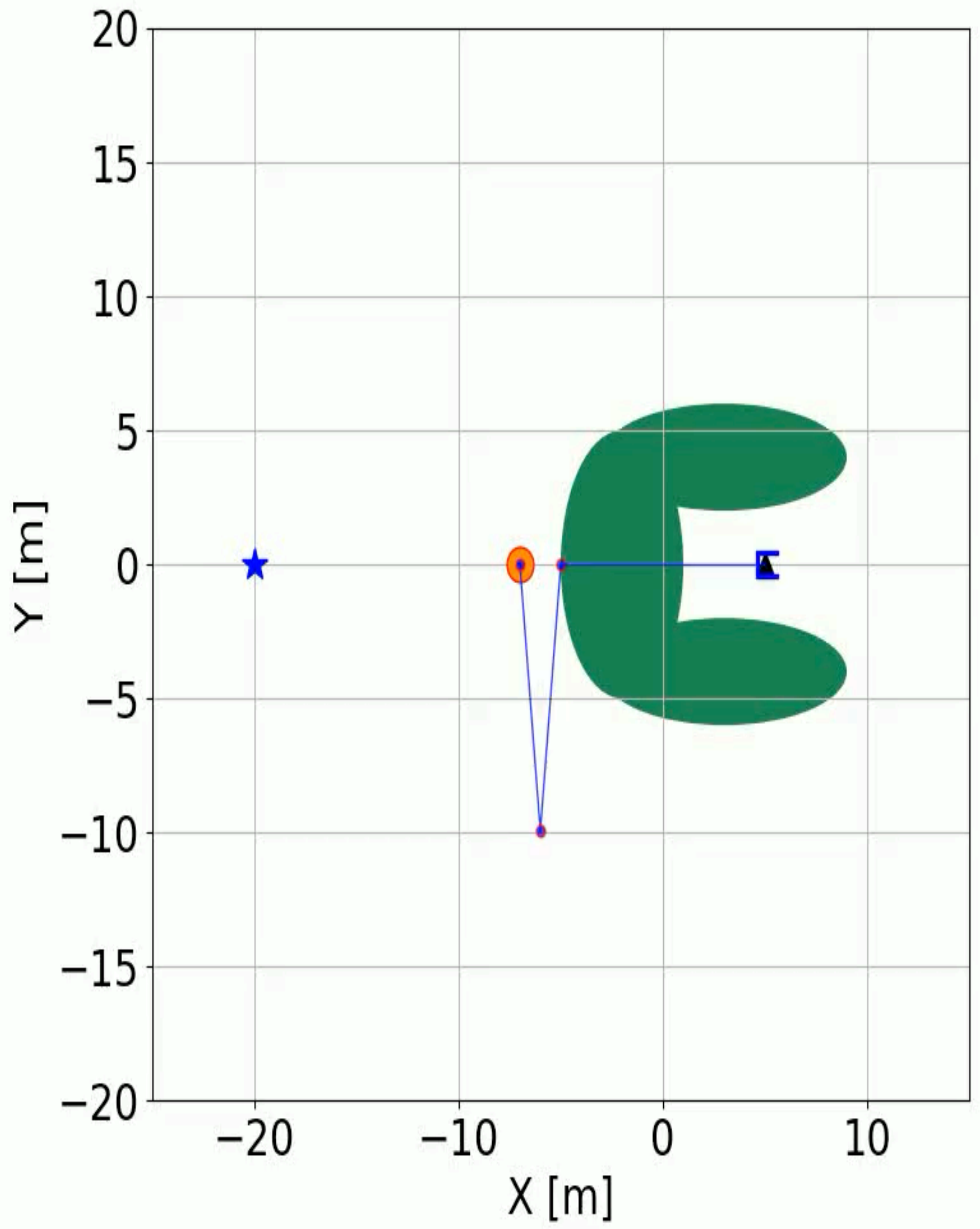
Results: 3-DoF Manipulator

Initial Conditions
warm-start



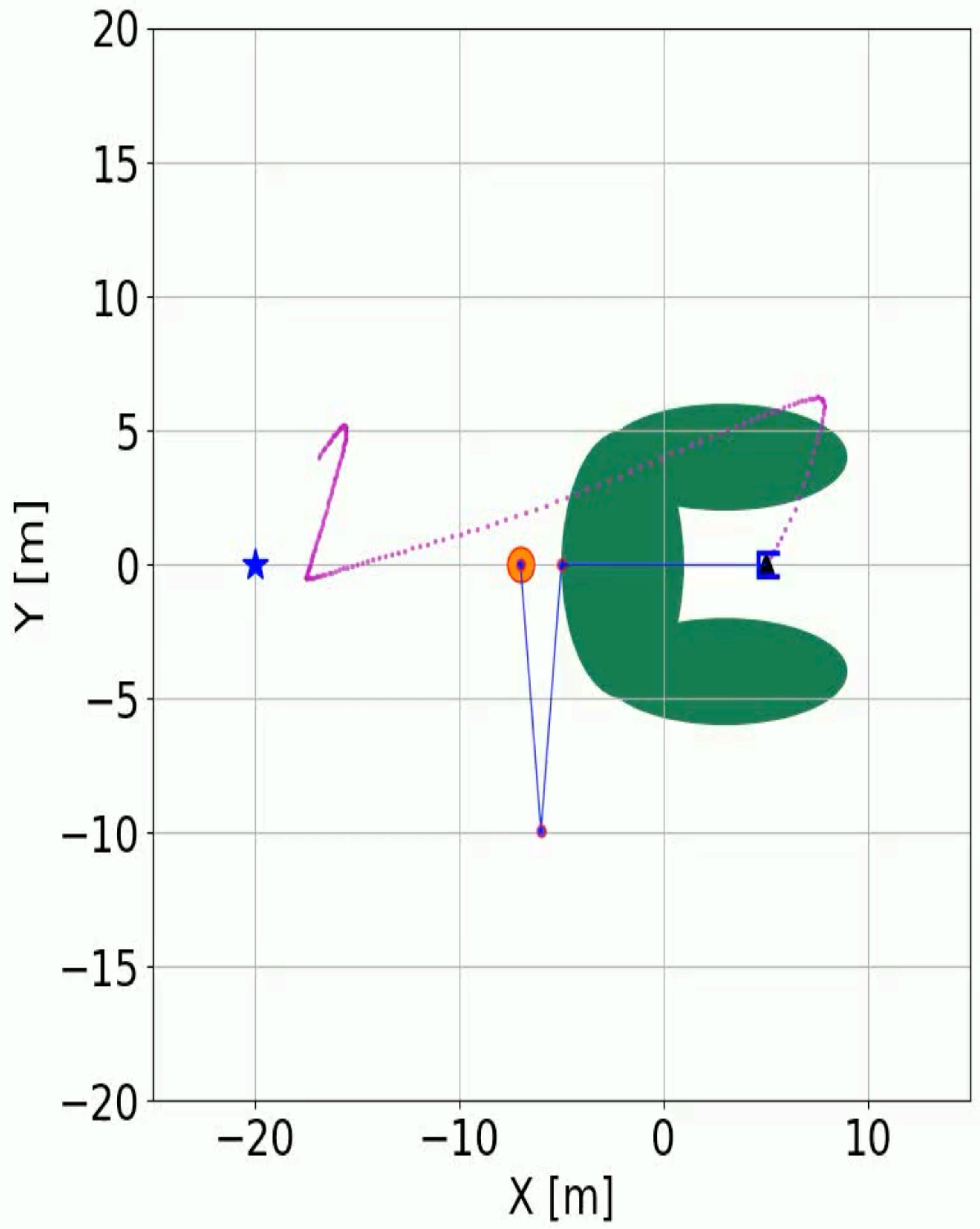
Cost = 70800

Random
warm-start



Cost = 88647

CACTO
warm-start



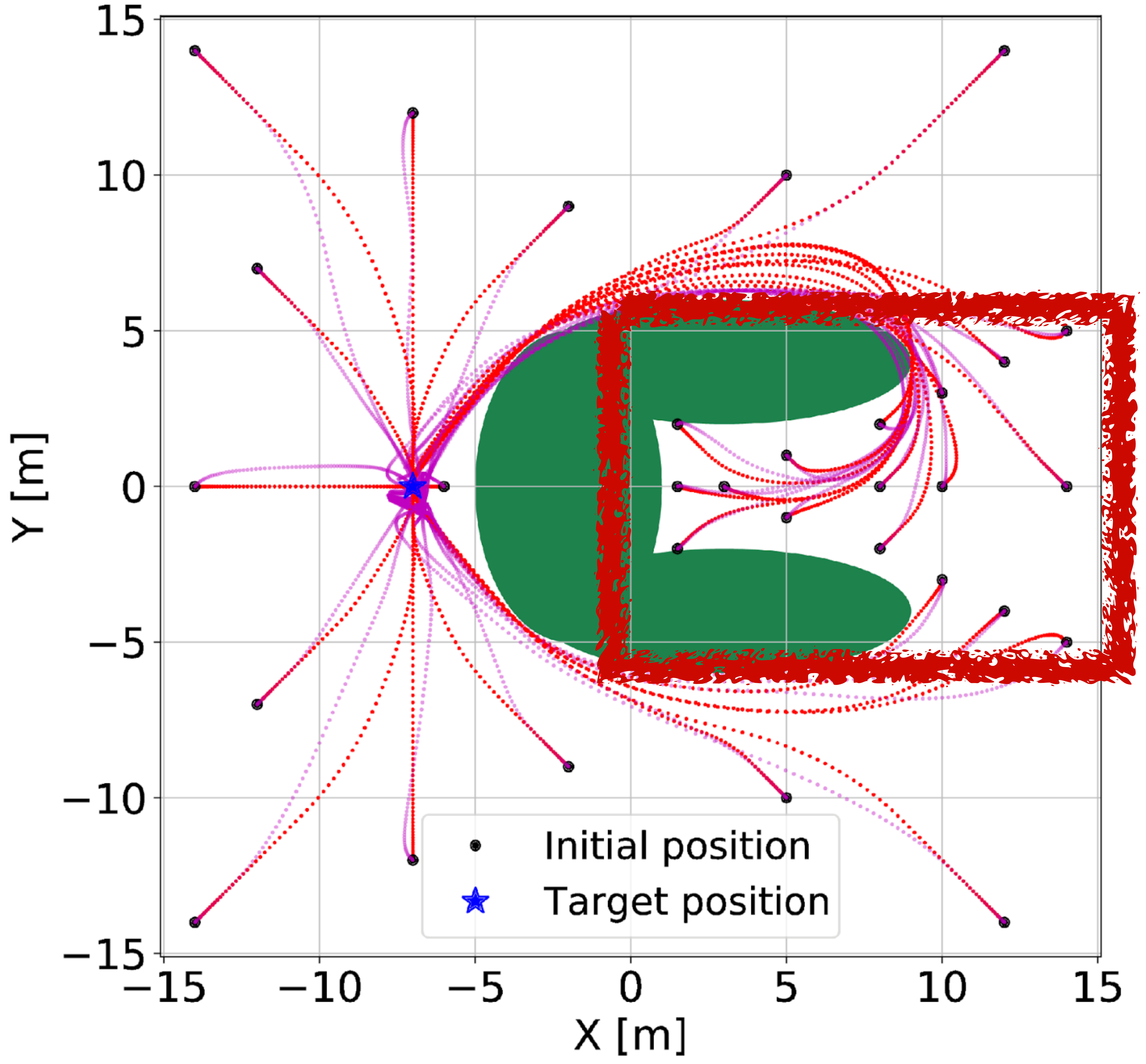
Cost = -145875

Comparison: CACTO vs TO

% of times TO finds better solution if warm-started with CACTO rather than:

- Random values
- Initial conditions (ICS) for states, zero for other variables

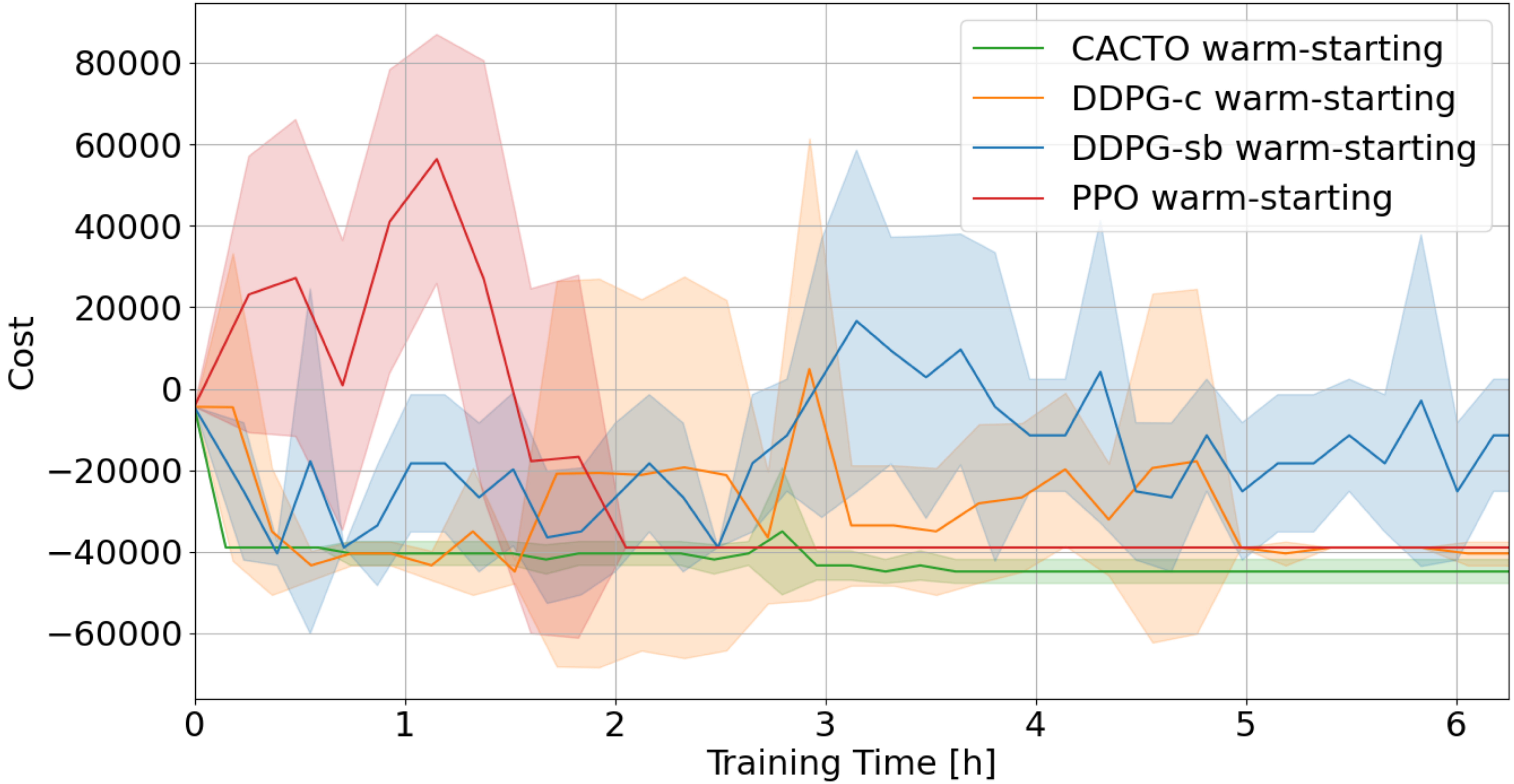
System	Hard Region	
	CACTO < (≤) Random	CACTO < (≤) ICS
2D Single Integrator	**99.1%** (99.1%)	**92%** (99.1%)
2D Double Integrator	**99.9%** (99.9%)	**92%** (99.1%)
Car	**100%** (100%)	**92.9%** (100%)
Manipulator	**87.5%** (87.5%)	**100%** (100%)



2D Double Integrator - CACTO warm-start

Comparison: CACTO, DDPG, PPO

Mean cost + std. dev. (across 5 runs) found by TO warm-started with different policies



Conclusions

- TO guides the RL exploration making it sample efficient
- Global convergence proof for discrete-space version of CACTO

Recent extension

- Improve data efficiency leveraging derivative of Value function [2]

Future work

- Bias initial episode state to improve data efficiency
- Parallelize on GPUs
- Handle non-differentiable dynamics

[2] Alboni, Grandesso, Rosati Papini, Carpentier, Del Prete (2024). CACTO-SL: Using Sobolev Learning to improve Continuous Actor-Critic with Trajectory Optimization. In Learning for Dynamics and Control Conference (L4DC)

Safe and Efficient robot control

Combining learning and trajectory optimization

Andrea Del Prete

UNIVERSITY OF TRENTO

Receding-Constraint Model Predictive Control

UNIVERSITY OF TRENTO

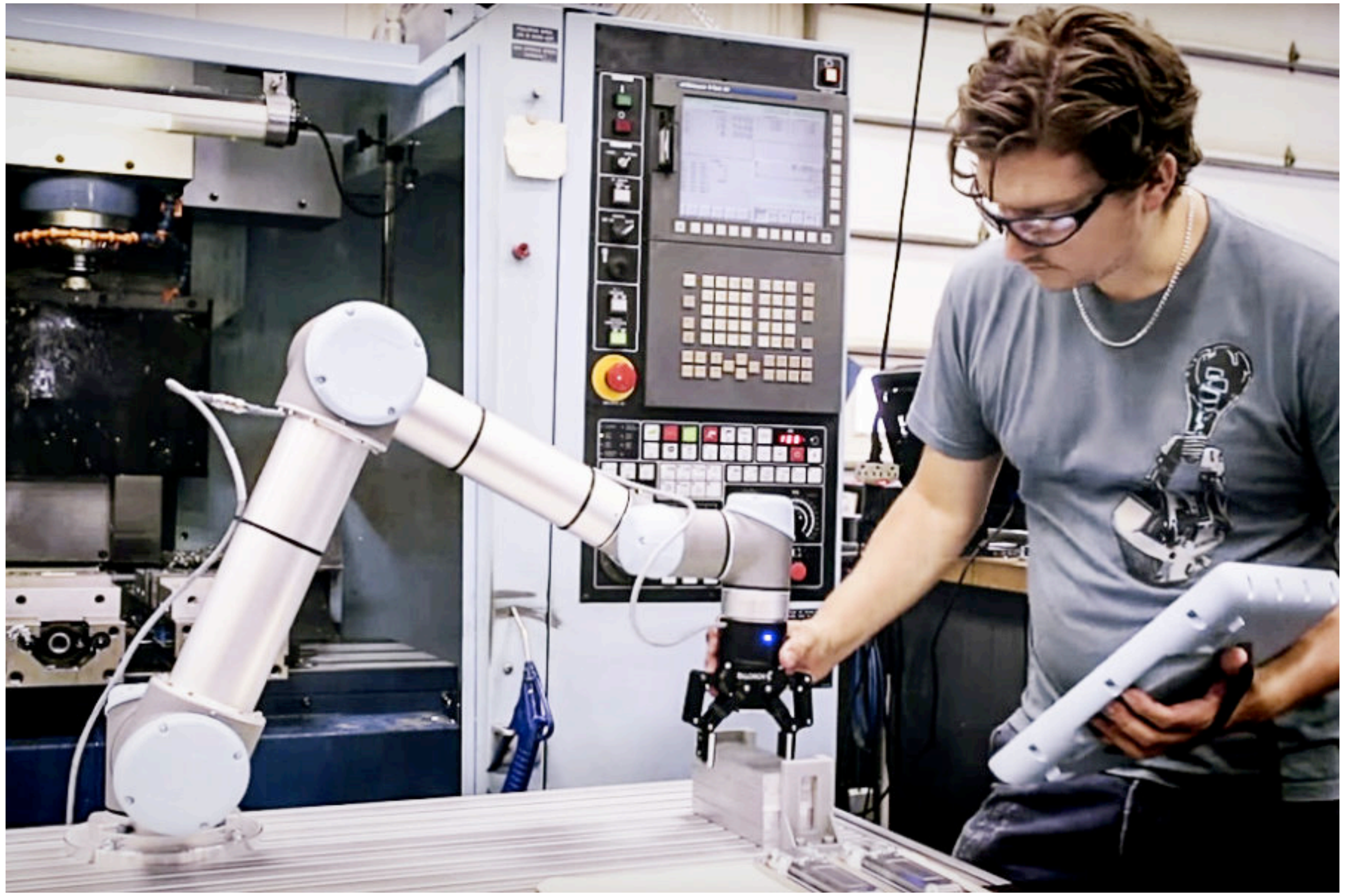
Gianni Lunardi
Asia La Rocca
Matteo Saveriano
Andrea Del Prete

Lunardi, La Rocca, Saveriano, Del Prete (2024). Receding-Constraint Model Predictive Control using a Learned Approximate Control-Invariant Set. IEEE ICRA.

Why Safety?

Today

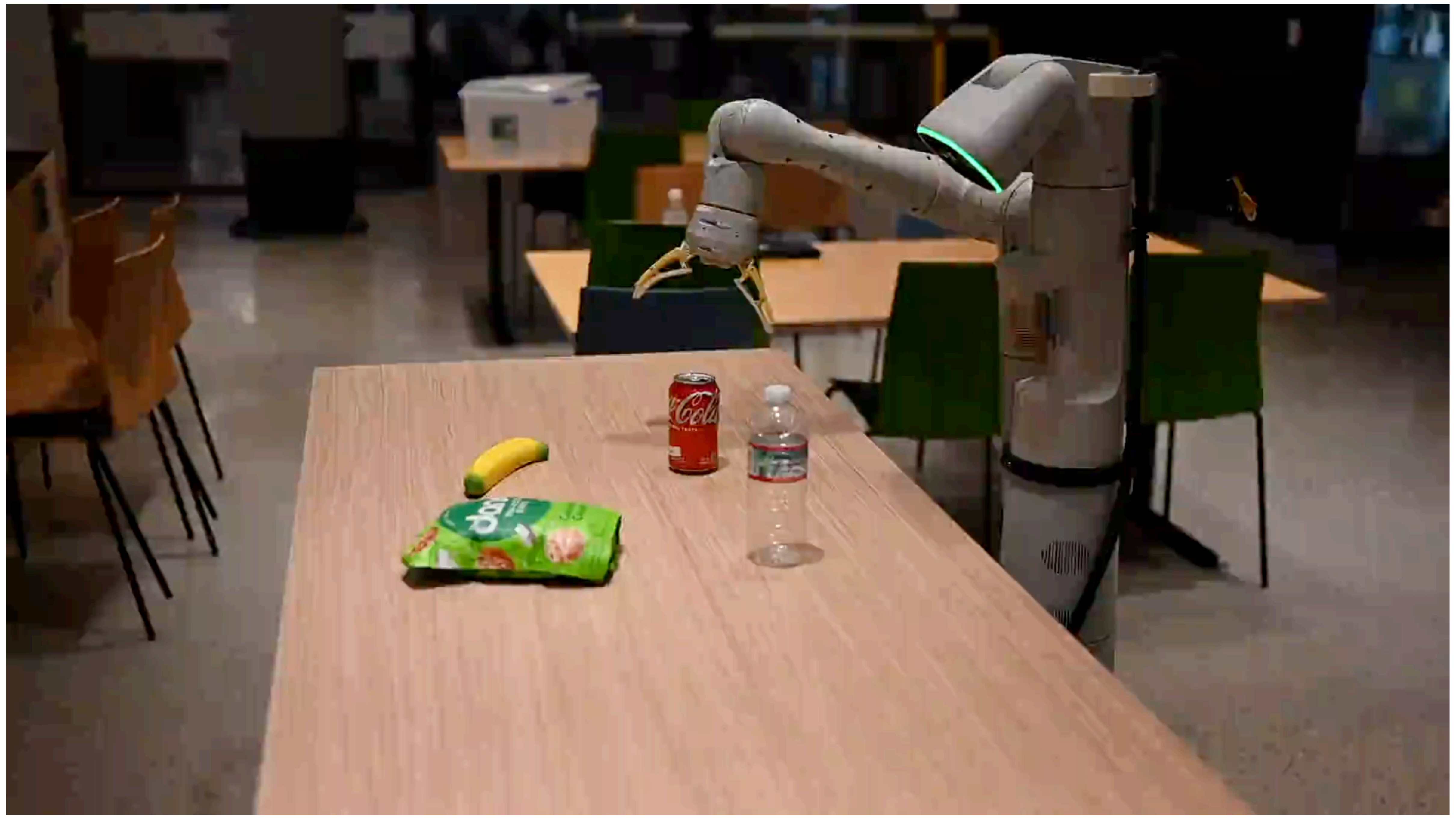
Human-Robot Collaboration in Industry



https://www.therobotreport.com/manufacturing/ria-osha-robot-safety/

Tomorrow

Black-box Data-Driven Control Policies



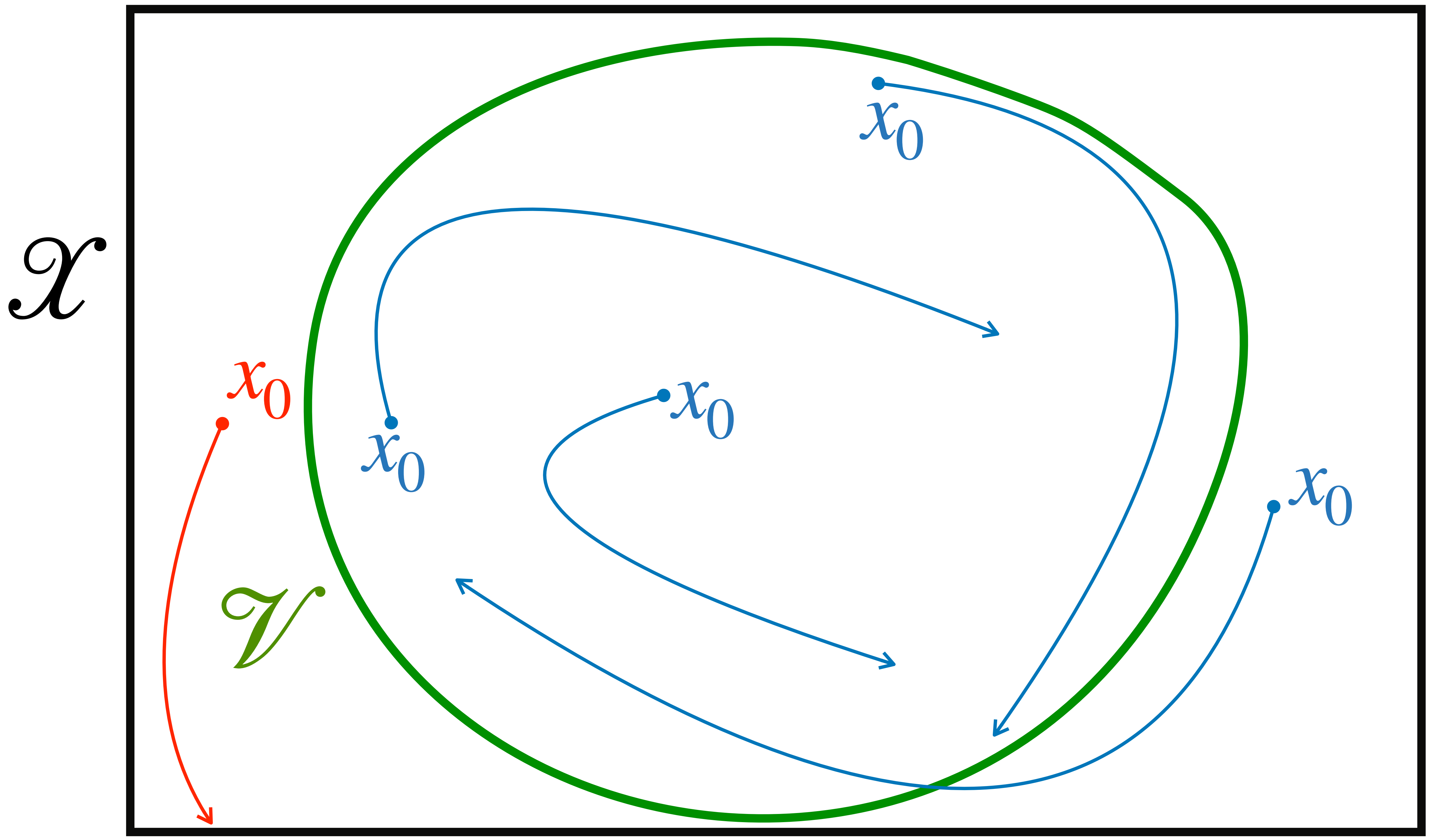
Zitkovich, Brianna, et al. "Rt-2: Vision-language-action models transfer web knowledge to robotic control." Conference on Robot Learning. PMLR, 2023.

Control Invariant Sets

Constrained discrete-time dynamical system:

$$x_{i+1} = f(x_i, u_i) \qquad x \in \mathscr{X}, \quad u \in \mathscr{U}$$

$\mathscr{V}$ is a control invariant set $\longleftrightarrow$ Once x is in $\mathscr{V}$, it can remain in $\mathscr{V}$



Recursive Feasibility

Model Predictive Control (MPC)

Using a CIS $\mathscr{V}$ as terminal set ensures recursive feasibility in MPC

$$
\begin{aligned}
\underset{\{x_i\}_0^N, \{u_i\}_0^{N-1}}{\text{minimize}} \quad & \sum_{i=0}^{N-1} \ell_i(x_i, u_i) + \ell_N(x_N) \\
\text{subject to} \quad & x_0 = x_{init} \\
& x_{i+1} = f(x_i, u_i) \qquad i = 0 \ldots N-1 \\
& x_i \in \mathcal{X}, u_i \in \mathcal{U} \qquad i = 0 \ldots N-1 \\
& \boxed{x_N \in \hat{\mathcal{V}}}
\end{aligned}
$$

What if the terminal set is an approximation of a CIS $\hat{\mathscr{V}} \approx \mathscr{V}$?

→

MPC problem can become unfeasible using $\hat{\mathscr{V}}$ instead of $\mathscr{V}$!

Idea #1: Safe Abort

Ensuring Safety

- Assume $\hat{\mathscr{V}} \subseteq \mathscr{V}$ = N-step backward reachable set of equilibrium states
  - → Even if $\hat{\mathscr{V}}$ is not a CIS, any state in $\hat{\mathscr{V}}$ is “safe”
- **Safe Abort:**
  - If MPC problem becomes unfeasible
  - Find (and follow) trajectory that:
    - starts from last predicted state in $\hat{\mathscr{V}}$
    - reaches an equilibrium state
- Such a trajectory is guaranteed to exist

Nice! This ensures SAFETY.

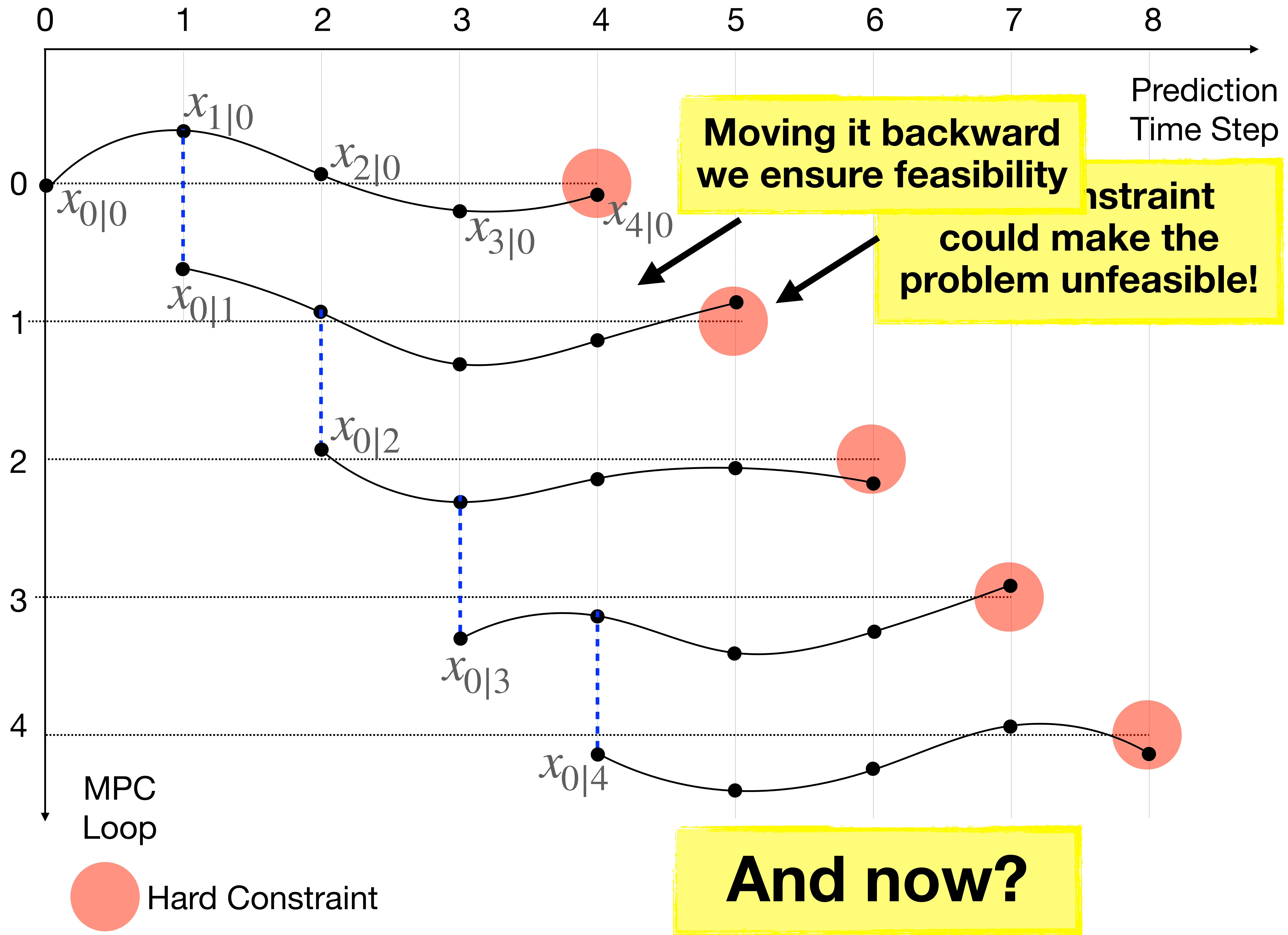
Can we also ensure
RECURSIVE FEASIBILITY?

Idea #2: Receding Constraint

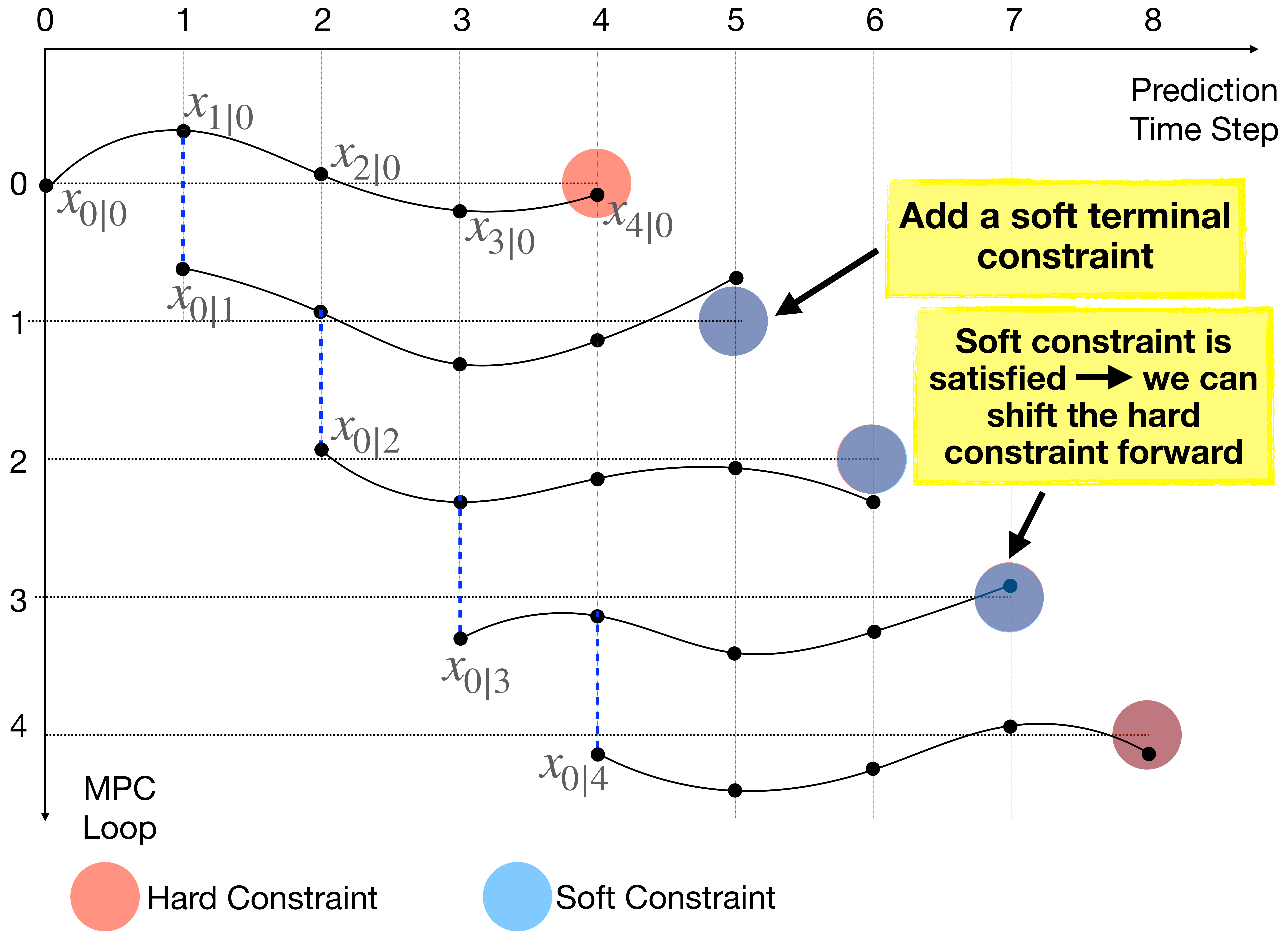
Ensuring Recursive Feasibility

- **Observation**
  - Having the terminal state in $\hat{\mathscr{V}}$ is not necessary to ensure safety
  - Having any future state in $\hat{\mathscr{V}}$ would be sufficient

- **Idea**
  - Adapt online the time step for which we constrain the state in $\hat{\mathscr{V}}$



0
1
2
3
4
5
6
7
8
Prediction
Time Step
0
1
2
3
4
MPC
Loop
$x_{0|0}$
$x_{1|0}$
$x_{2|0}$
$x_{3|0}$
$x_{4|0}$
$x_{0|1}$
$x_{0|2}$
$x_{0|3}$
$x_{0|4}$
Moving it backward
we ensure feasibility
nstraint
could make the
problem unfeasible!
And now?
Hard Constraint



0
1
2
3
4
5
6
7
8
Prediction Time Step
MPC Loop
$x_{0|0}$
$x_{1|0}$
$x_{2|0}$
$x_{3|0}$
$x_{4|0}$
$x_{0|1}$
$x_{0|2}$
$x_{0|3}$
$x_{0|4}$
Add a soft terminal constraint
Soft constraint is satisfied → we can shift the hard constraint forward
Hard Constraint
Soft Constraint

Simulation Results

- Comparing 5 MPC formulations
- 3 DoF robot manipulator
- Acados software library
- Setpoint regulation task
- 100 simulations from random initial joint configurations
- Horizon N=35 to ensure computation time < dt (5 ms)
- https://github.com/idra-lab/safe-mpc

Results

Safety Margin 2%

MPC Formulation	# Tasks Completed	# Tasks Safely Aborted	# Tasks Failed
Naive	69	-	31
Soft Terminal	69	-	31
Soft Terminal with Abort	70	11	19
Hard Terminal with Abort	70	8	22
Receding Constraint	**77**	**18**	**5**

Can we do better?

Results

Safety Margin 10%

MPC Formulation	# Tasks Completed	# Tasks Safely Aborted	# Tasks Failed
Naive	69	-	31
Soft Terminal	69	-	31
Soft Terminal with Abort	70	22	8
Hard Terminal with Abort	70	21	9
Receding Constraint	**77**	**20**	**3**

Cost & Computation Time

Safety Margin 10%

MPC Formulation	Cost Increase	Computation Times (99-Percentile)	
		MPC [ms]	Safe Abort [ms]
Naive	0%	3.8	-
Soft Terminal	0.05%	5.5	-
Soft Terminal with Abort	0.04%	3.7	130
Hard Terminal with Abort	0.04%	3.9	100
Receding Constraint	**0.02%**	**3.9**	**80**

Conclusions

- Novel MPC formulation ensuring
  - Recursive feasibility under weaker conditions (N-Step CIS)
  - Safety under even weaker conditions (inner approx. of CIS)

On-going/future work

- Learn safe-abort policy to warm-start safe-abort OCP solver
- Hardware implementation
- Computation/certification of N-Step CIS and inner approx. of CIS
- Handle dynamics uncertainties/obstacles
- Application as safety filter for RL policies

Take-Home Message

Globally Optimal and Safe Robot Control

- Using ideas from TO we can make RL efficient and safe
  - Use dynamics derivatives to guide RL exploration (CACTO)
  - Use Control Invariance to make control (RL) safe

Current challenges

- algorithms to compute $\hat{\mathscr{V}}$ do not scale and cannot certify set properties (e.g. N-Step Control Invariance)
- dynamics derivatives are ill-defined in contact-rich tasks

Safe and Efficient Robot Control

Combining learning and trajectory optimization

[1] Grandesso, Alboni, Papini, Wensing, Del Prete (2023). CACTO: Continuous Actor-Critic With Trajectory Optimization - Towards Global Optimality. IEEE Robotics and Automation Letters (RAL)

[2] Alboni, Grandesso, Rosati Papini, Carpentier, Del Prete (2024). CACTO-SL: Using Sobolev Learning to improve Continuous Actor-Critic with Trajectory Optimization. In Learning for Dynamics and Control Conference (L4DC)

[3] Lunardi, La Rocca, Saveriano, Del Prete, (2024). Receding-Constraint Model Predictive Control using a Learned Approximate Control-Invariant Set. In IEEE International Conference on Robotics and Automation (ICRA)

[4] La Rocca, Saveriano, Del Prete (2023). VBOC: Learning the Viability Boundary of a Robot Manipulator using Optimal Control. IEEE Robotics and Automation Letters (RAL)

Andrea Del Prete

UNIVERSITY OF TRENTO