A Physics-Informed Machine Learning Framework for Safe and Optimal Control of Autonomous Systems

Manan Tayal*[1], Aditya Singh*[1], Shishir Kolathaya[1], Somil Bansal[2]

1 INDIAN INSTITUTE OF SCIENCE
भारतीय विज्ञान संस्थान

2 Stanford University

ICML International Conference On Machine Learning 2025

ICML
International Conference
On Machine Learning

Motivation

Autonomous systems are becoming increasingly prevalent across various domains.

Problem ⇢ maintain an optimal balance between **Performance** and **Safety.**

E.g., autonomous delivery vehicle navigating a cluttered env., it is expected to,

- ➢ Safety: Avoid obstacles
- ➢ Performance: Reach target in time

Challenge: Existing methods lack scalability & guarantees

State Constrained Optimal Control Problem

$$\min_{\mathbf{u}} \int_t^T l(x(s))\,ds + \phi(x(T))$$

$$\text{s.t. } \dot{x} = f(x,u),$$

$$g(x(s)) \leq 0 \quad \forall s \in [t,T]$$

Running Cost

Terminal Cost

System Dynamics

State Constraint

How do we solve this SC-OCP?

Epigraph Form of the SC-OCP

We reformulate the problem in its epigraph form (Boyd & Vandenberghe, 2004), which transforms it into a two-stage optimization problem:

$$V(t, x(t)) = \min_{z \in \mathbb{R}^+} \; z$$

$$\text{s.t. } \hat{V}(t, x, z) \leq 0$$

Here, $\hat{V}$ is defined as (Altarovici et al., 2013):

$$\hat{V}(t, x(t), z) = \min_{\mathbf{u}} \max\{ \int_{s=t}^{T} l(x(s))\, ds + \phi(x(T)) - z, \; \max_{s \in [t, T]} g(x(s)) \}$$

Reference:

1. *Boyd, Stephen, and Lieven Vandenberghe. Convex Optimization. Cambridge University Press, 2004.*

2. *Altarovici, A., Bokanowski, O., and Zidani, H. A general hamilton-jacobi framework for non-linear state-constrained control problems.*
ESAIM: Control, Optimisation and Calculus of Variations, 19(2):337–357, 2013.

Learning the Auxiliary Value Function

HJB-PDE

$$\min\Big(-\partial_t \hat{V} - \min\langle \nabla_{\hat{x}} \hat{V}(t,\hat{x}), \hat{f}(\hat{x},u)\rangle, \hat{V} - g(x)\Big) = 0,$$

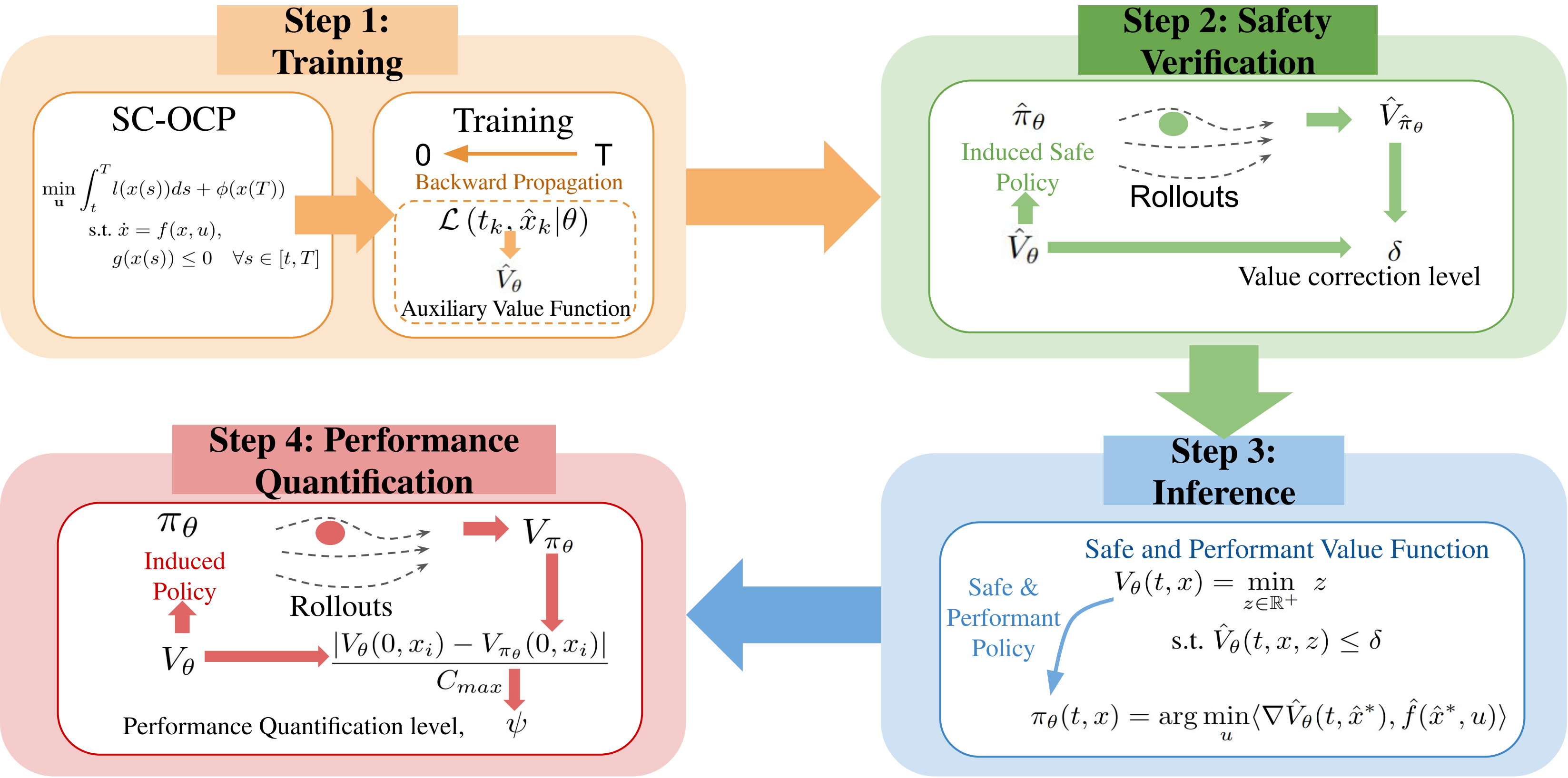
$$\forall t \in [0, T) \text{ and } \hat{x} \in \mathcal{X} \times \mathbb{R}.$$

Boundary Conditions

$$\hat{V}(T,\hat{x}) = \max\left(\phi(x(T)) - z, g(x)\right), \quad \hat{x} \in \mathcal{X} \times \mathbb{R}$$

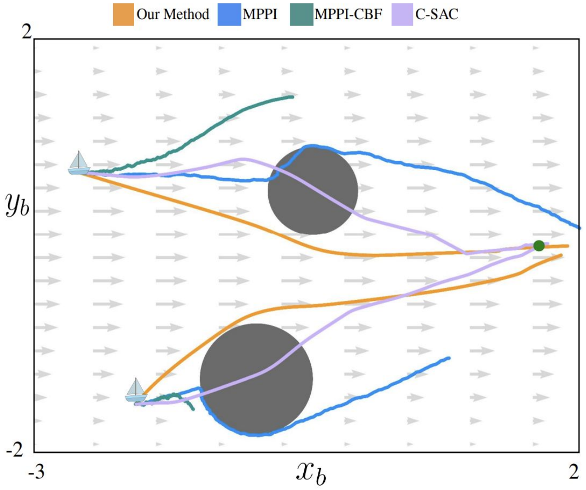
Curriculum Learning Approach → $\hat{V}_\theta$

Algorithm

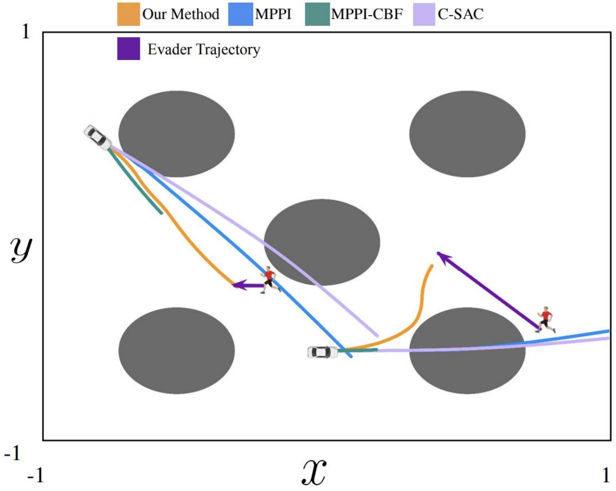


Step 1: Training
SC-OCP
$\min_{\mathbf{u}} \int_t^T l(x(s))ds + \phi(x(T))$
s.t. $\dot{x} = f(x, u)$,
$g(x(s)) \leq 0 \quad \forall s \in [t, T]$
Training
0
T
Backward Propagation
$\mathcal{L}\left(t_k, \hat{x}_k | \theta\right)$
$\hat{V}_\theta$
Auxiliary Value Function
Step 2: Safety Verification
$\hat{\pi}_\theta$
Induced Safe Policy
Rollouts
$\hat{V}_{\hat{\pi}_\theta}$
$\hat{V}_\theta$
δ
Value correction level
Step 3: Inference
Safe and Performant Value Function
Safe & Performant Policy
$V_\theta(t, x) = \min_{z \in \mathbb{R}^+} z$
s.t. $\hat{V}_\theta(t, x, z) \leq \delta$
$\pi_\theta(t, x) = \arg\min_u \langle \nabla \hat{V}_\theta(t, \hat{x}^*), \hat{f}(\hat{x}^*, u) \rangle$
Step 4: Performance Quantification
π_θ
Induced Policy
Rollouts
V_{π_θ}
V_θ
$\frac{|V_\theta(0, x_i) - V_{\pi_\theta}(0, x_i)|}{C_{max}}$
Performance Quantification level, ψ

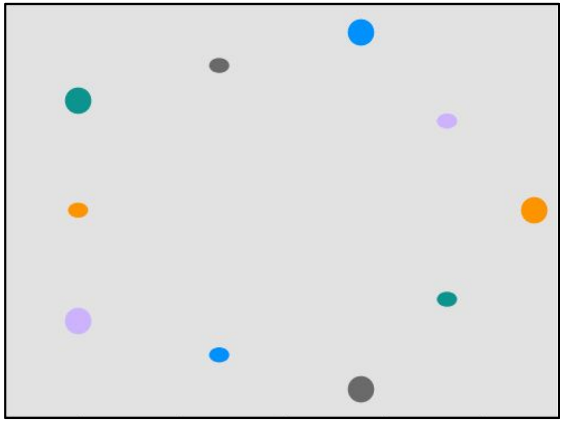
Experiments



Boat Navigation
(2D)



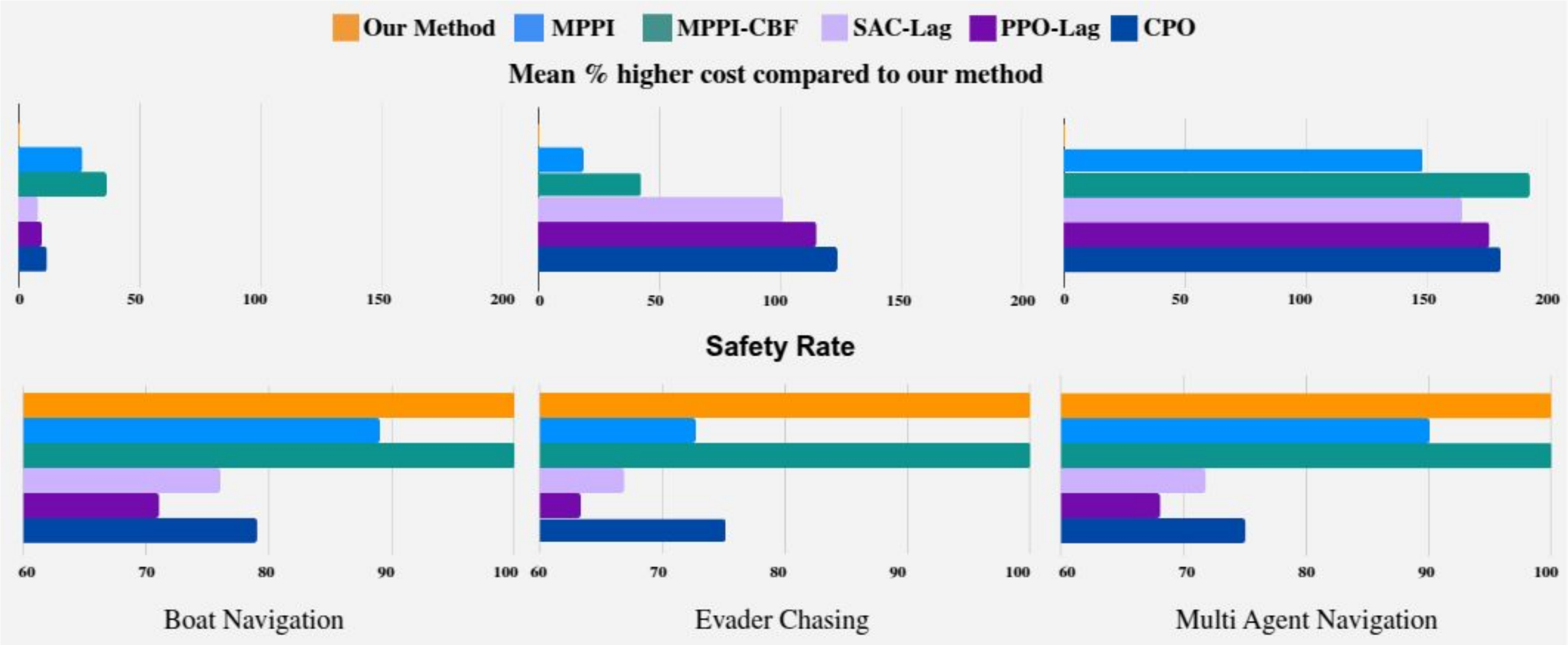
Pursuer Tracking an Evader
(8D)



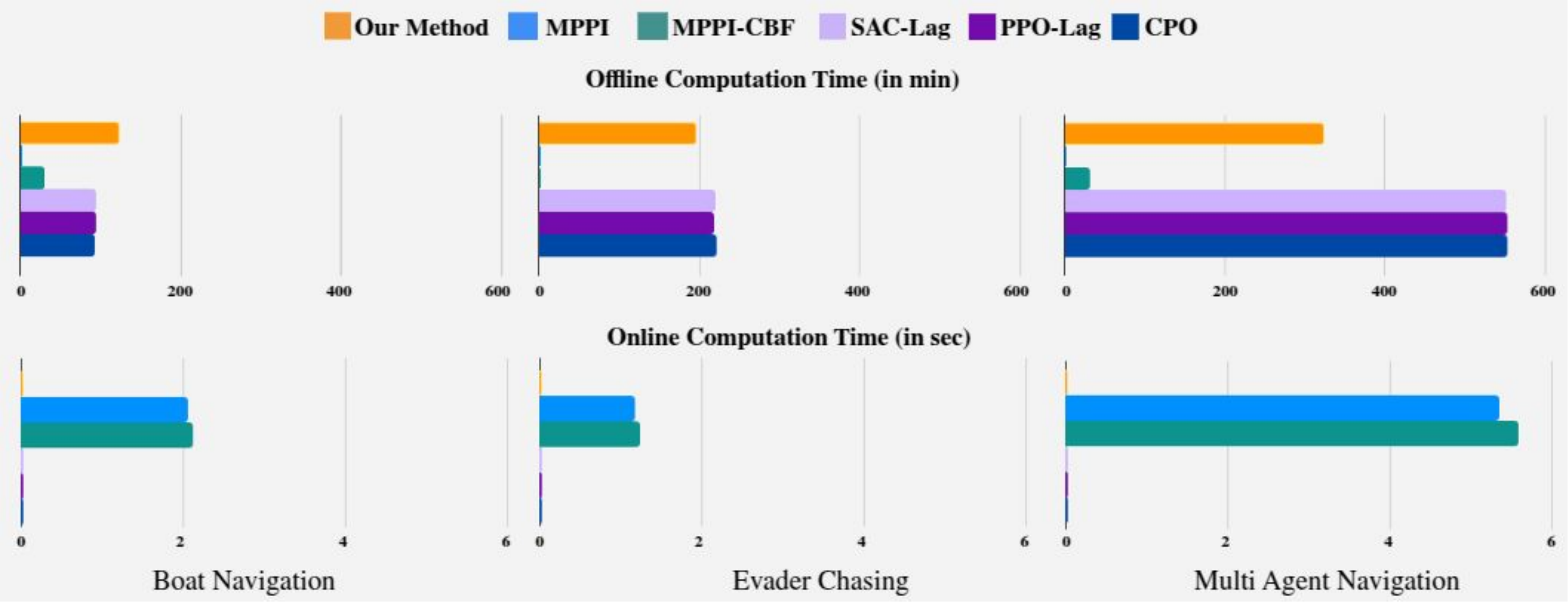
Multi-Agent Navigation
(20D)

International Conference On Machine Learning

Baseline Comparisons



Baseline Comparisons



Thank You

Paper

https://arxiv.org/pdf/2502.11057

Webpage

https://tayalmanan28.github.io/piml-soc/