Learning how to race using a predictive control approach: towards multi-agents racing

Ugo Rosolia

AMBER Lab
California Institute of Technology

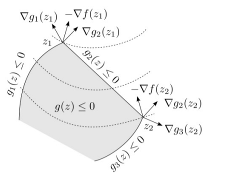
May 31st, 2021


Formula 1
ROLEX

Control design for autonomous racing

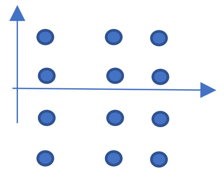
Compute the racing line

KKT Conditions



Optimization

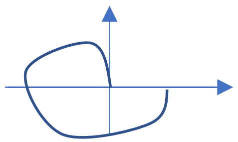
Bellman Recursion



Dynamic Programming

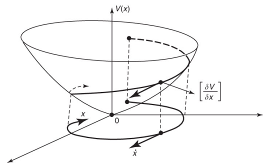
Control Theory

Nyquist plot



Frequency Domain

Lyapunov Function



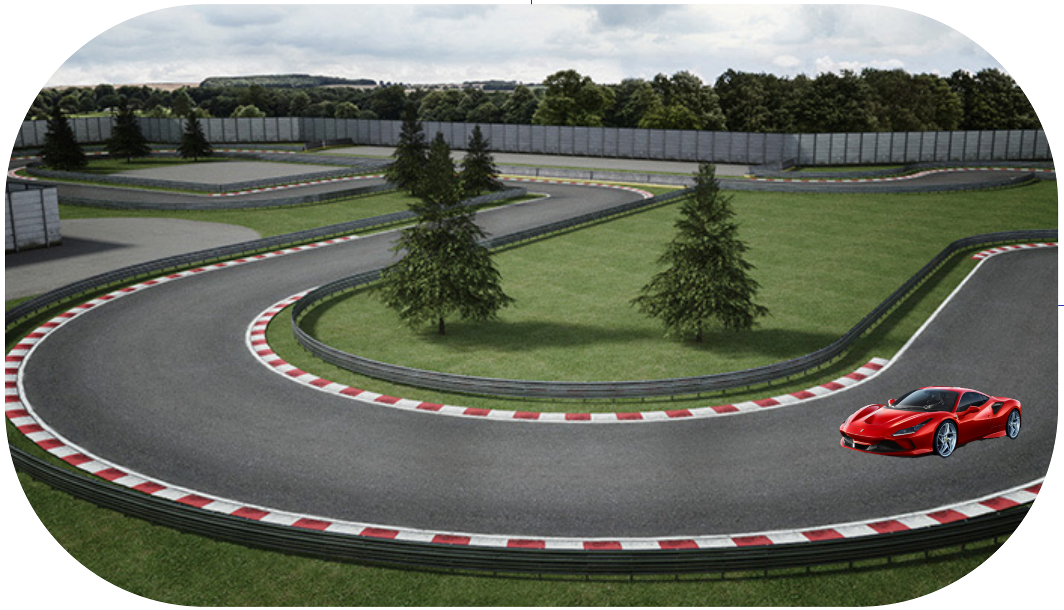
Nonlinear Control

Numerical Methods

$$\min \quad f(x)$$
$$\text{s.t.} \quad g(x) \leq 0$$

Predictive Control

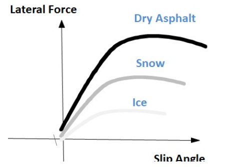
Track shape



Data

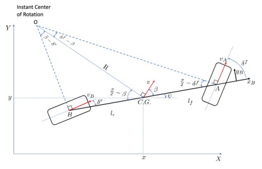
System Identification

Pajecka Formula



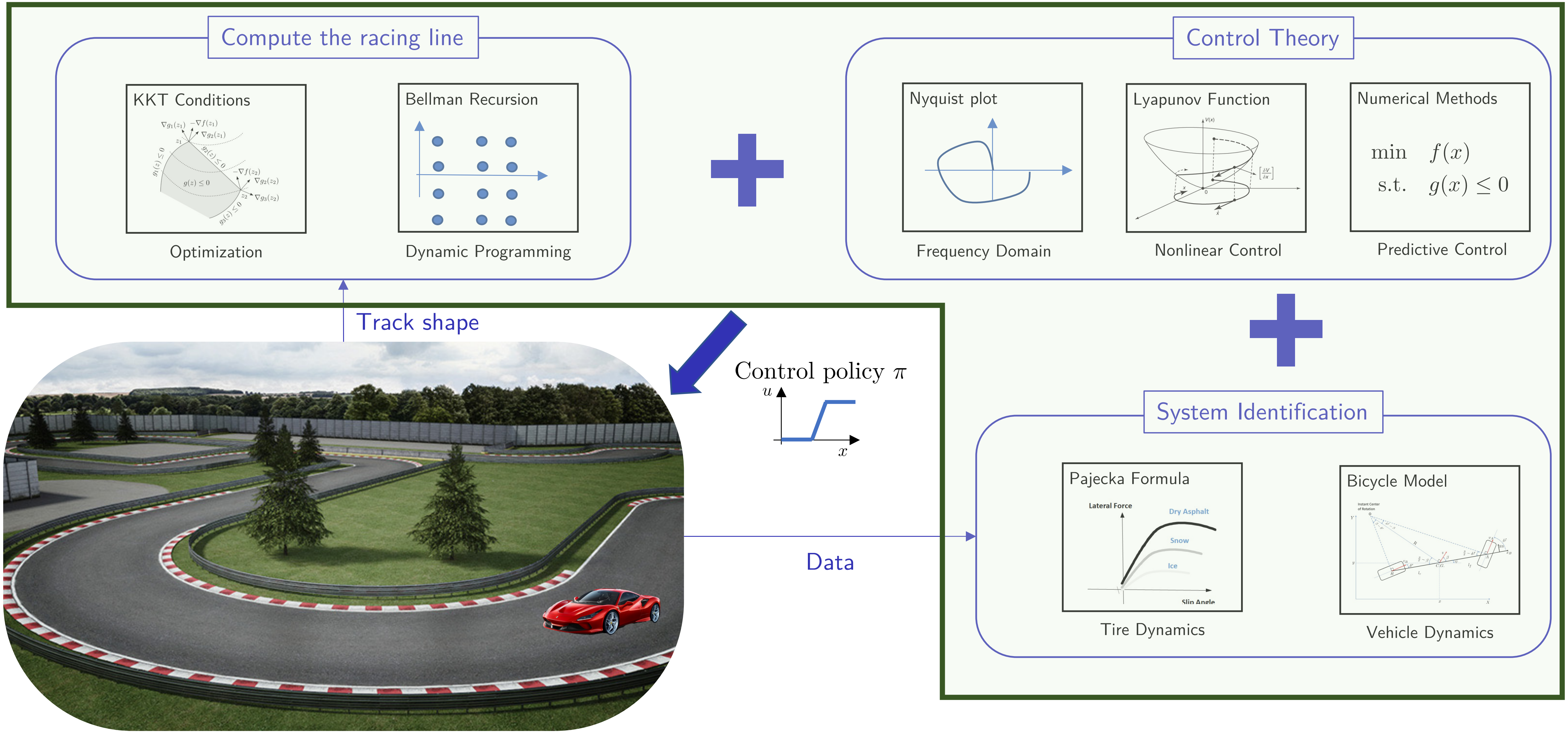
Tire Dynamics

Bicycle Model

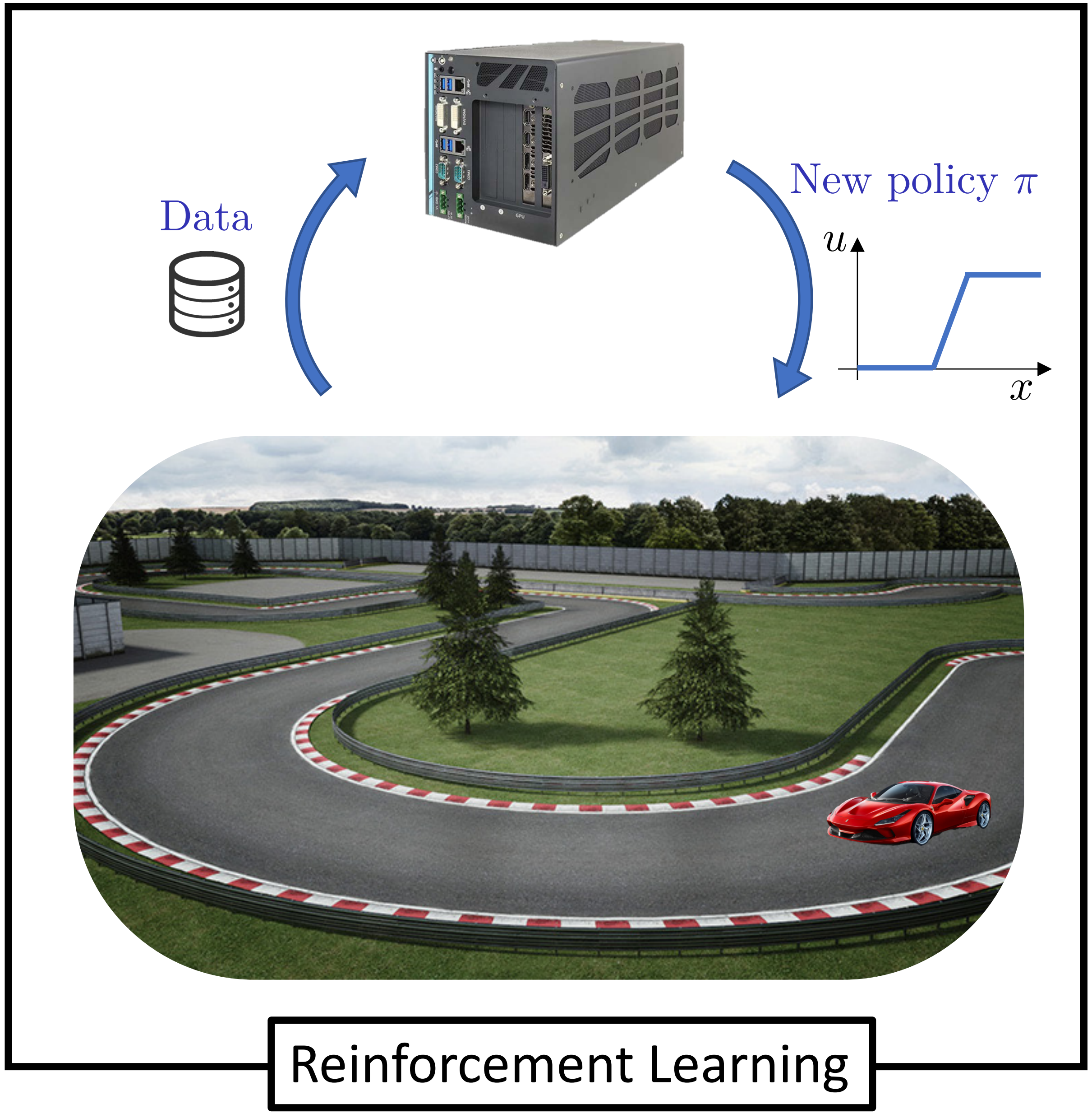


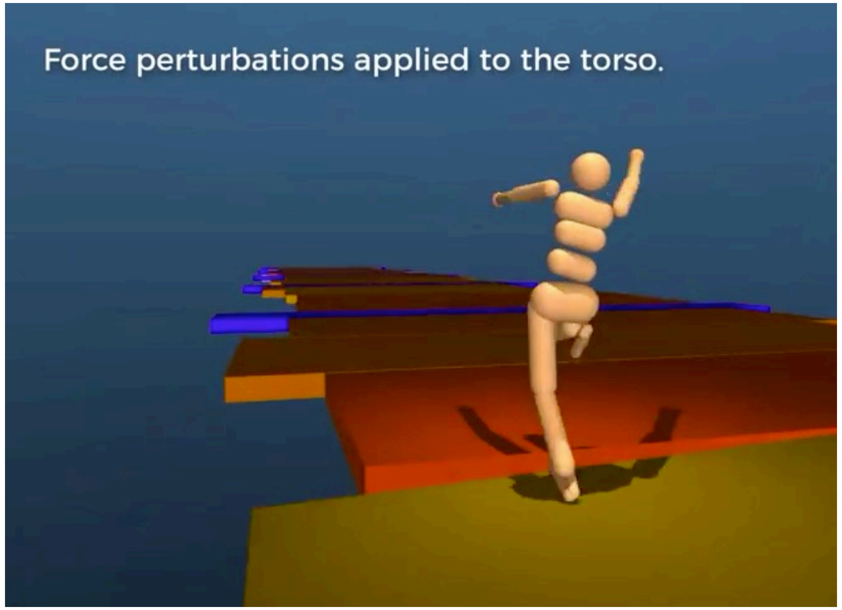
Vehicle Dynamics

Control design for autonomous racing

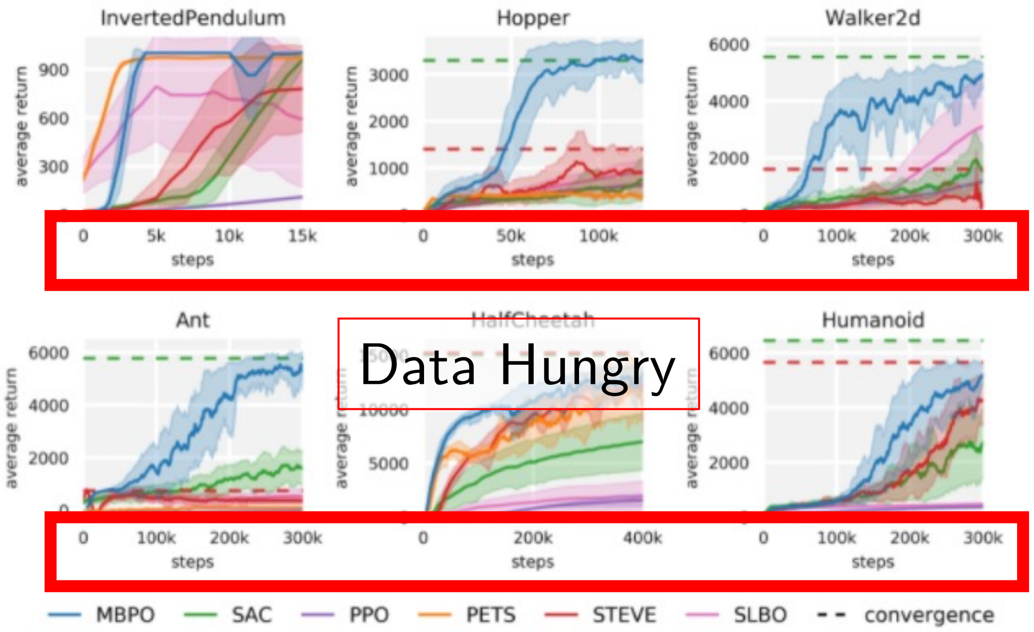


Can we simplify the control design?



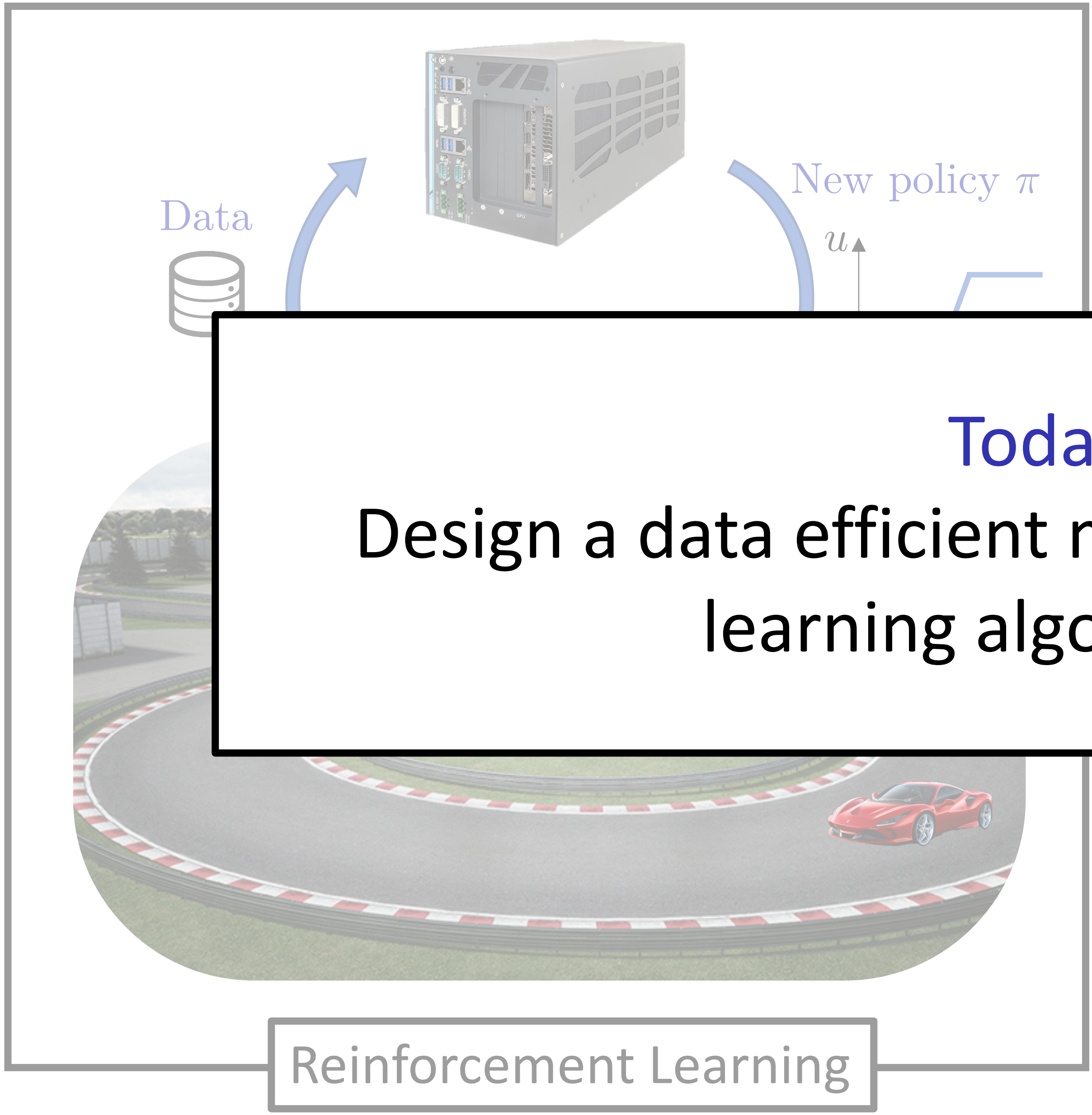


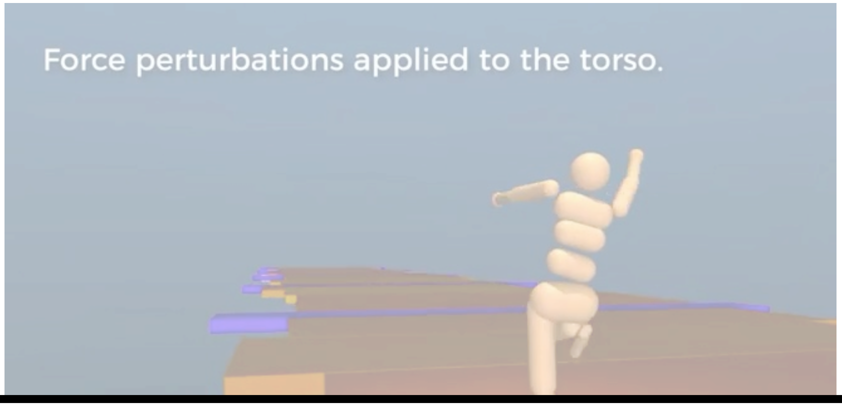
DeepMind



M. Janner, J. Fu, M. Zhang, and S. Levine. "When to trust your model: Model-based policy optimization." arXiv preprint arXiv:1906.08253 (2019).

Can we simplify the control design?

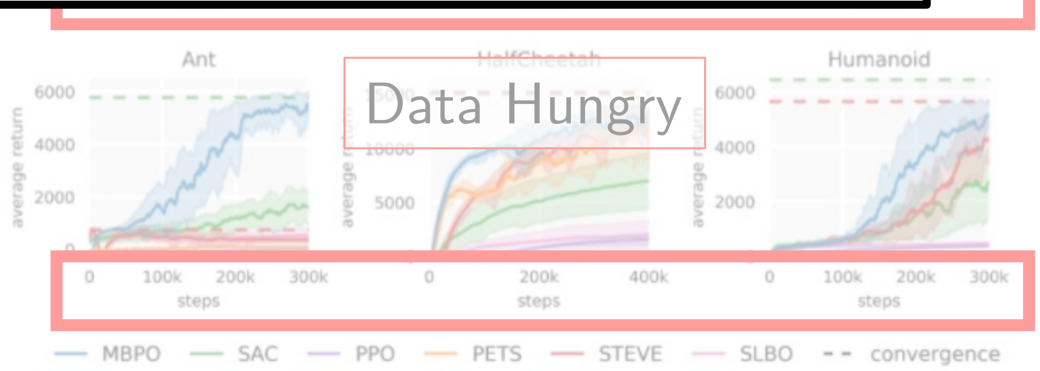




DeepMind

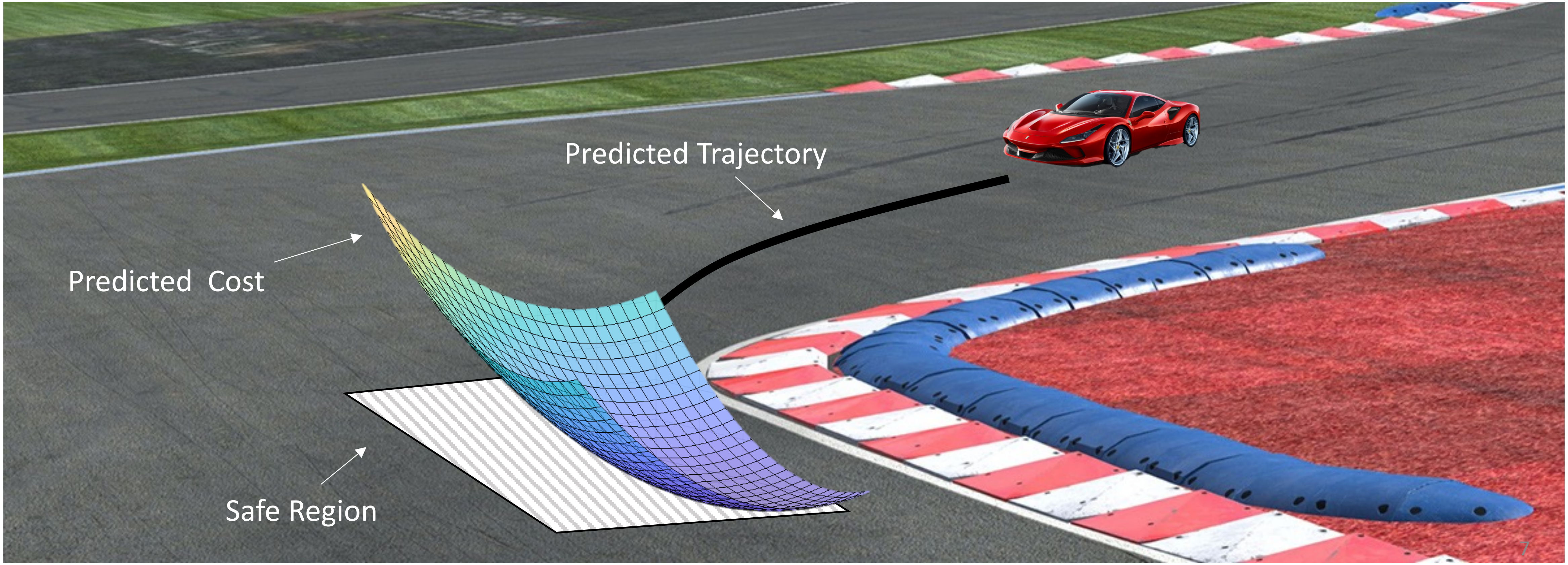
Today's goal:

Design a data efficient model-based reinforcement learning algorithm for racing



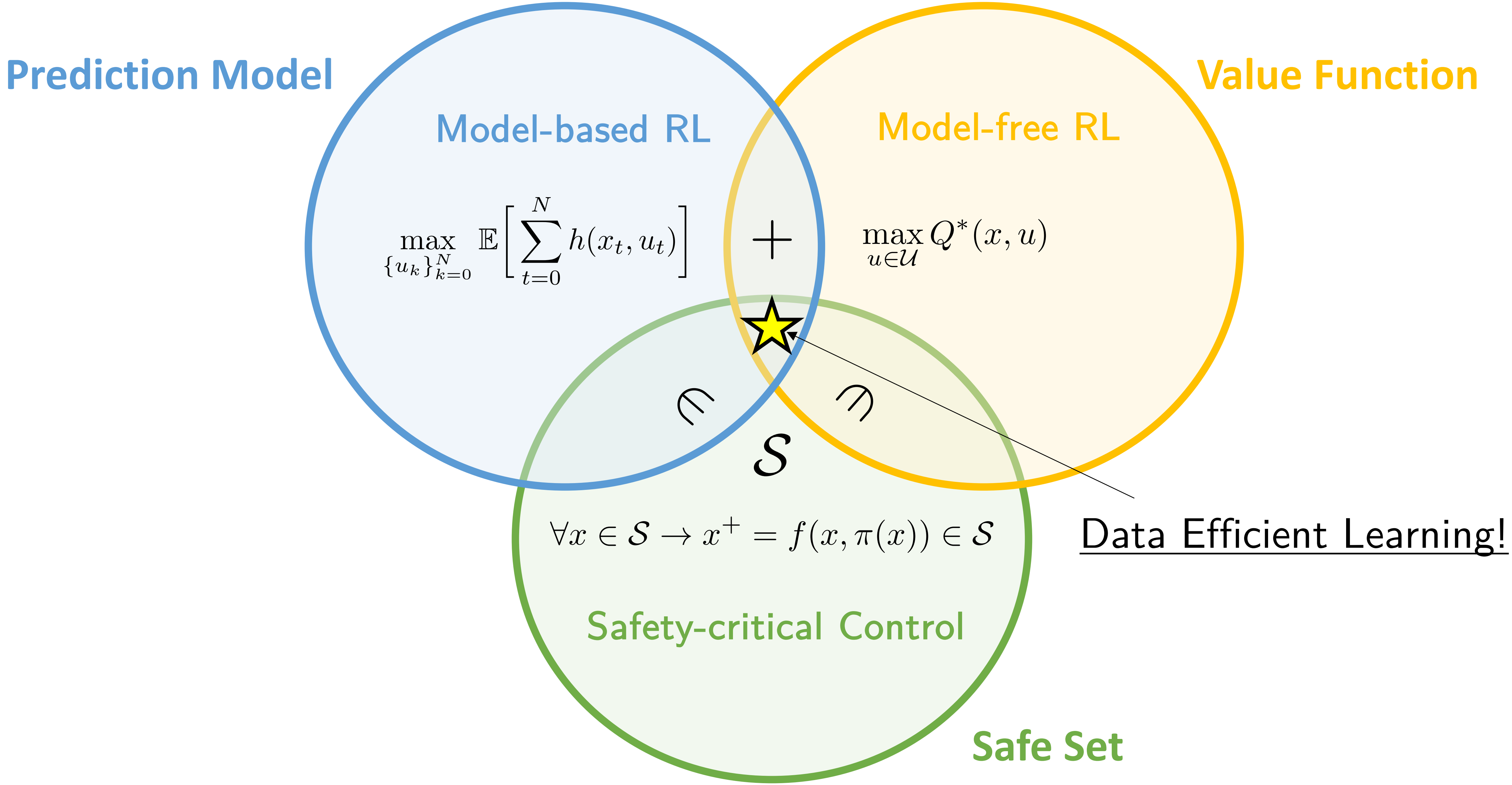
M. Janner, J. Fu, M. Zhang, and S. Levine. "When to trust your model: Model-based policy optimization." arXiv preprint arXiv:1906.08253 (2019).

How to compute control actions?



- Predicted trajectory given by Prediction Model
- Safe region estimated by the Safe Set
- Predicted cost estimated by Value Function

Three key components to learn



Problem Formulation

Minimum Time Control Problem

$$\min_{T,\mathbf{u}} \quad \boxed{T} \quad \text{Control objective}$$

$$\boxed{x_0 = x_s, \;\; x_T = \mathcal{X}_F} \quad \text{Start \& end position}$$

System dynamics
System constraints

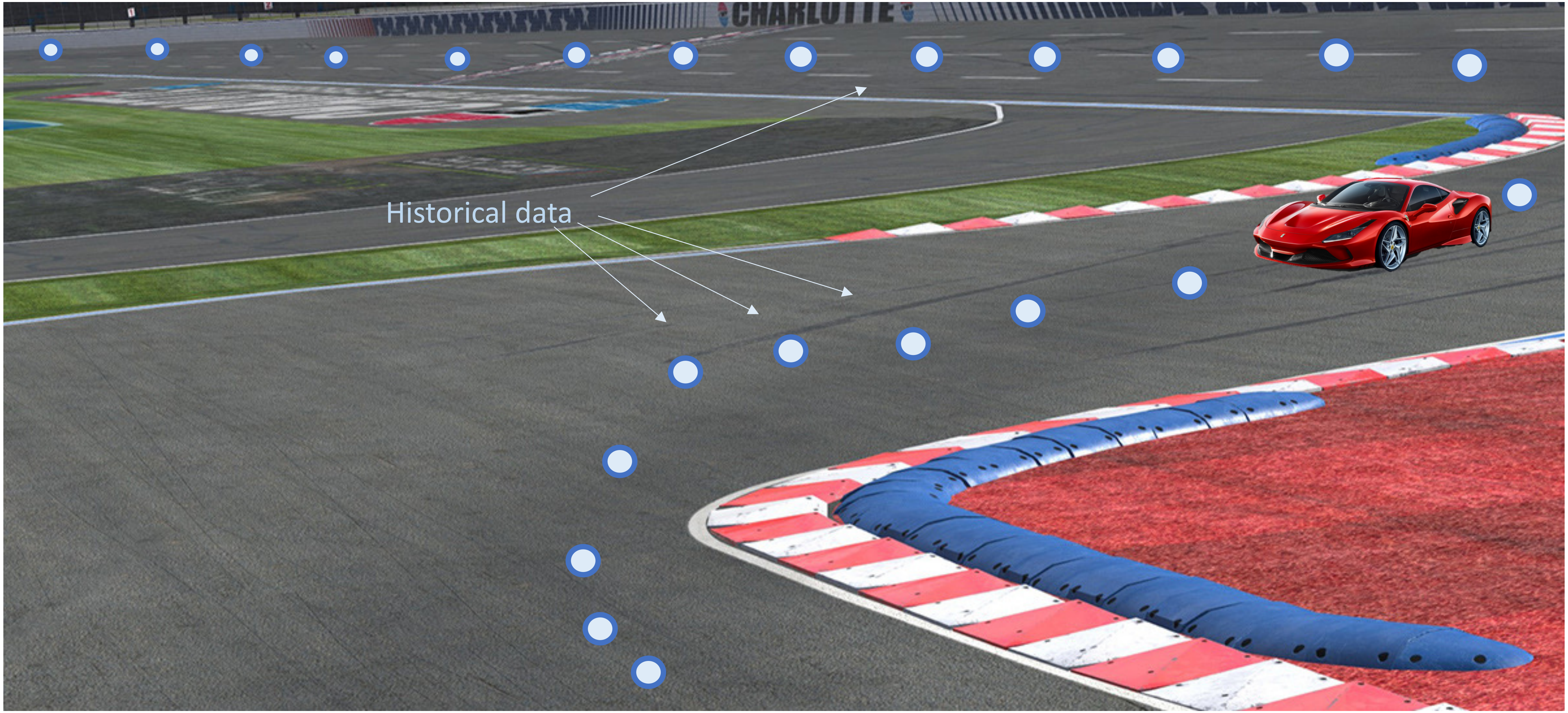
$$\boxed{x_{k+1} = f(x_k, u_k), \qquad \forall k \in \{0, \ldots, T-1\}}$$

Safety constraints

$$\boxed{x_k \in \mathcal{X}, \;\; u_k \in \mathcal{U}, \qquad \forall k \in \{0, \ldots, T-1\}}$$

Key Assumption

We are given a first feasible trajectory and/or controller



Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\text{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\ldots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \ \mathcal{CS}^{j-1}(x),$$

Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\text{LMPC},j}_{t\to t+N}(x^j_t) = \min_{u^j_{t|t},\dots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \mathcal{CS}^{j-1}(x),$$

Prediction Model

Value Function

Safe Set

Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\text{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\ldots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

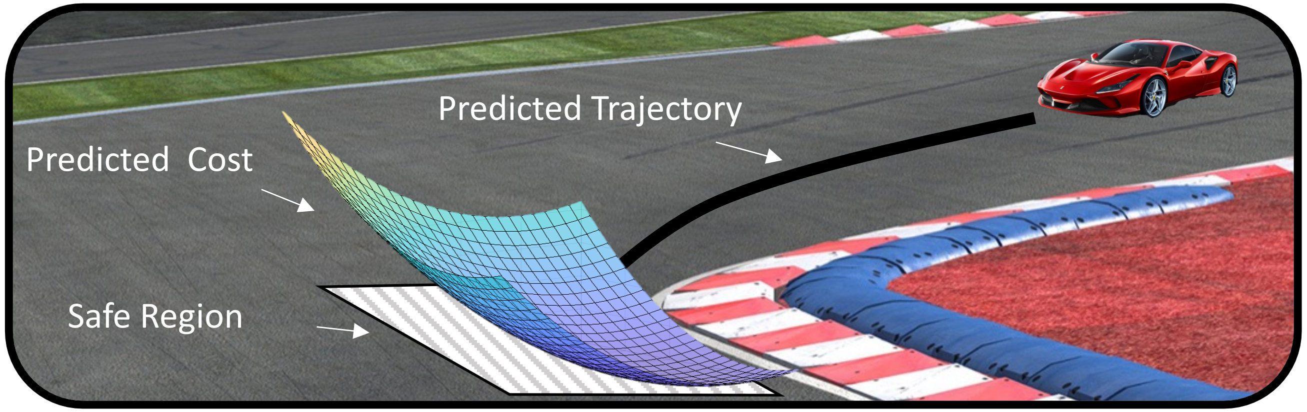
$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \mathcal{CS}^{j-1}(x),$$

Value Function

Prediction Model

Safe Set



Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\text{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\ldots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

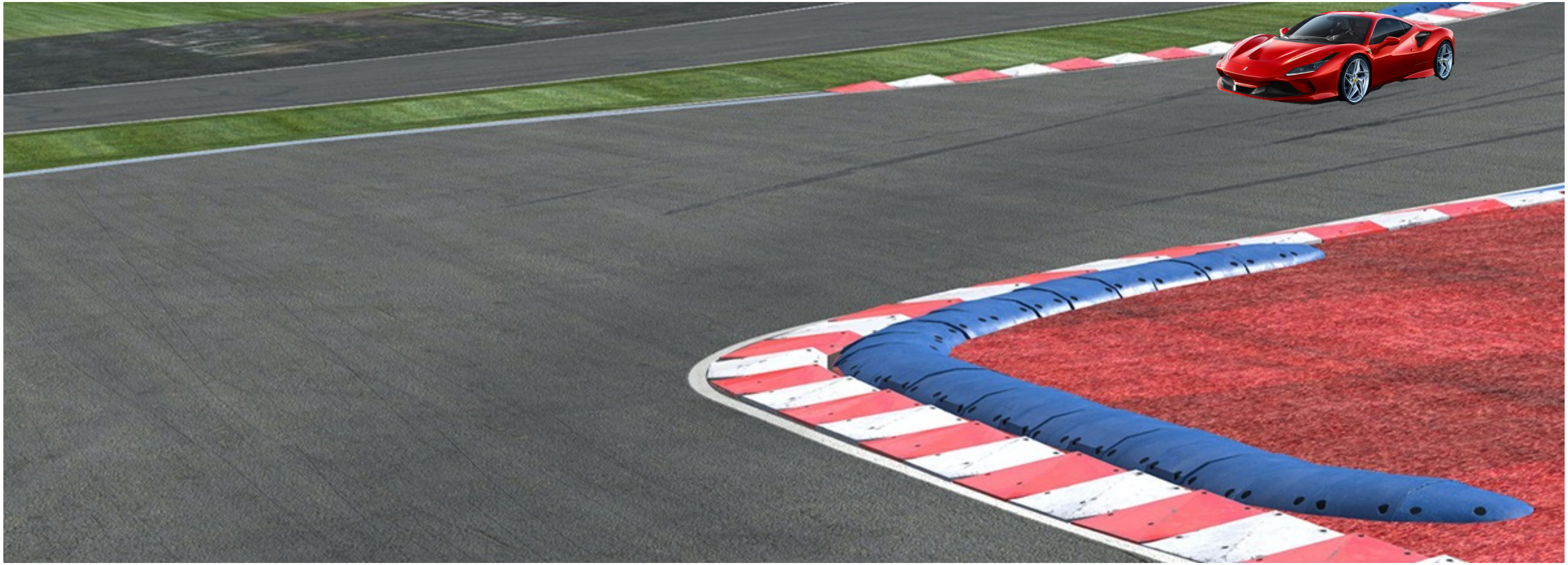
$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

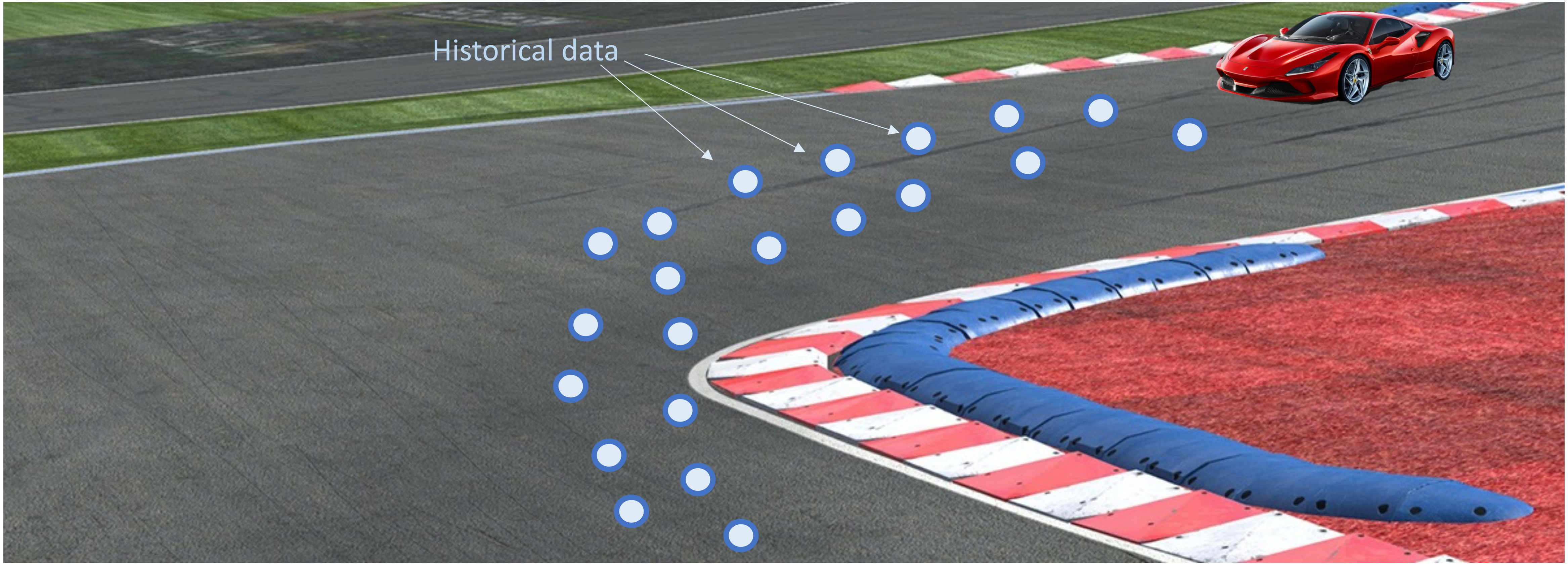
$$x^j_{t+N|t} \in \mathcal{CS}^{j-1}(x),$$

Safe Set

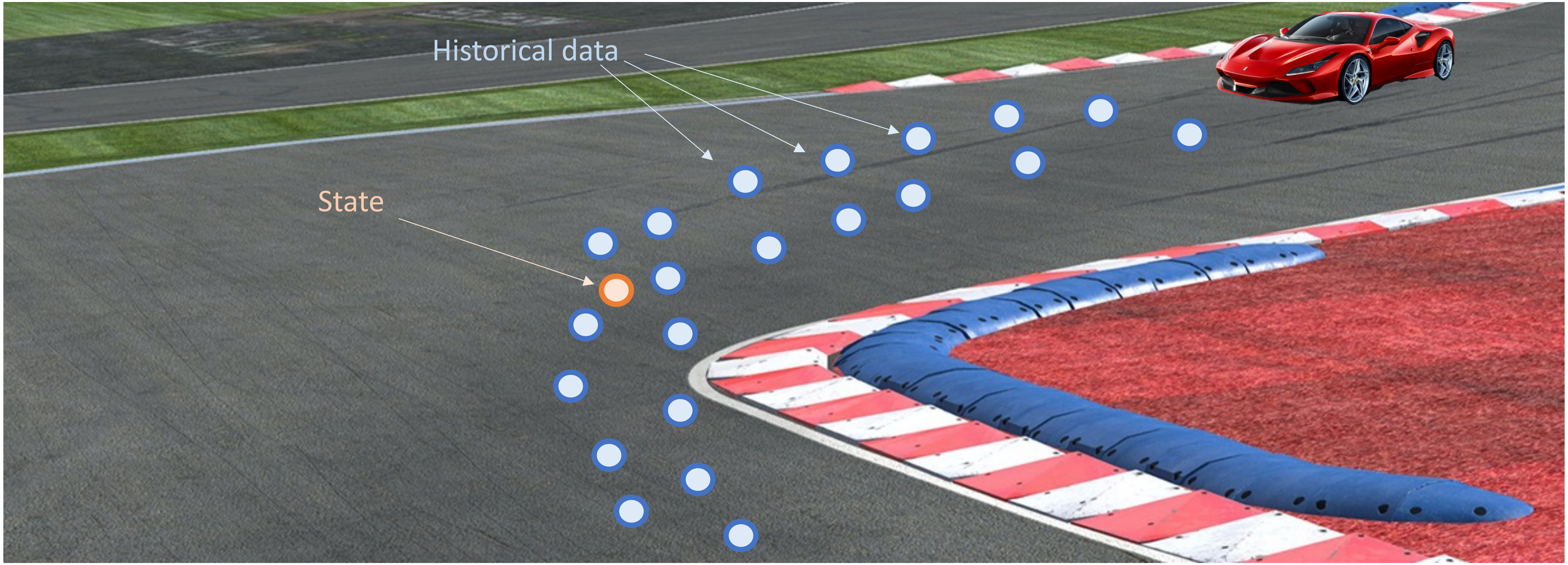
Safe Set Local Approximations



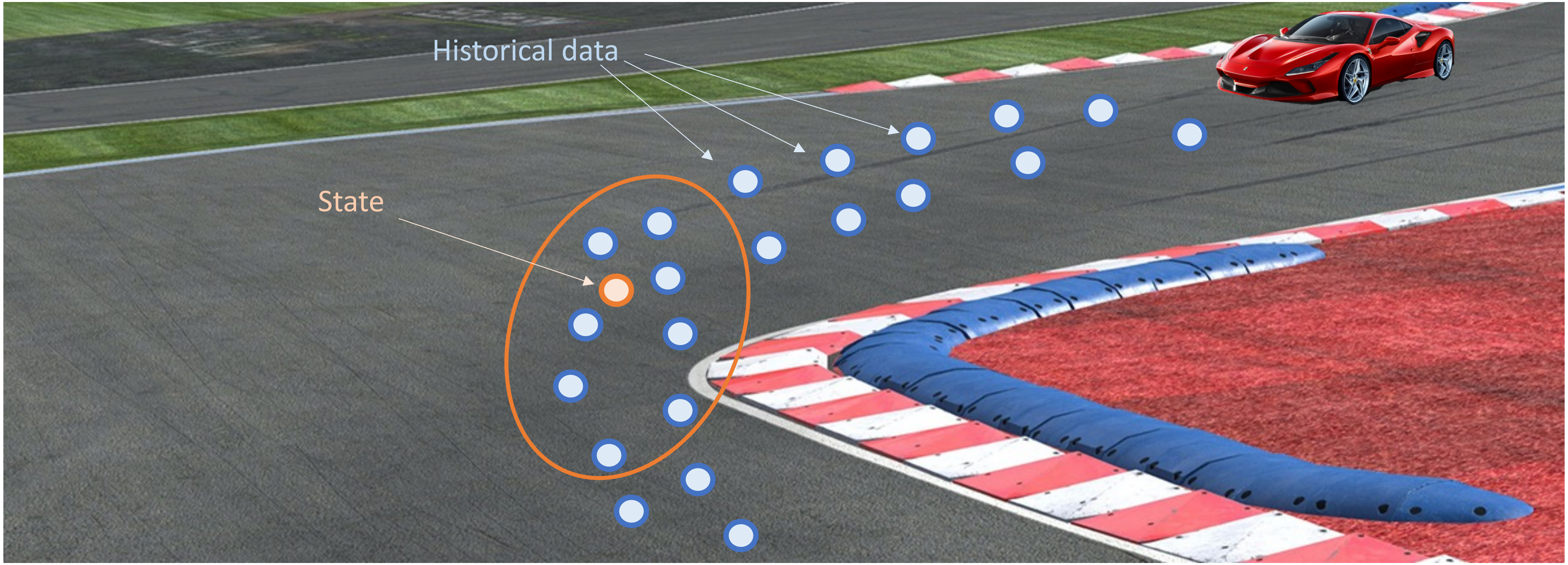
Safe Set Local Approximations



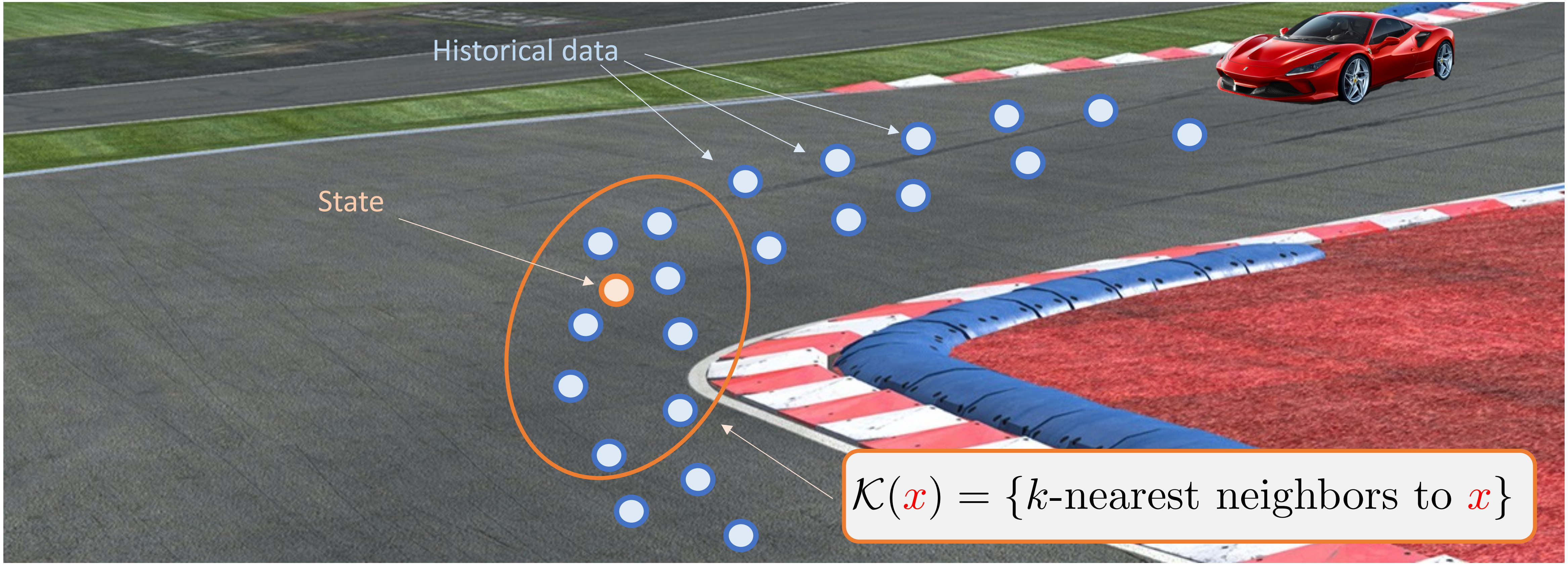
Safe Set Local Approximations



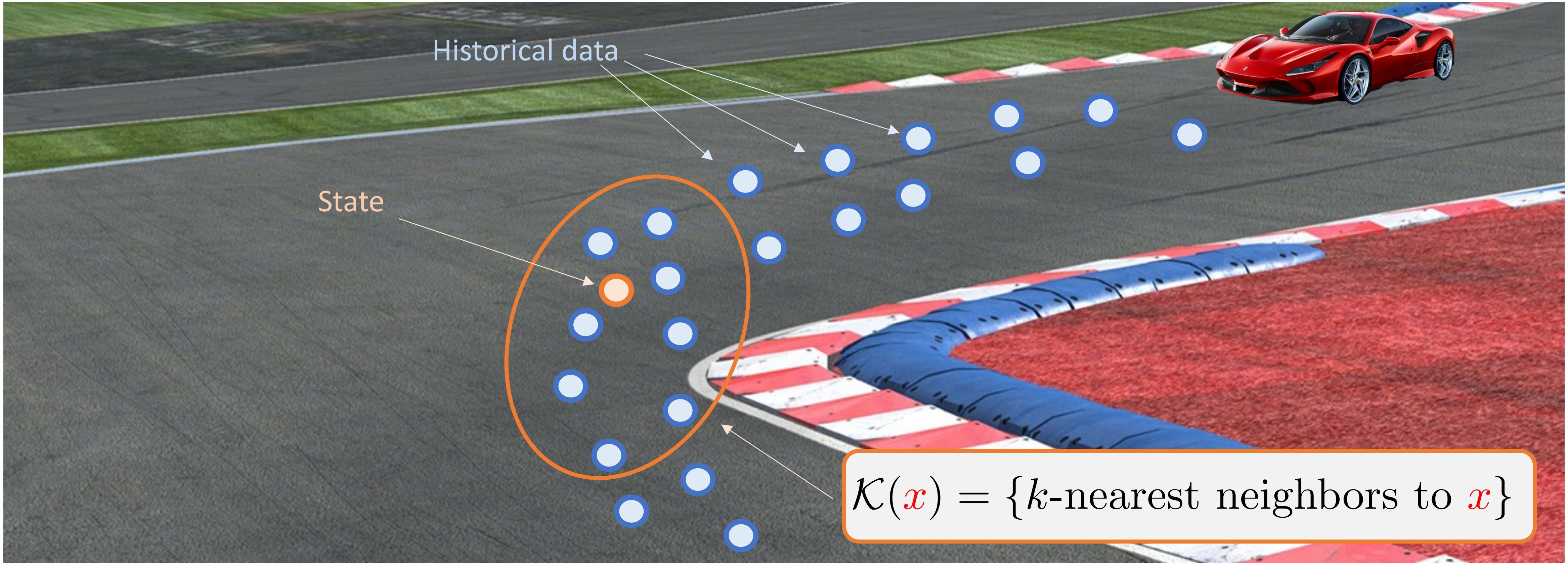
Safe Set Local Approximations



Safe Set Local Approximations



Safe Set Local Approximations



Local convex safe set approximation:

$$\mathcal{CS}^j(x) = \text{conv}\left(\cup_{x_t^j \in \mathcal{K}(x)} x_t^j \right)$$

Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\text{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\dots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \mathcal{CS}^{j-1}(x),$$

Safe Set

where $x = g$(Previous Optimal Trajectory)

Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\mathrm{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\ldots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

Value Function

s.t.

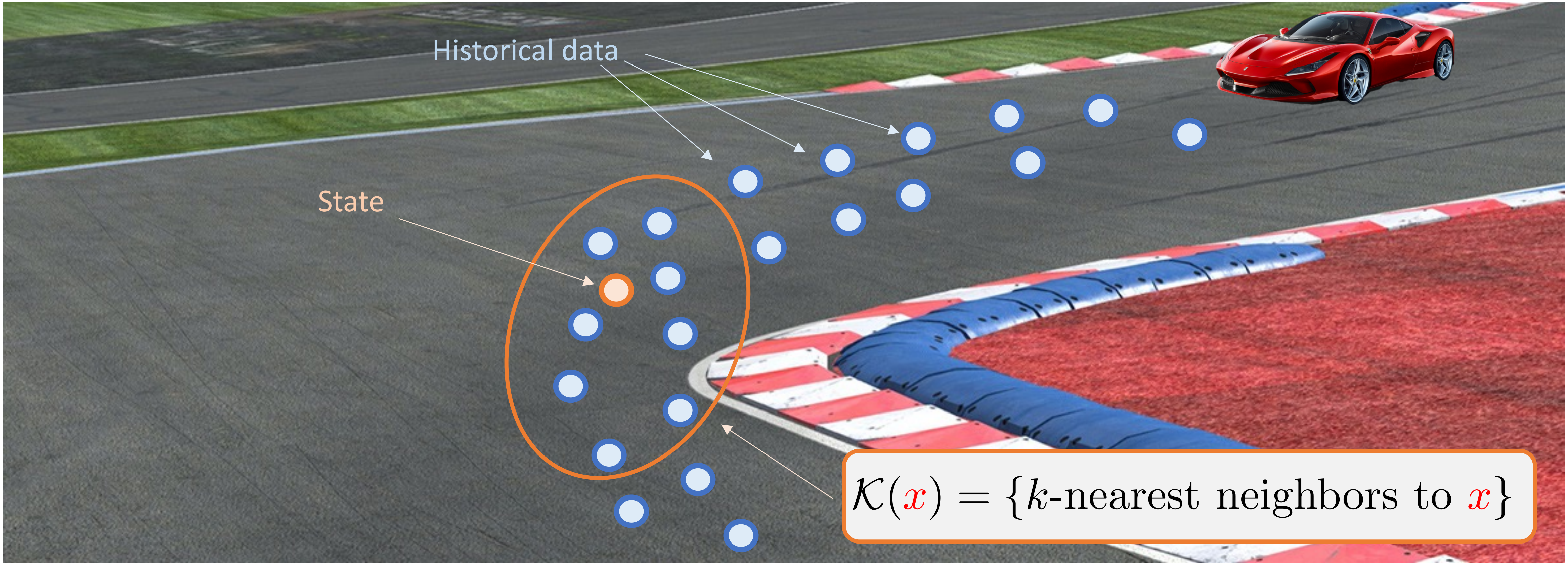
$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \ \mathcal{CS}^{j-1}(x),$$

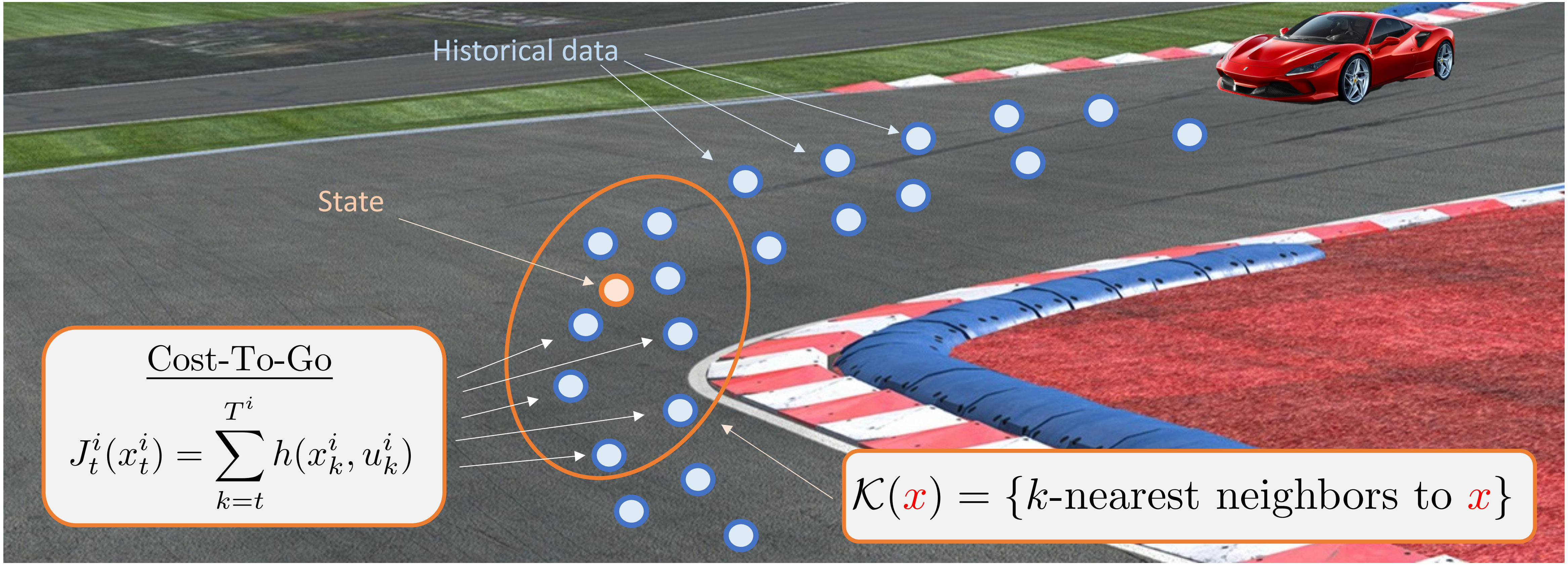
Value Function Local Approximations



Local convex safe set approximation:

$$\mathcal{CS}^j(x) = \text{conv}\left(\cup_{x_t^j \in \mathcal{K}(x)} x_t^j \right)$$

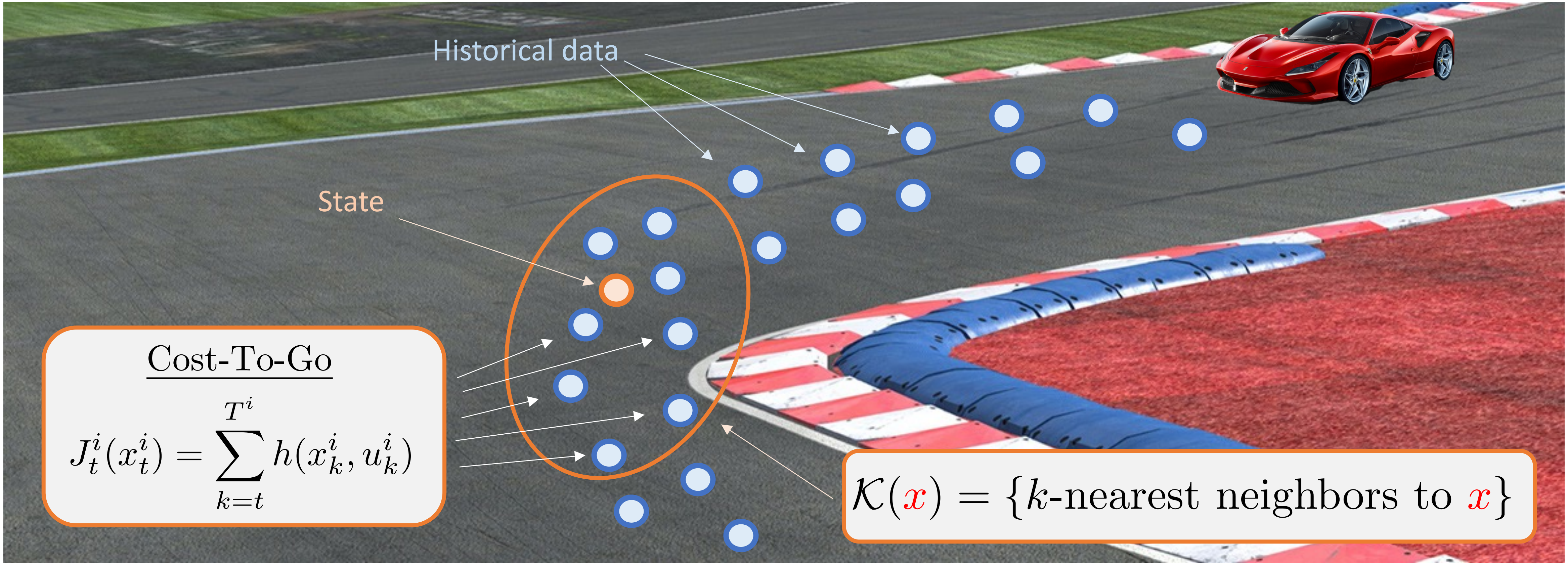
Value Function Local Approximations



Local convex safe set approximation:

$$\mathcal{CS}^j(x) = \text{conv}\Big(\cup_{x_t^j \in \mathcal{K}(x)} x_t^j \Big)$$

Value Function Local Approximations



Local value function approximation:

$$V^j(x, x) = \text{ Interpolation of the cost-to-go } J_t^i(x_t^i) = \sum_{k=t}^{T^i} h(x_k^i, u_k^i)$$

Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\mathrm{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\dots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, {\color{red}x})$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \ \mathcal{CS}^{j-1}({\color{red}x}),$$

Prediction Model

U. Rosolia, and F. Borrelli. "Learning how to autonomously race a car: a predictive control approach." IEEE Transactions on Control Systems Technology 28, no. 6 (2019): 2713-2719.

Learning Model Predictive Controller

At time t of iteration j solve the following Constrained Finite Time Optimal Control Problem (CFTOCP)

$$J^{\mathrm{LMPC},j}_{t\rightarrow t+N}(x^j_t) = \min_{u^j_{t|t},\ldots,u^j_{t+N-1|t}} \sum_{k=t}^{t+N-1} h(x^j_{k|t}, u^j_{k|t}) + V^{j-1}(x^j_{t+N|t}, x)$$

s.t.

$$x^j_{k+1|t} = A^j_{k|t} x^j_{k|t} + B^j_{k|t} u^j_{k|t} + C^j_{k|t}$$

$$x^j_{t|t} = x^j_t,$$

$$x^j_{k|t} \in \mathcal{X},\ u^j_{k|t} \in \mathcal{U},\ \forall k \in [t, \cdots, t+N-1]$$

$$x^j_{t+N|t} \in \mathcal{CS}^{j-1}(x),$$

Prediction Model

Value Function

Safe Set

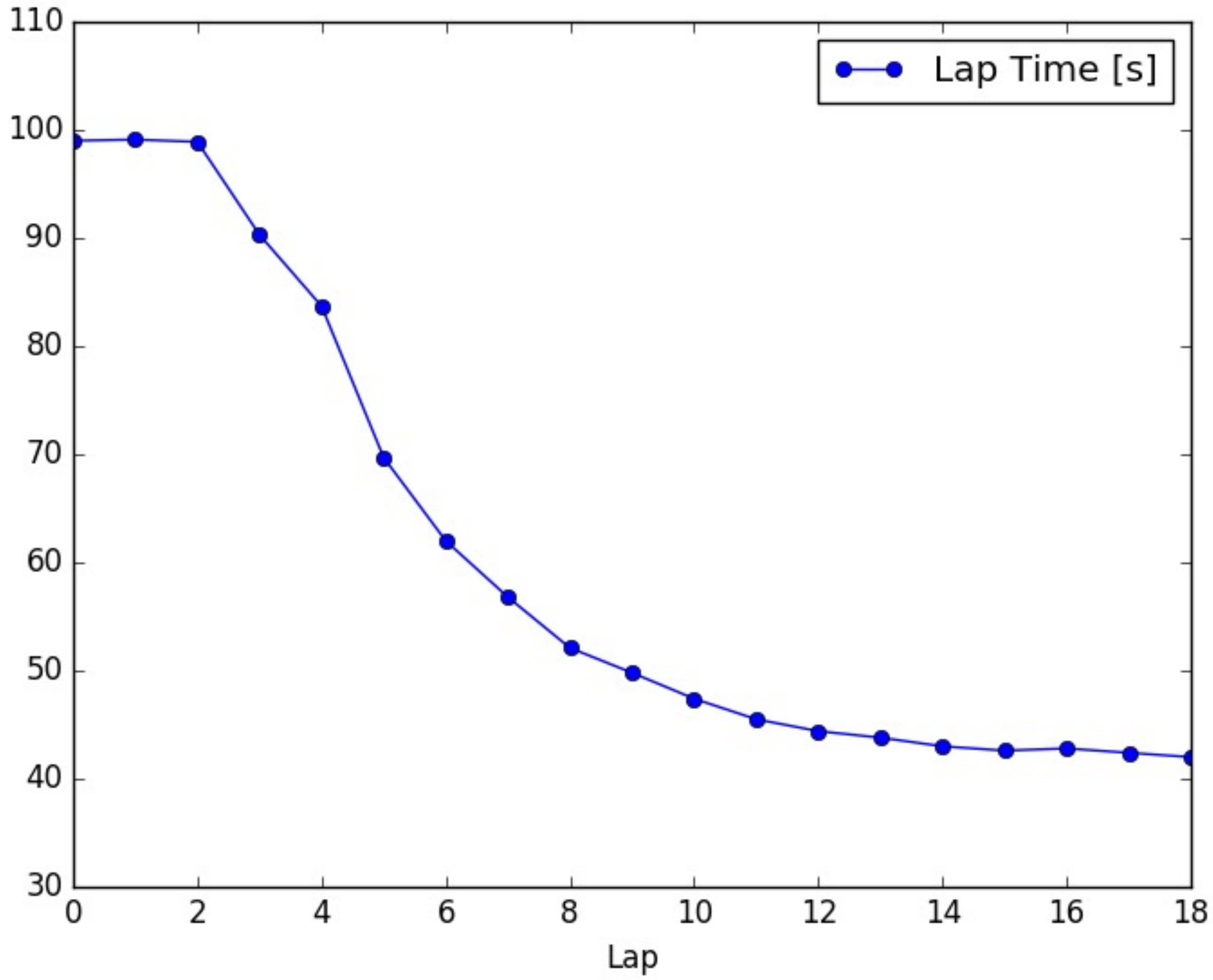
Learning Model Predictive Controller full-size vehicle experiments

Credits: Siddharth Nair, Nitin Kapania and Ugo Rosolia

Learning Model Predictive Controller full-size vehicle experiments

Credits: Siddharth Nair, Nitin Kapania and Ugo Rosolia

Lap Time



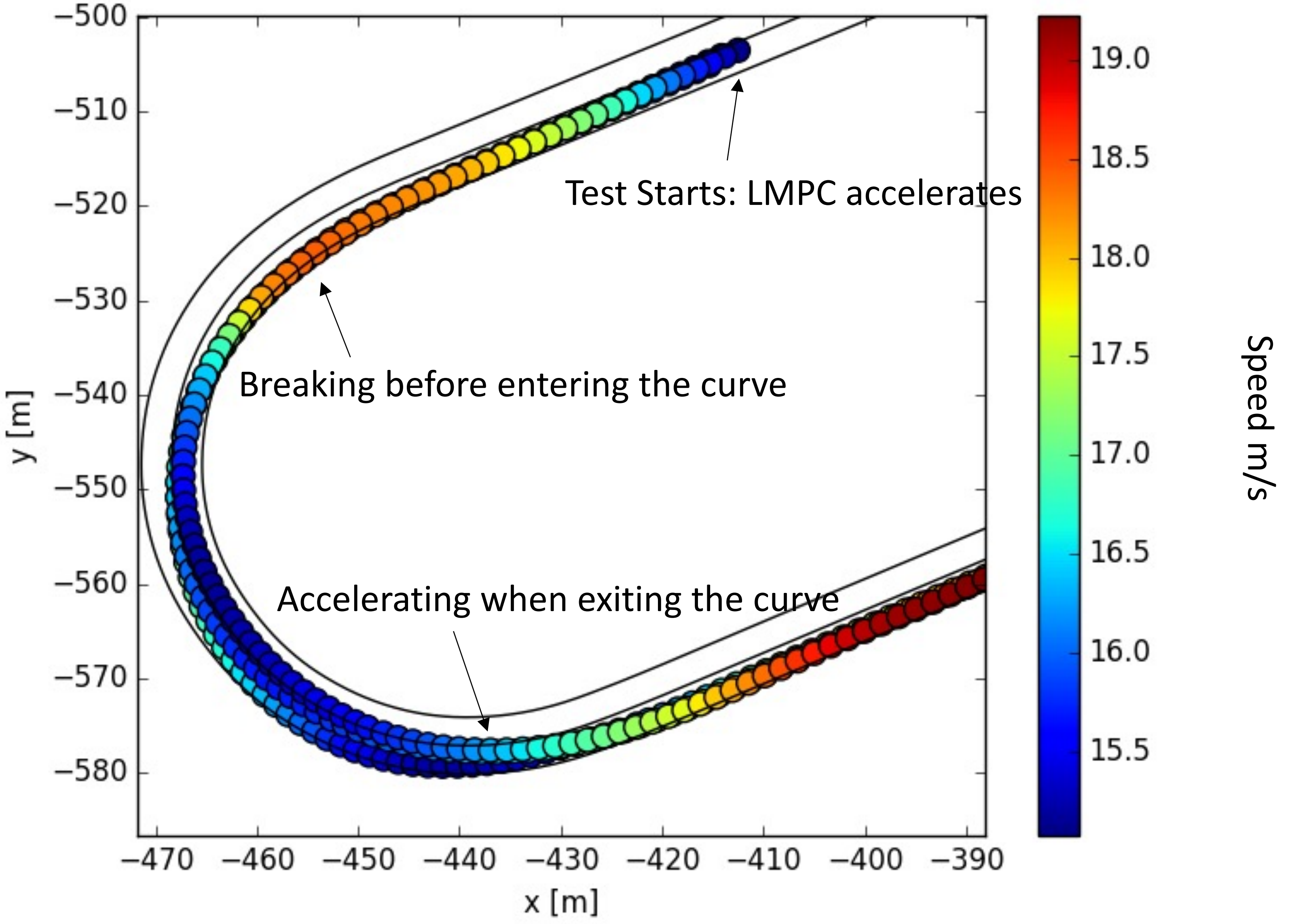
The control policy is constructed using ~1k data points (last 2 laps)

The control action is computed using ~100 data points

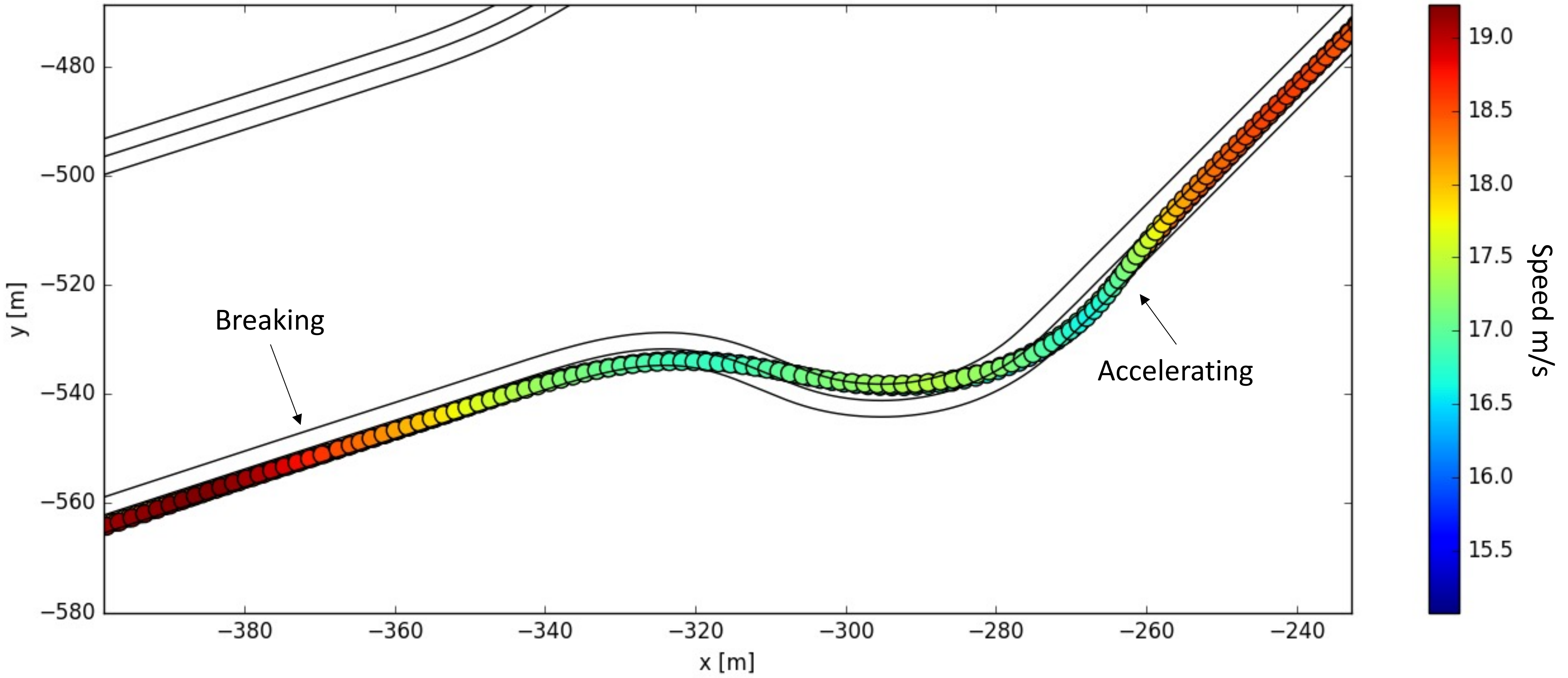
Learning Model Predictive Controller full-size vehicle experiments

Credits: Siddharth Nair, Nitin Kapania and Ugo Rosolia

Velocity Profile at Convergence (Curve 1)

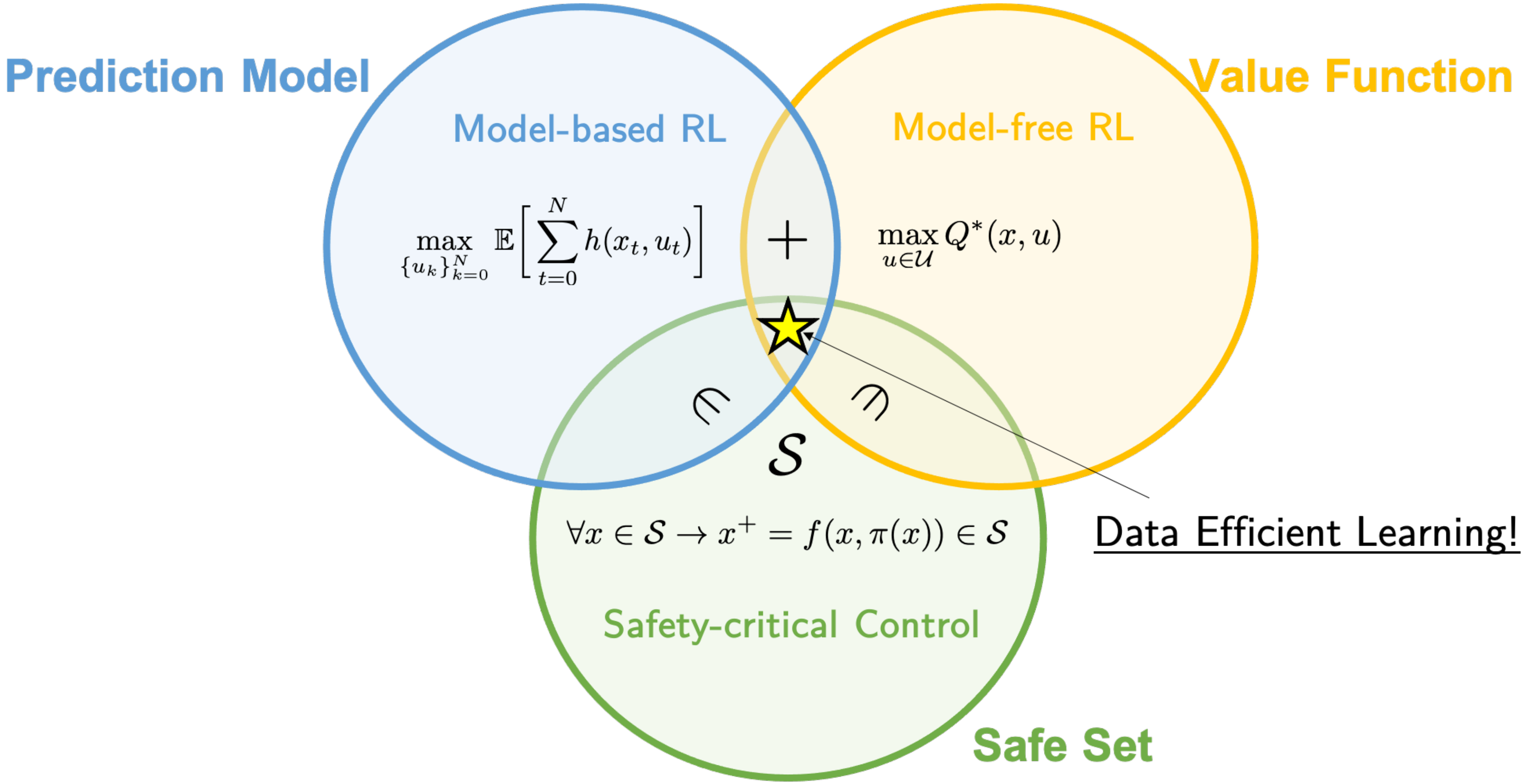


Velocity Profile at Convergence (Chicane)



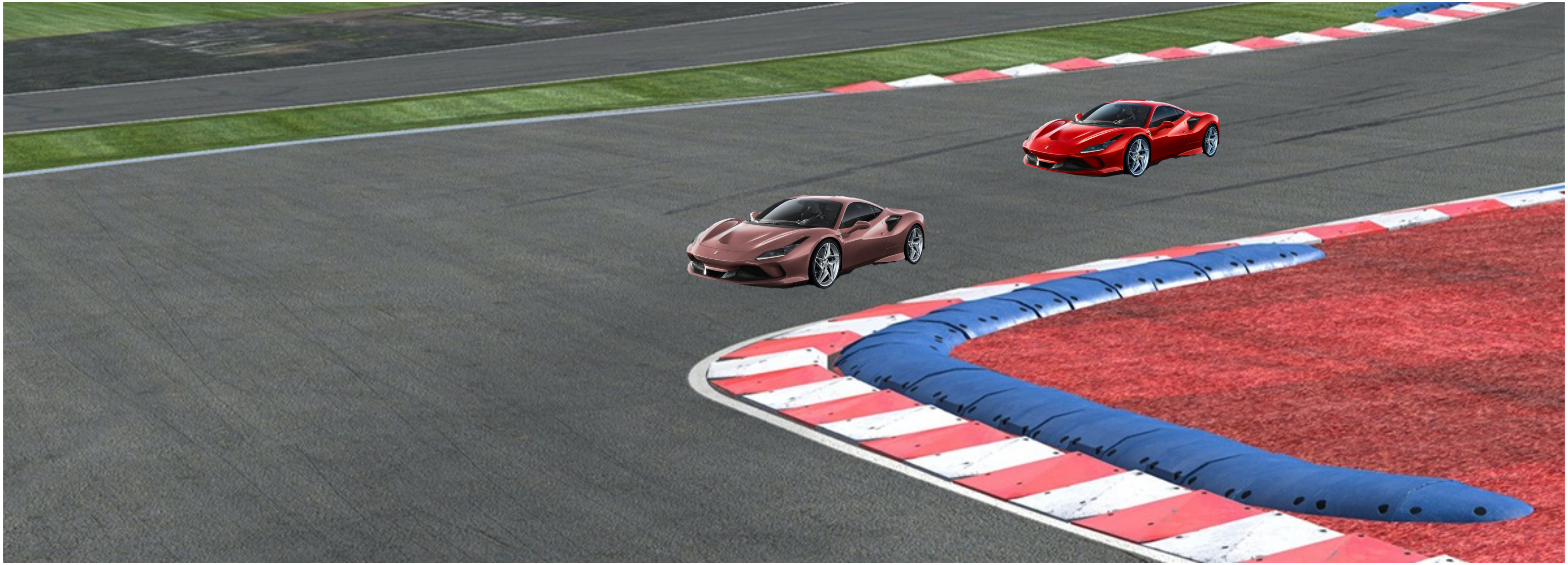
The key components

- Predicted trajectory given by prediction model
- Predicted cost estimated by value function
- Safe region estimated by the safe set



What is next?

What is next?



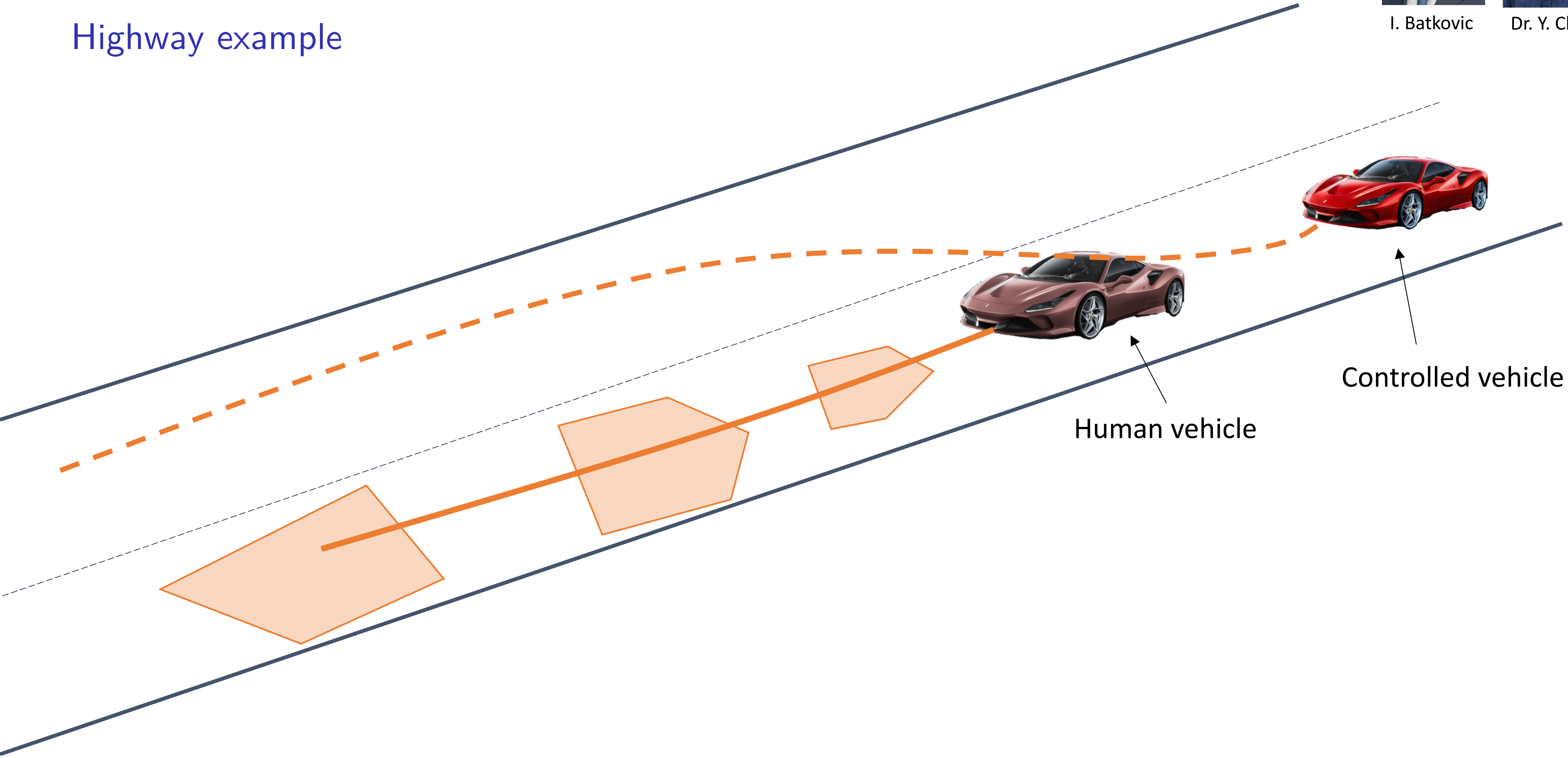
[Human-Machine Interaction]

Planning Under Uncertainty

I. Batkovic

Dr. Y. Chen

Highway example

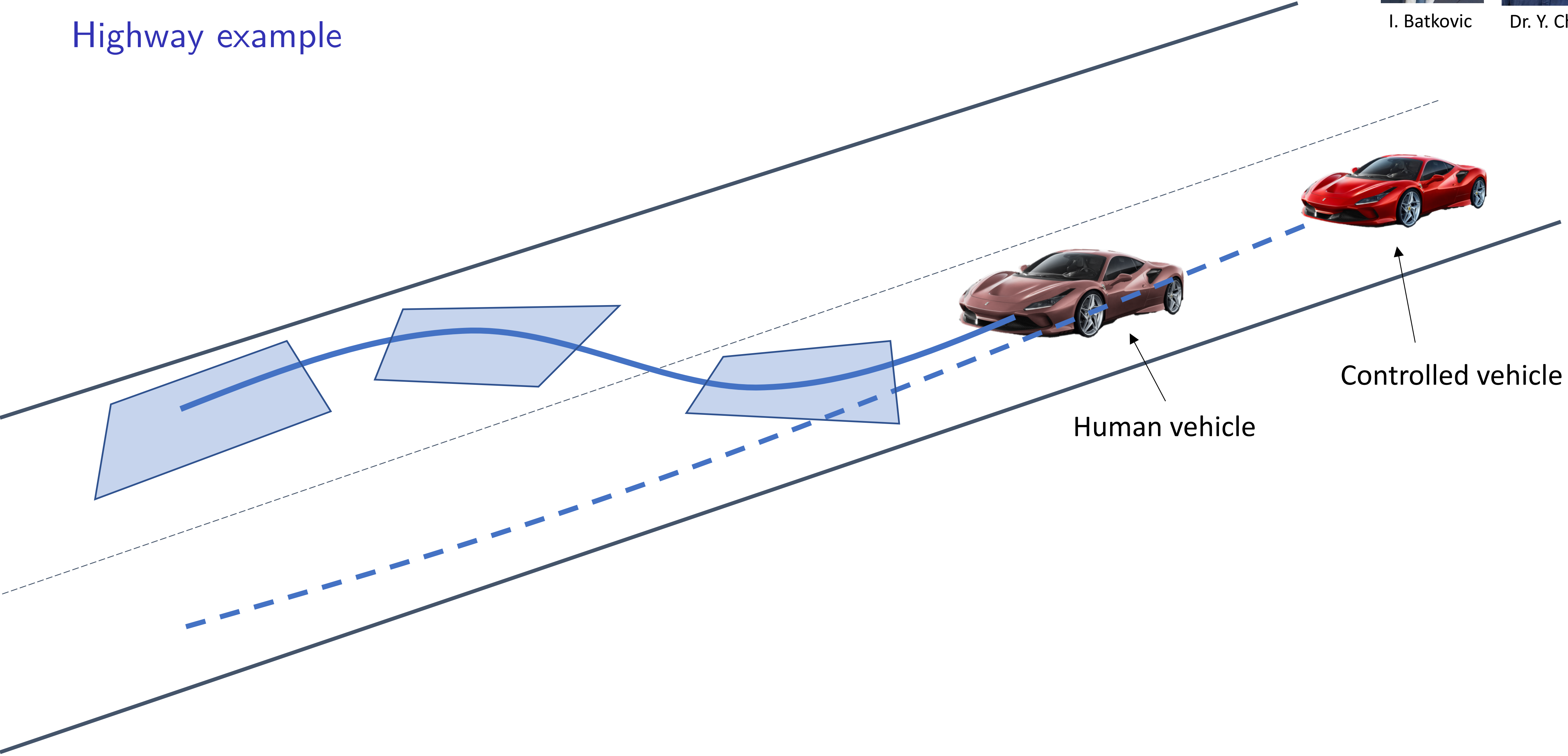


Planning Under Uncertainty

I. Batkovic

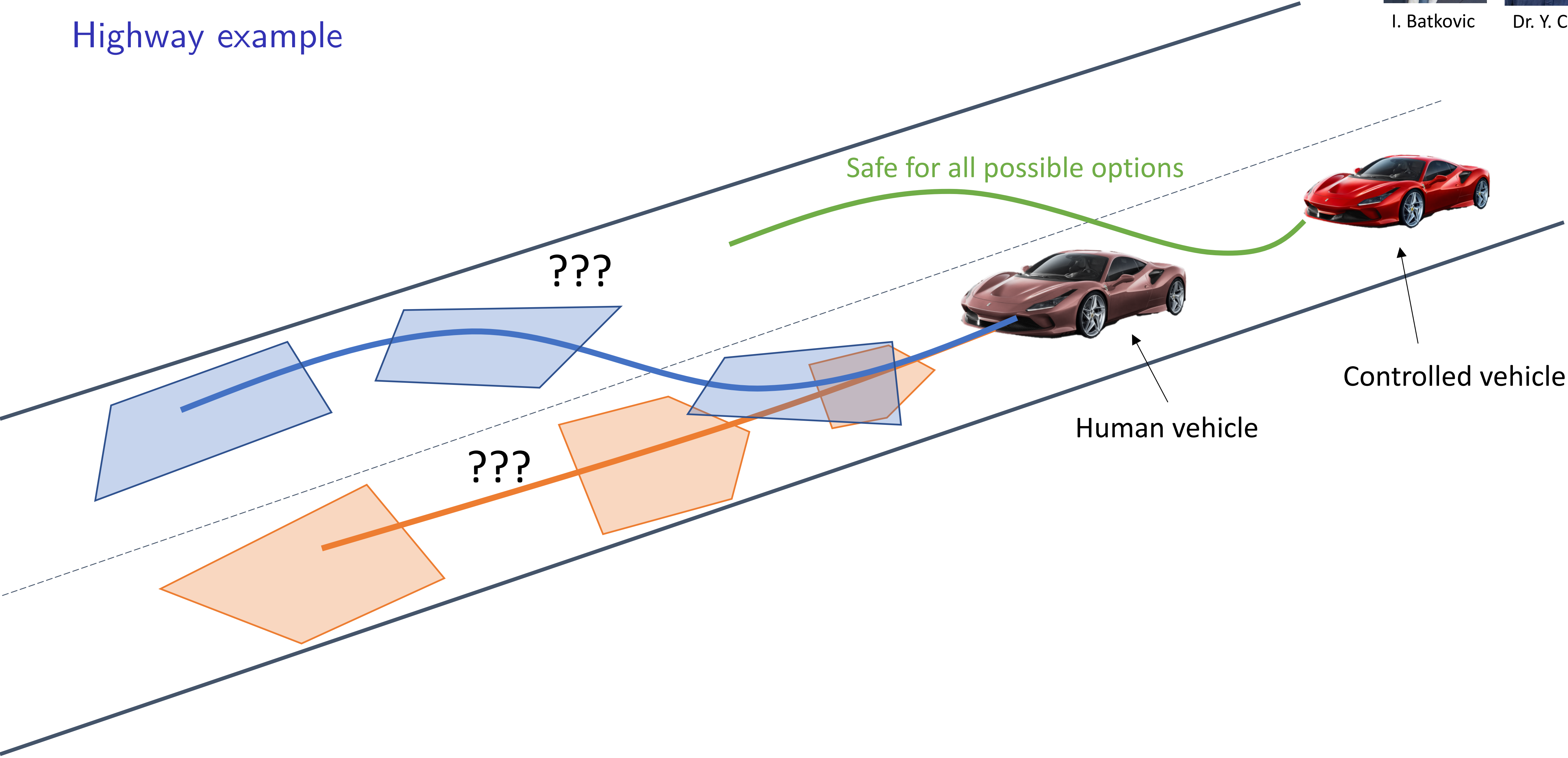
Dr. Y. Chen

Highway example



Planning Under Uncertainty

Highway example

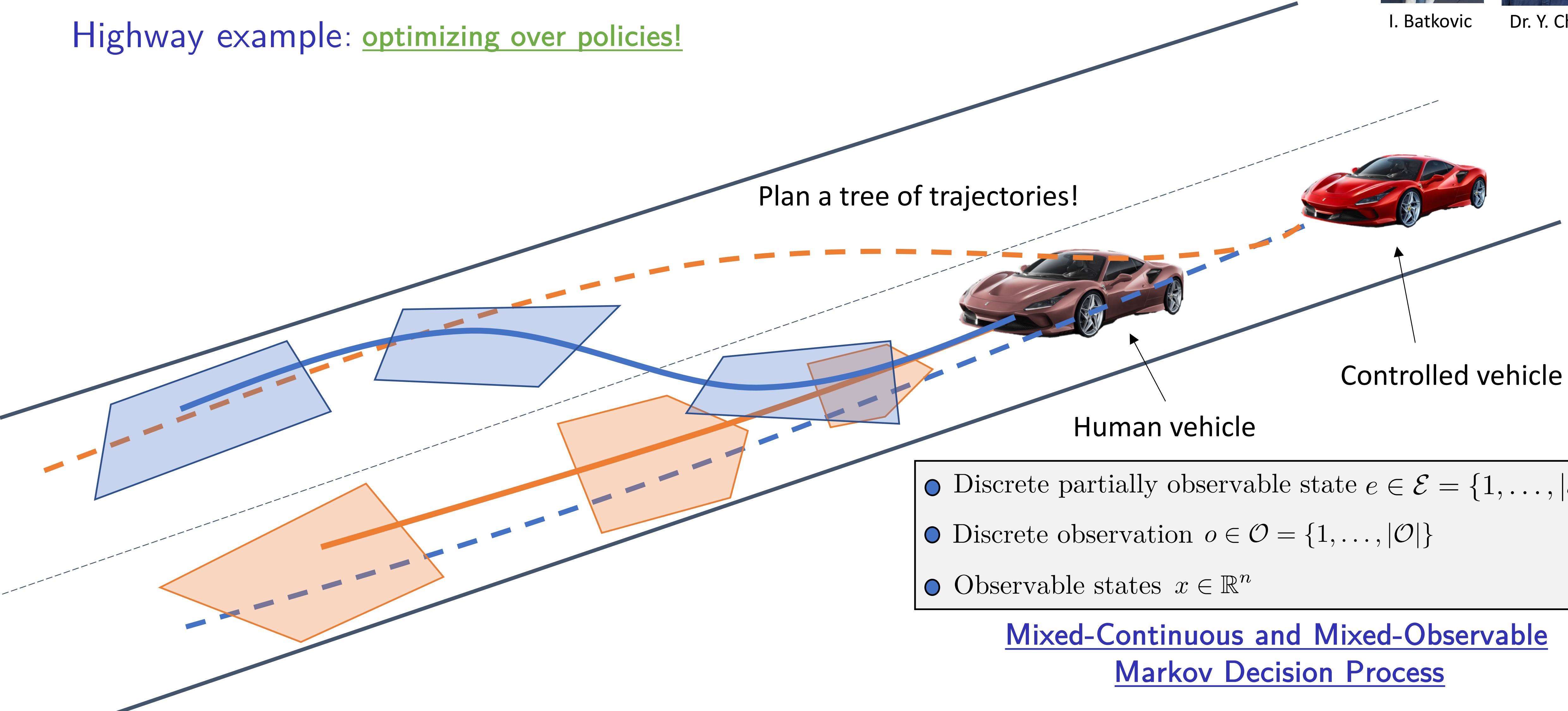


Planning Under Uncertainty

I. Batkovic

Dr. Y. Chen

Highway example: optimizing over policies!



- Discrete partially observable state $e \in \mathcal{E} = \{1, \ldots, |\mathcal{E}|\}$
- Discrete observation $o \in \mathcal{O} = \{1, \ldots, |\mathcal{O}|\}$
- Observable states $x \in \mathbb{R}^n$

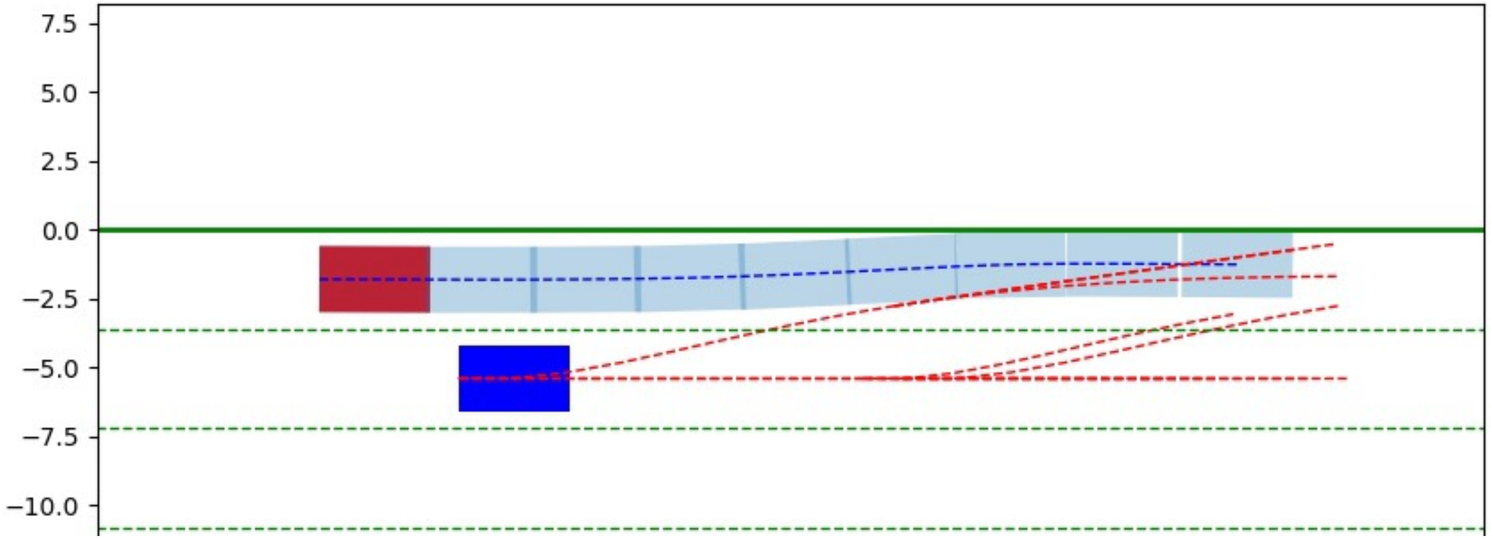
Mixed-Continuous and Mixed-Observable Markov Decision Process

I. Batkovic, U. Rosolia, M. Zanon, and P. Falcone. "A Robust Scenario MPC Approach for Uncertain Multi-modal Obstacles." IEEE Control Systems Letters 5, no. 3 (2020): 947-952.
P. OM Scokaert, and D. Q. Mayne. "Min-max feedback model predictive control for constrained linear systems." IEEE Transactions on Automatic control 43, no. 8 (1998): 1136-1142.

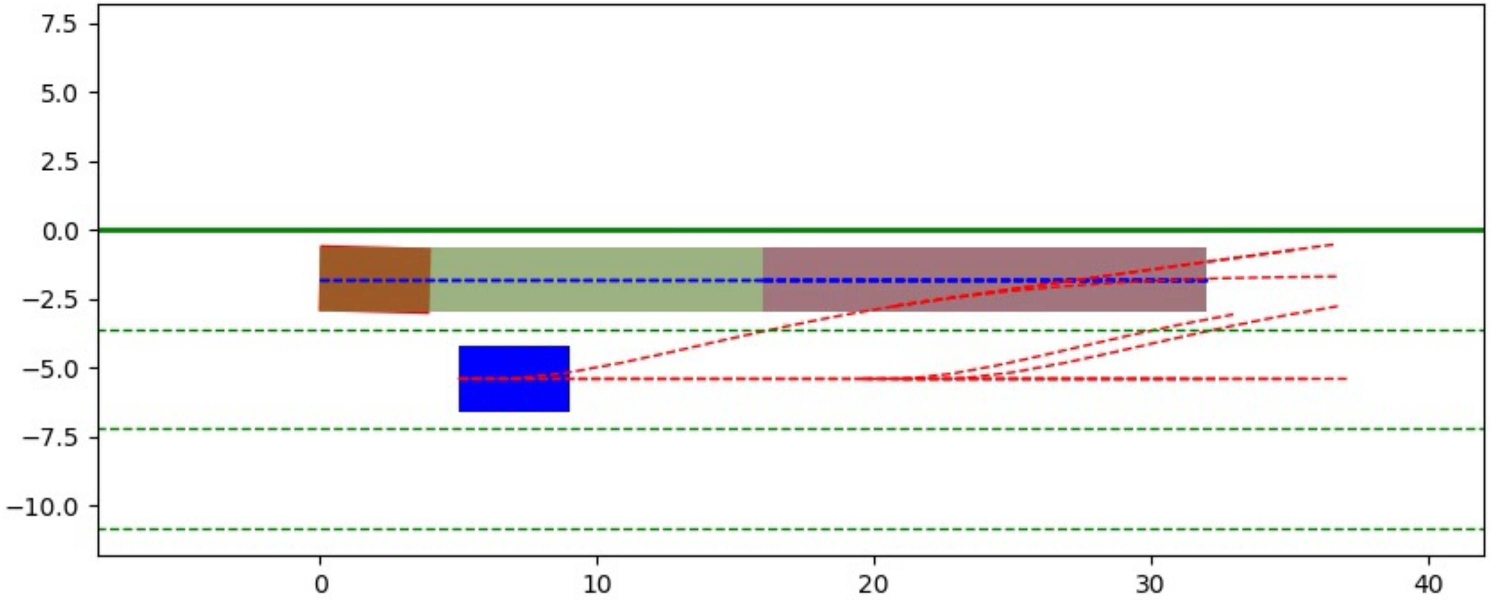
Planning Under Uncertainty: Highway driving

Dr. Y. Chen

Optimizing over
open-loop actions

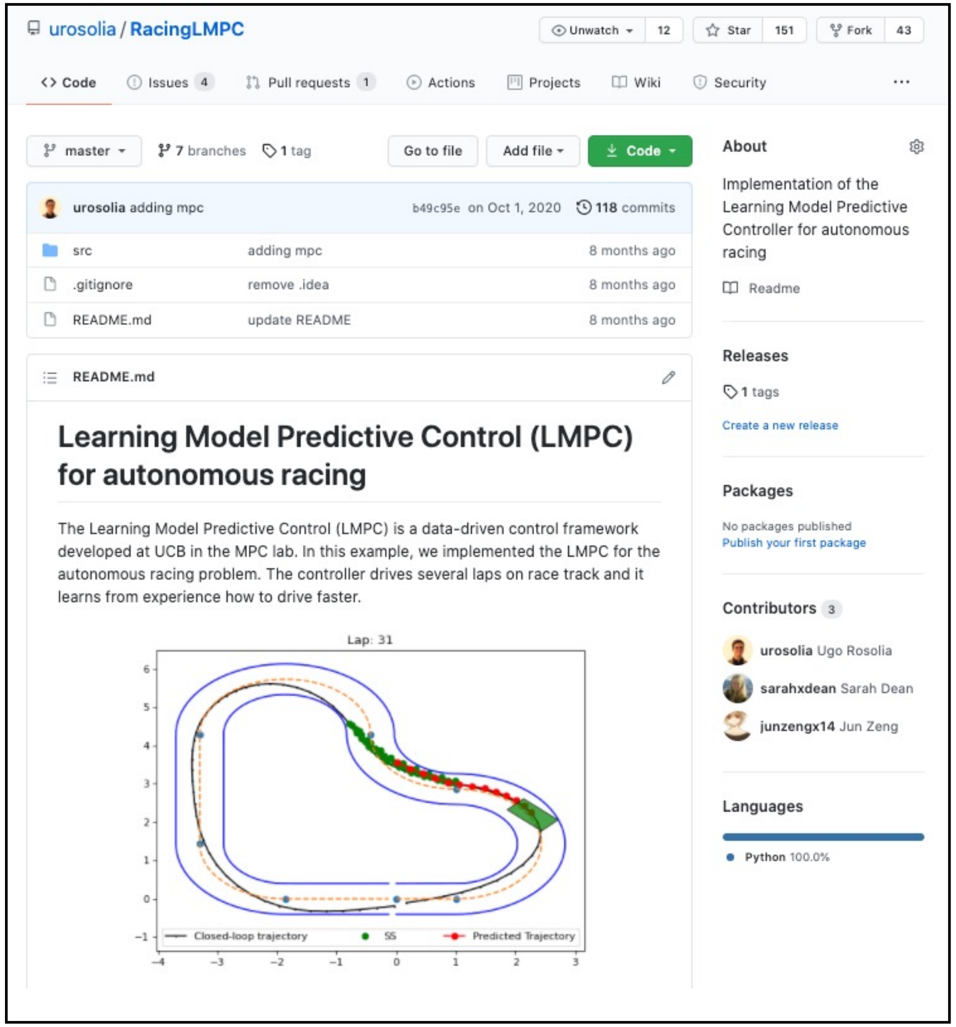


Optimizing over
closed-loop policies

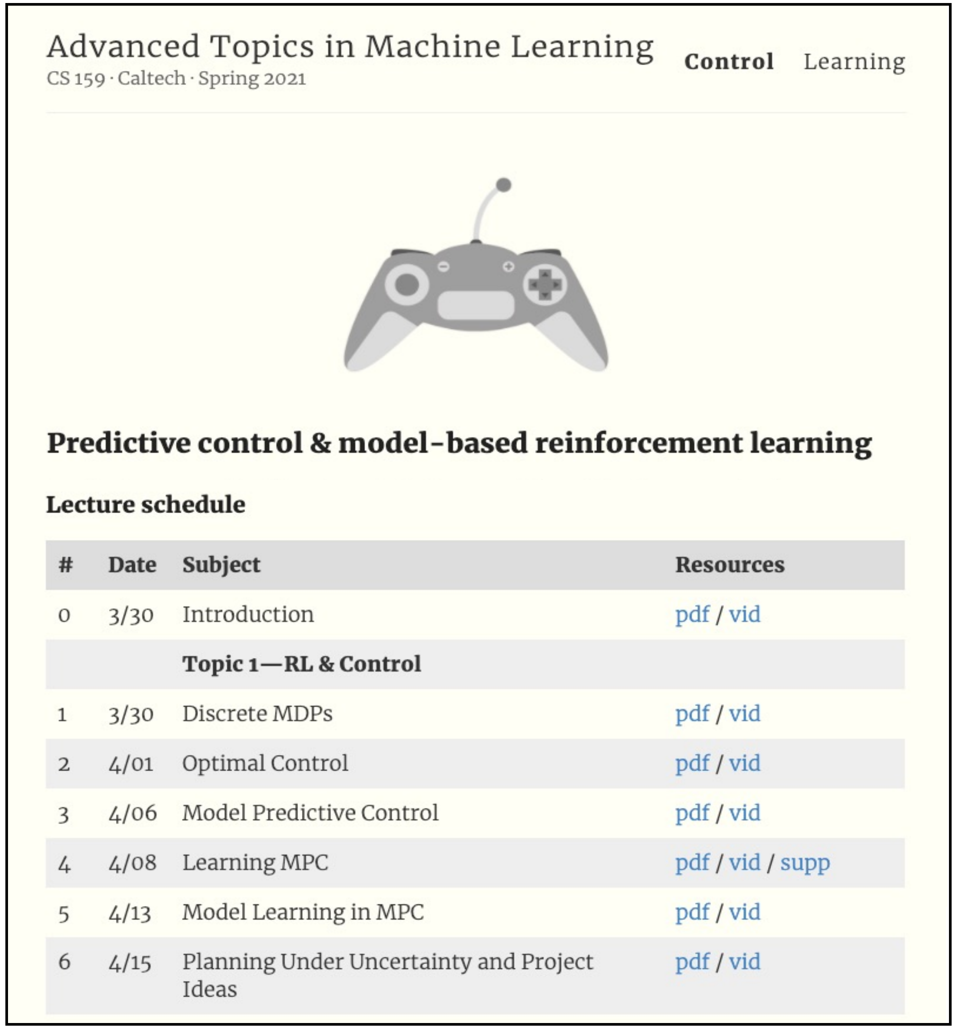


Thanks! Questions?

Code available online



Course material online



Advanced Topics in Machine Learning

CS 159 · Caltech · Spring 2021

Control Learning

Predictive control & model-based reinforcement learning

Lecture schedule

#	Date	Subject	Resources
0	3/30	Introduction	pdf / vid
		Topic 1—RL & Control	
1	3/30	Discrete MDPs	pdf / vid
2	4/01	Optimal Control	pdf / vid
3	4/06	Model Predictive Control	pdf / vid
4	4/08	Learning MPC	pdf / vid / supp
5	4/13	Model Learning in MPC	pdf / vid
6	4/15	Planning Under Uncertainty and Project Ideas	pdf / vid

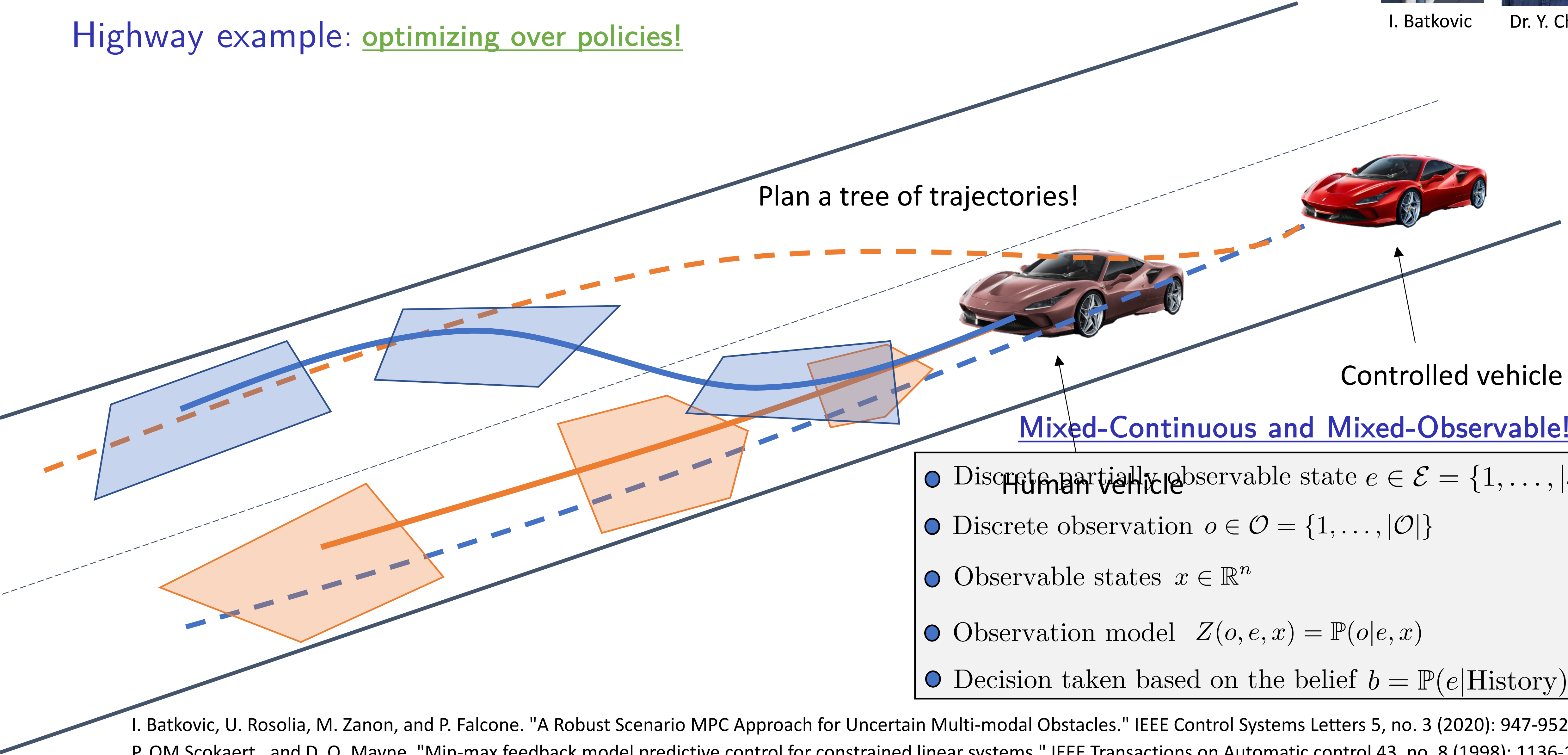
Backup slides

Planning Under Uncertainty


I. Batkovic


Dr. Y. Chen

Highway example: optimizing over policies!



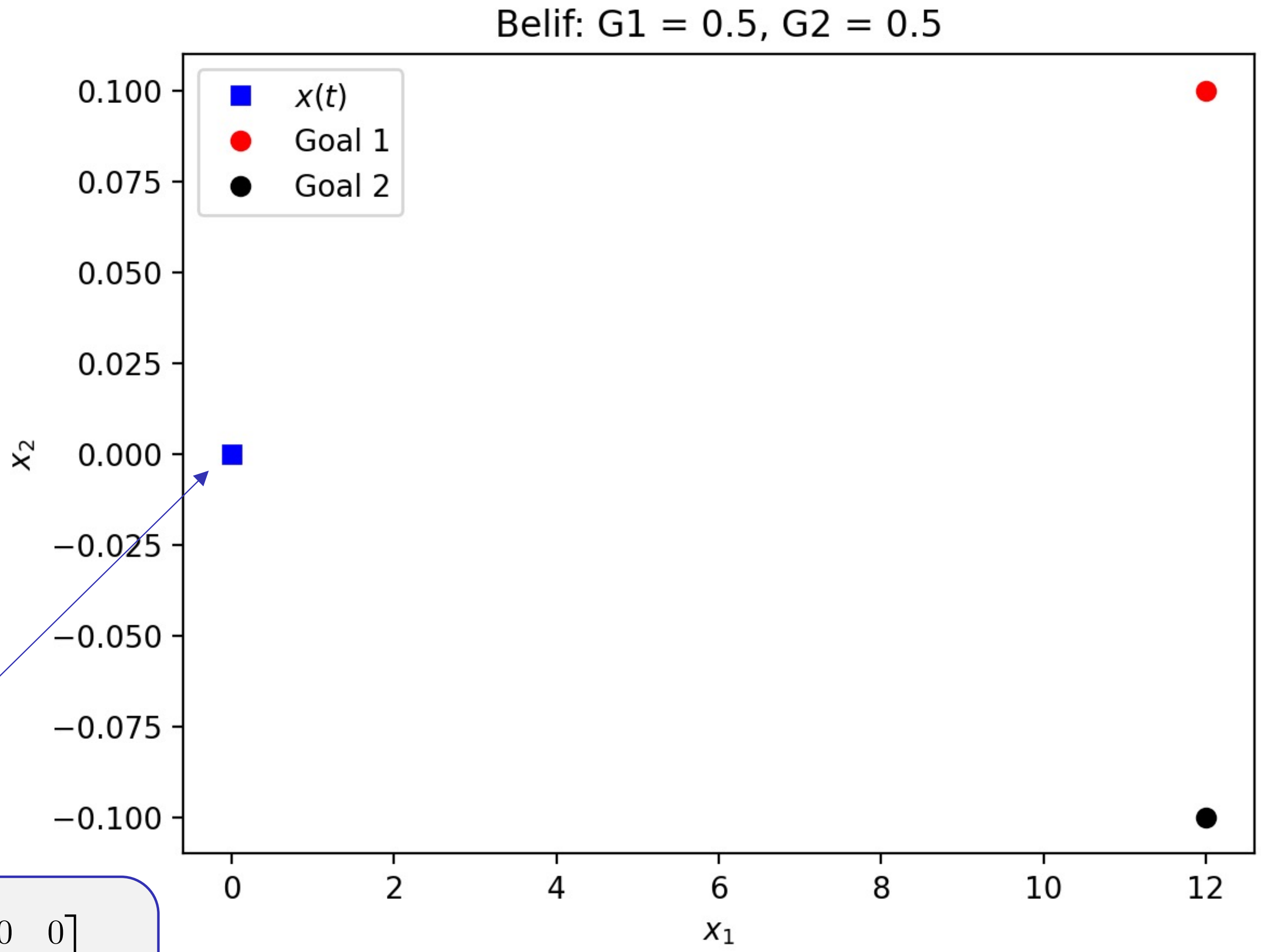
Mixed-Continuous and Mixed-Observable!

- Discrete partially observable state $e \in \mathcal{E} = \{1, \dots, |\mathcal{E}|\}$
- Discrete observation $o \in \mathcal{O} = \{1, \dots, |\mathcal{O}|\}$
- Observable states $x \in \mathbb{R}^n$
- Observation model $Z(o, e, x) = \mathbb{P}(o|e, x)$
- Decision taken based on the belief $b = \mathbb{P}(e|\text{History})$

I. Batkovic, U. Rosolia, M. Zanon, and P. Falcone. "A Robust Scenario MPC Approach for Uncertain Multi-modal Obstacles." IEEE Control Systems Letters 5, no. 3 (2020): 947-952.
P. OM Scokaert,, and D. Q. Mayne. "Min-max feedback model predictive control for constrained linear systems." IEEE Transactions on Automatic control 43, no. 8 (1998): 1136-1142.

Planning Under Uncertainty: Goal Exploration

Dr. Y. Chen



Observation Model

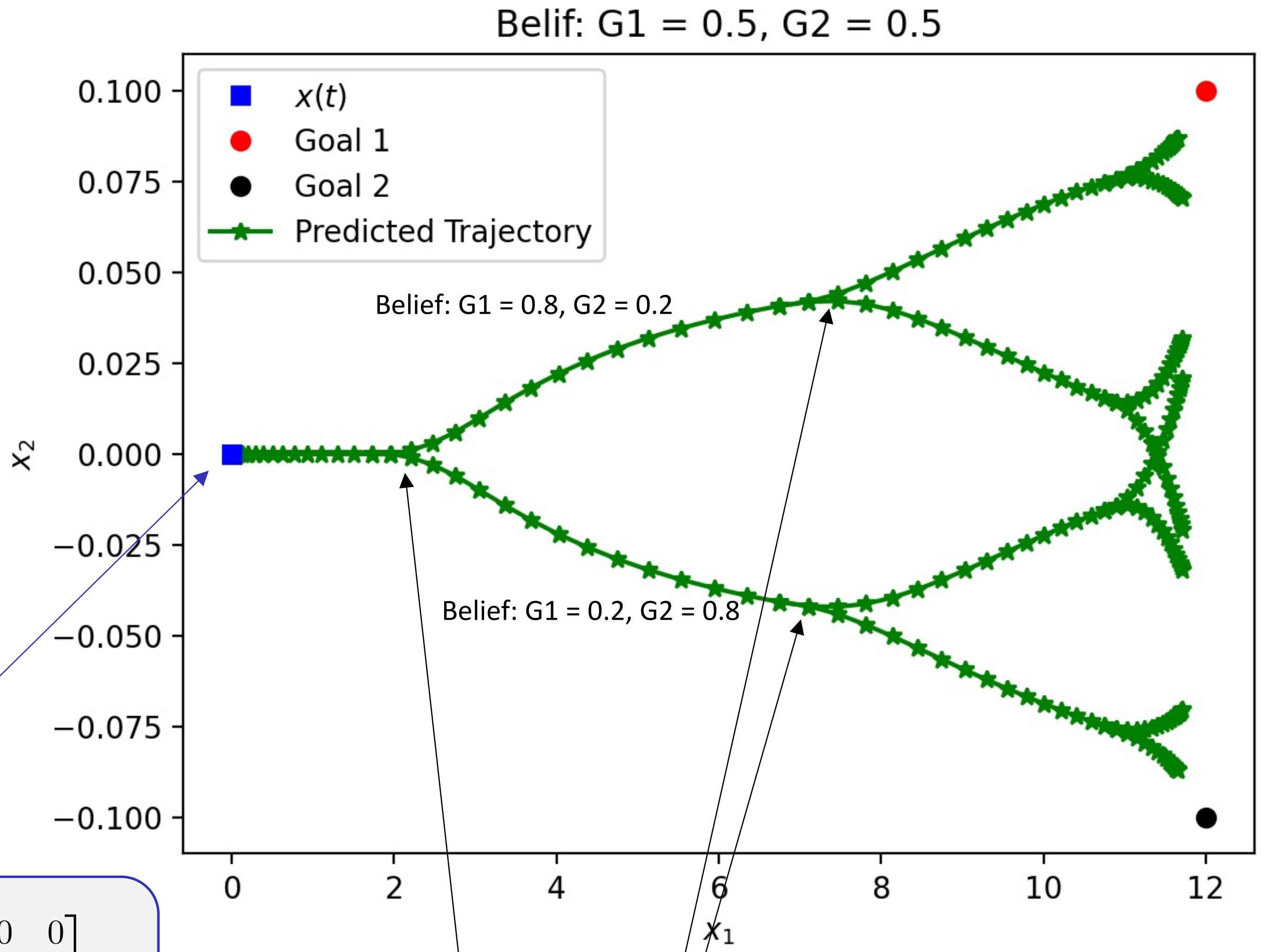
$$P(o = 1|\text{goal} = 1) = 0.8$$

$$P(o = 0|\text{goal} = 1) = 0.2$$

$$x_{k+1} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} u_k$$

Planning Under Uncertainty: Goal Exploration

Dr. Y. Chen



Observation Model

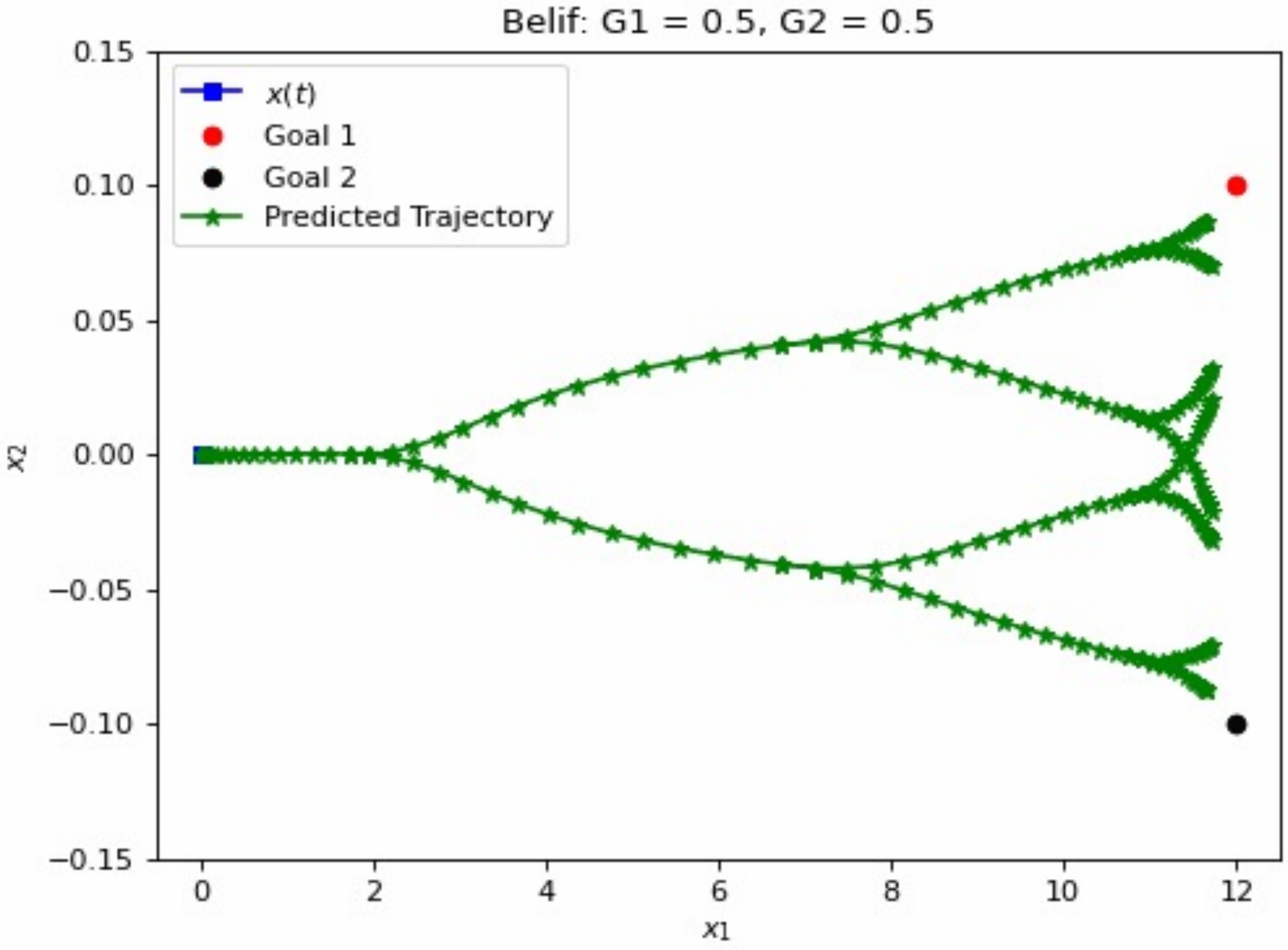
$$P(o = 1|\text{goal} = 1) = 0.8$$
$$P(o = 0|\text{goal} = 1) = 0.2$$

$$x_{k+1} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} u_k$$

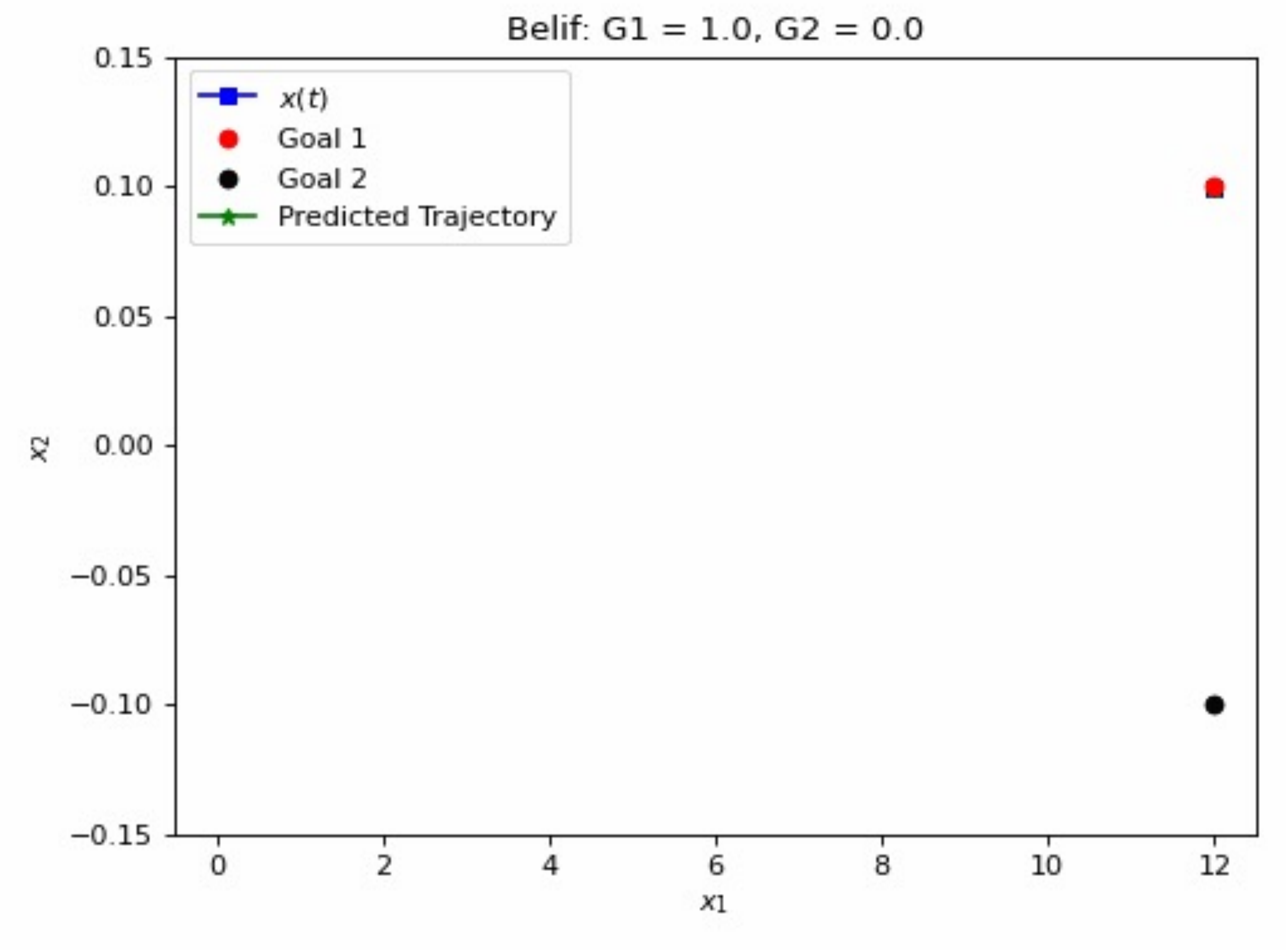
Branching based on discrete observation $o \in \mathcal{O} = \{1, \ldots, |\mathcal{O}|\}$

Planning Under Uncertainty: Goal Exploration

Dr. Y. Chen



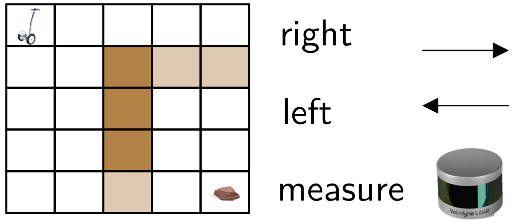
Planning Under Uncertainty: Goal Exploration



Dr. Y. Chen

Multi-Rate Hierarchical Control

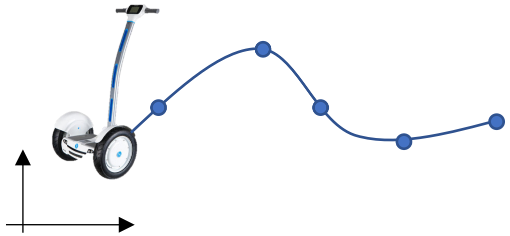
High-level
Decision Making



Abstraction with partial observations

Event-Based

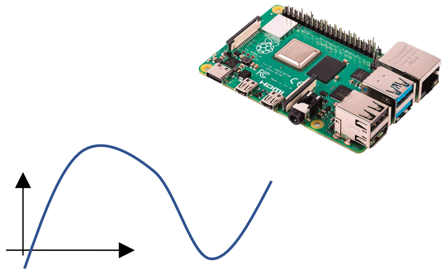
Mid-level
Trajectory Planning



Discrete time linear dynamics

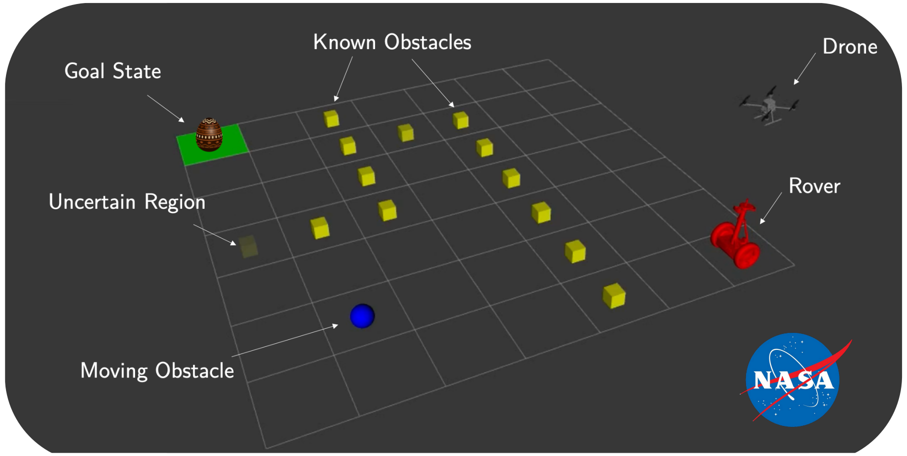
Frequency 1/T Hz

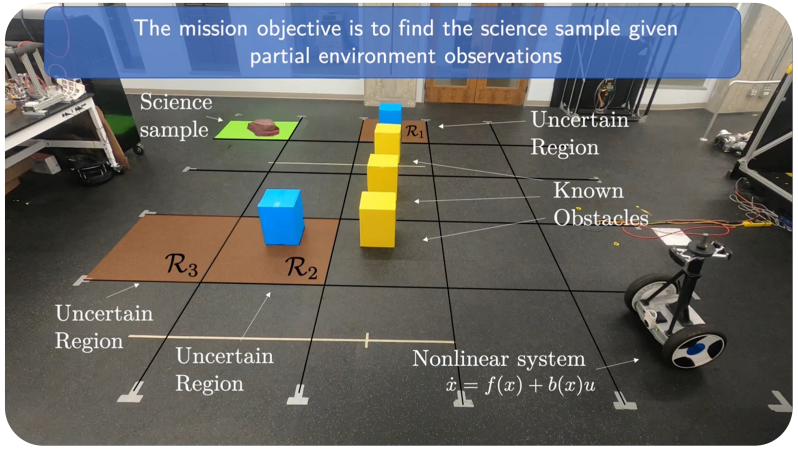
Low-level
Actuator Control



Continuous time nonlinear system

Continuous Time





U. Rosolia, and A. D. Ames, “Multi-rate control design leveraging control barrier functions and model predictive control policies”. *IEEE Control Systems Letters, (2020)*
U. Rosolia, A. Singletary, and A. D. Ames. "Unified Multi-Rate Control: from Low Level Actuation to High Level Planning." *arXiv preprint arXiv:2012.06558* (2020).

System ID in Autonomous Racing

- Nonlinear Dynamical System,

$$
\begin{aligned}
\ddot{x} &= \dot{y}\dot{\psi} + \frac{1}{m}\sum_i F_{x_i} \\
\ddot{y} &= -\dot{x}\dot{\psi} + \frac{1}{m}\sum_i F_{y_i} \\
\ddot{\psi} &= \frac{1}{I_z}(a(F_{y_{1,2}}) - b(F_{y_{2,3}}) + c(-F_{x_{1,3}} + F_{x_{2,4}}) \\
\dot{X} &= \dot{x}\cos\psi - \dot{y}\sin\psi, \quad \dot{Y} = \dot{x}\sin\psi + \dot{y}\cos\psi
\end{aligned}
$$

System ID in Autonomous Racing

- Nonlinear Dynamical System,

$$
\begin{aligned}
\ddot{x} &= \dot{y}\dot{\psi} + \frac{1}{m}\sum_i F_{x_i} \\
\ddot{y} &= -\dot{x}\dot{\psi} + \frac{1}{m}\sum_i F_{y_i} \\
\ddot{\psi} &= \frac{1}{I_z}(a(F_{y_{1,2}}) - b(F_{y_{2,3}}) + c(-F_{x_{1,3}} + F_{x_{2,4}}) \\
\dot{X} &= \dot{x}\cos\psi - \dot{y}\sin\psi, \quad \dot{Y} = \dot{x}\sin\psi + \dot{y}\cos\psi
\end{aligned}
$$

Kinematic Equations

System ID in Autonomous Racing

- Nonlinear Dynamical System,

$$\begin{aligned}
\ddot{x} &= \dot{y}\dot{\psi} + \frac{1}{m}\sum_i F_{x_i} \\
\ddot{y} &= -\dot{x}\dot{\psi} + \frac{1}{m}\sum_i F_{y_i} \\
\ddot{\psi} &= \frac{1}{I_z}(a(F_{y_{1,2}}) - b(F_{y_{2,3}}) + c(-F_{x_{1,3}} + F_{x_{2,4}}) \\
\dot{X} &= \dot{x}\cos\psi - \dot{y}\sin\psi, \quad \dot{Y} = \dot{x}\sin\psi + \dot{y}\cos\psi
\end{aligned}$$

Kinematic Equations

- Identifying the Dynamical System

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} x^j_{k|t} + \begin{bmatrix} \\ \\ \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

System ID in Autonomous Racing

- Nonlinear Dynamical System,

$$
\begin{aligned}
\ddot{x} &= \dot{y}\dot{\psi} + \frac{1}{m}\sum_i F_{x_i} \\
\ddot{y} &= -\dot{x}\dot{\psi} + \frac{1}{m}\sum_i F_{y_i} \\
\ddot{\psi} &= \frac{1}{I_z}(a(F_{y_{1,2}}) - b(F_{y_{2,3}}) + c(-F_{x_{1,3}} + F_{x_{2,4}}) \\
\dot{X} &= \dot{x}\cos\psi - \dot{y}\sin\psi, \quad \dot{Y} = \dot{x}\sin\psi + \dot{y}\cos\psi
\end{aligned}
$$

Dynamic Equations

Kinematic Equations

- Identifying the Dynamical System

$$
x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} x^j_{k|t} + \begin{bmatrix} \\ \\ \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}
$$

Linearization around predicted trajectory

System ID in Autonomous Racing

- Nonlinear Dynamical System,

$$\begin{aligned}\ddot{x} &= \dot{y}\dot{\psi} + \frac{1}{m}\sum_i F_{x_i} \\ \ddot{y} &= -\dot{x}\dot{\psi} + \frac{1}{m}\sum_i F_{y_i} \\ \ddot{\psi} &= \frac{1}{I_z}(a(F_{y_{1,2}}) - b(F_{y_{2,3}}) + c(-F_{x_{1,3}} + F_{x_{2,4}}) \\ \dot{X} &= \dot{x}\cos\psi - \dot{y}\sin\psi, \quad \dot{Y} = \dot{x}\sin\psi + \dot{y}\cos\psi\end{aligned}$$

Dynamic Equations

Kinematic Equations

- Identifying the Dynamical System

Local Linear Regression

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \left[\begin{array}{c|c} \multicolumn{2}{c}{\arg\min \sum_{i,s} K(x^j_{k|t} - x^i_s)||\Lambda_y \begin{bmatrix} x^j_{k|t} \\ u^j_{k|t} \\ 1 \end{bmatrix} - y^i_{s+1}||, \forall y \in \{\dot{x}, \dot{y}, \ddot{\psi}\}} \\ \begin{matrix}\text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics}\end{matrix} \; x^j_{k|t} + & \begin{matrix}\text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics}\end{matrix} \end{array} \right] \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

System Identification – Design Steps

Identifying the Dynamical System

Local Linear Regression

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \left[\begin{bmatrix} \arg\min \sum_{i,s} K(x^j_{k|t} - x^i_s) ||\Lambda_y \begin{bmatrix} x^j_{k|t} \\ u^j_{k|t} \\ 1 \end{bmatrix} - y^i_{s+1}||, \forall y \in \{\dot{x}, \dot{y}, \ddot{\psi}\} \\ \begin{matrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{matrix} \quad x^j_{k|t} + \quad \begin{matrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{matrix} \end{bmatrix} \right] \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

Implementation Details

System Identification – Design Steps

Identifying the Dynamical System

Local Linear Regression

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \left[\begin{matrix} \arg\min \sum_{i,s} K(x^j_{k|t} - x^i_s) \left\| \Lambda_y \begin{bmatrix} x^j_{k|t} \\ u^j_{k|t} \\ 1 \end{bmatrix} - y^i_{s+1} \right\|, \forall y \in \{\dot{x}, \dot{y}, \ddot{\psi}\} \\ \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} x^j_{k|t} + \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} \end{matrix} \right] \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

Implementation Details

- Compute trajectory to linearize around from previous optimal inputs and stored data

System Identification – Design Steps

Identifying the Dynamical System

Local Linear Regression

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \begin{bmatrix} \multicolumn{2}{c}{\arg\min \sum_{i,s} K(x^j_{k|t} - x^i_s) ||\Lambda_y \begin{bmatrix} x^j_{k|t} \\ u^j_{k|t} \\ 1 \end{bmatrix} - y^i_{s+1}||, \forall y \in \{\dot{x}, \dot{y}, \ddot{\psi}\}} \\ \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} x^j_{k|t} + & \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} \end{bmatrix} \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

Implementation Details

- Compute trajectory to linearize around from previous optimal inputs and stored data
- Enforce model-based sparsity in local linear regression

System Identification – Design Steps

Identifying the Dynamical System

Local Linear Regression

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \begin{bmatrix} \multicolumn{3}{c}{\arg\min \sum_{i,s} K(x^j_{k|t} - x^i_s)\|\Lambda_y \begin{bmatrix} x^j_{k|t} \\ u^j_{k|t} \\ 1 \end{bmatrix} - y^i_{s+1}\|, \forall y \in \{\dot{x}, \dot{y}, \ddot{\psi}\}} \\ \begin{matrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{matrix} \end{bmatrix} x^j_{k|t} + \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

Implementation Details

- Compute trajectory to linearize around from previous optimal inputs and stored data
- Enforce model-based sparsity in local linear regression
- Perform local linear regression at the linearization points using a subset of the stored data

System Identification – Design Steps

Identifying the Dynamical System

Local Linear Regression

$$x^j_{k+1|t} = \begin{bmatrix} \dot{x}_{k+1|t} \\ \dot{y}_{k+1|t} \\ \ddot{\psi} \\ \psi_{k+1|t} \\ X_{k+1|t} \\ Y_{k+1|t} \end{bmatrix} = \left[\begin{matrix} \arg\min \sum_{i,s} K(x^j_{k|t} - x^i_s) \| \Lambda_y \begin{bmatrix} x^j_{k|t} \\ u^j_{k|t} \\ 1 \end{bmatrix} - y^i_{s+1} \|, \forall y \in \{\dot{x}, \dot{y}, \ddot{\psi}\} \\ \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} x^j_{k|t} + \begin{bmatrix} \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \\ \text{Linearized Kinematics} \end{bmatrix} \end{matrix} \right] \begin{bmatrix} u^j_{k|t} \\ 1 \end{bmatrix}$$

Linearization around predicted trajectory

Implementation Details

- Compute trajectory to linearize around from previous optimal inputs and stored data
- Enforce model-based sparsity in local linear regression
- Perform local linear regression at the linearization points using a subset of the stored data
- Use kernel *K()* to weight differently data as a function of distance to the linearization trajectory

Problem Formulation

Minimum Time Control Problem

$$\min_{T,\mathbf{u}} \; \boxed{T}$$ Control objective

$$\boxed{x_0 = x_s, \;\; x_T = \mathcal{X}_F}$$ Start & end position

System dynamics
System constraints

$$\boxed{x_{k+1} = f(x_k, u_k), \quad \forall k \in \{0, \ldots, T-1\}}$$

Safety constraints

$$\boxed{x_k \in \mathcal{X}, \;\; u_k \in \mathcal{U}, \quad \forall k \in \{0, \ldots, T-1\}}$$

Problem Formulation: Assumption 1

Minimum Time Control Problem

$$\min_{T,\mathbf{u}} \boxed{T}$$ Control objective

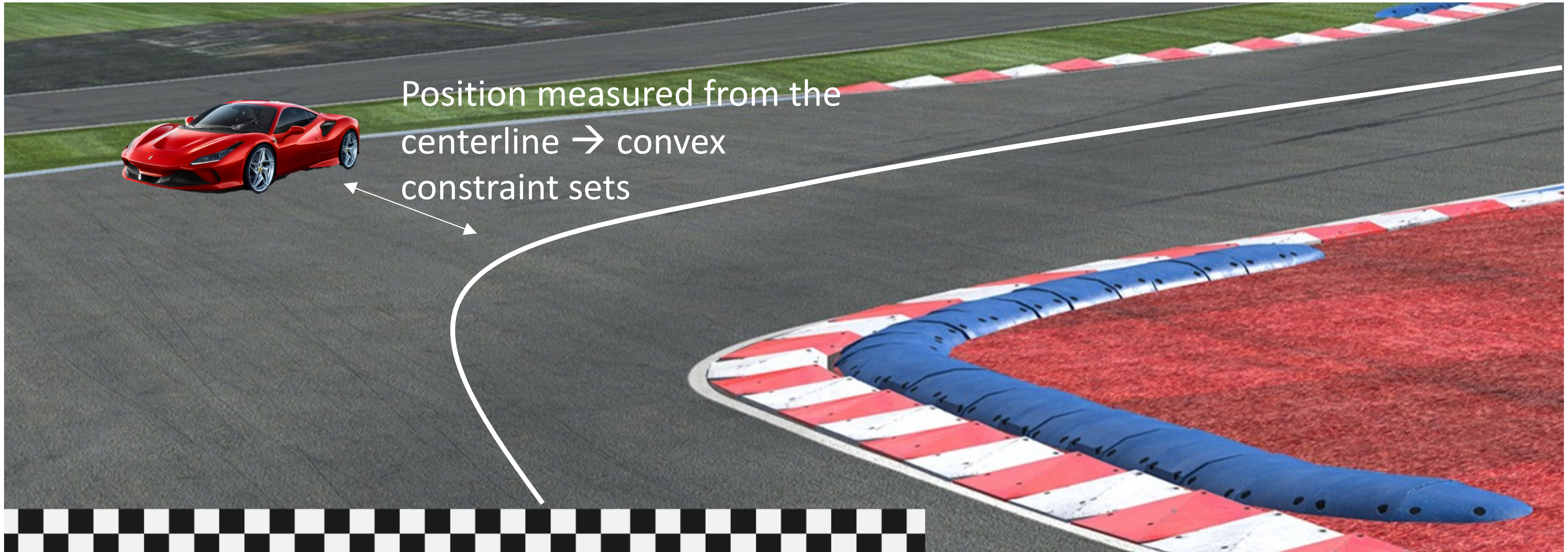
$$\boxed{x_0 = x_s,\ x_T = \mathcal{X}_F}$$ Start & end position

System dynamics
System constraints

$$\boxed{x_{k+1} = f(x_k, u_k), \quad \forall k \in \{0, \ldots, T-1\}}$$

Safety constraints

$$\boxed{x_k \in \mathcal{X},\ u_k \in \mathcal{U}, \quad \forall k \in \{0, \ldots, T-1\}}$$



$$
\begin{aligned}
J(x(t), b(t)) = \min_{\pi_0,\ldots,\pi_{N-1}} \quad & \mathbb{E}_{\mathbf{o_{N-1}}}\left[\sum_{k=0}^{N-1} h(x_k, u_k, e_k) \middle| b(t)\right] \\
\text{s.t.} \quad & x_{k+1} = Ax_k + Bu_k, \\
& u_k = \pi_k(x_0, \ldots, x_k, \mathbf{o_k}, b(t)), \\
& x_0 = x(t), \\
& u_k \in \mathcal{U}, x_k \in \mathcal{X}, \\
& \forall k \in \{0, \ldots, N-1\},
\end{aligned} \tag{3}
$$

$$
\begin{aligned}
J(x(t), b(t)) = \min_{\mathbf{u}} \quad & \sum_{k=0}^{N-1} \sum_{\mathbf{o_k} \in \mathcal{O}^k} \sum_{e \in \mathcal{E}} v_k^{\mathbf{o_k}}[e] h(x_k^{\mathbf{o_k}}, u_k^{\mathbf{o_k}}, e) \\
\text{s.t.} \quad & x_{k+1}^{\mathbf{o_k}} = A x_k^{\mathbf{o_{k-1}}} + B u_k^{\mathbf{o_k}}, \\
& x_0^{\mathbf{o_{-1}}} = x(t), v_0^{\mathbf{o_0}} = b(t), \\
& v_k^{\mathbf{o_{k+1}}} = A_e(o_{k+1}, x_{k+1}^{\mathbf{o_k}}) v_k^{\mathbf{o_k}}, \\
& u_k^{\mathbf{o_k}} \in \mathcal{U}, x_{k+1}^{\mathbf{o_k}} \in \mathcal{X}, \\
& \forall \mathbf{o_k} \in \mathcal{O}^k, \forall k \in \{0, \ldots, N-1\},
\end{aligned} \tag{8}
$$