

Balancing Safety and Performance in Learning-Based Optimal Control

VAIA Seminar series — Machine Learning in Control and Automation

Panos Patrinos

January 2023

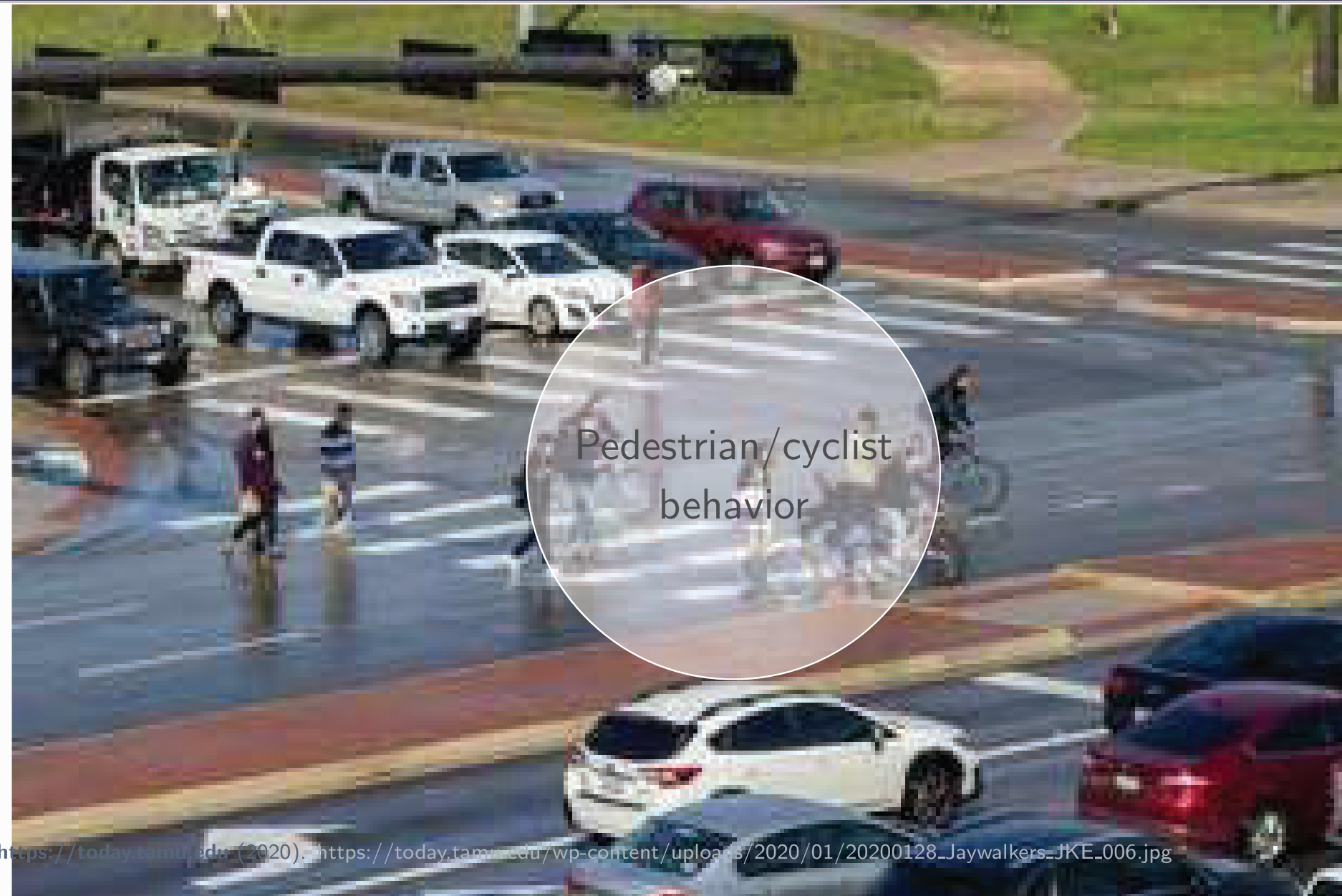
ESAT-STADIUS, KU Leuven

Introduction– The modern-day control system



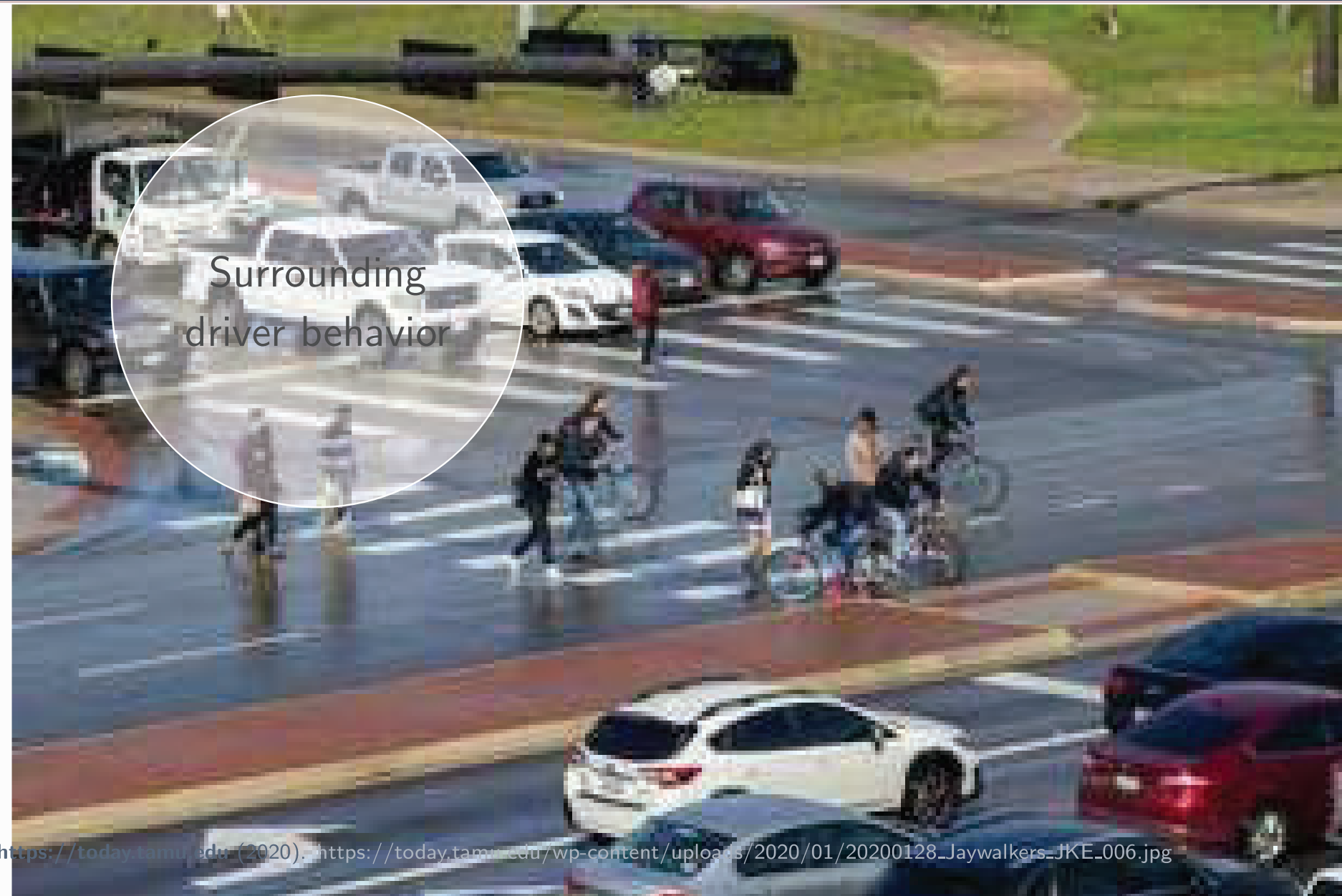
<https://today.tamuedu> (2020). https://today.tamuedu/wp-content/uploads/2020/01/20200128_Jaywalkers-JKE-006.jpg

Introduction– The modern-day control system

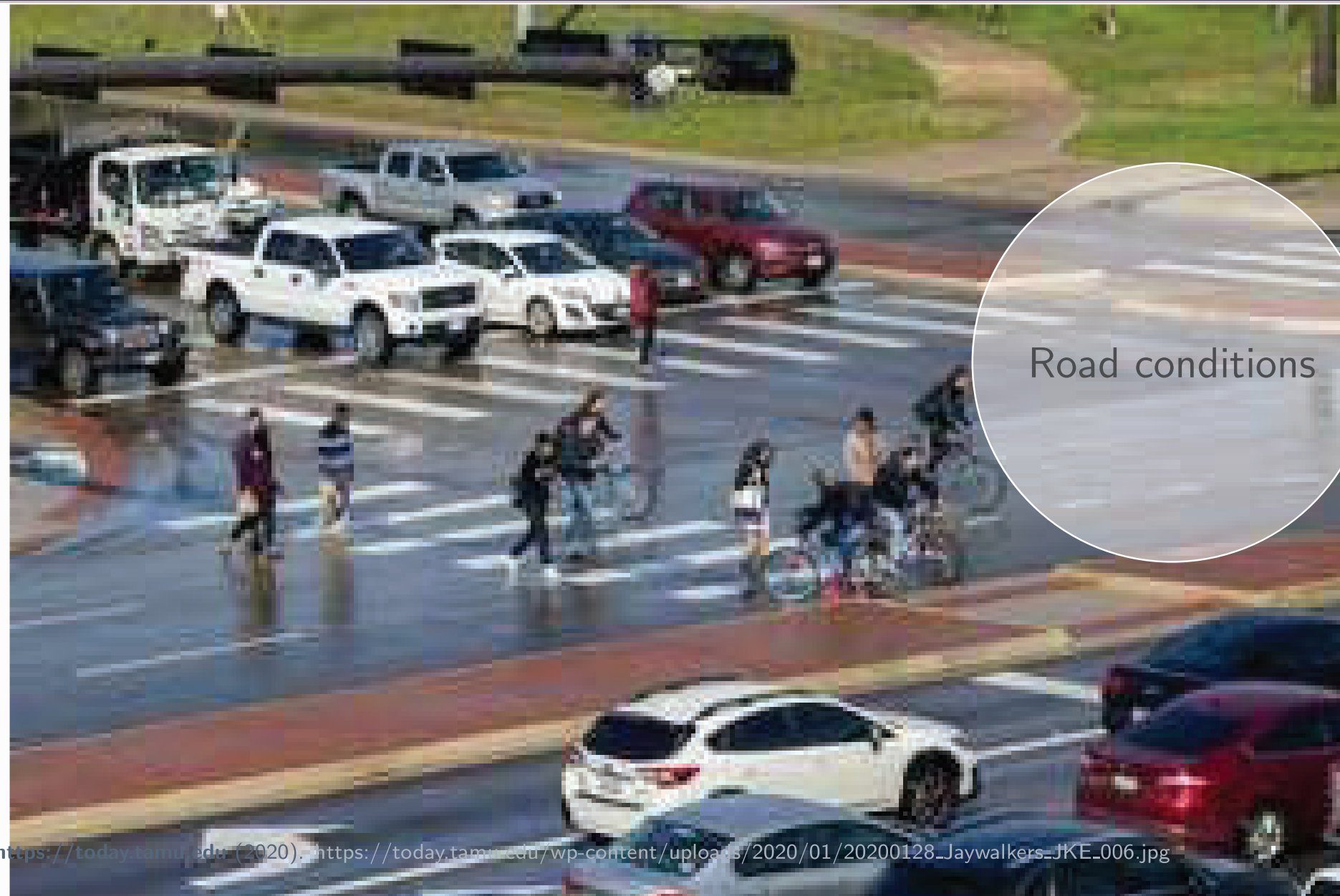


<https://today.tamuedu> (2020). https://today.tamuedu/wp-content/uploads/2020/01/20200128_Jaywalkers-JKE-006.jpg

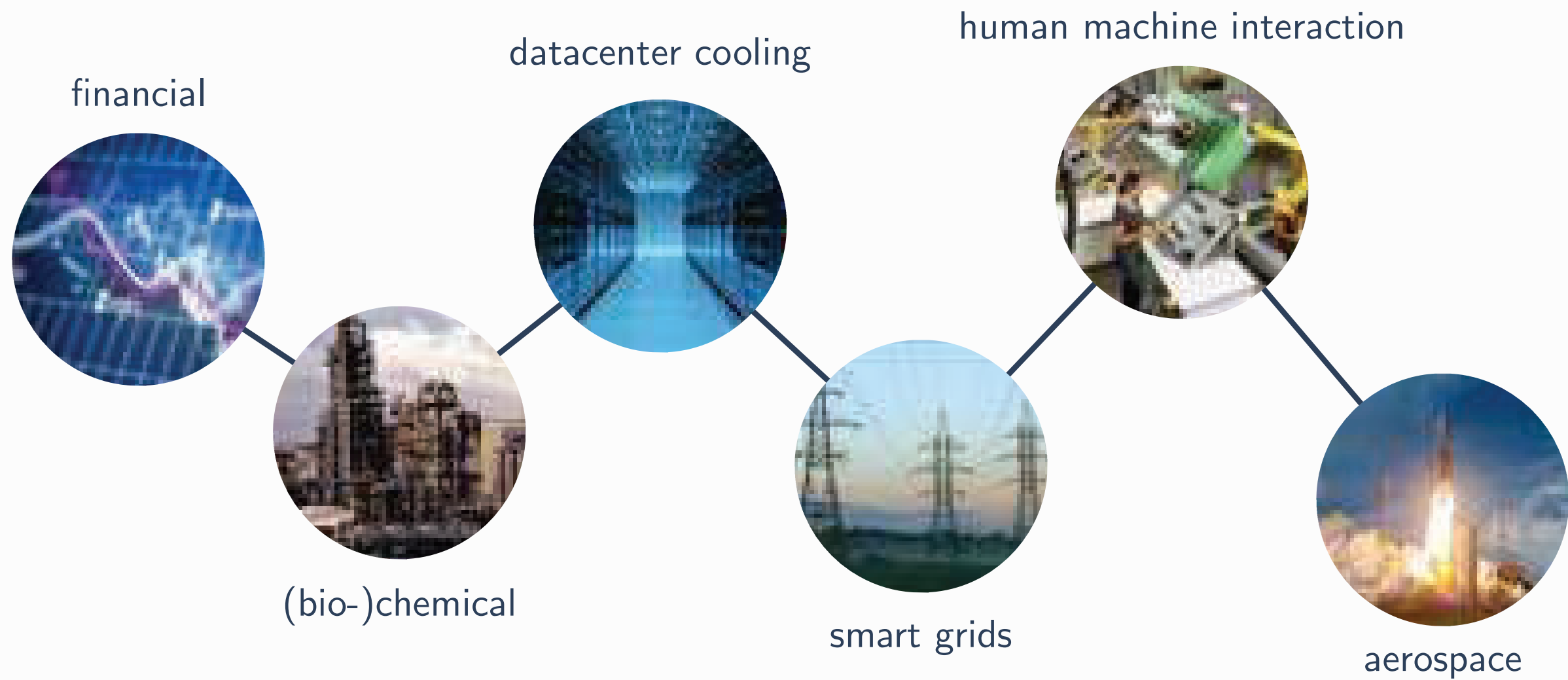
Introduction– The modern-day control system



Introduction– The modern-day control system

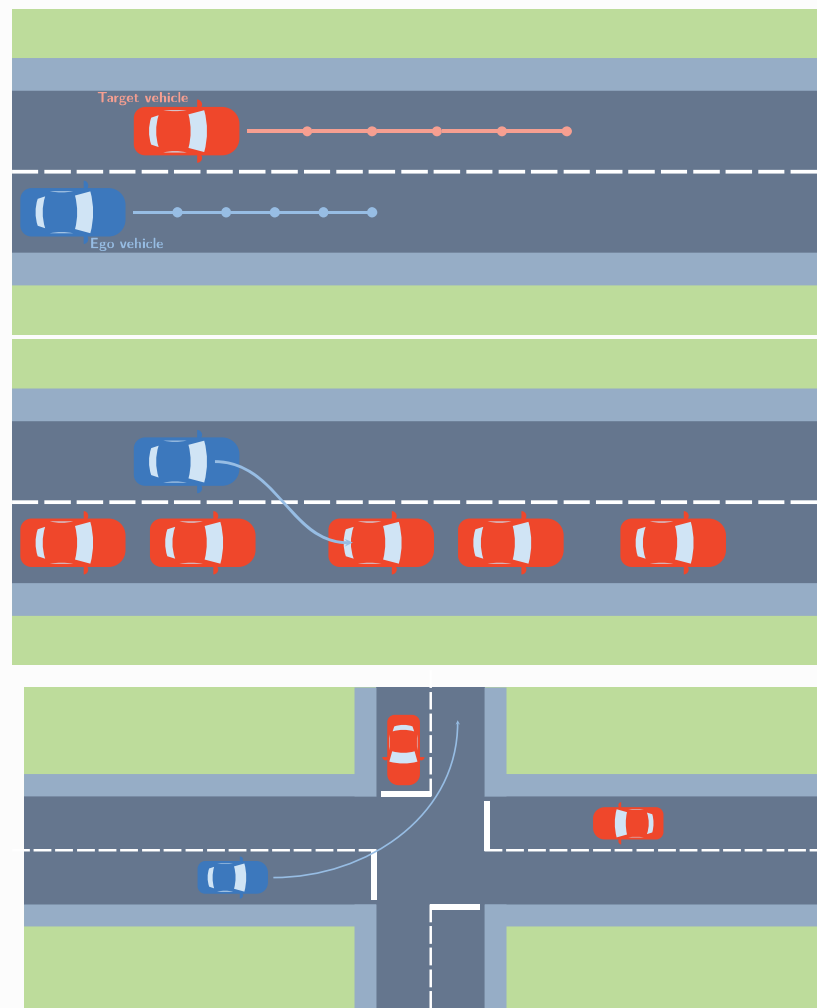


Introduction – Uncertain Systems



Introduction– The necessity of online adaptation

Planning requires prediction under uncertainty

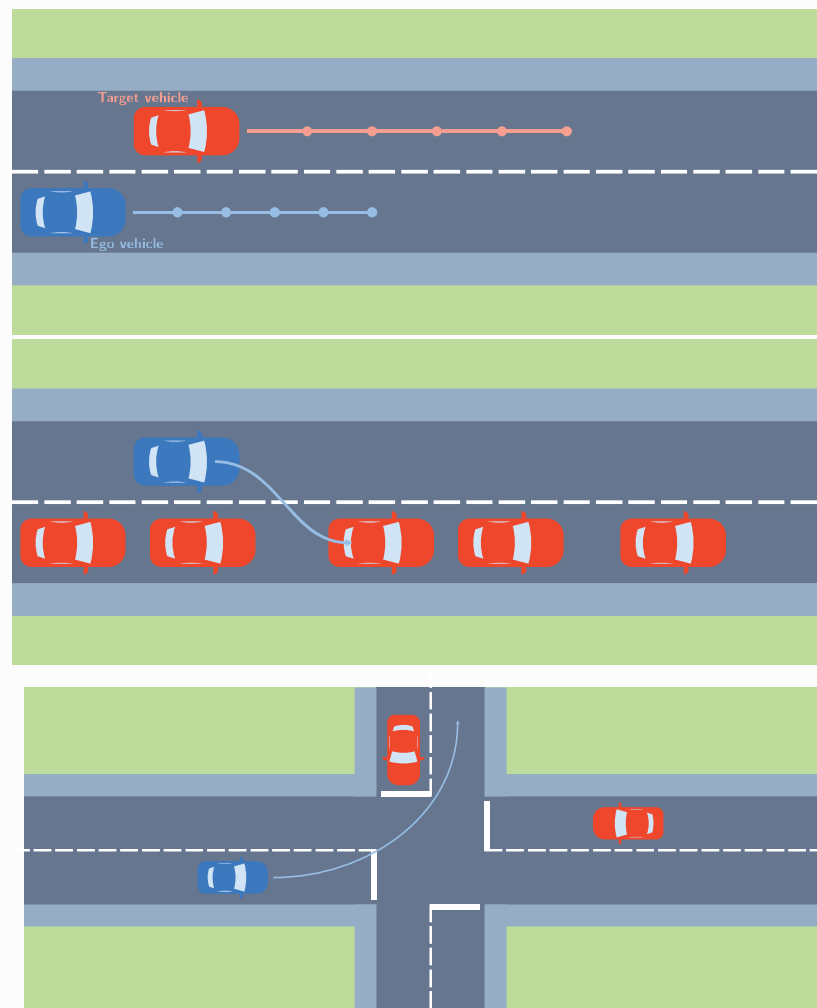


Uncertain elements **change over time**

- **weather** and **road** conditions vary over time

Introduction– The necessity of online adaptation

Planning requires prediction under uncertainty

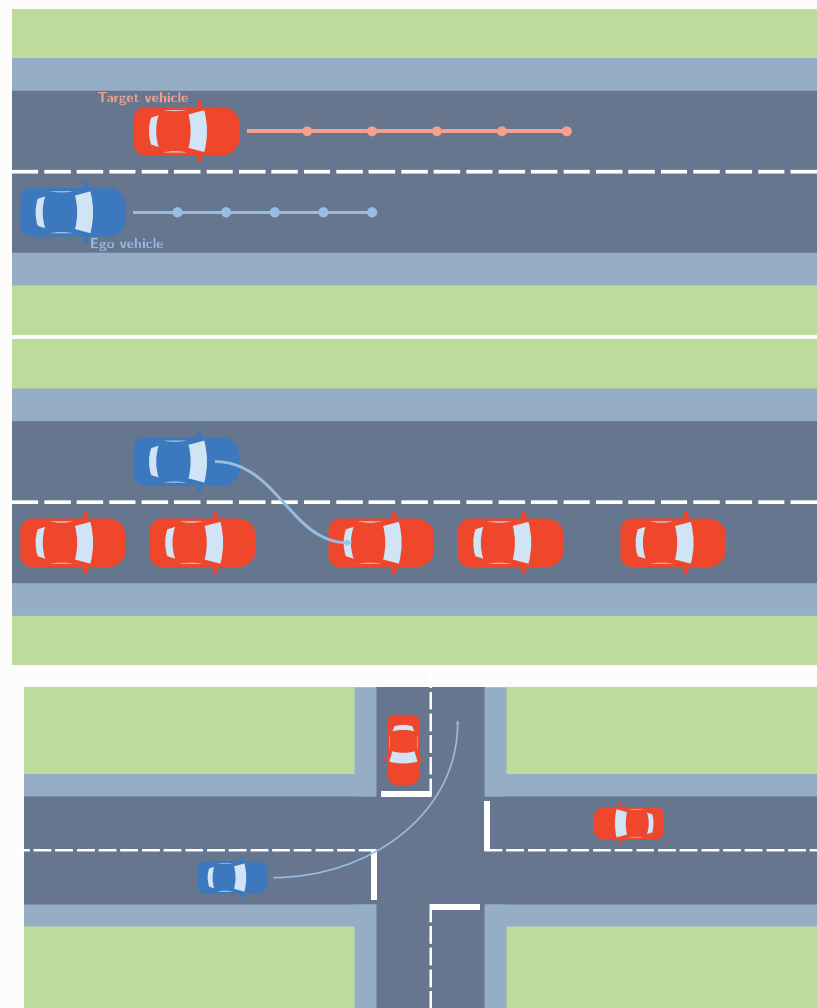


Uncertain elements **change over time**

- **weather** and **road** conditions vary over time
- variety of traffic **scenarios**

Introduction– The necessity of online adaptation

Planning requires prediction under uncertainty

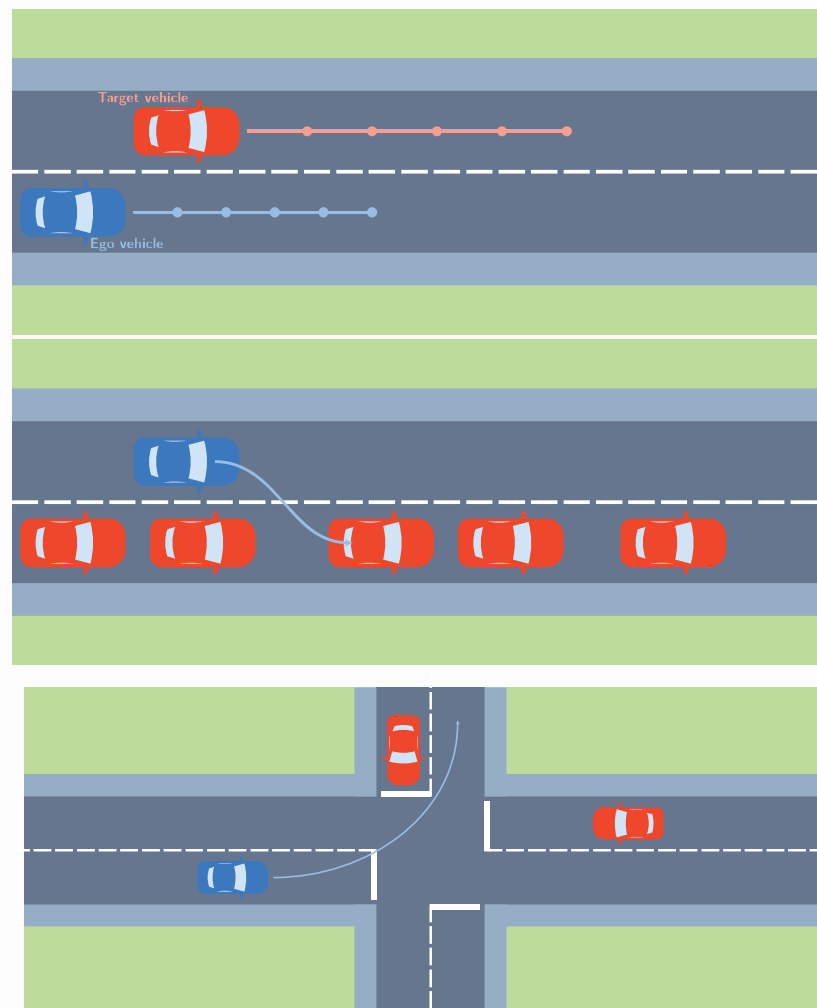


Uncertain elements **change over time**

- **weather** and **road** conditions vary over time
- variety of traffic **scenarios**
- we **interact only briefly** with any driver

Introduction– The necessity of online adaptation

Planning requires prediction under uncertainty



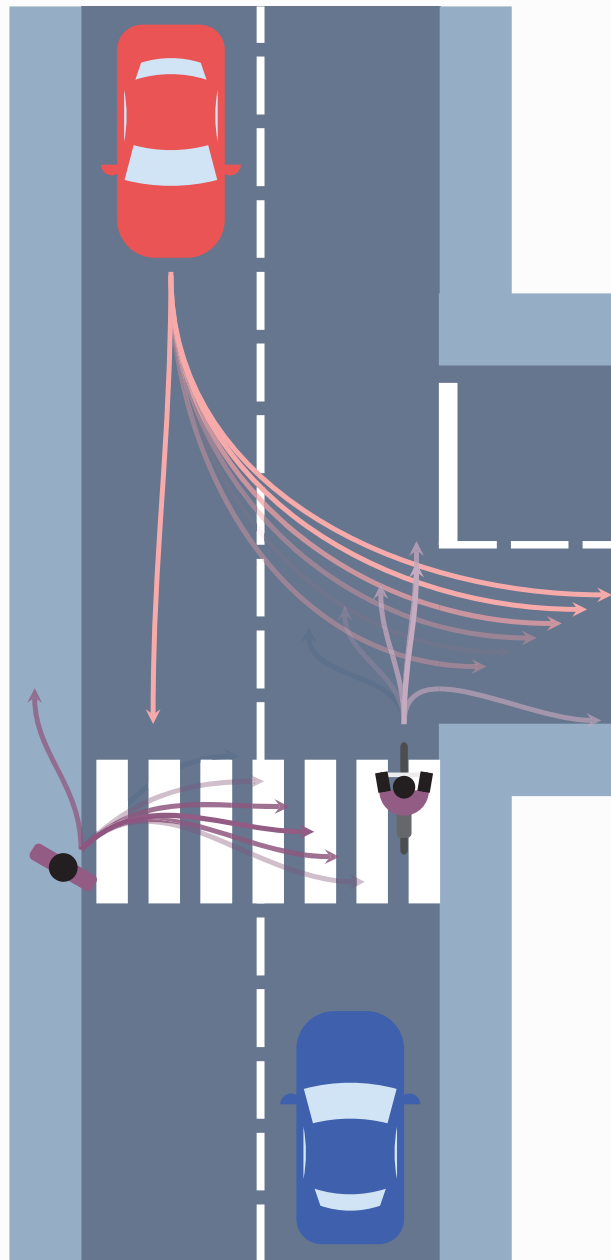
Uncertain elements **change over time**

- **weather** and **road** conditions vary over time
- variety of traffic **scenarios**
- we **interact only briefly** with any driver

→ need to **learn** from experience

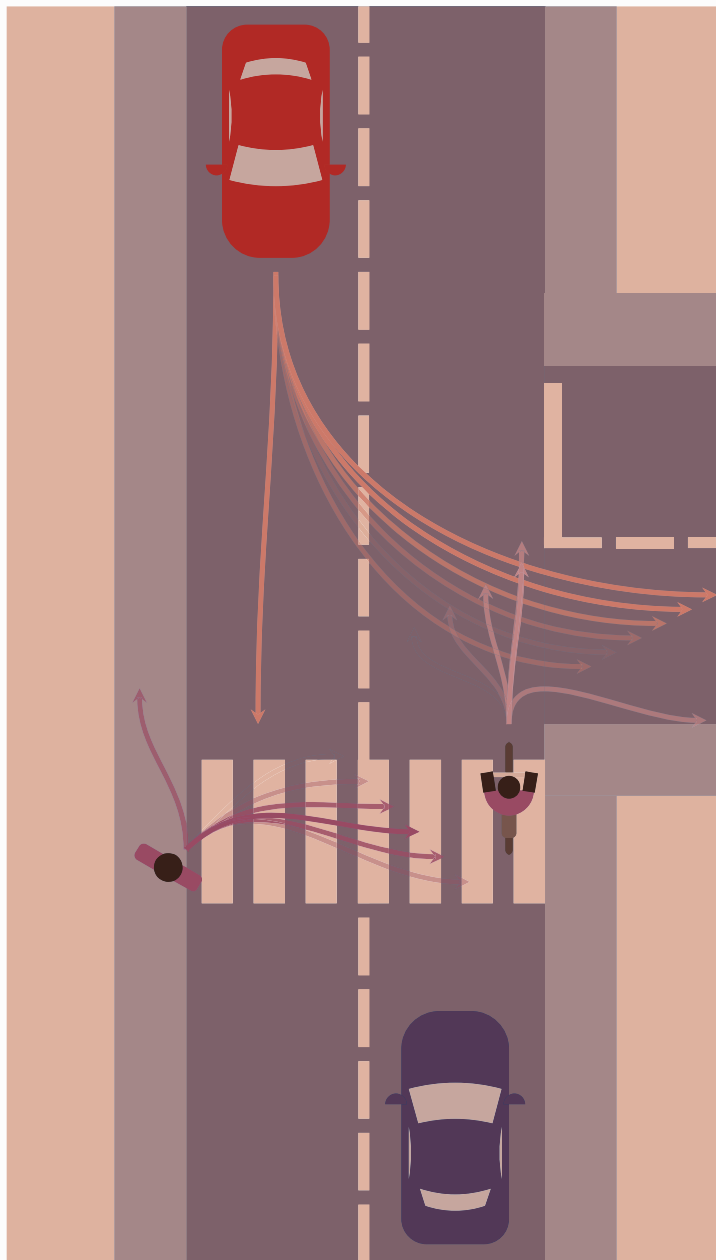
→ need to learn **online** and in **real-time**

Introduction – End-to-end learning



Why not just learn everything?

Introduction – End-to-end learning



Why not just learn everything?

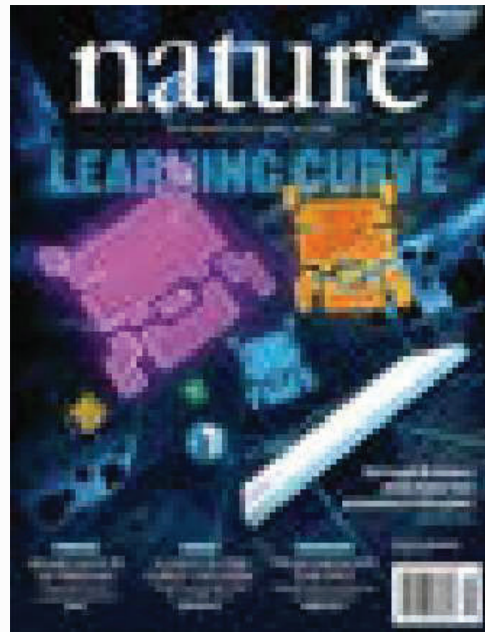

sensors

offline + online
learning

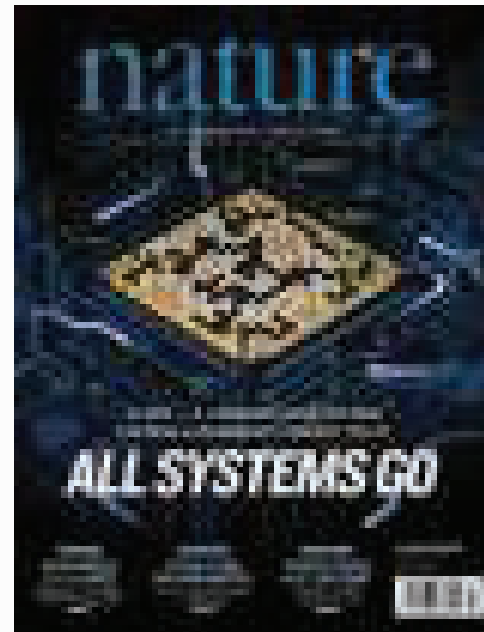

actuators

Full environment + control system as a **black box**

Introduction – End-to-end learning

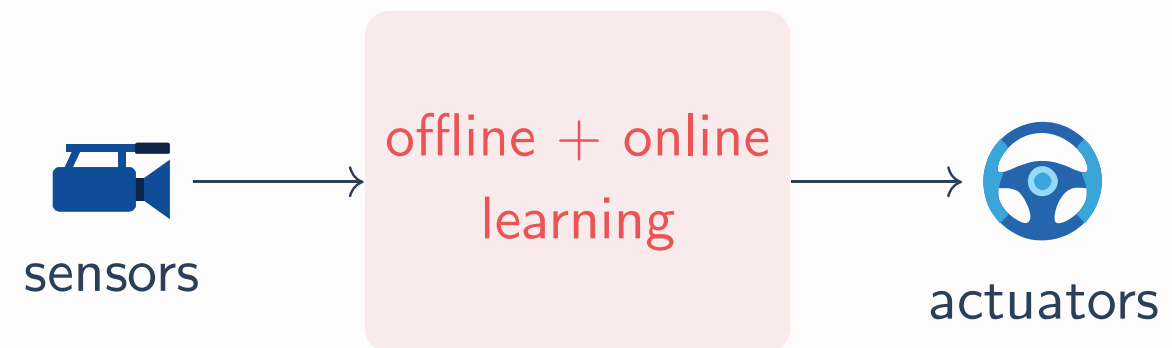


[Mnih et al. 2015]



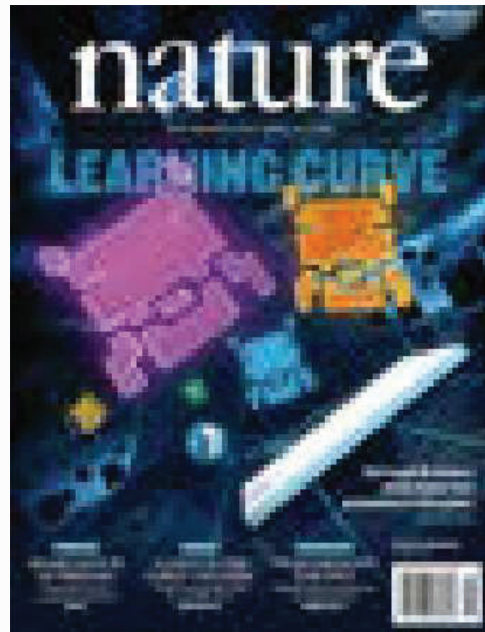
[Silver et al. 2016]

Why not just learn everything?

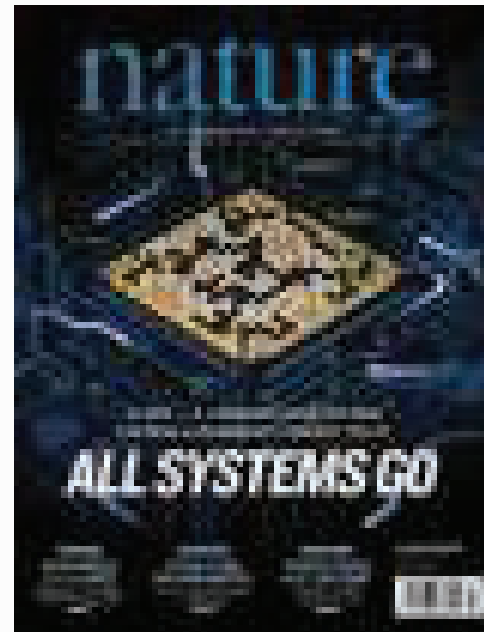


Full environment + control system as a **black box**

Introduction – End-to-end learning



[Mnih et al. 2015]

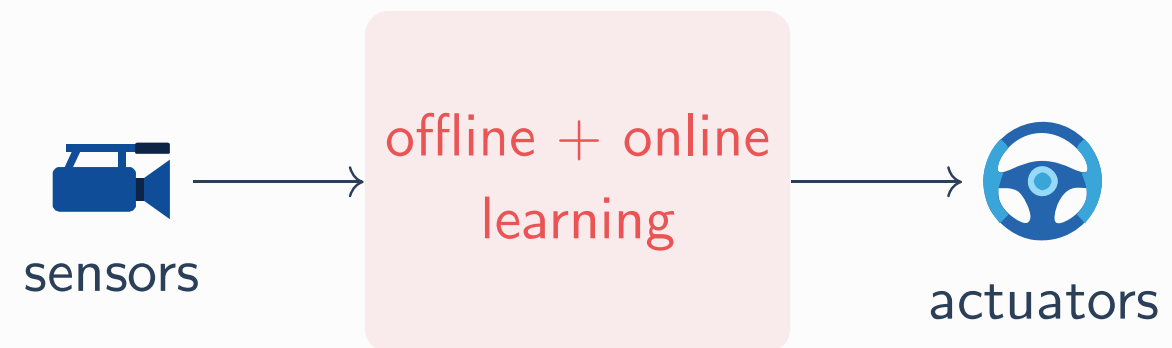


[Silver et al. 2016]

Game environments

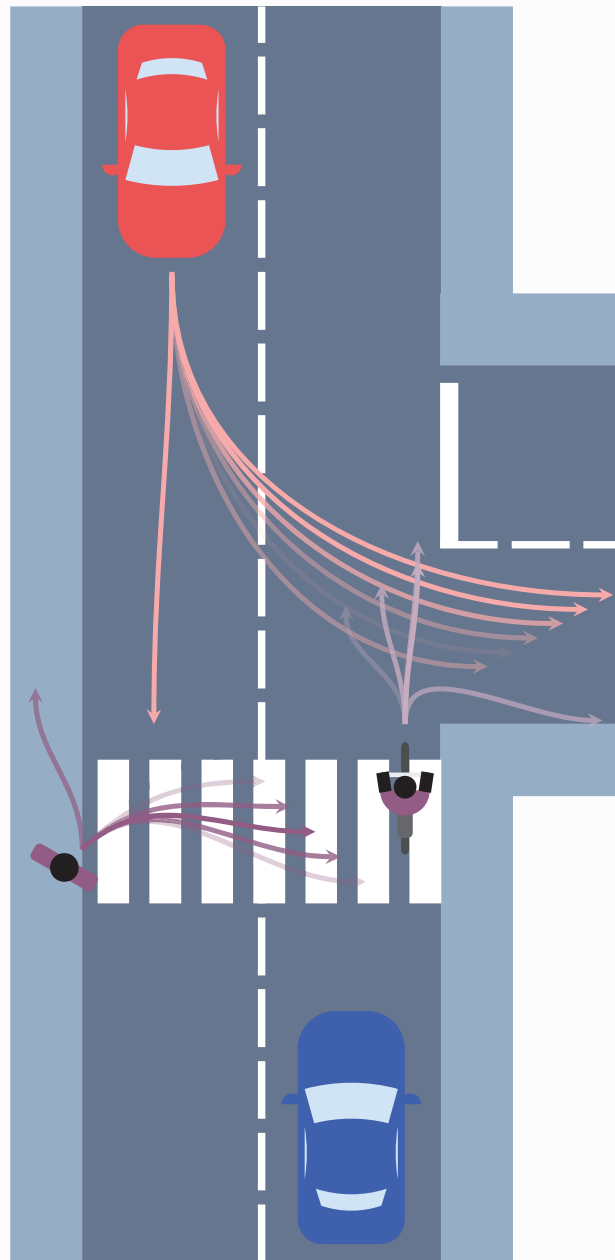
- **discrete** states/actions
- abundant **interactive data**
- **no safety** constraints

Why not just learn everything?

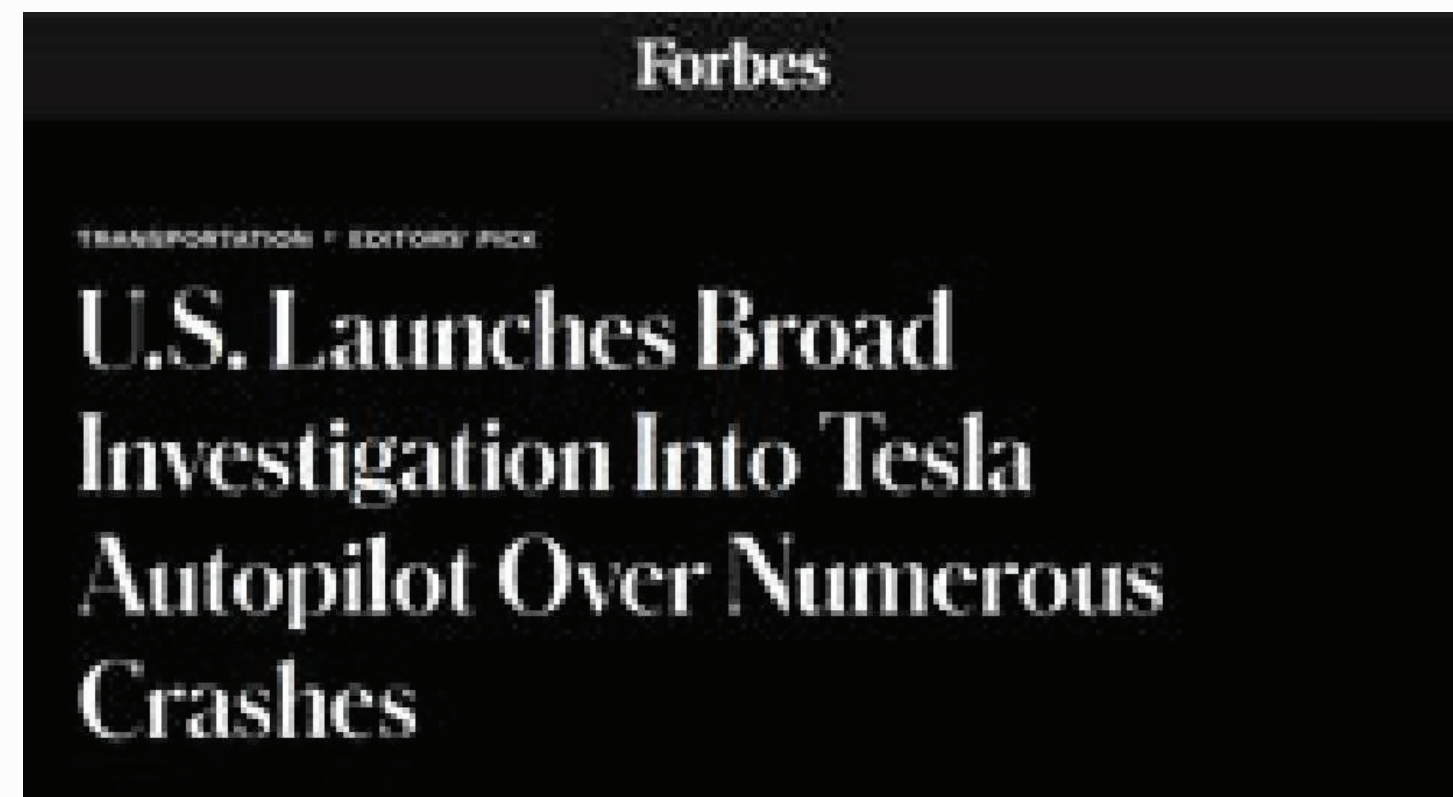


Full environment + control system as a **black box**

Introduction – End-to-end learning

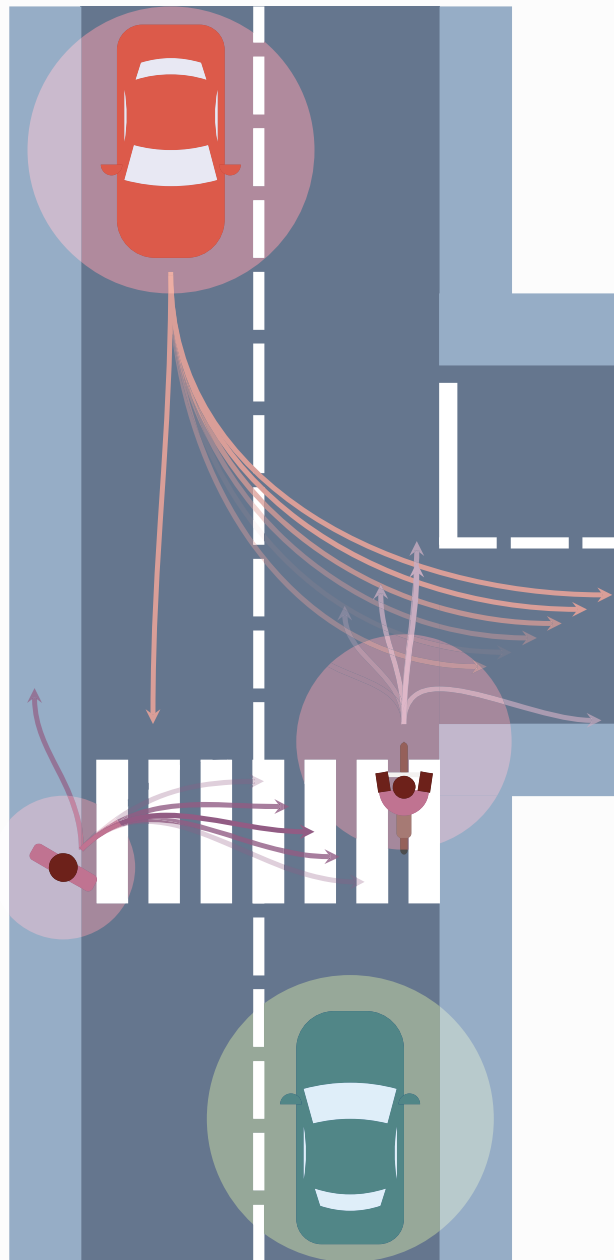


Real-world robotic systems

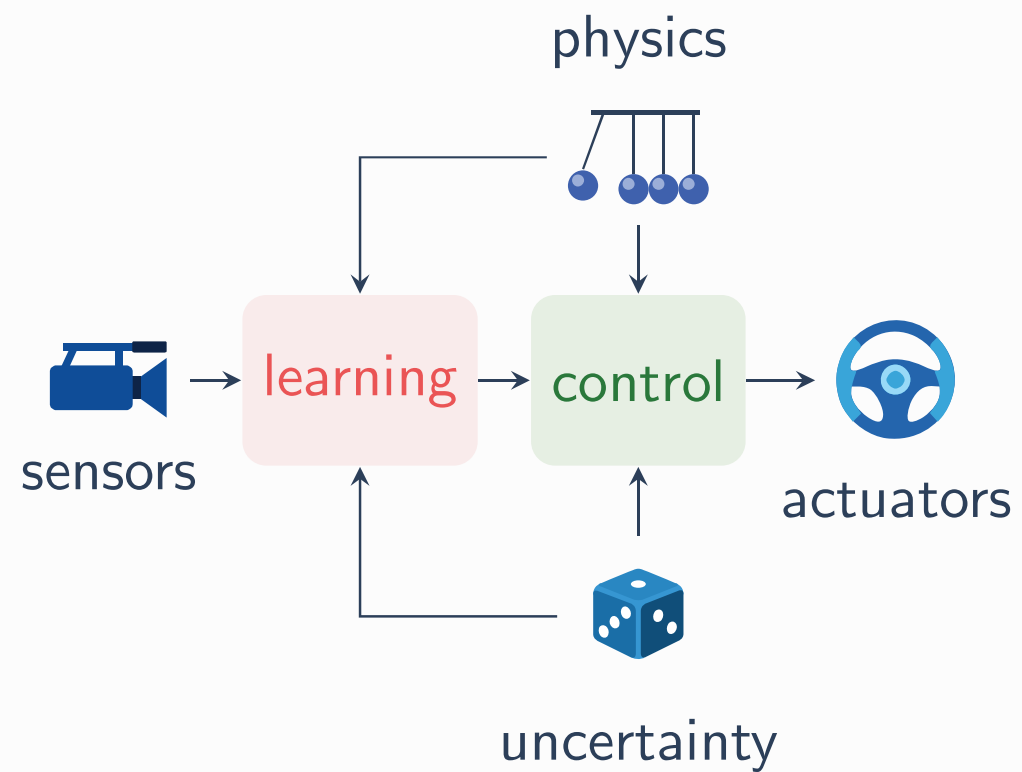


- end-to-end learning requires massive commercial **scale**
- **safety guarantees** are critical

Introduction – Combining Learning with Control Theory



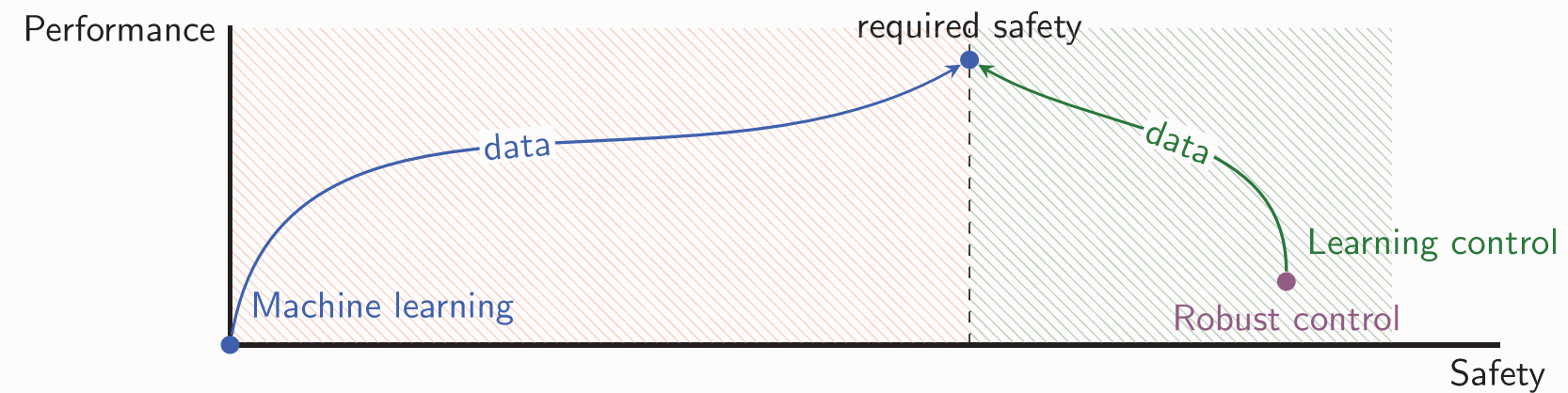
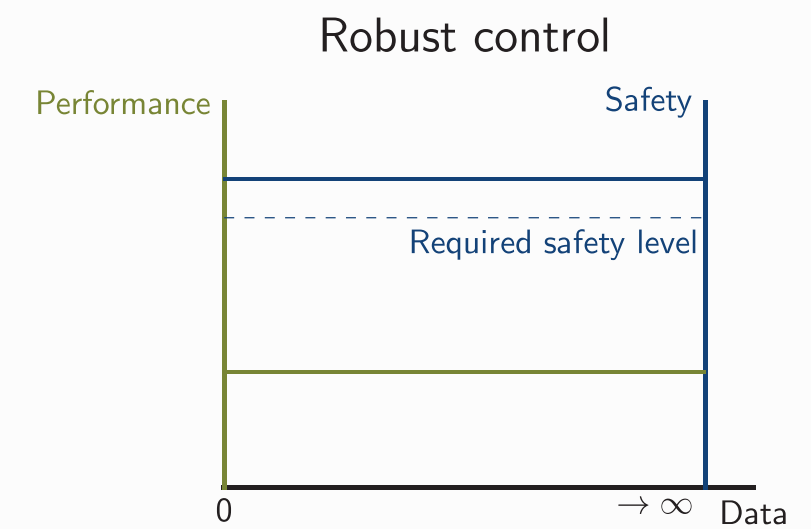
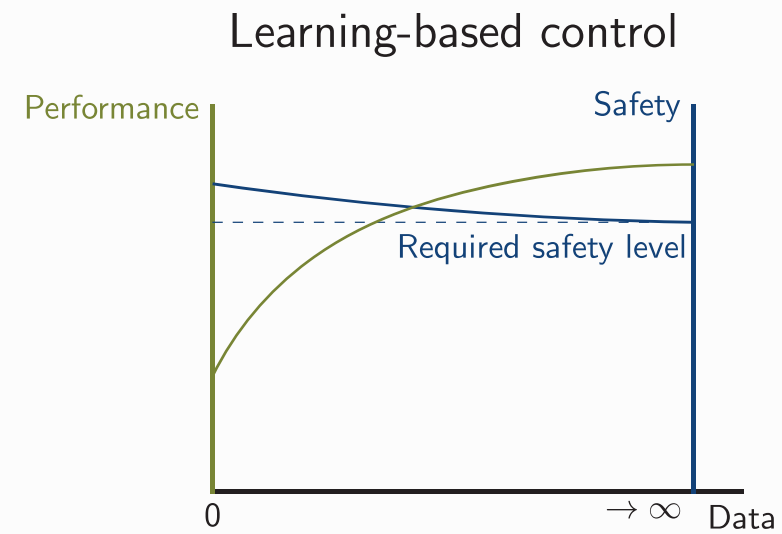
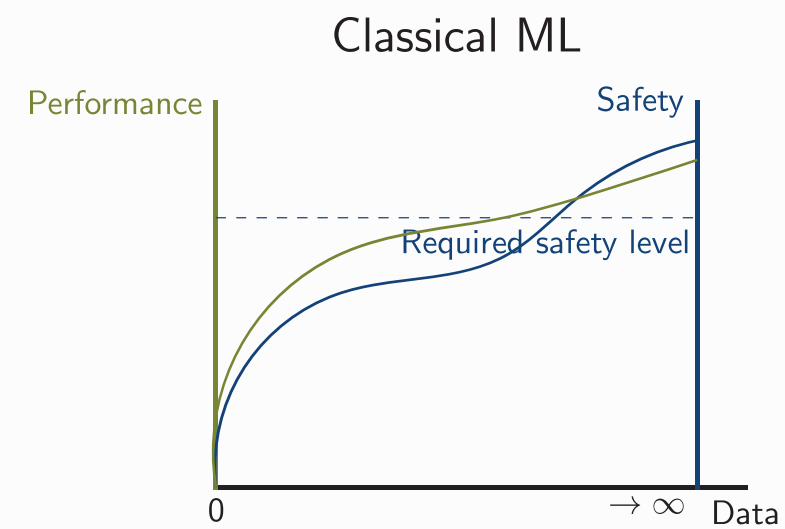
Exploit what we can model, learn what we can't



Sample-efficiency



Adaptation



Overview – Optimal Control

The diagram illustrates an optimal control problem. The main equation is $V_*(x_0) = \min_{\pi} \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right]$. A purple arrow labeled "stage cost (designer)" points from the text above to the term $\ell(x_t, \pi(x_t))$ inside the summation. A purple arrow labeled "stage constraints (designer)" points from the text below to the term $x_t \in X$ in the constraints. A green arrow labeled "dynamics (modeler)" points from the text below to the term $f(x_t, \pi(x_t), w_t)$ in the state transition equation. The constraints are given as $\text{s. t. } x_t \in X, \quad t \in \mathbb{N}$ and $x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N}$.

$$V_*(x_0) = \min_{\pi} \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right]$$

stage cost (designer)

stage constraints (designer)

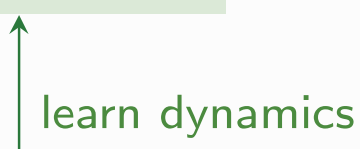
s. t. $x_t \in X, \quad t \in \mathbb{N}$

$x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N}$

dynamics (modeler)

Overview – Optimal Control

$$\begin{aligned} V_{\star}(x_0) = & \min_{\pi} \quad \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right] \\ \text{s. t.} \quad & x_t \in X, \quad t \in \mathbb{N} \\ & x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N} \end{aligned}$$


learn dynamics

Overview – Optimal Control

$$\begin{aligned} V_{\star}(x_0) = \min_{\pi} \quad & \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right] \\ \text{s. t.} \quad & x_t \in X, \quad t \in \mathbb{N} \\ & x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N} \end{aligned}$$

Diagram illustrating the optimal control problem with learning annotations:

- The objective function is $V_{\star}(x_0) = \min_{\pi} \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right]$.
- The constraints are $x_t \in X, \quad t \in \mathbb{N}$ and $x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N}$.
- The function f is highlighted in green, and the disturbance w_t is highlighted in pink.
- A purple arrow labeled "learn disturbance" points from the expectation operator $\mathbb{E}$ to the disturbance w_t .
- A green arrow labeled "learn dynamics" points from the function f to the state x_t .

Overview – Optimal Control

$$\begin{aligned}
 &\text{learn value} \rightarrow V_{\star}(x_0) = \min_{\pi} \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right] \\
 &\quad \text{s. t. } x_t \in X, \quad t \in \mathbb{N} \\
 &\quad x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N} \\
 &\quad \text{learn disturbance} \rightarrow w_t \\
 &\quad \text{learn dynamics} \rightarrow f
 \end{aligned}$$

$$\pi_{\star}(x) = \operatorname{argmin}_u \underbrace{\ell(x, u) + \mathbb{E}[V_{\star}(f(x, u, w))]}_{Q\text{-function}}.$$

P. Coppens and P. Patrinos (2023). “Policy iteration using Q-functions: Linear dynamics with multiplicative noise”.
In: IFAC (submitted). arXiv: 2212.01192.

Overview – Optimal Control

The diagram illustrates the optimal control problem with four learning annotations: 'learn value' (blue arrow to $V_*(x_0)$), 'learn policy' (red arrow to $\pi(x_t)$), 'learn disturbance' (purple arrow to w_t), and 'learn dynamics' (green arrow to the function f). The equations are as follows:

$$V_*(x_0) = \min_{\pi} \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right]$$

s. t. $x_t \in X, \quad t \in \mathbb{N}$

$$x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N}$$

Overview – Optimal Control

$$\begin{aligned}
 &\text{learn value} \rightarrow V_*(x_0) = \min_{\pi} \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right] \\
 &\text{s. t. } x_t \in X, \quad t \in \mathbb{N} \\
 &x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N} \\
 &\text{learn policy} \rightarrow \pi(x_t) \\
 &\text{learn disturbance} \rightarrow w_t \\
 &\text{learn dynamics} \rightarrow f(x_t, \pi(x_t), w_t)
 \end{aligned}$$

modeling

- optimize
- assumptions
- + prior info
- + safe



learning

- + evaluate
- + flexible
- data hungry
- unsafe

Overview

Introduction

Overview

Learning Distribution

Learning Dynamics

Learning Policies

Numerical Algorithms

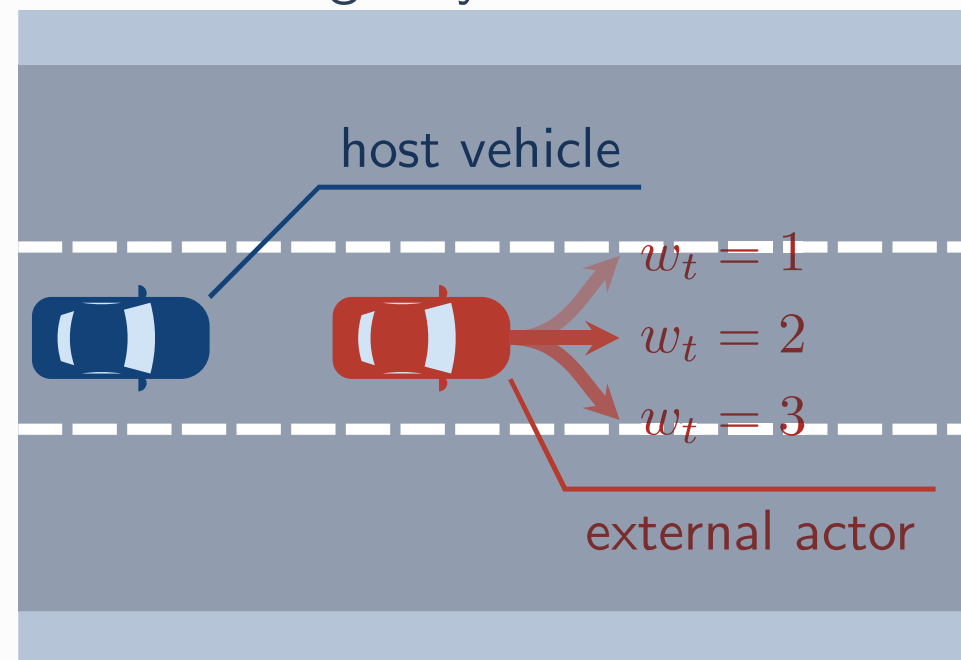
Conclusion

References

Learning Distribution – Multi-modal MPC



Ex.: highway lane selection



Driver selects a discrete *target* lane $w_t \in W$

Discrete-valued uncertainty:

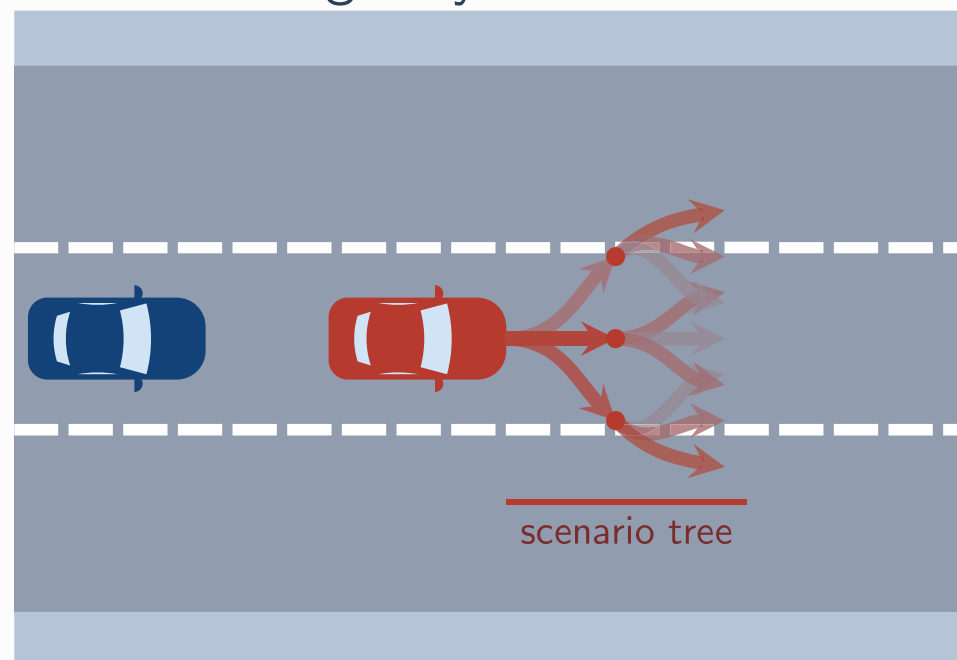
$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC

Ex.: highway lane selection



Driver selects a discrete *target* lane $w_t \in W$

Discrete-valued uncertainty:

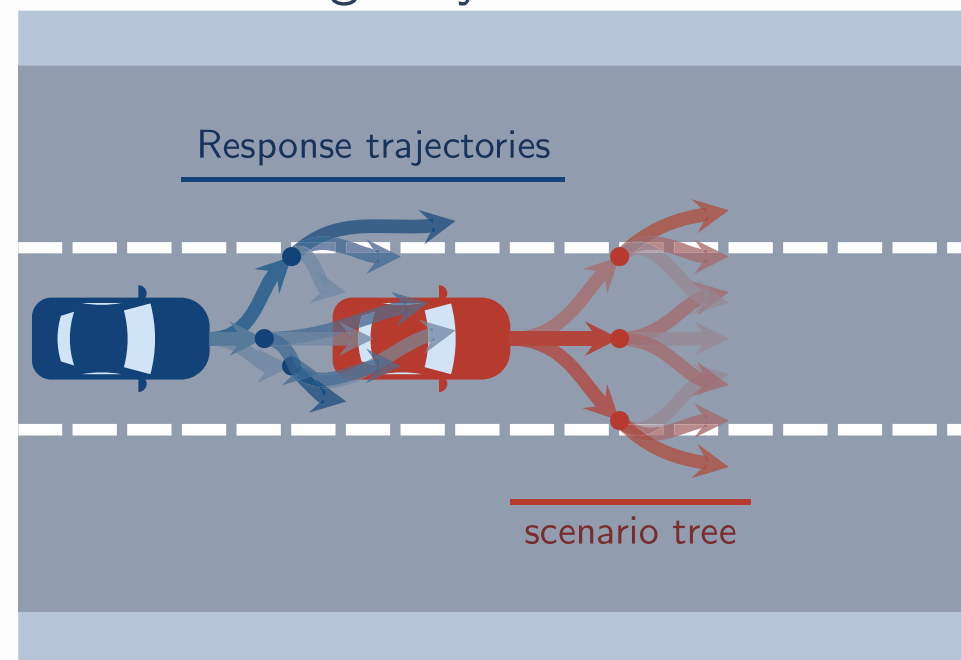
$$x_{t+1} = f(x_t, \pi(x_t), \boxed{w_t})$$

$w_t \in W = \{1, \dots, d\}$

Scenario tree. representation of predicted trajectories (captures multi-modal distributions!)

Learning Distribution – Multi-modal MPC

Ex.: highway lane selection



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

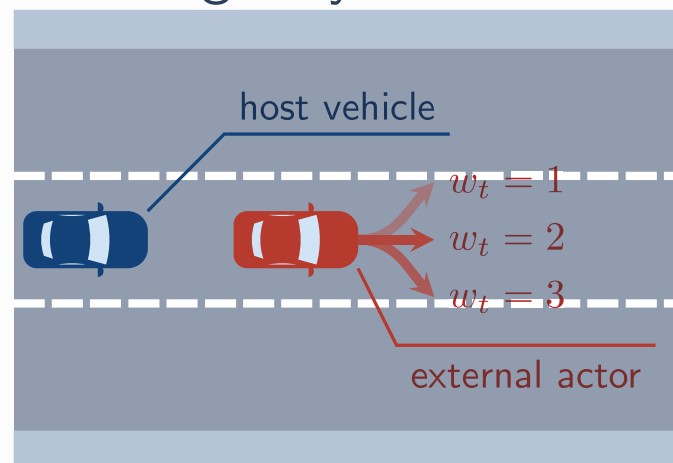
Scenario tree. representation of predicted trajectories (captures multi-modal distributions!)

Goal. Plan safe and efficient policy accounting for uncertainty in joint dynamics f

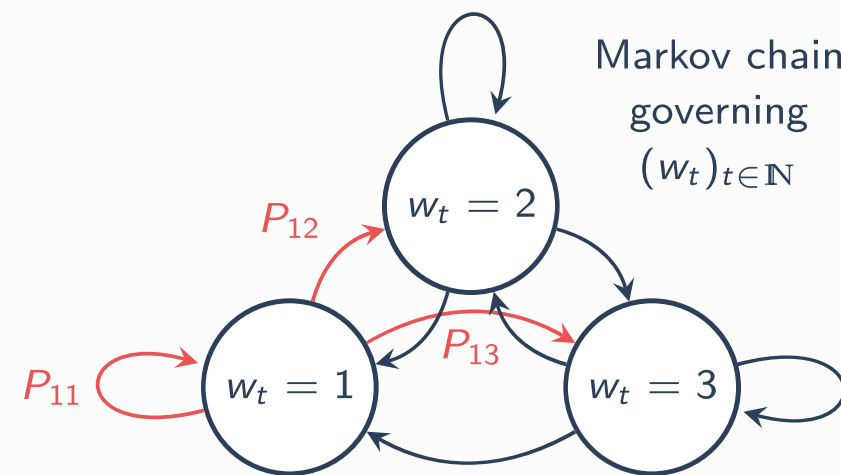
Driver selects a discrete *target* lane $w_t \in W$

Learning Distribution – Multi-modal MPC

Ex.: highway lane selection



Driver selects a discrete *target* lane $w_t \in W$



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

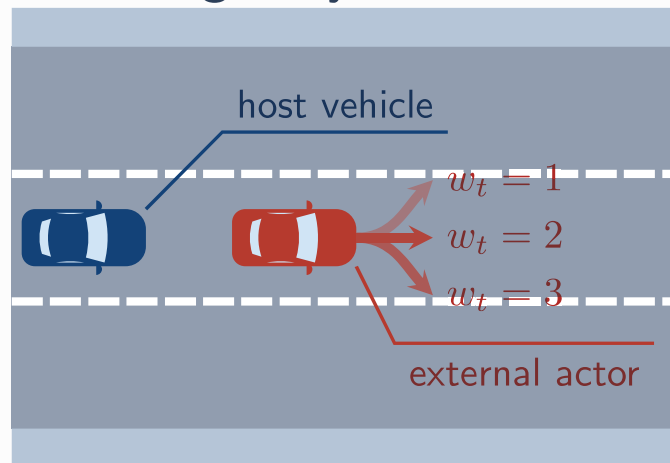
$(w_t)_{t \in \mathbb{N}}$ may be governed by

- i.i.d. process
- Markov chain
- state-dependent distribution (See later ...)

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC

Ex.: highway lane selection



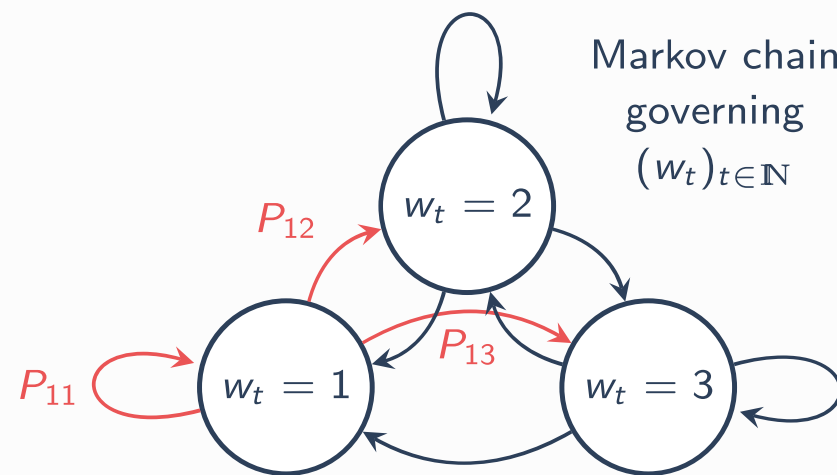
Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), \boxed{w_t})$$

$w_t \in W = \{1, \dots, d\}$

Stochastic optimal control

Driver selects a discrete *target* lane $w_t \in W$



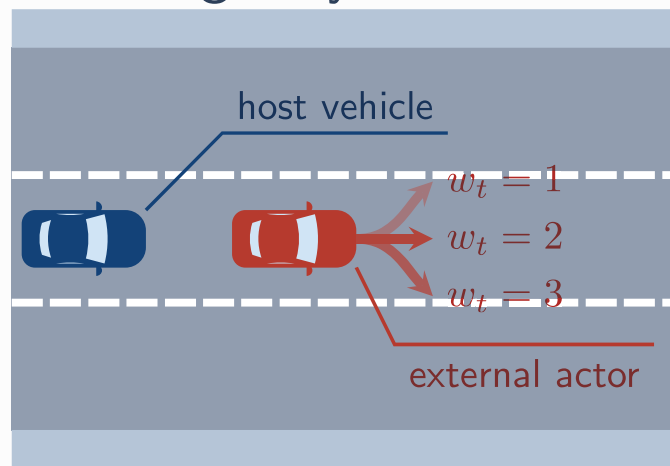
$$V_{\star}(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim \boxed{P_w}} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim \boxed{P_w}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

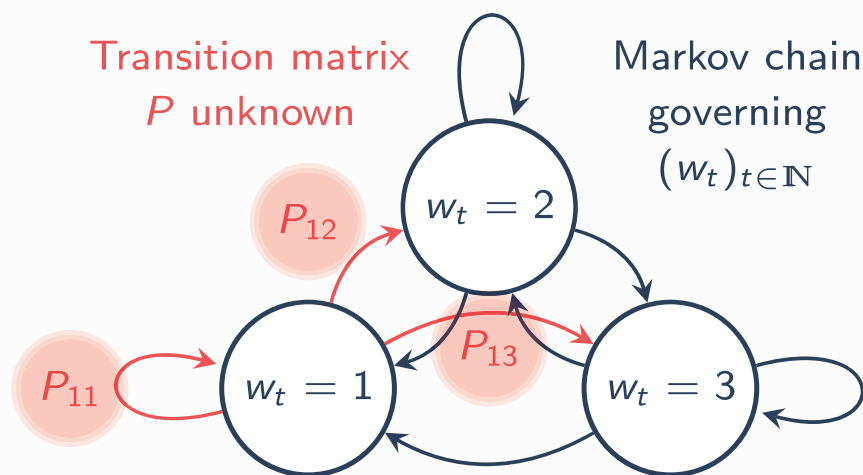
M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC

Ex.: highway lane selection



Driver selects a discrete *target* lane $w_t \in W$



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

Stochastic optimal control

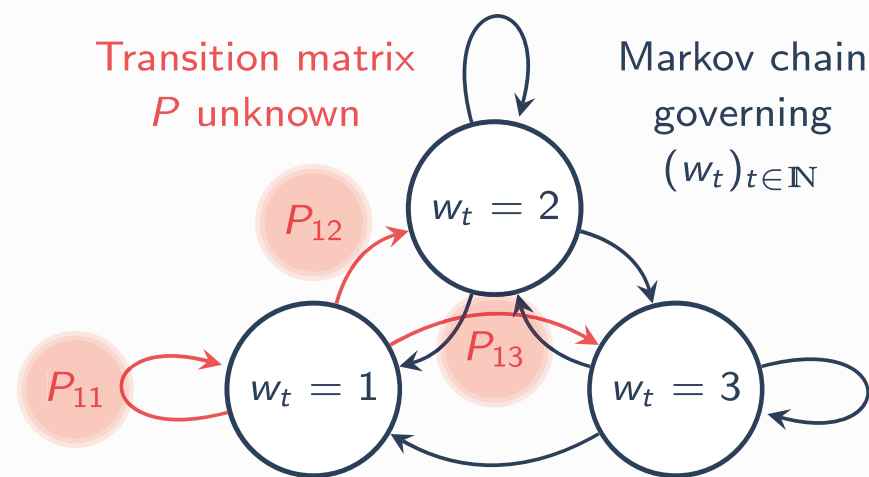
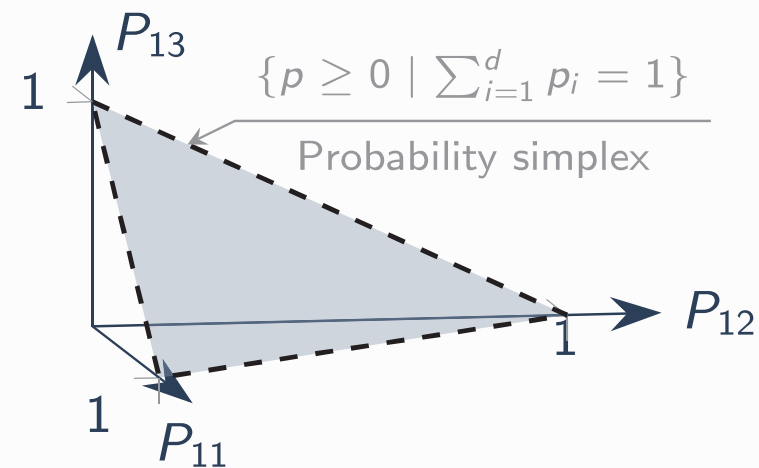
$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim P_w} [\hat{V}(f(x, u, v), v)]$$

Unknown!

$$\text{subj. to } \mathbb{P}_{v \sim P_w} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

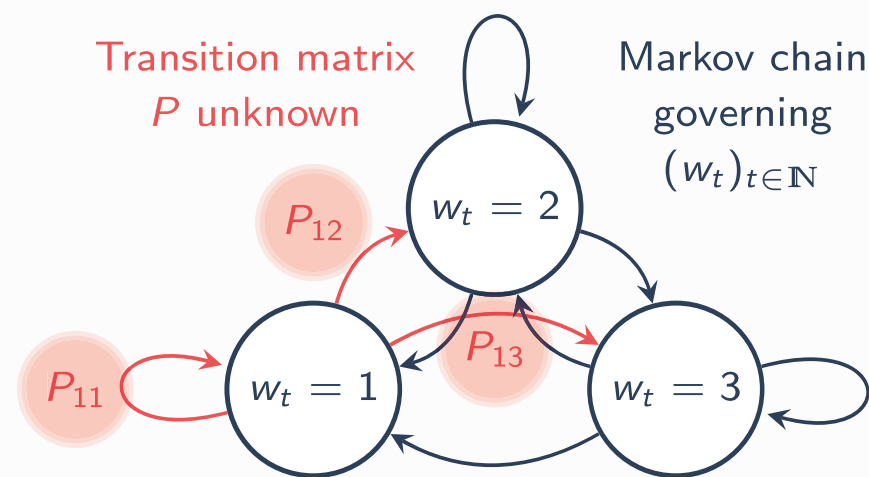
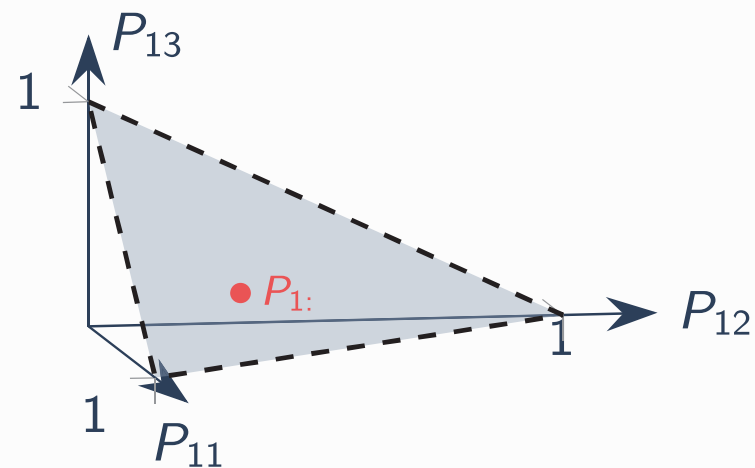
Stochastic optimal control

$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim P_w} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim P_w} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

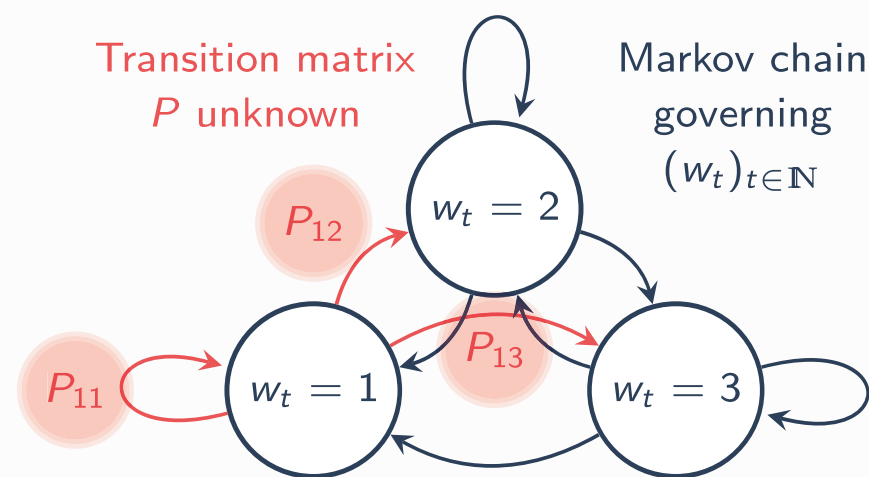
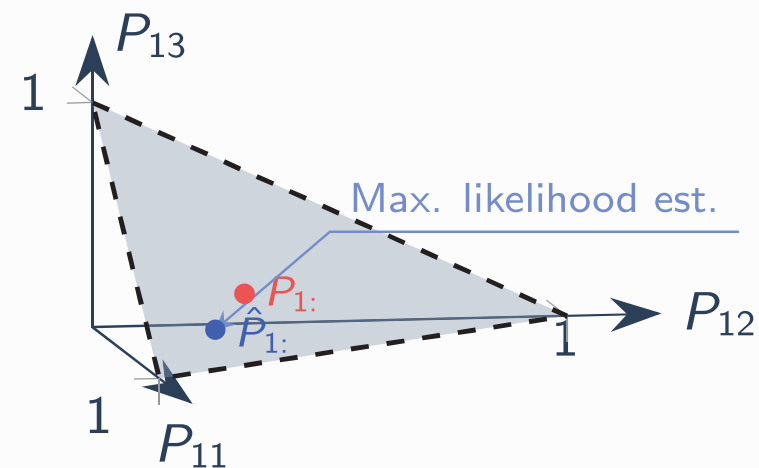
Stochastic optimal control

$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim P_w} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim P_w} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC



Discrete-valued uncertainty:

$$w_t \in W = \{1, \dots, d\}$$

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

Stochastic optimal control

$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim P_{w:}} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim P_{w:}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

$$\approx \min_u \ell(x, u) + \mathbb{E}_{v \sim \hat{P}_{w:}} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim \hat{P}_{w:}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC

Discrete-valued uncertainty:

$$w_t \in W = \{1, \dots, d\}$$

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

Stochastic optimal control

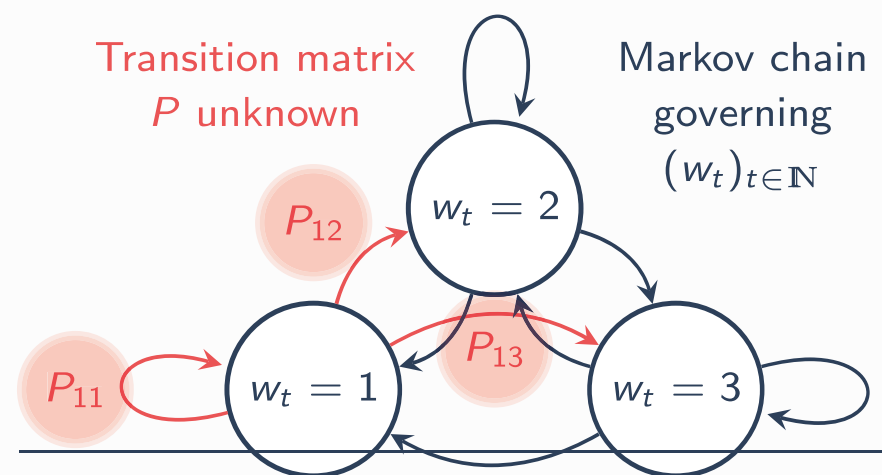
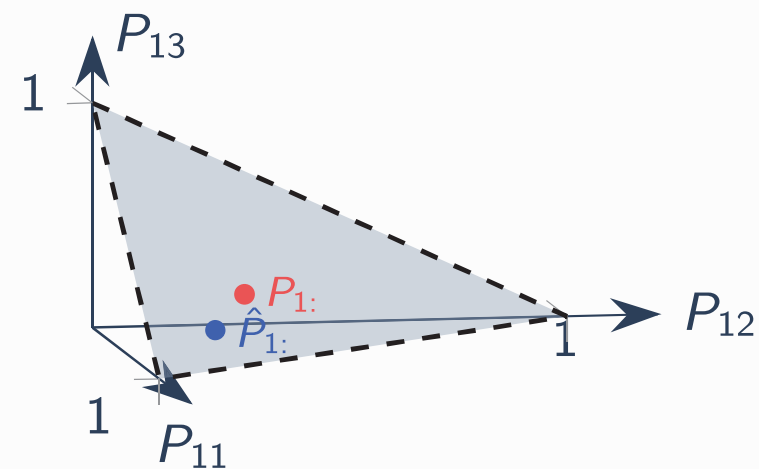
$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim P_{w:}} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim P_{w:}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

$$\approx \min_u \ell(x, u) + \mathbb{E}_{v \sim \hat{P}_{w:}} [\hat{V}(f(x, u, v), v)]$$

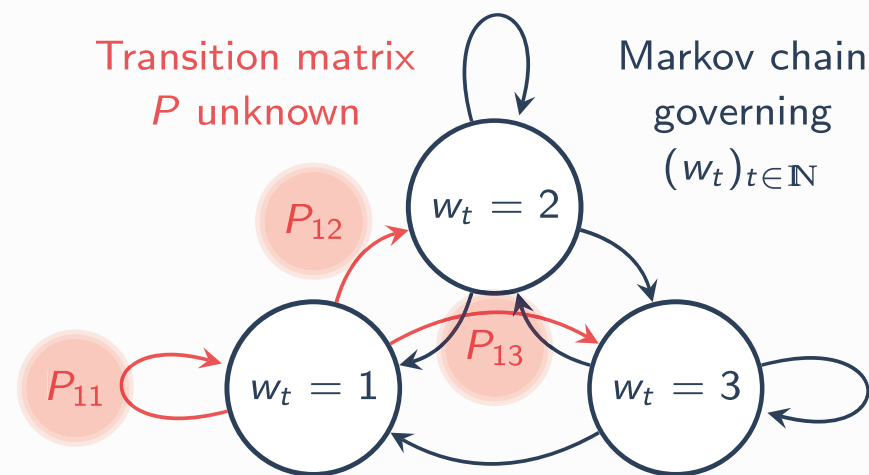
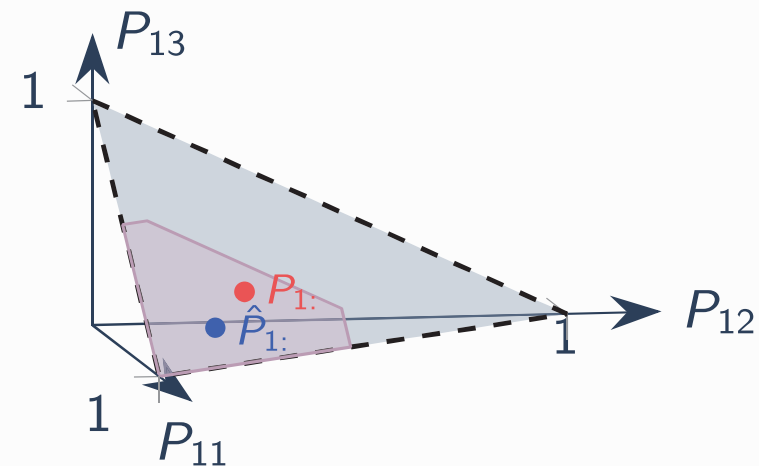
$$\text{subj. to } \mathbb{P}_{v \sim \hat{P}_{w:}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

No guarantees!



M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$w_t \in W = \{1, \dots, d\}$

Stochastic optimal control

$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim P_{w:}} [\hat{V}(f(x, u, v), v)]$$

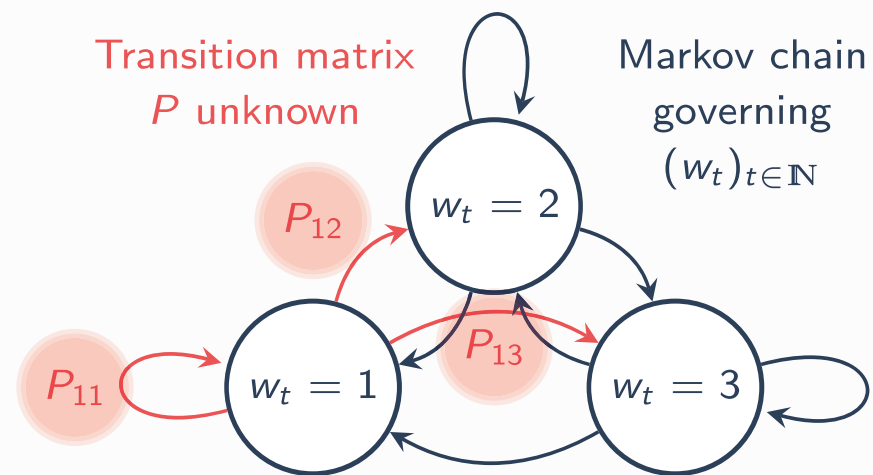
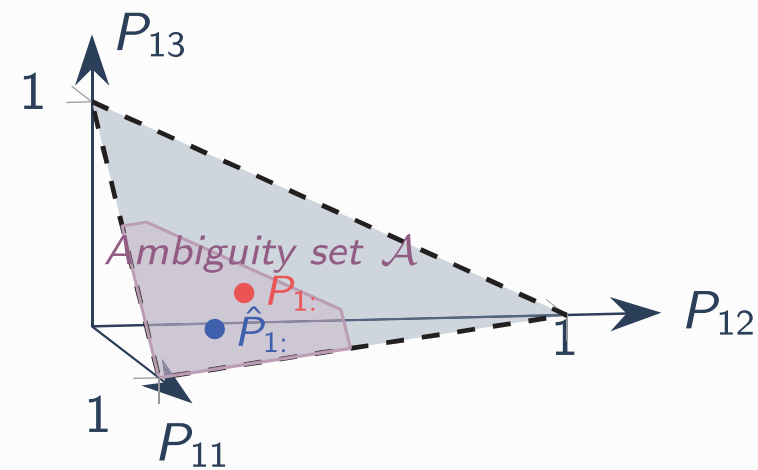
$$\text{subj. to } \mathbb{P}_{v \sim P_{w:}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

$$\approx \min_u \ell(x, u) + \mathbb{E}_{v \sim \hat{P}_{w:}} [\hat{V}(f(x, u, v), v)]$$

$$\text{subj. to } \mathbb{P}_{v \sim \hat{P}_{w:}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), \boxed{w_t})$$

$w_t \in W = \{1, \dots, d\}$

Stochastic optimal control

$$V_*(x, w) = \min_u \ell(x, u) + \mathbb{E}_{v \sim \boxed{P_{w:}}} [\hat{V}(f(x, u, v), v)]$$

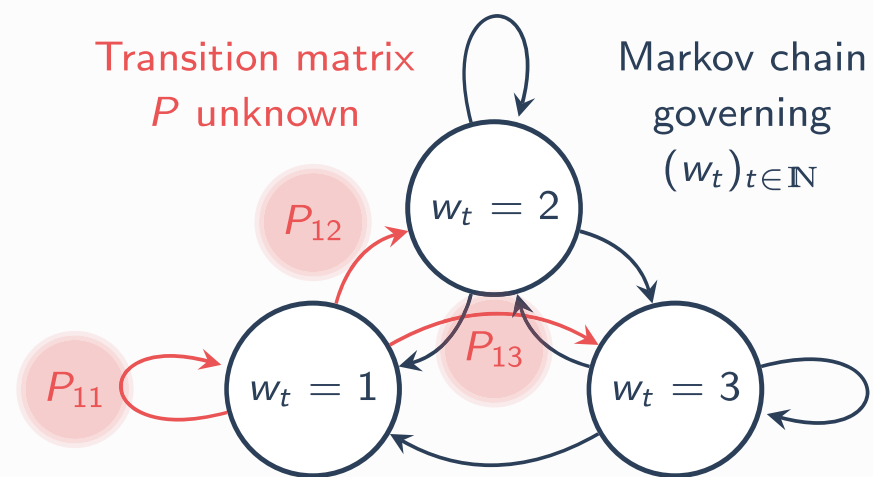
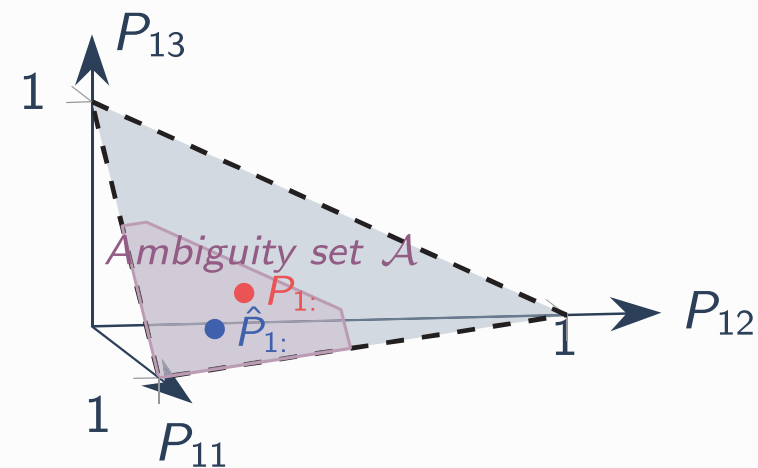
$$\text{subj. to } \mathbb{P}_{v \sim \boxed{P_{w:}}} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

$$\leq \min_u \ell(x, u) + \max_{p \in \mathcal{A}} \mathbb{E}_{v \sim p} [\hat{V}'(f(x, u, v), v)]$$

$$\text{subj. to } \min_{p \in \mathcal{A}} \mathbb{P}_{v \sim p} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Learning Distribution – Multi-modal MPC



Discrete-valued uncertainty:

$$x_{t+1} = f(x_t, \pi(x_t), \boxed{w_t})$$

$w_t \in W = \{1, \dots, d\}$

Stochastic optimal control

$$V_*(x, w) \leq \min_u \ell(x, u) + \max_{p \in \mathcal{A}} \mathbb{E}_{v \sim p} [\hat{V}'(f(x, u, v), v)]$$

$$\text{subj. to } \min_{p \in \mathcal{A}} \mathbb{P}_{v \sim p} [g(f(x, u, v)) \leq 0] \geq 1 - \epsilon \quad (\text{Safety})$$

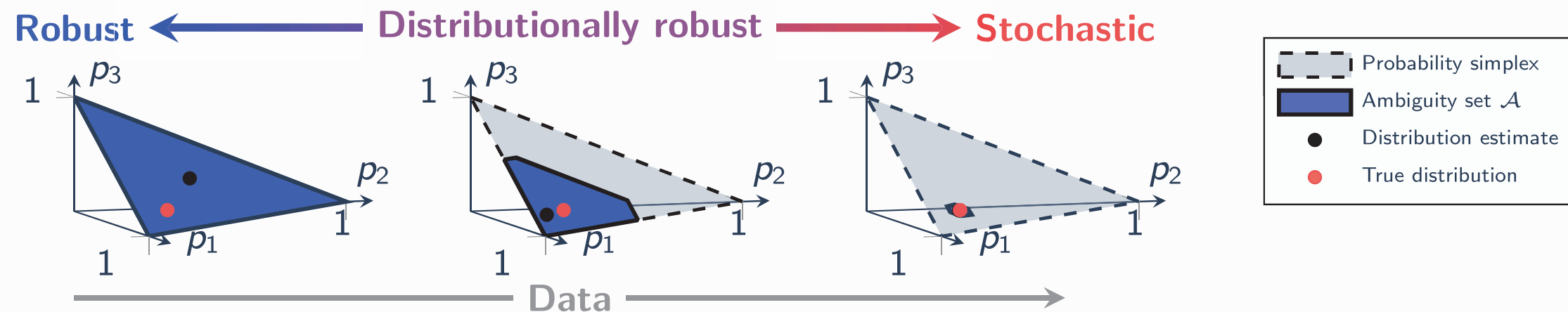
Robustifies against misestimations of the distribution.

- Safety (constraint satisfaction)
- Stability



M. Schuurmans and P. Patrinos (2023). "A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems". In: **TAC**. to appear. arXiv: 2106.00561

Distributionally Robust Optimization – Between Stochastic and Robust



Robust

$$\mathcal{A} = \Delta$$

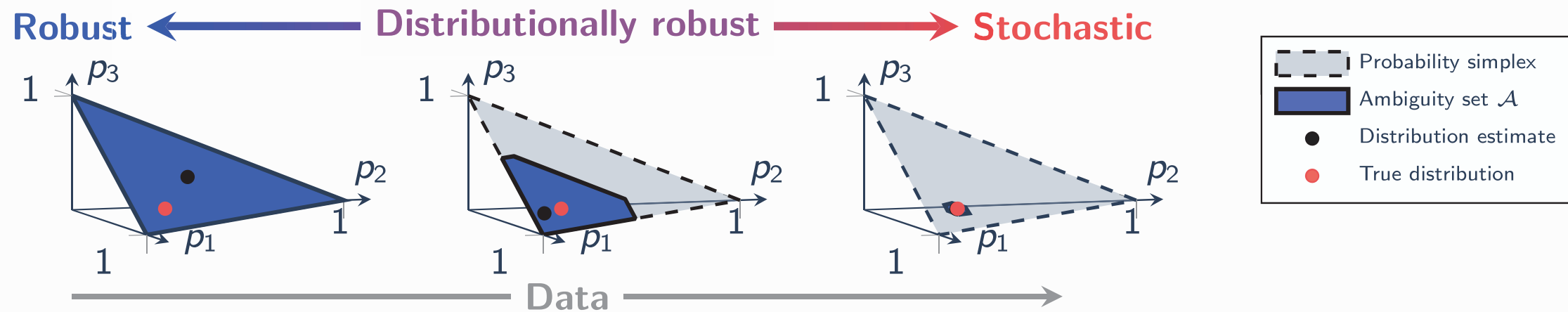
assumes **no** statistical information
often overly **conservative**

Stochastic (sample-average approx.)

$$\mathcal{A} = \{\hat{p}\} \text{ (}\hat{p} \text{ the empirical distribution)}$$

assumes **perfect** statistical information
optimizer's curse (**unsafe**)

Distributionally Robust Optimization – Between Stochastic and Robust



Robust

$$\mathcal{A} = \Delta$$

assumes **no** statistical information
often overly **conservative**

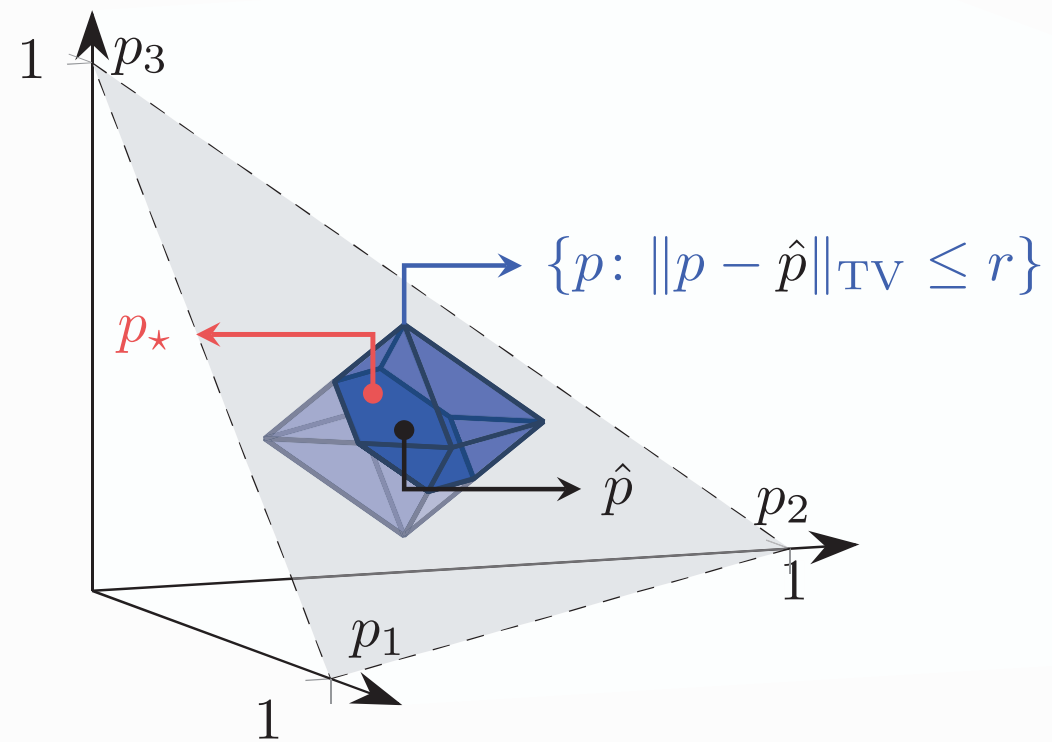
Stochastic (sample-average approx.)

$$\mathcal{A} = \{\hat{p}\} \text{ } (\hat{p} \text{ the empirical distribution})$$

assumes **perfect** statistical information
optimizer's curse (**unsafe**)

⇒ **DR** approach: safely interpolate between both based on amount of available information

Distributionally Robust Optimization – Between Stochastic and Robust



Distributionally Robust

- carefully calibrate ambiguity set
 $\mathcal{A} = \{p \in \Delta : \|p - \hat{p}\|_{\text{TV}} \leq r\};$
- safety certificate: $\mathbb{P}[p_* \in \mathcal{A}] \geq 1 - \epsilon$

$$r = \sqrt{\frac{2(d \log(2) - \log(\epsilon))}{\# \text{ samples}}}$$

Multi-modal MPC – Illustrative results

Deterministic MPC

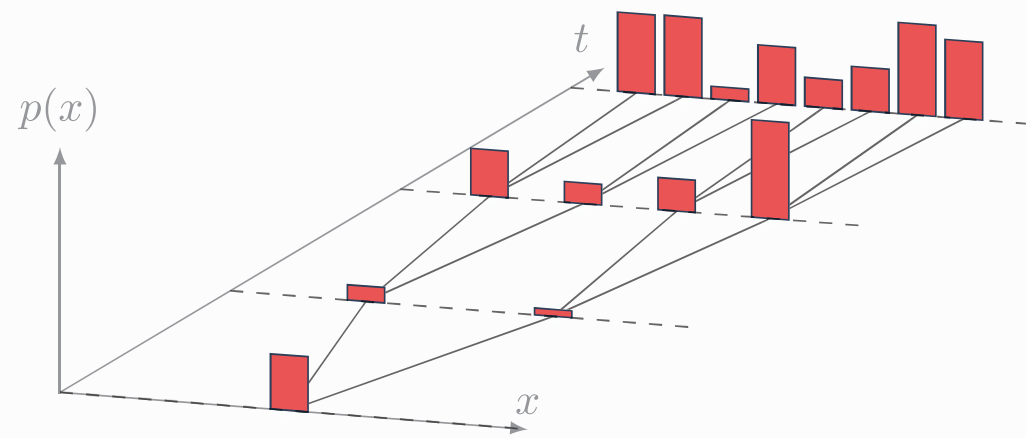


DR MPC



M. Schuurmans, A. Katriniok, et al. (2022). “Safe, Learning-Based MPC for Highway Driving under Lane-Change Uncertainty: A Distributionally Robust Approach”. In: **JAI**. Submitted. arXiv: 2206.13319.

Learning Distribution – Moments



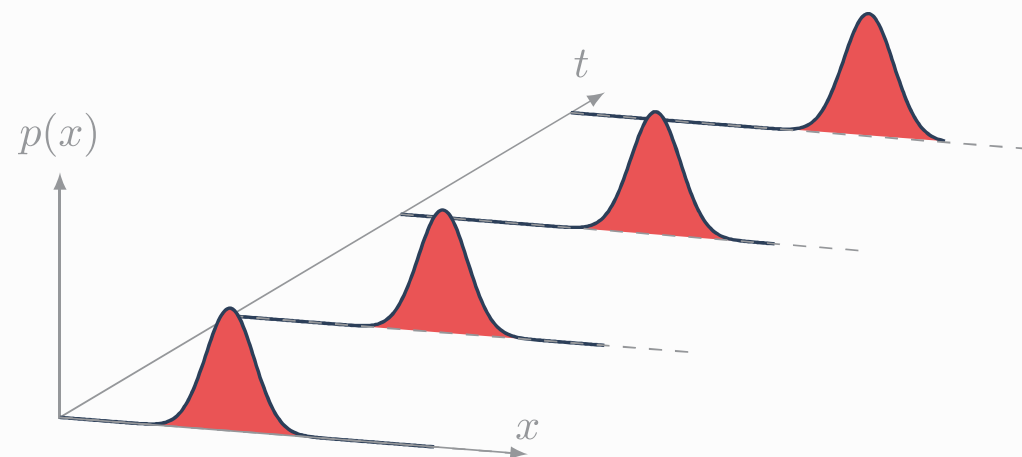
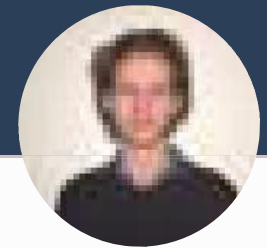
discrete disturbance

- can handle general nonlinear dynamics

$$x_{t+1} = f(x_t, \pi(x_t), w_t)$$

$$w_t \in W = \{1, \dots, d\}$$

- continuous disturbance
affects: uncertainty propagation, tractability, confidence intervals, ...



continuous distributions

- propagate uncertainty by restricting model class

$$x_+ = \left(\sum_{i=1}^r A_i w^{(i)} \right) x$$

- propagate moment dynamics

$$\mathbb{E}[x_+ x_+^\top] = \mathcal{E}(\mathbb{E}[x x^\top]) .$$

↑
bilinear in $\mathbb{E}[x x^\top]$
 $\mathbb{E}[w w^\top]$

- learn second moment $\mathbb{E}[w w^\top]$
sufficient when $\ell(x, u) = x^\top Q x + u^\top R u$.

Multiplicative – DR Synthesis

distributionally robust

$$\underset{u_0, u_1, \dots}{\text{minimize}} \quad \max_{\mu \in \mathcal{A}} \quad \mathbb{E}_{\mu} \left[\sum_{t=0}^{\infty} x_t^{\top} Q x_t + u_t^{\top} R u_t \right]$$

↑ defined by $\|\mathbb{E}[ww^{\top}] - \hat{W}\|_2 \leq r$

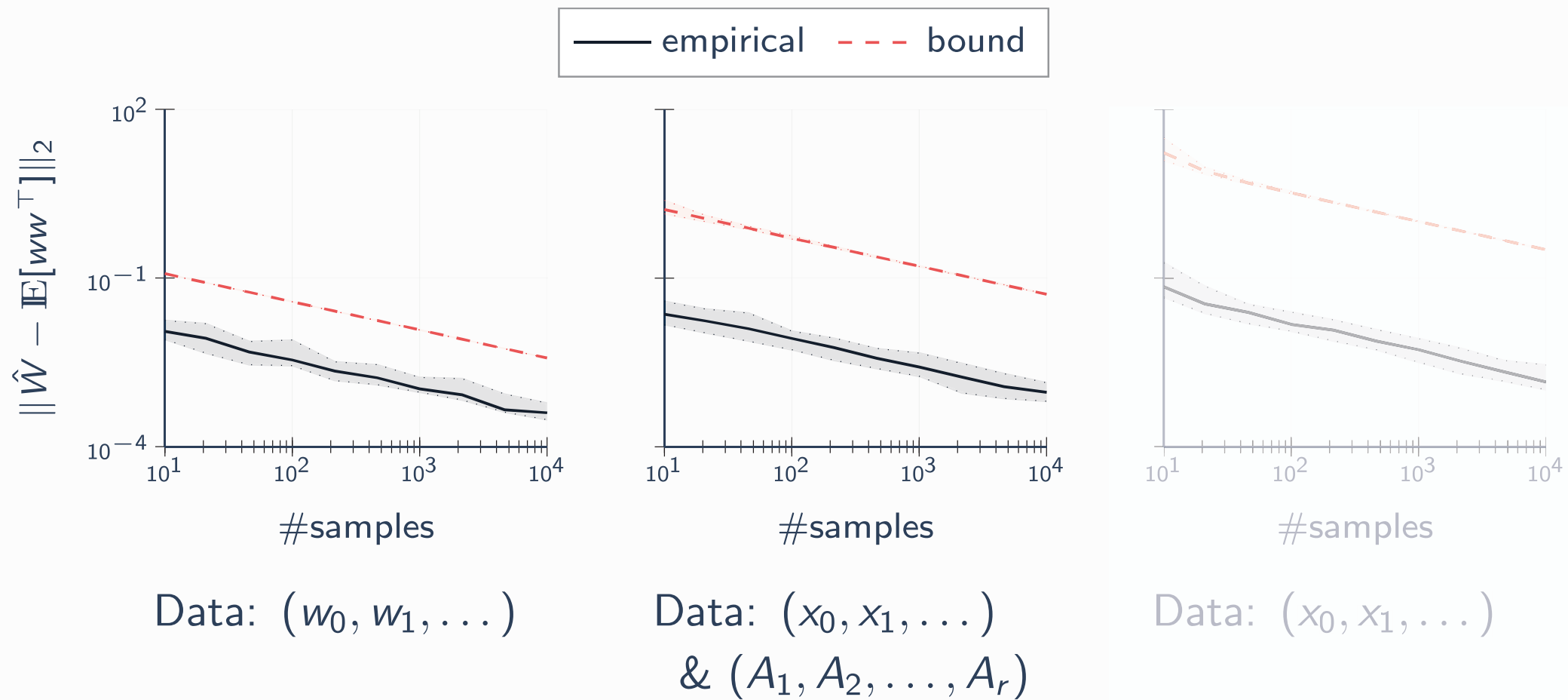
$$\text{subject to} \quad x_{t+1} = \left(\sum_{i=1}^r A_i w^{(i)} \right) x + \left(\sum_{i=1}^r B_i w^{(i)} \right) u, \quad t = 0, 1, \dots$$

- convex–concave saddle point problem
- can be solved using convex optimization tools (SDP)

P. Coppens and P. Patrinos (2022). “Safe learning LQR of linear dynamics with multiplicative noise”. In: **TAC** (submitted). arXiv: 2207.06062.

Learning Distribution – Moments

$$x_+ = \left(\sum_{i=1}^r A_i w^{(i)} \right) x$$

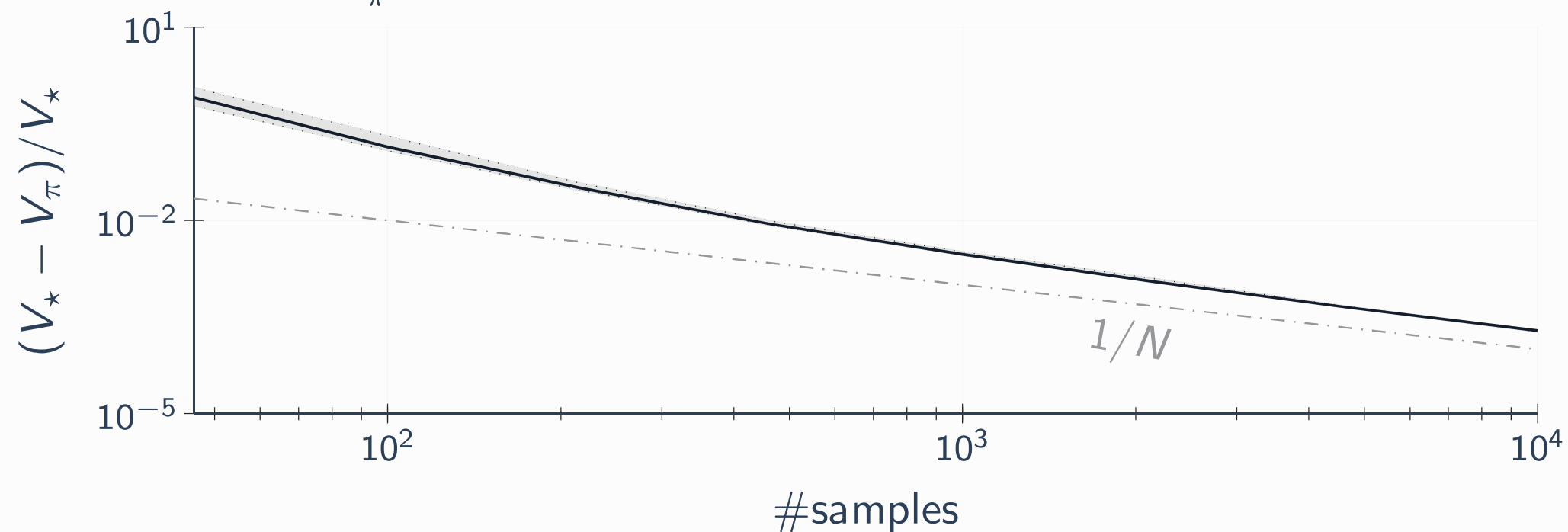


Multiplicative – Performance

Performance Evaluation

$$V_{\pi}(x_0) = \left\{ \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right] : x_{t+1} = f(x_t, \pi(x_t), w_t), t \in \mathbb{N} \right\} \quad (\text{actual performance})$$

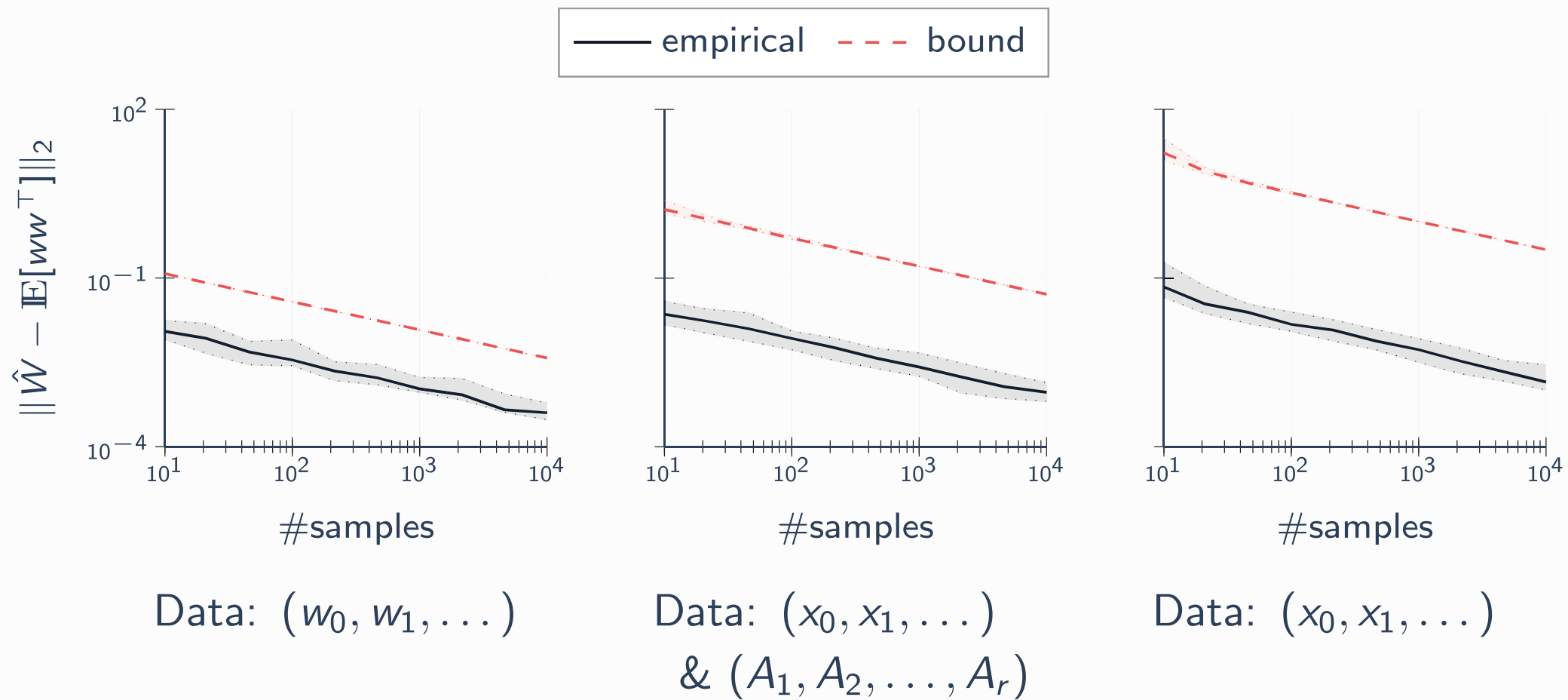
$$V_{\star}(x_0) = \min_{\pi} V_{\pi}(x_0) \quad (\text{best performance})$$



P. Coppens and P. Patrinos (2020). "Sample Complexity of Data-Driven Stochastic LQR with Multiplicative Uncertainty". In: CDC.

Learning Distribution – Moments

$$x_+ = \left(\sum_{i=1}^r A_i w^{(i)} \right) x$$



Learning Dynamics – Extending Moment Learning

$$x_+ = \left(\sum_{i=1}^r A_i w^{(i)} \right) x$$

- Learning the dynamics as a special case of learning the disturbance;

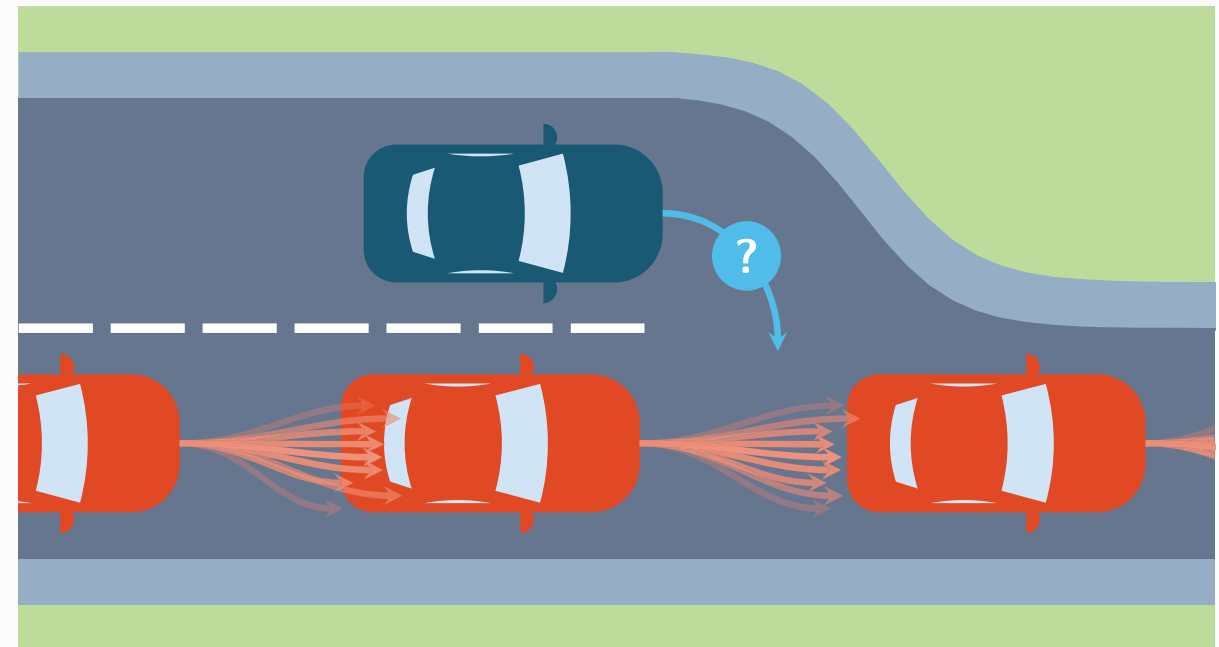
$$A_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}, \quad \dots$$

- More general learning = more conservativeness;
- How to include structure in other learning architecture?

Learning Dynamics – Interaction

Learning exogenous distributions (Until now)

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(w)}_{\text{learned}}$$



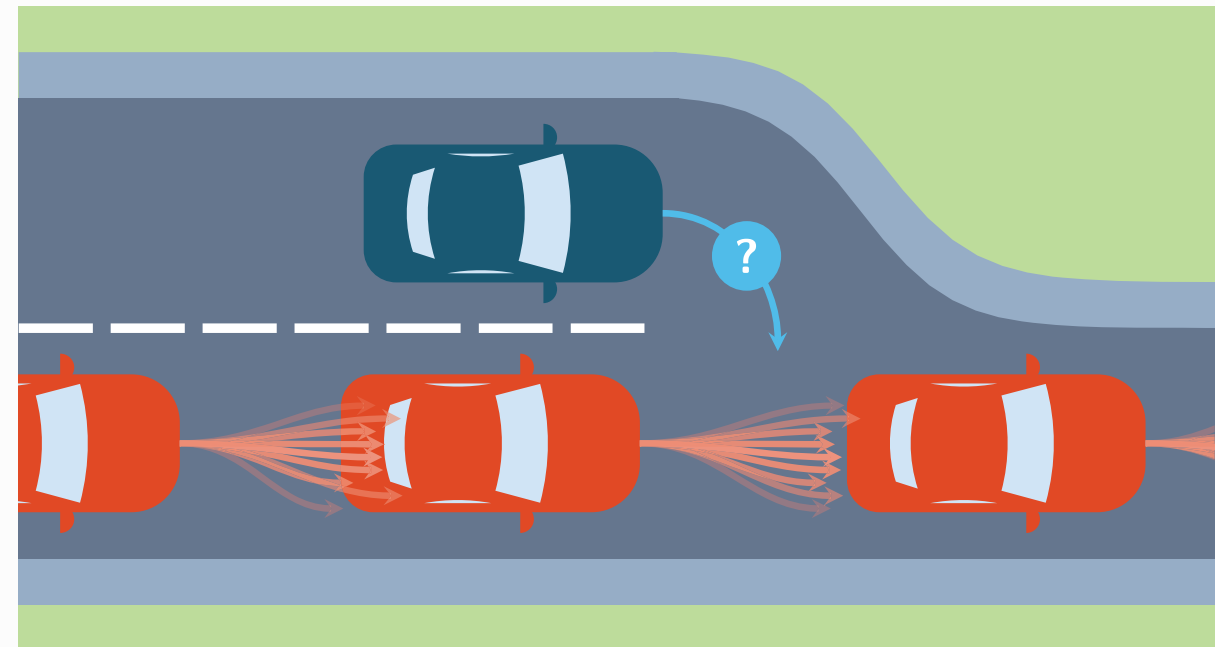
Example: unknown behavior of external actors is assumed independent of the controlled vehicle's state

Learning Dynamics – Interaction

Learning exogenous distributions (Until now)

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(w)}_{\text{learned}}$$

+ *learning* and *control* problem **decoupled**



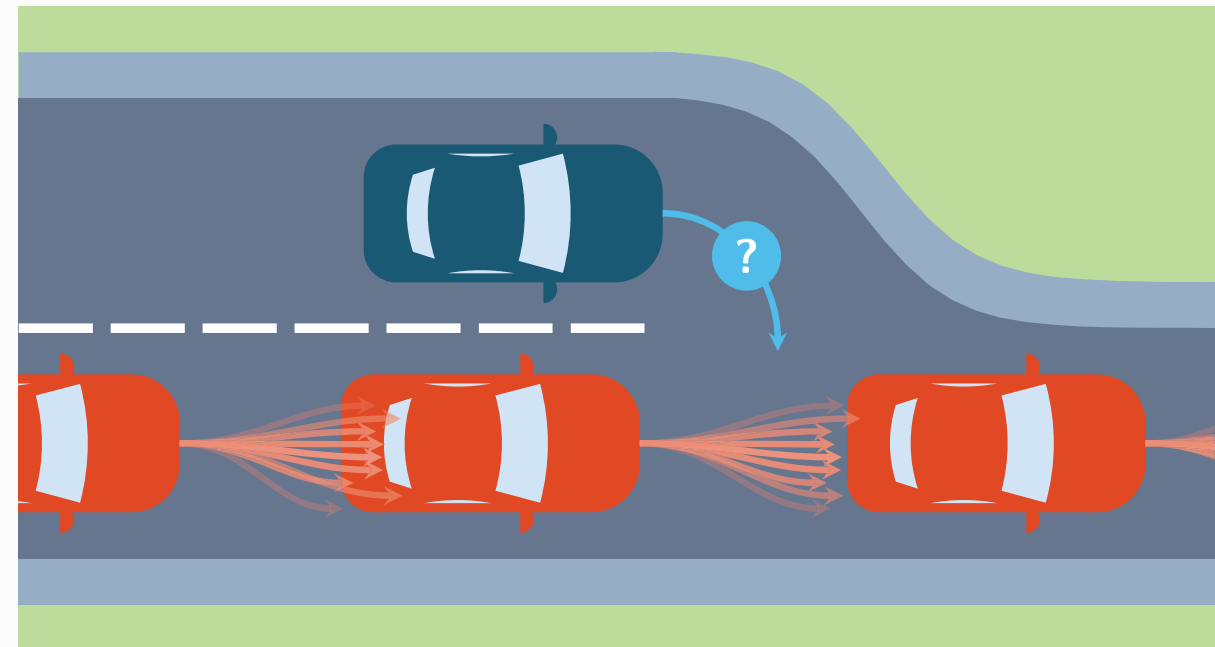
Example: unknown behavior of external actors is assumed independent of the controlled vehicle's state

Learning Dynamics – Interaction

Learning exogenous distributions (Until now)

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(w)}_{\text{learned}}$$

- + *learning* and *control* problem **decoupled**
- + sound **theory**, strong guarantees



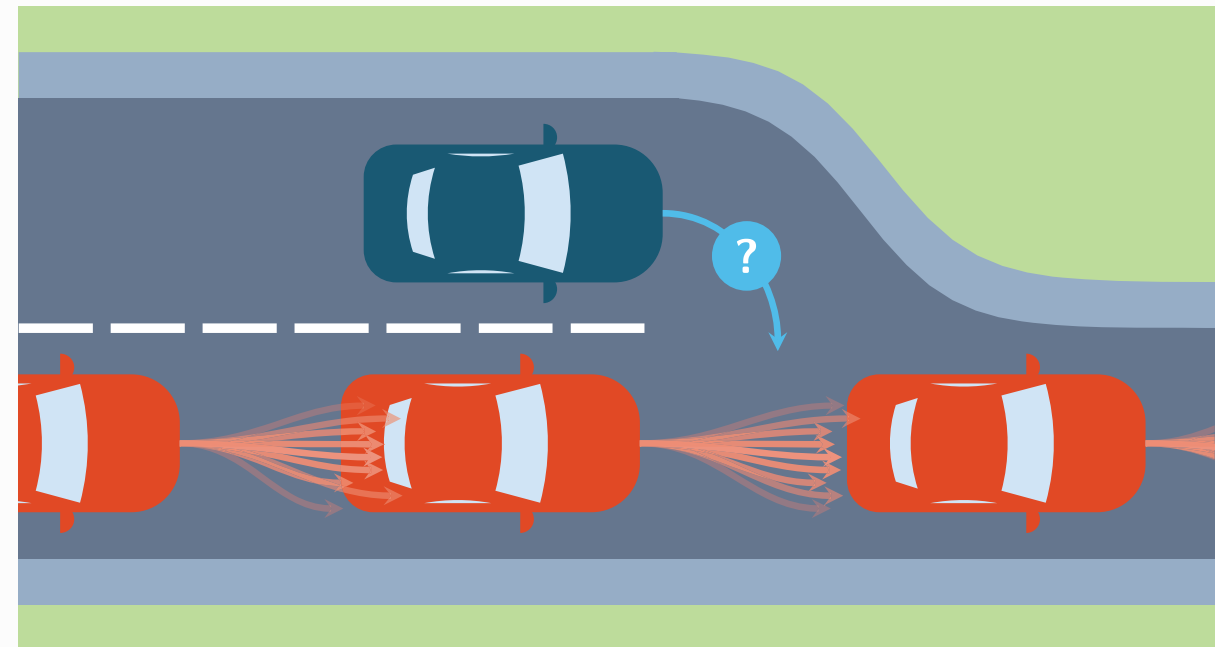
Example: unknown behavior of external actors is assumed independent of the controlled vehicle's state

Learning Dynamics – Interaction

Learning exogenous distributions (Until now)

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(w)}_{\text{learned}}$$

- + *learning* and *control* problem **decoupled**
- + sound **theory**, strong guarantees
- + efficient learning + uncertainty propagation



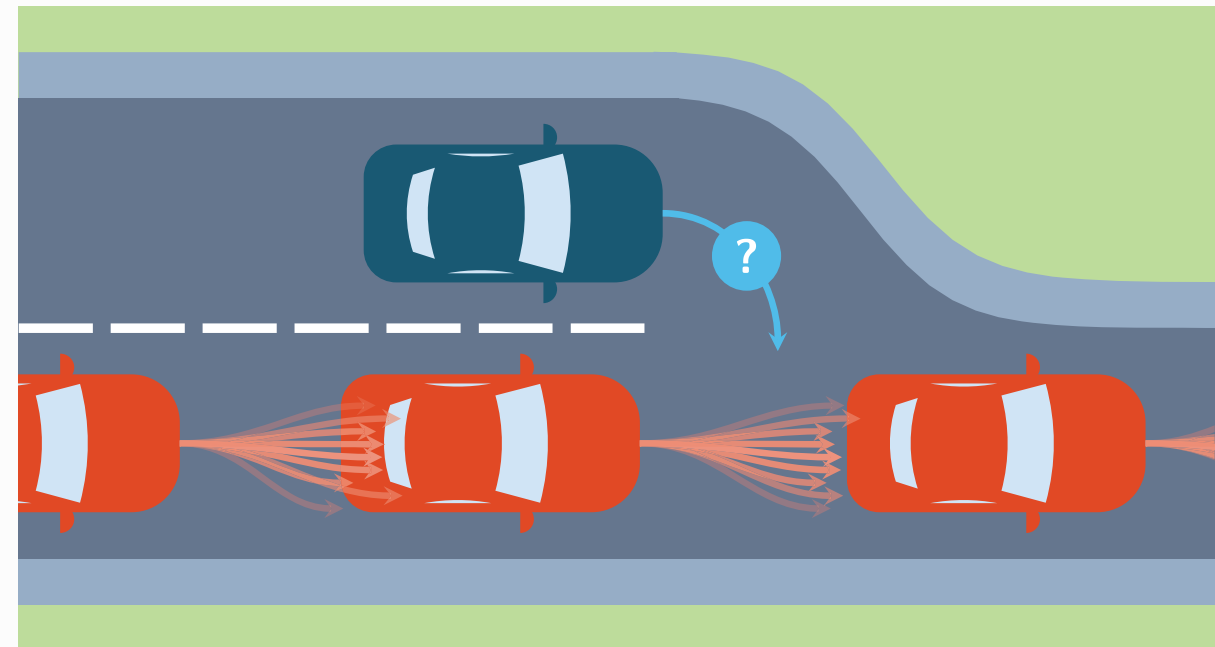
Example: unknown behavior of external actors is assumed independent of the controlled vehicle's state

Learning Dynamics – Interaction

Learning exogenous distributions (Until now)

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(w)}_{\text{learned}}$$

- + *learning* and *control* problem **decoupled**
- + sound **theory**, strong guarantees
- + efficient learning + uncertainty propagation
- limited modeling capacity



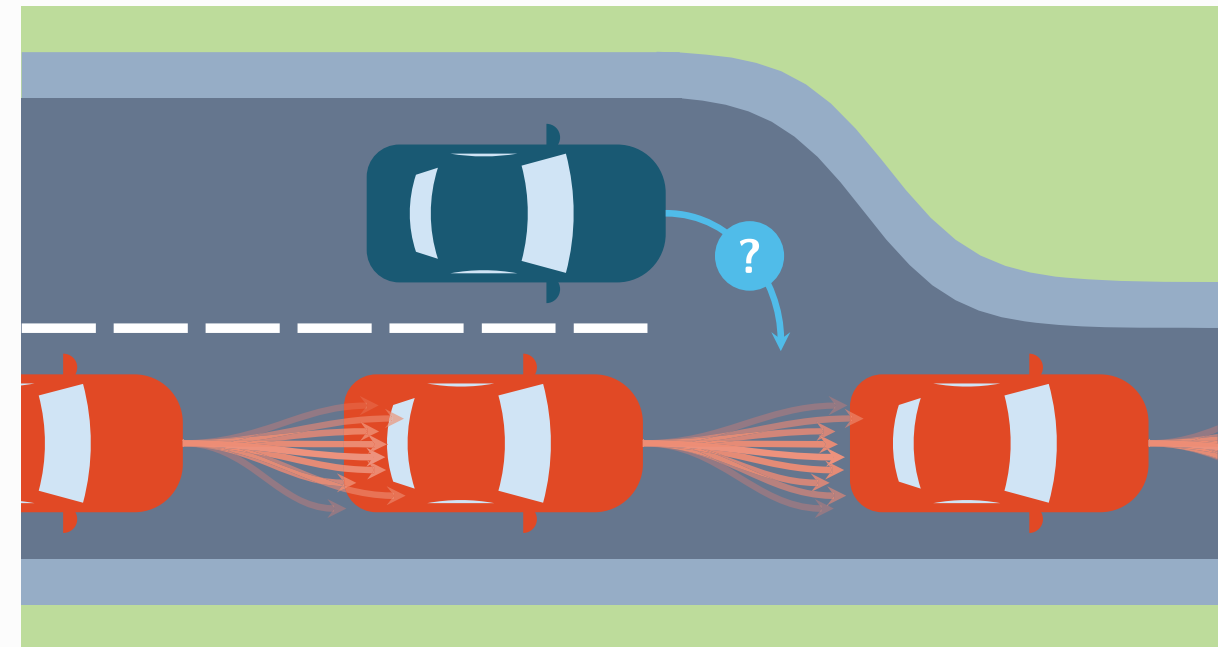
Example: unknown behavior of external actors is assumed independent of the controlled vehicle's state

Learning Dynamics – Interaction

Learning exogenous distributions (Until now)

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(w)}_{\text{learned}}$$

- + *learning* and *control* problem **decoupled**
- + sound **theory**, strong guarantees
- + efficient learning + uncertainty propagation
- limited modeling capacity
- combined with safety → *frozen robot problem*

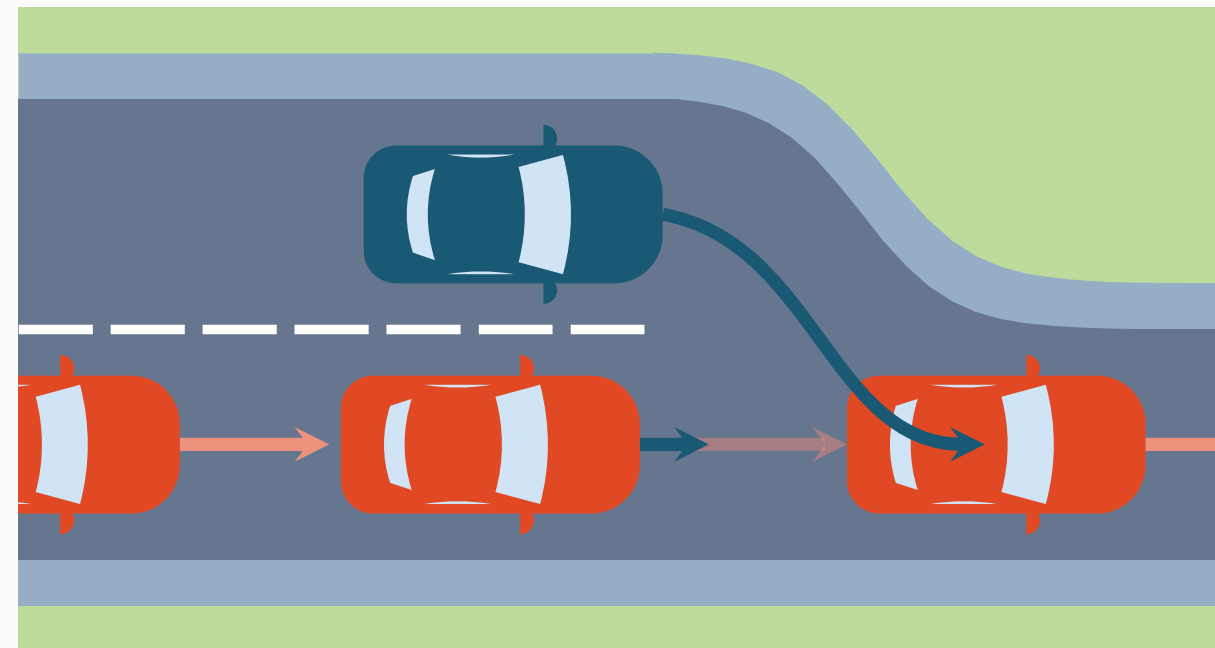


Example: unknown behavior of external actors is assumed independent of the controlled vehicle's state

Learning Dynamics – Interaction

Learning decision-dependent distributions

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(x, u)}_{\text{learned}}$$



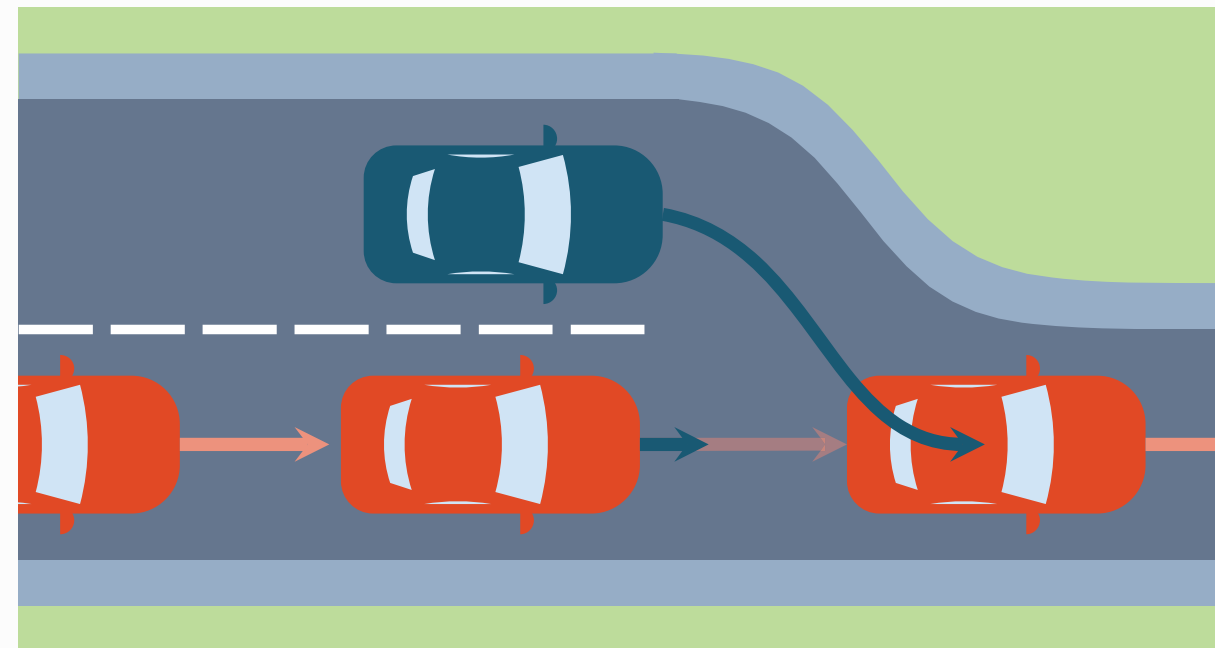
Example: effect of controlled vehicle's behavior on surrounding vehicle's behavior is included in the learned model

Learning Dynamics – Interaction

Learning decision-dependent distributions

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(x, u)}_{\text{learned}}$$

- Data depends on states and actions: *learning* and control are **coupled**



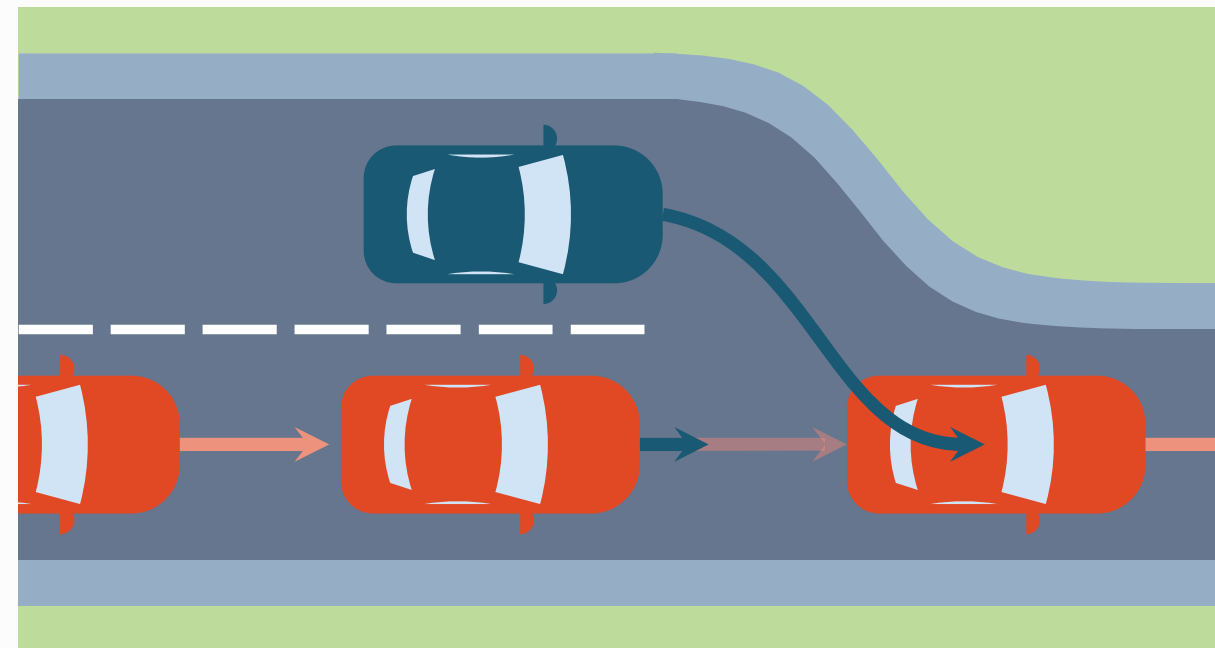
Example: effect of controlled vehicle's behavior on surrounding vehicle's behavior is included in the learned model

Learning Dynamics – Interaction

Learning decision-dependent distributions

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(x, u)}_{\text{learned}}$$

- Data depends on states and actions: *learning* and control are **coupled**
- Theory more challenging



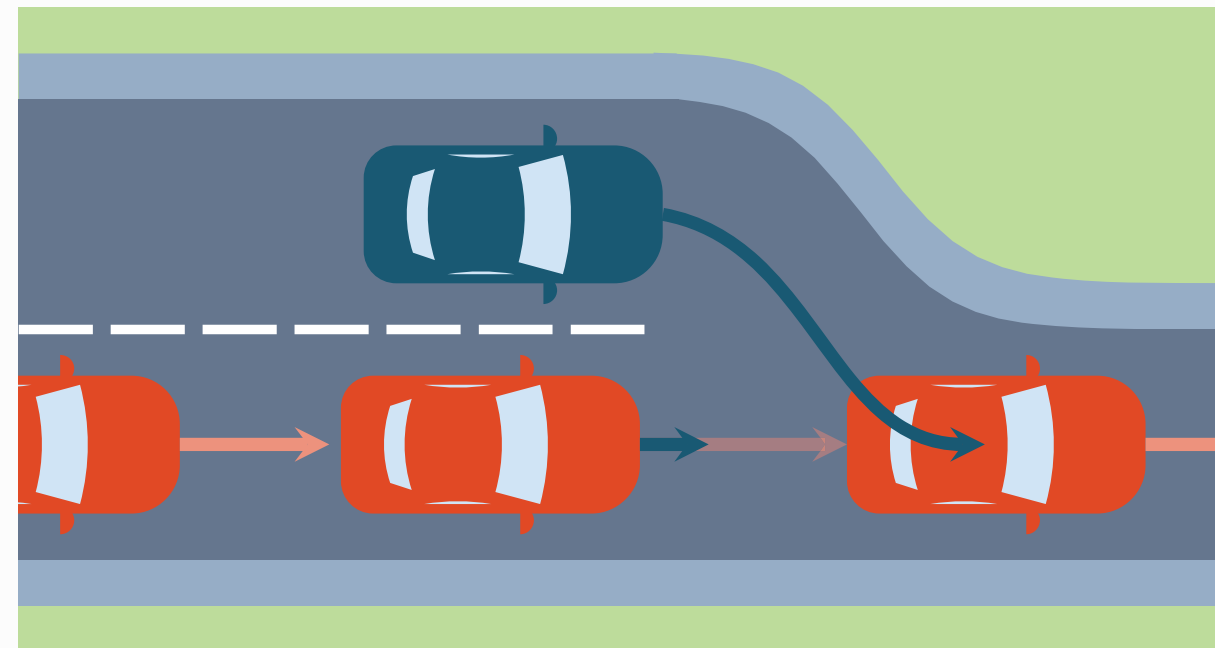
Example: effect of controlled vehicle's behavior on surrounding vehicle's behavior is included in the learned model

Learning Dynamics – Interaction

Learning decision-dependent distributions

$$x_+ = \underbrace{f(x, u, w_+)}_{\text{modeled}}, \quad w_+ \sim \underbrace{p(x, u)}_{\text{learned}}$$

- Data depends on states and actions: *learning* and control are **coupled**
- Theory more challenging
- + Can handle more complex, realistic scenarios



Example: effect of controlled vehicle's behavior on surrounding vehicle's behavior is included in the learned model

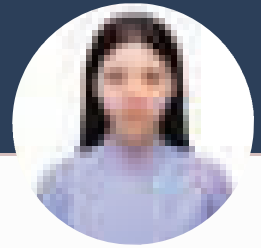
Dual and adaptive control – History

Update dynamical model while controller is deployed

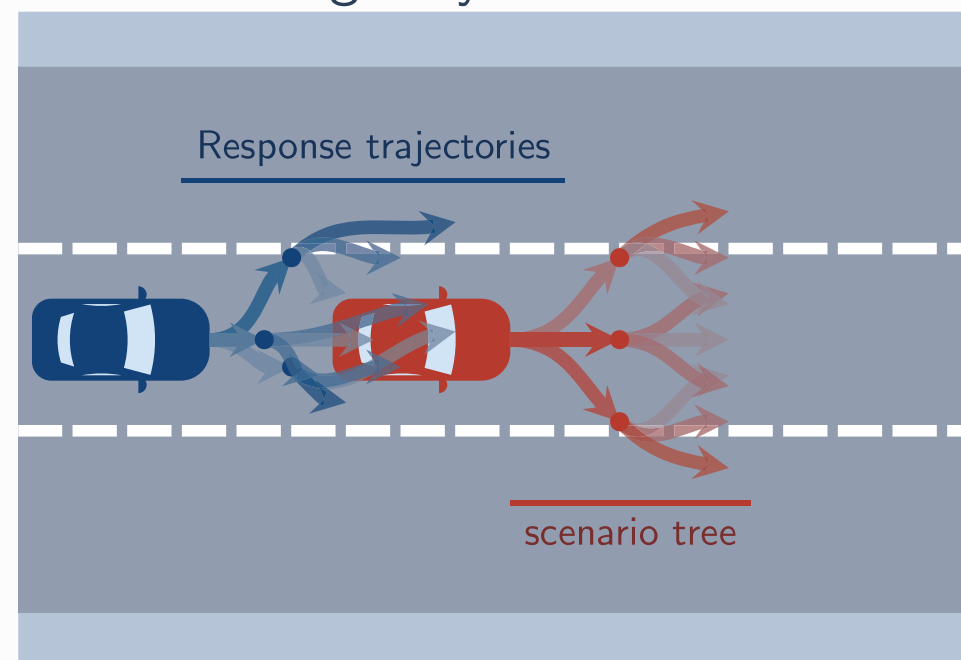
Central challenge. Future estimates depend on control actions/states

→ Minimize the **control cost** (exploitation), while obtaining **informative** data (exploration)

- *Adaptive/dual control* [Feldbaum 1961]
- Exact solution tractable in few cases [Åström and Helmersson 1986; Sternby 1976]
- More recently, approximate solutions: **explicit** vs. **implicit** [Mesbah 2018]
 - **explicit.** add exploration term to controller [Brüggemann and Bitmead 2022; Houska et al. 2016]
 - **implicit.** approximately solve dual control problem [Arcari et al. 2019; Hu and Fisac 2023]



Ex.: highway lane selection



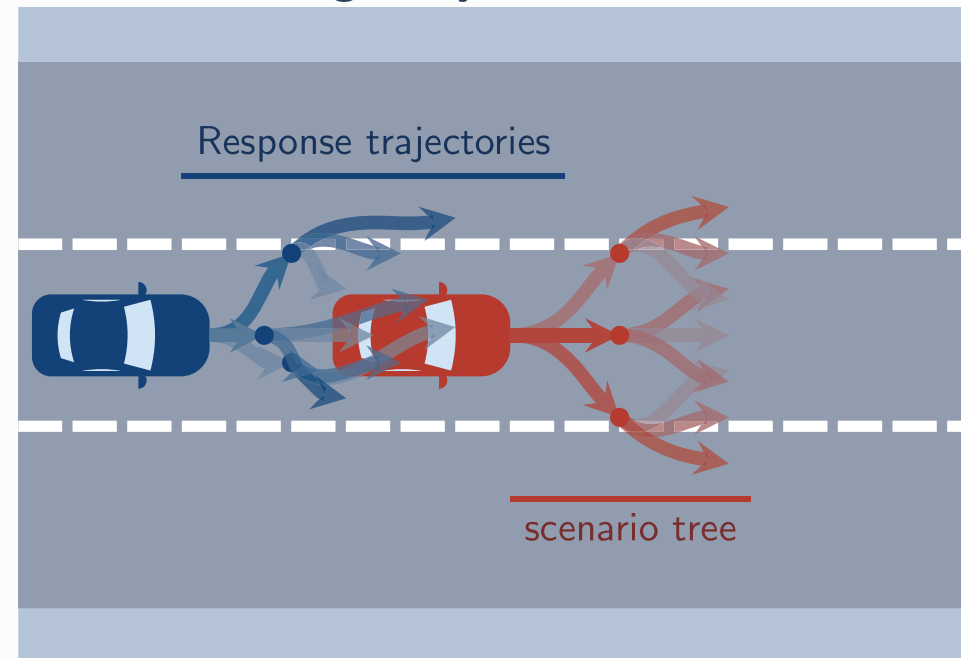
Discrete-valued uncertainty, now with **state-dependent distribution**

$$x_{t+1} = f(x_t, \pi(x_t), w_t), \quad w_t \sim p(x_t)$$

$w_t \in W = \{1, \dots, d\}$ learned

Learning Dynamics – Multi-Modal MPC Revisited

Ex.: highway lane selection



Discrete-valued uncertainty, now with **state-dependent distribution**

$$x_{t+1} = f(x_t, \pi(x_t), w_t), \quad w_t \sim p(x_t)$$

$$w_t \in W = \{1, \dots, d\}$$

learned

Multinomial logistic regression model

$$\hat{p}(x; \Theta) = \left(\frac{\exp(\theta_i^\top \varphi(x))}{\sum_{j=1}^d \exp(\theta_j^\top \varphi(x))} \right)_{i=1}^d$$

Parameters Features

R. Wang, M. Schuurmans, and P. Patrinos (2022). **Interaction-Aware Model Predictive Control for Autonomous Driving.** arXiv: 2211.17053

Multi-Modal MPC – passive learning

Multinomial logistic regression model

$$\hat{p}(x; \Theta) = \left(\frac{\exp(\theta_i^\top \varphi(x))}{\sum_{j=1}^d \exp(\theta_j^\top \varphi(x))} \right)_{i=1}^d$$

Estimate Θ using maximum likelihood

$$\hat{\Theta}_t \in \underset{\Theta}{\operatorname{argmin}} \mathcal{L}(\Theta; X_t, W_t)$$

negative log-likelihood

data $(x_k, w_k)_{k=0}^{t-1}$

Passive learning

at every time t ,

- Solve multi-modal MPC for **fixed** $\hat{\Theta}_t$
- Obtain measurement $\rightarrow$ update $\hat{\Theta}_{t+1}$

Results

method	control cost
Logistic model	139.50
Markov chain	154.85

Multi-Modal MPC – passive learning

Multinomial logistic regression model

$$\hat{p}(x; \Theta) = \left(\frac{\exp(\theta_i^\top \varphi(x))}{\sum_{j=1}^d \exp(\theta_j^\top \varphi(x))} \right)_{i=1}^d$$

Estimate Θ using maximum likelihood

$$\hat{\Theta}_t \in \underset{\Theta}{\operatorname{argmin}} \mathcal{L}(\Theta; X_t, W_t)$$

negative log-likelihood

data $(x_k, w_k)_{k=0}^{t-1}$

Passive learning

at every time t ,

- Solve multi-modal MPC for **fixed** $\hat{\Theta}_t$
- Obtain measurement $\rightarrow$ update $\hat{\Theta}_{t+1}$

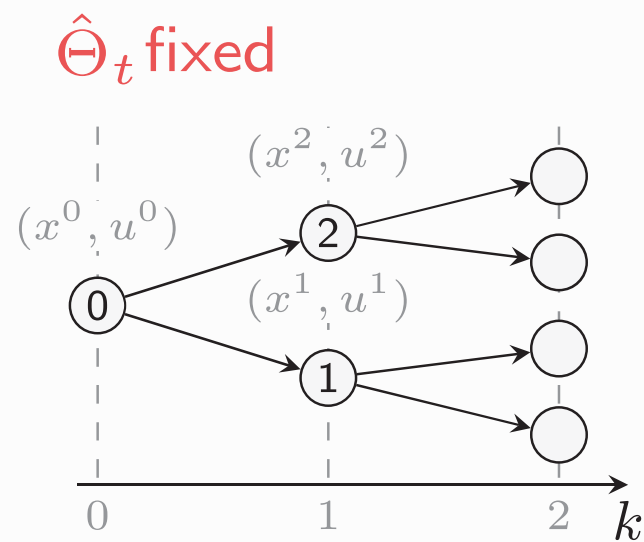
Results

method	control cost
Logistic model	139.50
Markov chain	154.85

even with passive learning: **improved performance due to interaction-aware model**

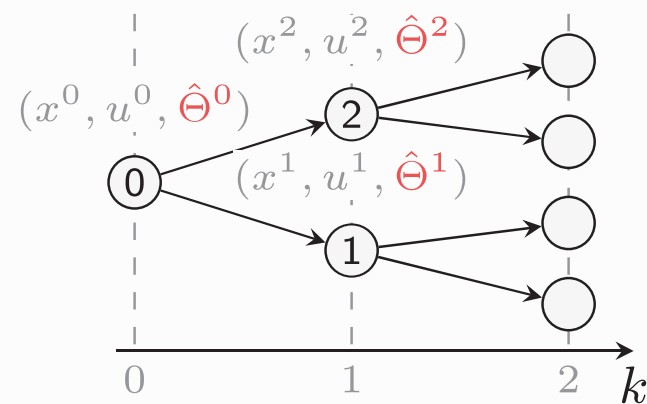
Multi-Modal MPC – passive vs. active learning

Passive learning



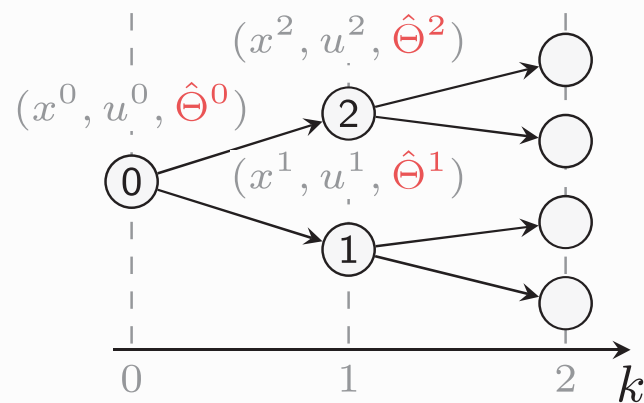
- $\hat{\Theta}_t$ **fixed** over pred. horizon
- No *dual control*

Active learning



- future estimates $\hat{\Theta}^i$ predicted for each node i
- controller is aware of online learning
- **bilevel optimization**

Multi-Modal MPC – Bilevel formulation



$$\underset{\mathbf{u} \in \mathbb{R}^{n_u N}}{\text{minimize}} \quad J_N(\mathbf{x}, \mathbf{u}, \hat{\Theta})$$

$$\text{subj. to} \quad \mathbf{x}_0 = \bar{\mathbf{x}},$$

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k), \quad \forall k \in \mathbb{N}_{[0, N-1]},$$

$$\hat{\Theta}_k \in \underset{\Theta}{\text{argmin}} \mathcal{L}(\Theta; \underbrace{\mathbf{x}_k, \mathbf{w}_k}_{\substack{\mathbf{x}_0, \dots, \mathbf{x}_k \\ \mathbf{w}_0, \dots, \mathbf{w}_k}}, X, W),$$

$$\forall k \in \mathbb{N}_{[1, N]}$$

where

$$J_N(\mathbf{x}, \mathbf{u}, \hat{\Theta}) = \sum_{\substack{\mathbf{w} \\ p_{\mathbf{w}_0}(\mathbf{x}_0, \hat{\Theta}_0) \times \dots \times p_{\mathbf{w}_N}(\mathbf{x}_N, \hat{\Theta}_N)}} \underbrace{P_{\mathbf{w}}(\mathbf{x}, \hat{\Theta})}_{\substack{\mathbf{w} \\ p_{\mathbf{w}_0}(\mathbf{x}_0, \hat{\Theta}_0) \times \dots \times p_{\mathbf{w}_N}(\mathbf{x}_N, \hat{\Theta}_N)}} \left(\ell_N(\mathbf{x}_N) + \sum_{k=0}^{N-1} \ell(\mathbf{x}_k, \mathbf{u}_k) \right)$$

Learning Dynamics – Multi-modal

Preliminary results

<i>Learning method</i>	<i>Closed-loop cost</i>
implicit active (bilevel)	13.08
explicit active	13.64
passive	14.54
omniscient	7.08

minimize $J_N(\mathbf{x}, \mathbf{u}, \hat{\Theta})$
 $\mathbf{u} \in \mathbb{R}^{n_u N},$

$\hat{\Theta} \in \mathbb{R}^{n_\theta \times d \times N}$

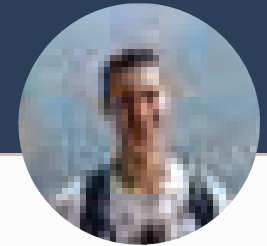
subj. to

$\mathbf{x}_0 = \bar{\mathbf{x}},$

$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k), \forall k \in \mathbb{N}_{[0, N-1]},$

$\nabla_{\Theta} \mathcal{L}(\hat{\Theta}_k; \mathbf{x}_k, \mathbf{w}_k, X, W) = 0_{n_\theta \times d}$

$\forall k \in \mathbb{N}_{[1, N]}$

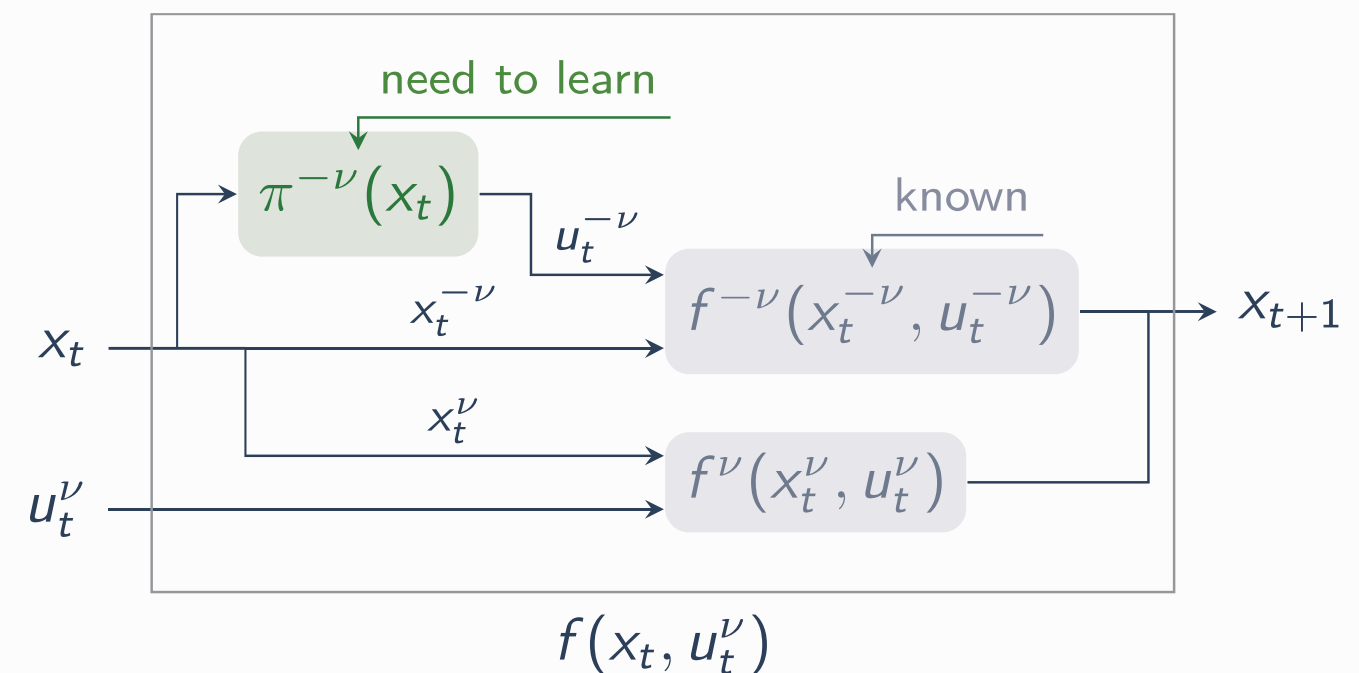


Learning intentions of other agents

Notation

- M players, labeled by $\mathcal{P} := \{1, \dots, M\}$
- u_t^ν – input (player ν)
- x_t^ν – state (player ν)
- $u_t^{-\nu} := (u_t^i)_{i \in \mathcal{P} \setminus \{\nu\}}$ (all except ν)
- $x_t^{-\nu} := (x_t^i)_{i \in \mathcal{P} \setminus \{\nu\}}$ (all except ν)
- $x_t := (x_t^i)_{i \in \mathcal{P}}$ (all players)
- ...

Unknown intentions = unknown **dynamics**



→ Need to learn **behavior** of other agents

Learning Dynamics – Game Theoretic Approach

Key assumption: other agents are **expected utility maximizing**

→ Each agent tries to solve a **coupled** optimization problem

$$\begin{aligned}
 & \overset{\text{first entry}}{\pi^\nu(x_0) \in} \underset{u_0^\nu}{\operatorname{argmin}} \min_{u_1^\nu, \dots, u_{N-1}^\nu} \sum_{t=0}^{N-1} \overset{\text{stage cost}}{\ell_t^\nu(x_t, u_t^\nu, u_t^{-\nu})} + \overset{\text{terminal cost}}{\ell_N^\nu(x_N)} \\
 & \text{subj. to } x_{t+1} = f(x_t, u_t^\nu, u_t^{-\nu}) \quad t \in \mathbb{N}_{[0, N-1]} \\
 & \quad u_t^\nu \in \overset{\text{input constraints}}{U^\nu} \quad t \in \mathbb{N}_{[0, N-1]} \\
 & \quad x_t^\nu \in \overset{\text{state constraints}}{X^\nu(x_t^{-\nu})} \quad t \in \mathbb{N}_{[0, N]}
 \end{aligned}$$

= **Generalized** Nash equilibrium problem (GNEP)

Learning Dynamics – Game Theoretic Approach

Key assumption: other agents are **expected utility maximizing**

→ Each agent tries to solve a **coupled** optimization problem

$$\begin{aligned}
 & \pi^\nu(x_0) \in \underset{u_0^\nu}{\operatorname{argmin}} \min_{u_1^\nu, \dots, u_{N-1}^\nu} \sum_{t=0}^{N-1} \ell_t^\nu(x_t, u_t^\nu, u_t^{-\nu}) + \ell_N^\nu(x_N) \\
 & \text{subj. to } x_{t+1} = f(x_t, u_t^\nu, u_t^{-\nu}) \quad t \in \mathbb{N}_{[0, N-1]} \\
 & \quad u_t^\nu \in U^\nu \quad t \in \mathbb{N}_{[0, N-1]} \\
 & \quad x_t^\nu \in X^\nu(x_t^{-\nu}) \quad t \in \mathbb{N}_{[0, N]}
 \end{aligned}$$

Diagram annotations:

- first entry:** points to $\pi^\nu(x_0)$
- stage cost:** points to $\ell_t^\nu(x_t, u_t^\nu, u_t^{-\nu})$
- terminal cost:** points to $\ell_N^\nu(x_N)$
- input constraints:** points to U^ν
- state constraints:** points to $X^\nu(x_t^{-\nu})$

= **Generalized** Nash equilibrium problem (GNEP)

Problem 1: Coupling of both **cost** and **constraints**

→ Need dedicated solvers (e.g. KKT conditions, splitting methods,...)

Problem 2: Need to learn **stage costs**, **terminal costs**, **inputs constraints** and **state constraints**

Learning Dynamics – Game Theoretic Learning

Problem 2: Need to learn stage costs, terminal costs, inputs constraints and state constraints

General learning methodology

1. Parametrize cost functions and constraints
 - **Ex.:** sum of relevant driving **features** (desired speed, target destination, input bounds, ...)
 - Leads to parametrized policy $\pi^\nu(x_t; \theta^\nu)$
2. Learn parameters $\theta^{-\nu}$ using data (**offline** and/or **online**)

Learning Dynamics – Game Theoretic Learning

Problem 2: Need to learn **stage costs**, **terminal costs**, **inputs constraints** and **state constraints**

General learning methodology

1. Parametrize cost functions and constraints
 - **Ex.:** sum of relevant driving **features** (desired speed, target destination, input bounds, ...)
 - Leads to parametrized policy $\pi^\nu(x_t; \theta^\nu)$
2. Learn parameters $\theta^{-\nu}$ using data (**offline** and/or **online**)

How to do this **in practice**?

1. Obtain measurements $(x_k, \hat{u}_k^{-\nu})_{k=0}^{t-1}$
2. Update $\theta^{-\nu}$ by solving bilevel optimization problem (can be performed **offline** and **online**)

$$\begin{aligned} \theta_\star^{-\nu} \in \underset{\theta^{-\nu}}{\operatorname{argmin}} \quad & \min_{u_0^{-\nu}, \dots, u_{t-1}^{-\nu}} \sum_{k=0}^{t-1} \|u_k^{-\nu} - \hat{u}_k^{-\nu}\|^2 \\ \text{subj. to} \quad & u_k^{-\nu} \in \pi^{-\nu}(x_k, \theta^{-\nu}) \quad t \in \mathbb{N}_{[0, t-1]} \end{aligned}$$

Learning Dynamics – Game Theoretic Learning

Problem 2: Need to learn **stage costs**, **terminal costs**, **inputs constraints** and **state constraints**

General learning methodology

1. Parametrize cost functions and constraints
 - **Ex.:** sum of relevant driving **features** (desired speed, target destination, input bounds, ...)
 - Leads to parametrized policy $\pi^\nu(x_t; \theta^\nu)$
2. Learn parameters $\theta^{-\nu}$ using data (**offline** and/or **online**)

How to do this **in practice**?

1. Obtain measurements $(x_k, \hat{u}_k^{-\nu})_{k=0}^{t-1}$
2. Update $\theta^{-\nu}$ by solving bilevel optimization problem (can be performed **offline** and **online**)

$$\begin{aligned} \theta_\star^{-\nu} \in \underset{\theta^{-\nu}}{\operatorname{argmin}} \quad & \min_{u_0^{-\nu}, \dots, u_{t-1}^{-\nu}} \sum_{k=0}^{t-1} \|u_k^{-\nu} - \hat{u}_k^{-\nu}\|^2 \\ \text{subj. to} \quad & u_k^{-\nu} \in \pi^{-\nu}(x_k, \theta^{-\nu}) \quad t \in \mathbb{N}_{[0, t-1]} \end{aligned}$$

No constraints $\approx$ **inverse reinforcement learning**



courteous human and stubborn belief

without learning



with learning



B. Evens, M. Schuurmans, and P. Patrinos (2022). “Learning MPC for Interaction-Aware Autonomous Driving: A Game-Theoretic Approach”. In: **2022 European Control Conference (ECC)**, pp. 34–39.



stubborn human and courteous belief

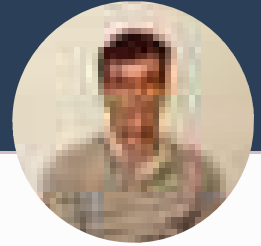
without learning



learning



B. Evens, M. Schuurmans, and P. Patrinos (2022). “Learning MPC for Interaction-Aware Autonomous Driving: A Game-Theoretic Approach”. In: **2022 European Control Conference (ECC)**, pp. 34–39.



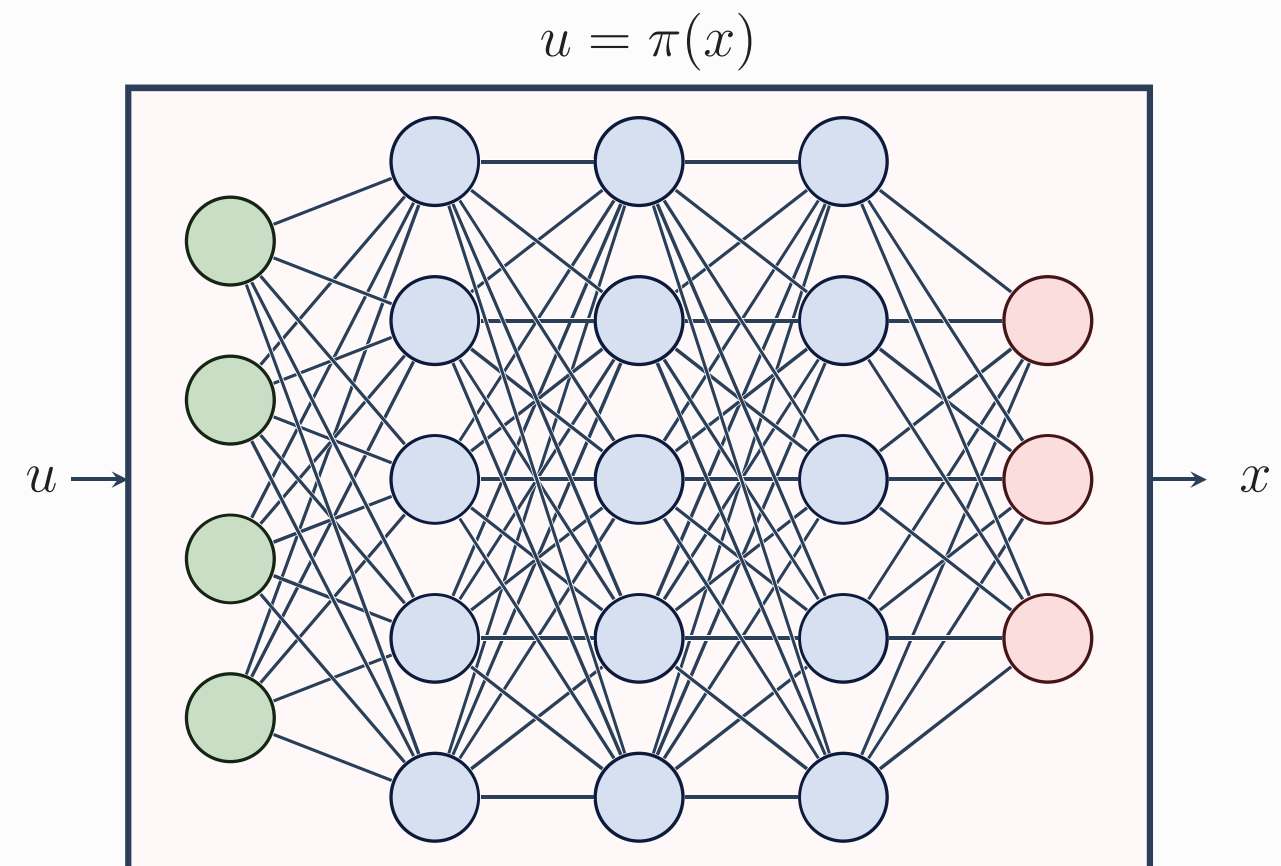
Learning Policies – Direct Search

$$\begin{aligned} \min_{\pi} \quad & \mathbb{E} \left[\sum_{t=0}^{\infty} \ell(x_t, \pi(x_t)) \right] \\ \text{s. t.} \quad & x_t \in X, \quad t \in \mathbb{N} \\ & x_{t+1} = f(x_t, \pi(x_t), w_t), \quad t \in \mathbb{N} \end{aligned}$$

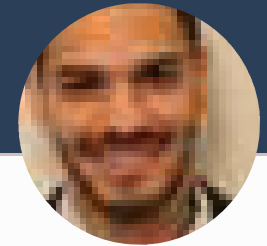
learn policy

Learn policy directly

- pick a parametrization (e.g. neural network);
- zero-order methods for optimization;
- can we include prior knowledge?



Learning Policies – Embedding Models



SIEMENS

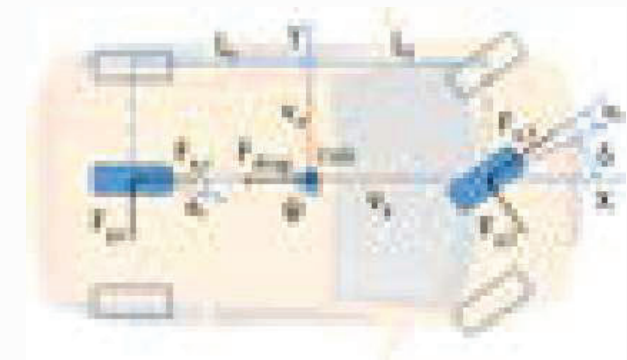
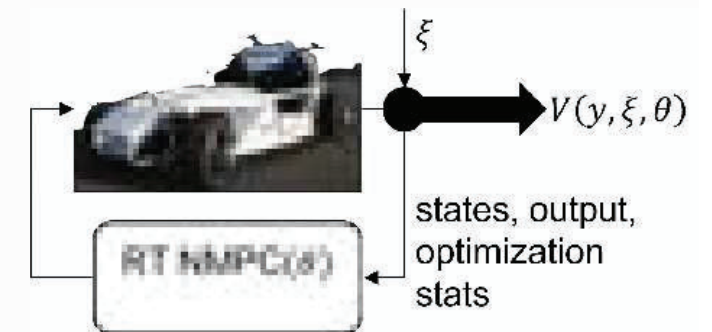
Goal: optimize the CL performance in the target domain while having limited access to it

Target
domain
autonomous
driving

$$\begin{aligned} & \underset{\theta}{\text{minimize}} \mathbb{E}_{\xi} \left[V(y, \xi, \theta) \right] \\ & \text{subj. to } x_{ego}(t+1) = f(x_{ego}(t), \pi_{\theta}(x_{ego}(t)), \xi) \quad t = t_0, \dots, T \\ & \theta \in \mathbf{C}_{\theta} \end{aligned}$$

NMPC
policy

$$\begin{aligned} \pi_{\theta}(x) = \underset{x(\cdot), u(\cdot)}{\operatorname{argmin}} & \left[J(x, u, \theta) = \int_t^{t+T_H(\theta)} (x^{\top} Q(\theta)x + u^{\top} R(\theta)u) dt \right] \\ & x(0) = x_{ego}(t) \\ & \dot{x} = \hat{f}(x, u, \theta) \\ & x \in \mathbf{X}(\theta) \end{aligned}$$



Learning Policies – Embedding Models

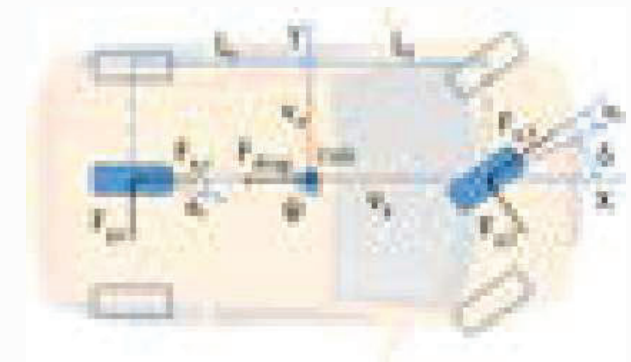
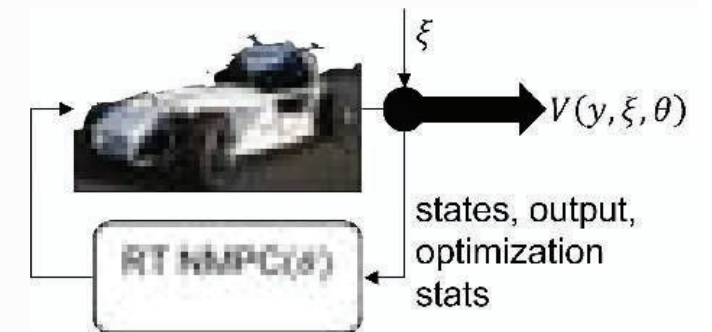
Goal: optimize the CL performance in the target domain while having limited access to it

**Target
domain
autonomous
driving**

$$\begin{aligned} & \underset{\theta}{\text{minimize}} \mathbb{E}_{\xi} \left[\overset{\text{performance measure}}{V(y, \xi, \theta)} \right] \\ & \text{subj. to } x_{ego}(t+1) = f(x_{ego}(t), \pi_{\theta}(x_{ego}(t)), \xi) \quad t = t_0, \dots, T \\ & \theta \in \mathbf{C}_{\theta} \end{aligned}$$

**NMPC
policy**

$$\begin{aligned} \pi_{\theta}(x) = \underset{x(\cdot), u(\cdot)}{\text{argmin}} & \left[J(x, u, \theta) = \int_t^{t+T_H(\theta)} (x^{\top} Q(\theta)x + u^{\top} R(\theta)u) dt \right] \\ & x(0) = x_{ego}(t) \\ & \dot{x} = \hat{f}(x, u, \theta) \\ & x \in \mathbf{X}(\theta) \end{aligned}$$



Learning Policies – Embedding Models

Goal: optimize the CL performance in the target domain while having limited access to it
 perturbations to the simulator's parameters –domain randomization

Target
domain
autonomous
driving

$$\underset{\theta}{\text{minimize}} \mathbb{E}_{\xi} \left[V(y, \xi, \theta) \right]$$

$$\text{subj. to } x_{ego}(t+1) = f(x_{ego}(t), \pi_{\theta}(x_{ego}(t)), \xi) \quad t = t_0, \dots, T$$

$$\theta \in \mathbf{C}_{\theta}$$

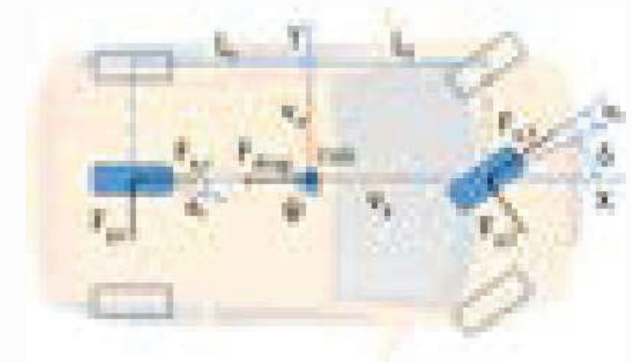
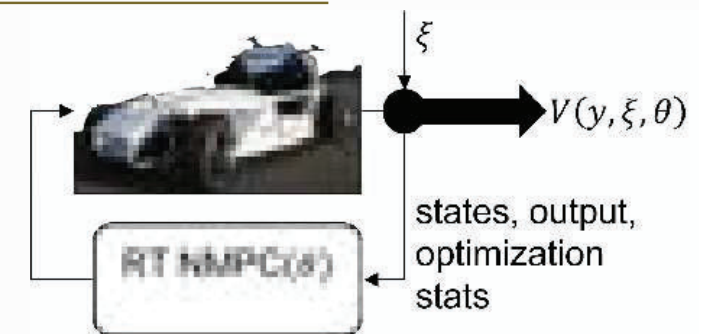
NMPC
policy

$$\pi_{\theta}(x) = \underset{x(\cdot), u(\cdot)}{\text{argmin}} \left[J(x, u, \theta) = \int_t^{t+T_H(\theta)} (x^{\top} Q(\theta)x + u^{\top} R(\theta)u) dt \right]$$

$$x(0) = x_{ego}(t)$$

$$\dot{x} = \hat{f}(x, u, \theta)$$

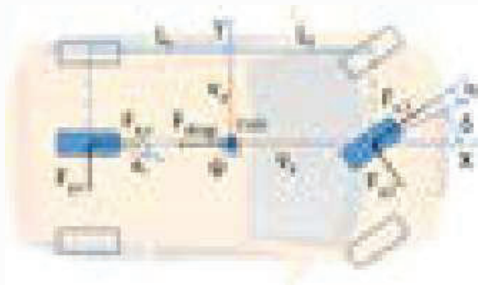
$$x \in \mathbf{X}(\theta)$$



Learning Policies – Embedding Models

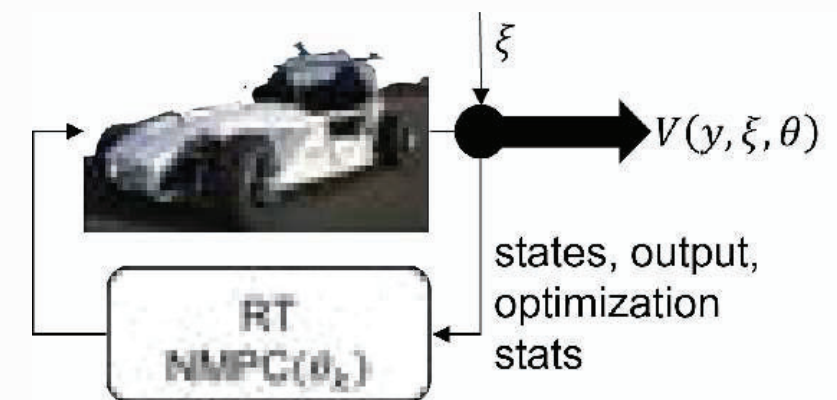
NMPC policy

$$\pi(x, \theta) = \underset{x(\cdot), u(\cdot)}{\operatorname{argmin}} [J(x, u, \theta)]$$



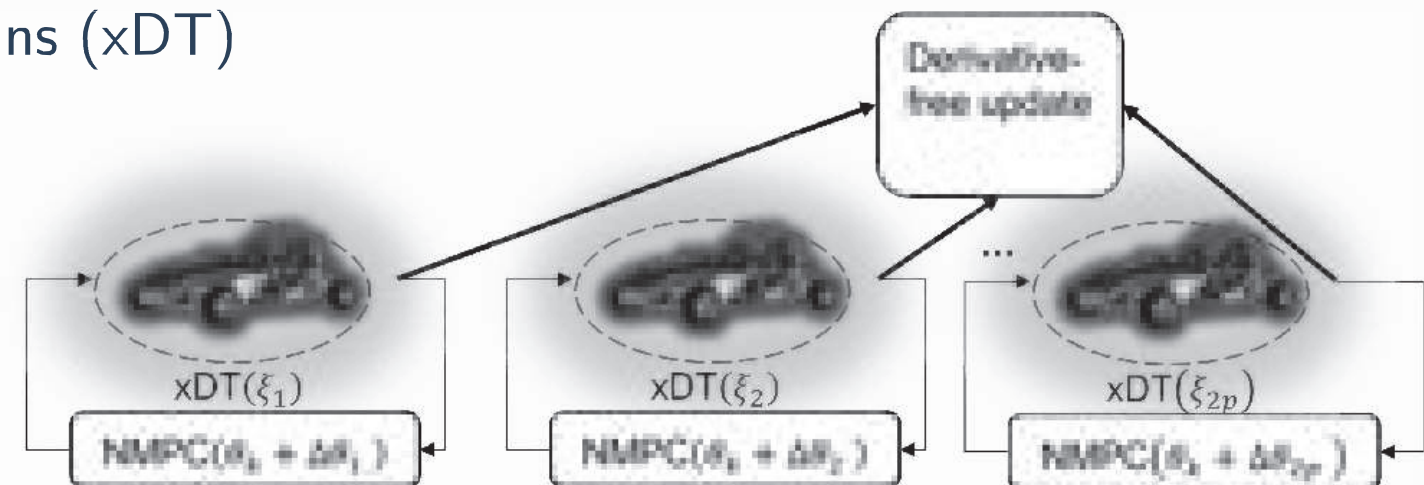
Target domain autonomous driving

- Run for T seconds
- Measure V
- k : tuning iteration index



Train using parallel executable digital twins (xDT)

- Run each xDT for T seconds
- Stochastic gradient by sampling xDTs and combining with real-world data (Zero-th order optimization)



J. P. Allamaa et al. (2022). "Sim2real for Autonomous Vehicle Control using Executable Digital Twin". In: vol. 55. 24. 10th IFAC Symposium on Advances in Automotive Control AAC 2022, pp. 385–391. DOI: 10.1016/j.ifacol.2022.10.314.

Learning Policies – Automatic controller adaptation

- Automatic controller tuning/adaptation
- Facilitates transfer from simulation to real-world and minimizes manual human intervention
- Indirectly compensates for uncertainties with no additional complexity on the NMPC formulation

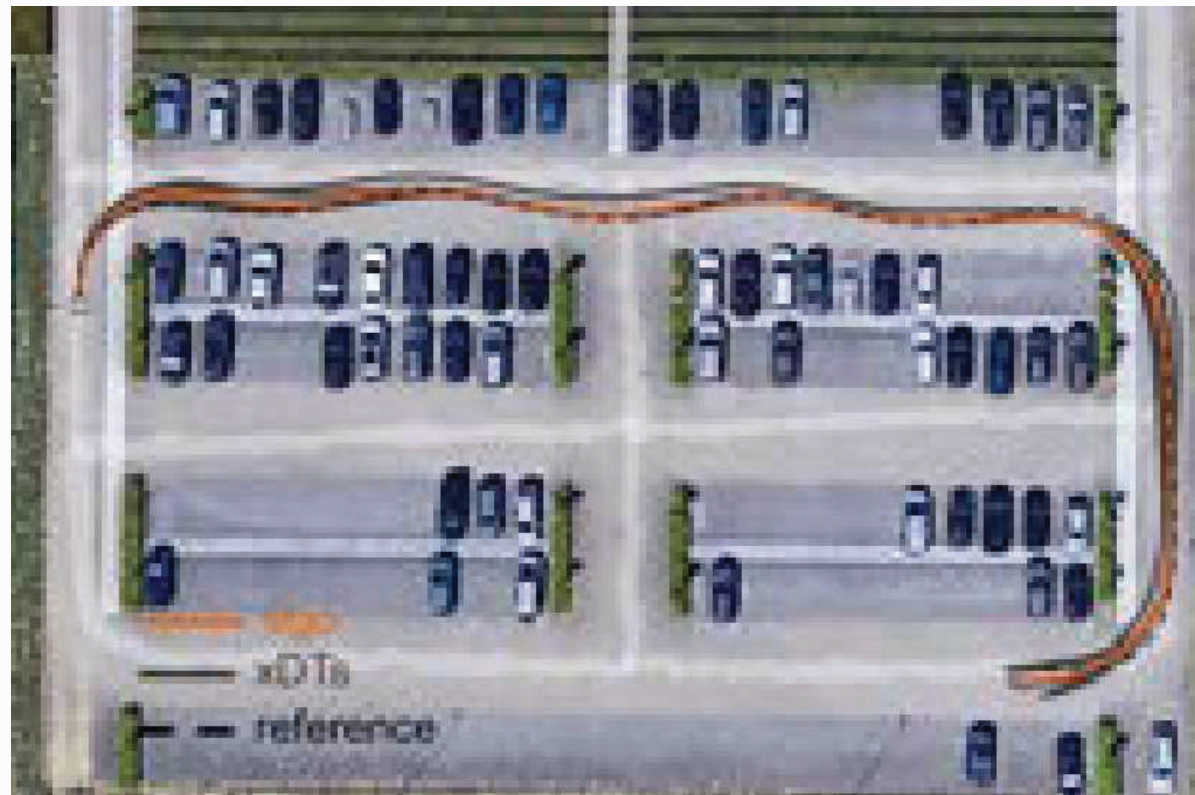


Figure 1: Tuning process: starting with a unity Q and R matrices

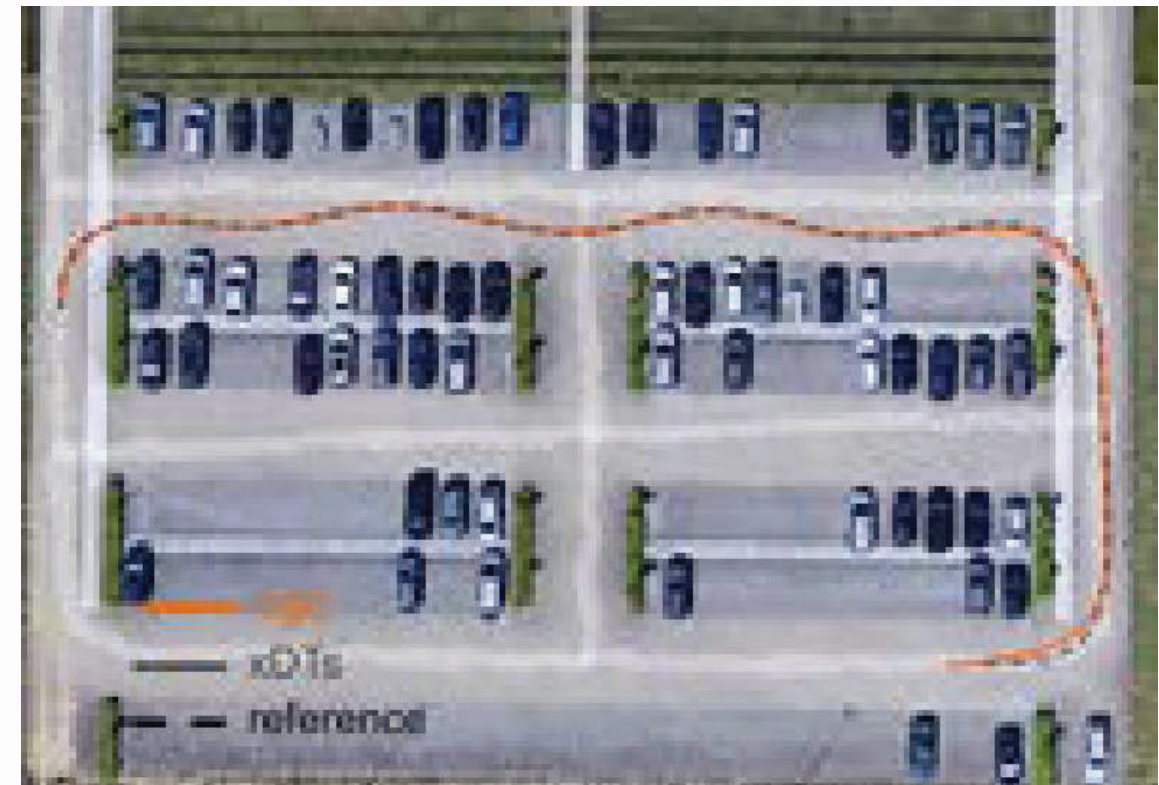


Figure 2: Tuning process: after one single iteration, performance is improved and the xDTs have a smaller variance

Learning Policies – Automatic controller adaptation

Sim2Real transfer is a major challenge for control engineers as we tend to **overtune** in simulation



Figure 3: Policy tuned in simulation to reach less than 10cm of tracking accuracy, fails to produce satisfactory performance in real-world

The policy has never encountered steering delays and large path deviation in simulation: overtuned for exact path tracking



Learning Policies – Automatic controller adaptation

- Real world demonstrations: tuning completed within 10 minutes
- We include domain randomization on the chosen path, vehicle parameters, input/output noise to obtain a generalizable controller

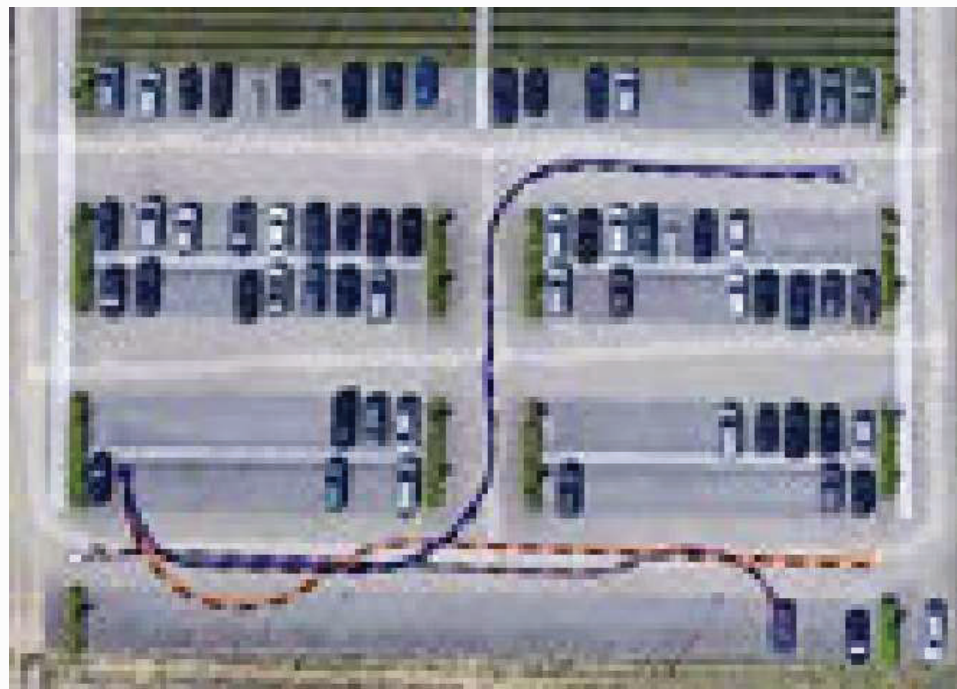


Figure 4: ViL tuning to simultaneously improve tracking performance and minimize the NMPC cost over a multitude of trajectories





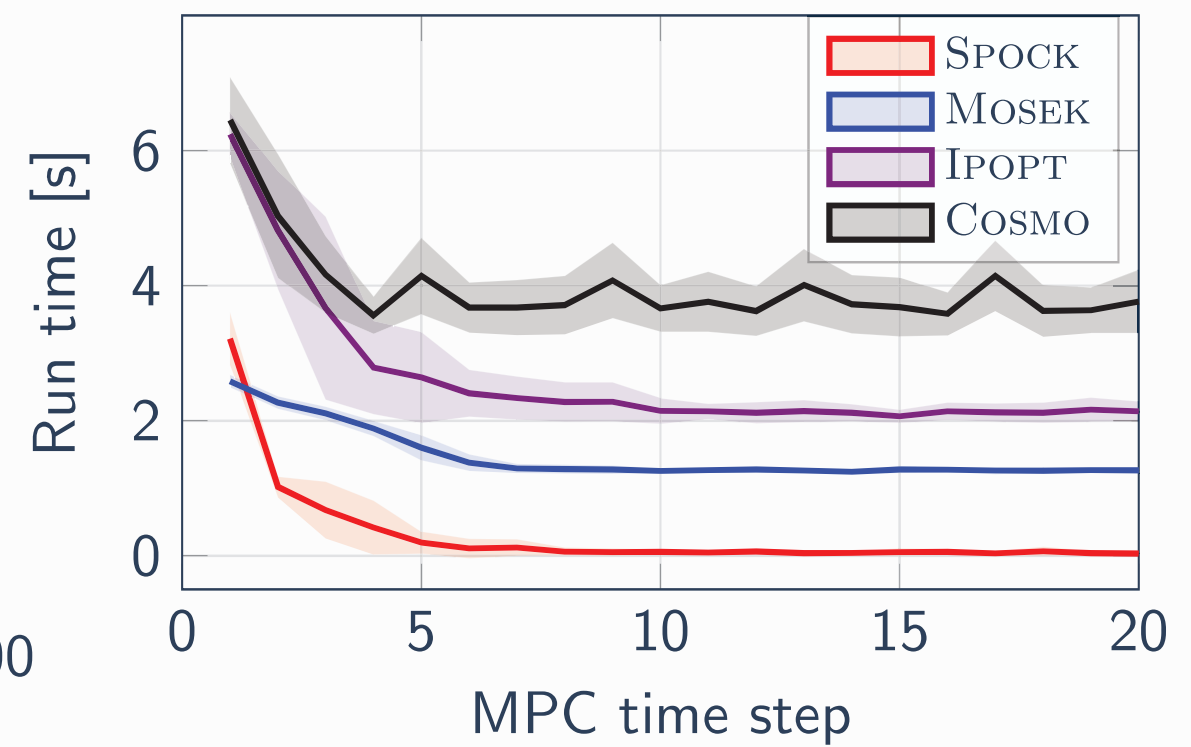
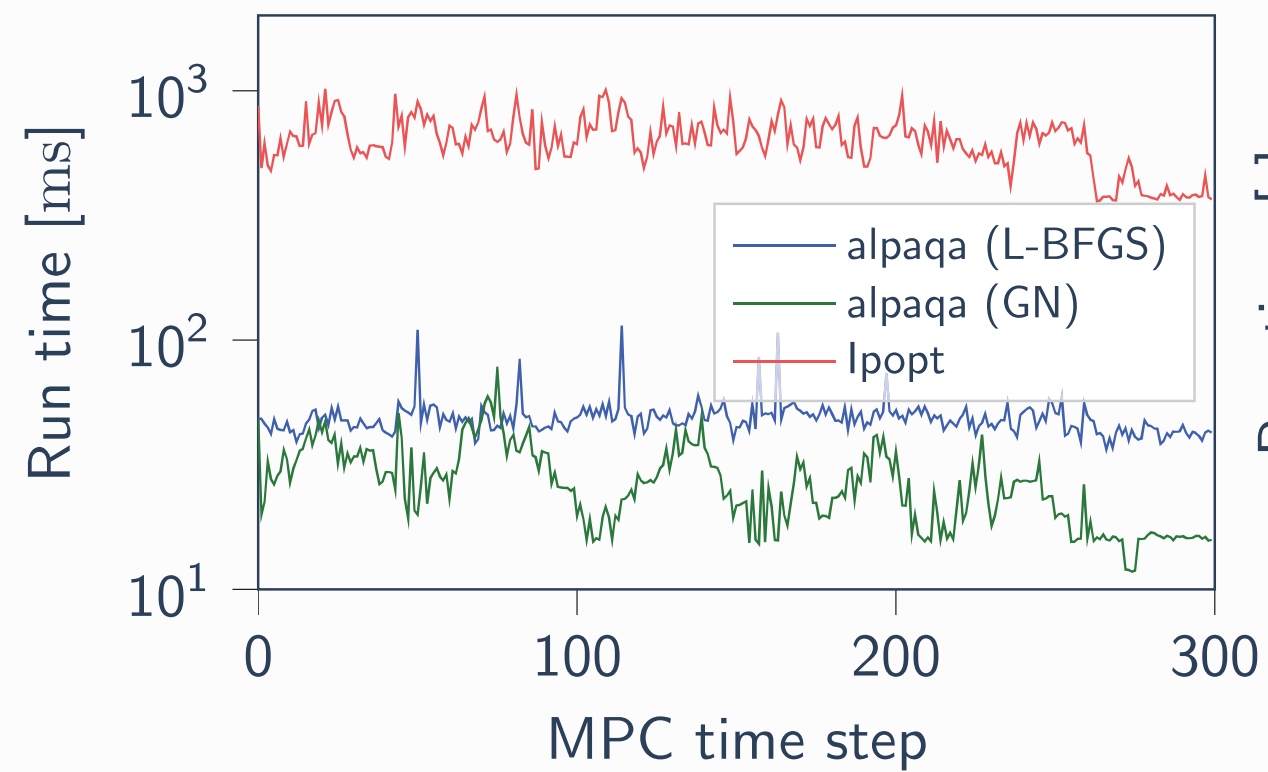
MPC require real-time solutions to potentially non-convex, large scale, optimization problems

Alpaqa	Large-scale, nonconvex nonlinear programs. Tailored to MPC	github.com/kul-optec/alpaqa
QPALM	(Nonconvex) quadratic programs	github.com/kul-optec/QPALM
SPOCK	multi-stage, multi-modal distributionally robust optimal control	github.com/kul-optec/spock.jl

... and more on our Github page

-
- P. Pas, M. Schuurmans, and P. Patrinos (2022). “Alpaqa: A matrix-free solver for nonlinear MPC and large-scale nonconvex optimization”. In: **2022 European Control Conference (ECC)**, pp. 417–422
- B. Hermans, A. Themelis, and P. Patrinos (2022). “QPALM: A Proximal Augmented Lagrangian Method for Nonconvex Quadratic Programs”. In: **Mathematical Programming Computation** 14.3
- A. Bodard et al. (2022). “SPOCK: A Proximal Method for Multistage Risk-Averse Optimal Control Problems”. In: arXiv: 2212.01110

Numerical algorithms – Benefits of specialized solvers

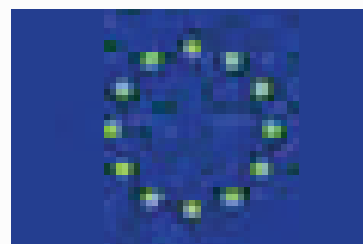


Conclusions

Autonomous systems: closing the loop between learning and control

- challenging problem: sim2real gap, need to learn from own experience
- learning in the real world: slow and dangerous
- safety should be priority # 1—use learning to improve performance
- off-the-shelf, end-to-end AI tools will not solve the problem
- combination of offline learning with real-time adaptation and optimization
- challenging optimization problems—need for better embedded solvers

Contributors



References i

- Allamaa, J. P. et al. (2022). “Sim2real for Autonomous Vehicle Control using Executable Digital Twin”. In: vol. 55. 24. 10th IFAC Symposium on Advances in Automotive Control AAC 2022, pp. 385–391. DOI: 10.1016/j.ifacol.2022.10.314.
- Arcari, E. et al. (Dec. 2019). “Dual Stochastic MPC for Systems with Parametric and Structural Uncertainty”. In: **arXiv:1912.10114 [cs, eess]**. arXiv: 1912.10114 [cs, eess].
- Åström, K. and A. Helmersson (June 1986). “Dual Control of an Integrator with Unknown Gain”. In: **Computers & Mathematics with Applications** 12.6, pp. 653–662. ISSN: 08981221.
- Bodard, A. et al. (2022). “SPOCK: A Proximal Method for Multistage Risk-Averse Optimal Control Problems”. In: **arXiv: 2212.01110**.
- Brüggemann, S. and R. R. Bitmead (Feb. 2022). “Forward-Looking Persistent Excitation in Model Predictive Control”. In: **Automatica** 136, p. 110033. ISSN: 0005-1098.
- Coppens, P. and P. Patrinos (2020). “Sample Complexity of Data-Driven Stochastic LQR with Multiplicative Uncertainty”. In: **CDC**.
- (2022). “Safe learning LQR of linear dynamics with multiplicative noise”. In: **TAC (submitted)**. arXiv: 2207.06062.

References ii

- Coppens, P. and P. Patrinos (2023). “Policy iteration using Q-functions: Linear dynamics with multiplicative noise”. In: **IFAC (submitted)**. arXiv: 2212.01192.
- Evens, B., M. Schuurmans, and P. Patrinos (2022). “Learning MPC for Interaction-Aware Autonomous Driving: A Game-Theoretic Approach”. In: **2022 European Control Conference (ECC)**, pp. 34–39.
- Feldbaum, A. (1961). “Theory of dual control”. In: **Avtomat. i Telemekh** 21.
- Hermans, B., A. Themelis, and P. Patrinos (2022). “QPALM: A Proximal Augmented Lagrangian Method for Nonconvex Quadratic Programs”. In: **Mathematical Programming Computation** 14.3.
- Houska, B. et al. (Oct. 2016). “Self-Reflective Model Predictive Control”. In: **arXiv:1610.03228 [math]**. arXiv: 1610.03228 [math].
- <https://today.tamu.edu>** (2020).
https://today.tamu.edu/wp-content/uploads/2020/01/20200128_Jaywalkers_JKE_006.jpg.
- Hu, H. and J. F. Fisac (2023). “Active Uncertainty Reduction for Human-Robot Interaction: An Implicit Dual Control Approach”. In: **Algorithmic Foundations of Robotics XV - Proceedings of the Fifteenth Workshop on the Algorithmic Foundations of Robotics**. Springer Nature, pp. 385–401.
- Mesbah, A. (2018). “Stochastic Model Predictive Control with Active Uncertainty Learning: A Survey on Dual Control”. In: **Annual Reviews in Control** 45, pp. 107–117. ISSN: 13675788.

References iii

- Mnih, V. et al. (Feb. 2015). “Human-Level Control through Deep Reinforcement Learning”. In: **Nature** 518.7540, pp. 529–533. ISSN: 1476-4687.
- Pas, P., M. Schuurmans, and P. Patrinos (2022). “Alpaqa: A matrix-free solver for nonlinear MPC and large-scale nonconvex optimization”. In: **2022 European Control Conference (ECC)**, pp. 417–422.
- Schuurmans, M., A. Katriniok, et al. (2022). “Safe, Learning-Based MPC for Highway Driving under Lane-Change Uncertainty: A Distributionally Robust Approach”. In: **JAI**. Submitted. arXiv: 2206.13319.
- Schuurmans, M. and P. Patrinos (2023). “A General Framework for Learning-Based Distributionally Robust MPC of Markov Jump Systems”. In: **TAC**. to appear. arXiv: 2106.00561.
- Silver, D. et al. (Jan. 2016). “Mastering the Game of Go with Deep Neural Networks and Tree Search”. In: **Nature** 529.7587, pp. 484–489. ISSN: 1476-4687.
- Sternby, J. (Dec. 1976). “A Simple Dual Control Problem with an Analytical Solution”. In: **IEEE Transactions on Automatic Control** 21.6, pp. 840–844. ISSN: 1558-2523.
- Wang, R., M. Schuurmans, and P. Patrinos (2022). **Interaction-Aware Model Predictive Control for Autonomous Driving**. arXiv: 2211.17053.