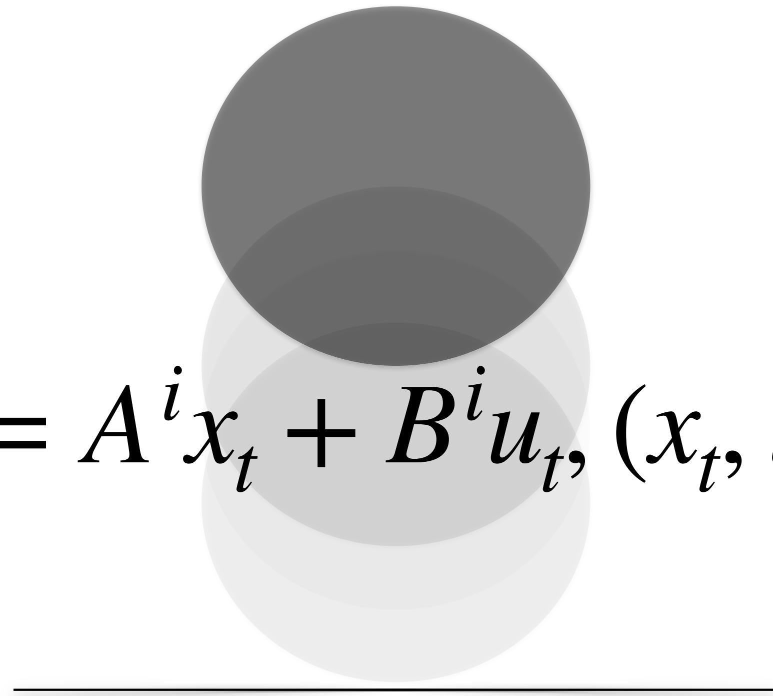
# Fast Non-Parametric Learning to Accelerate Mixed-Integer Programming for Hybrid MPC

Jia-Jie Zhu
Empirical Inference Department
Max Planck Institute for Intelligent Systems
Tübingen, Germany

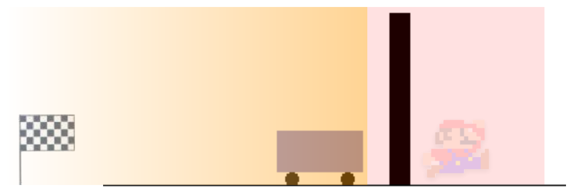
Georg Martius
Autonomous Learning Group
Max Planck Institute for Intelligent Systems
Tübingen, Germany



$$x_{t+1} = A^i x_t + B^i u_t, (x_t, u_t) \in \mathscr{C}^i$$

# Piecewise affine formulation (PWA) of Hybrid system

$$
\begin{cases}
x_1^+ = x_1 + x_2 \; \Delta t \\
x_2^+ = x_2 + \dfrac{\Delta t}{m} \; u, & \text{if } x \in \mathcal{C}_1 \\
x_1^+ = x_1 \\
x_2^+ = -\epsilon x_2, & \text{if } x \in \mathcal{C}_2
\end{cases}
$$

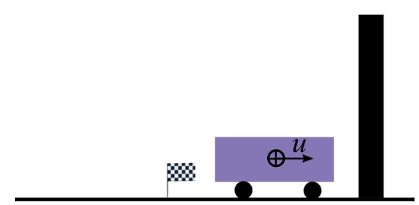


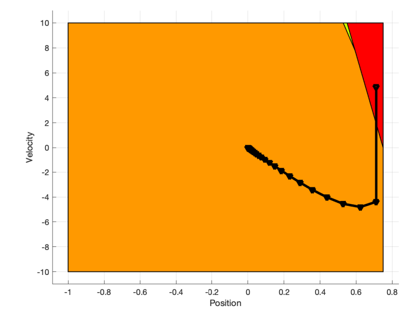
# Exact solution of mixed-integer program

"Big-M" formulation of MIP

$$
\begin{aligned}
\underset{u_t,\mu_t,t=0,\dots,N-1}{\text{minimize}} \quad & \sum_{t=1}^{N-1} x_t^\top Q x_t + u_t^\top R u_t + x_N^\top P x_N \\
\text{subject to} \quad & |x_{t+1} - A^i x_t - B^i u_t| \leq (1-\mu_t^i)M \\
& h(x_t, u_t) \leq (1-\mu_t^i)M \\
& \sum_i \mu_t^i = 1,\ \mu_t^i \in \{0,1\}, \quad \forall i, t, \\
& x_0 = x_p.
\end{aligned}
$$

1: Get state estimation $x_p$ .
2: Solve MIQP.
3: Apply the first control solution $u_0^*$.

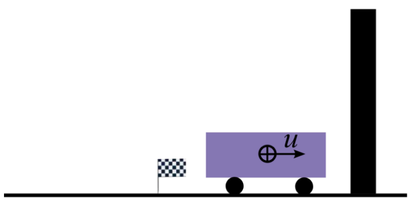


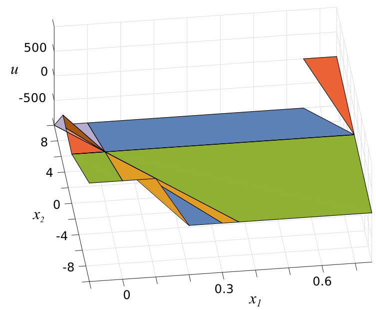




**But…**

☹️ Solving MIP online is (often) out of the question

# Offline Explicit HMPC




$$u^*(x) = F^i x + G^i,\ x \in \mathcal{R}_i,$$

where $\mathcal{R}_i$ is a critical region

**Offline**

1: Enumerate possible mode sequences.
2: For each mode sequence, solve multi-parametric program to characterize critical regions $R_i$.
3: For each region, store the PWA control law $u^*(x_p) = F^i x_p + G^i$ and the corresponding region.

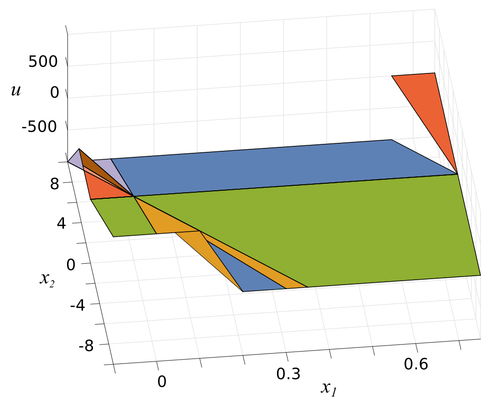
**Online**

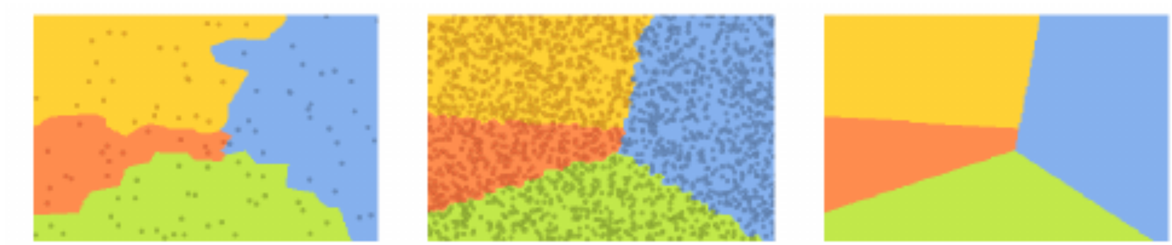
1: Get state estimation $x_p$
2: Locate the region $\mathcal{R}_i$ that contains $x_p \in \mathcal{R}_i$
3: Apply PWA control $u^*(x_p) = F^i x_p + G^i$

But…

- ☹️ The number of critical regions can be huge
- ☹️ For a given $x$, finding the critical region $\mathcal{R}_i$ can be slow

# Main idea



Use Voronoi tessellation induced by the nearest neighbor classifier to approximate feasible regions for hybrid MPC.
The NN classifier is trained using supervised learning to classify the mode sequence switching mode sequences based on the *sampled* states.

As we have more and more samples,
the region approximates feasible regions for mode sequences

# Computation reduction

$$\underset{u_t,\mu_t,t=0,\ldots,N-1}{\text{minimize}} \quad \sum_{t=1}^{N-1} x_t^\top Q x_t + u_t^\top R u_t + x_N^\top P x_N$$

$$\text{subject to} \quad |x_{t+1} - A^i x_t - B^i u_t| \leq (1-\mu_t^i)M$$

$$h(x_t, u_t) \leq (1-\mu_t^i)M$$

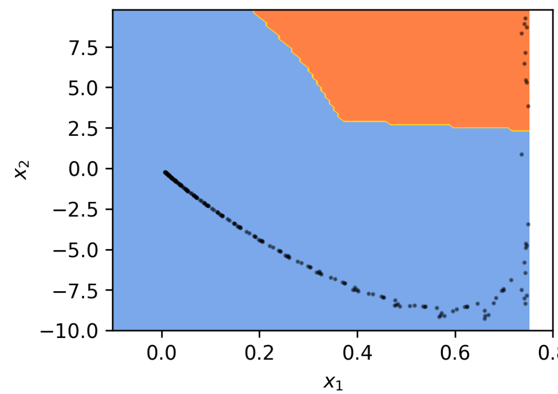
$$\sum_i \mu_t^i = 1,\ \mu_t^i \in \{0,1\}, \quad \forall i, t,$$

$$x_0 = x_p.$$

If we fix the mode sequences, the computational cost is drastically reduced.

LNMS

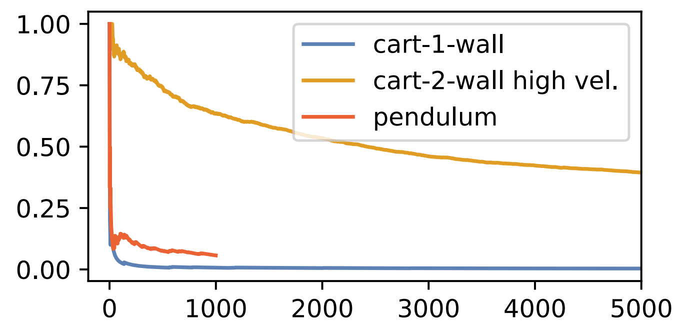
1: Get state estimation $x_p$ .
2: Query the nearest neighbor classifier (with dataset $\mathcal{D}$) for the mode sequence $\mathcal{M} = \{m_i\}_{i=1}^N$
3: Solve hybrid MPC with integer variables $\{m_i\}_{i=1}^N$ as warm-start solution. Terminate when computational budget reached.
4: Add the $(x_p, \mathcal{M}^*)$-pair to the dataset $\mathcal{D}$, where $\mathcal{M}^*$ is the integer solution obtained last step.
5: Apply the first control solution $u_0^*$.

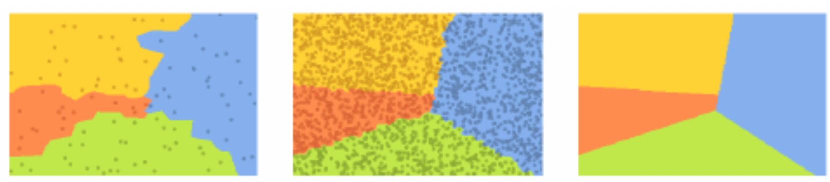


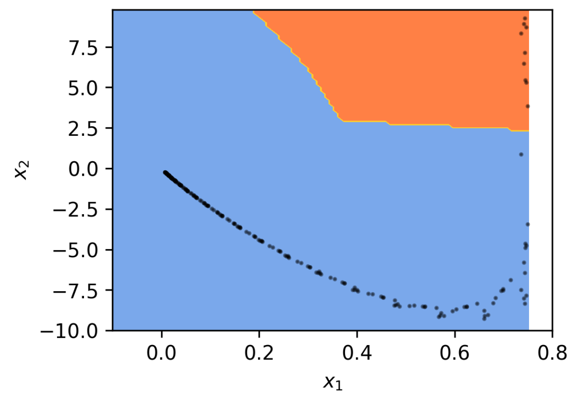
# Computation reduction

**Idea**
Given the dataset $\mathcal{D}$ of size $n$, let $P_n^{\mathrm{MIP}}$ denote the probability of having to execute MIP solver in our algorithm. Then

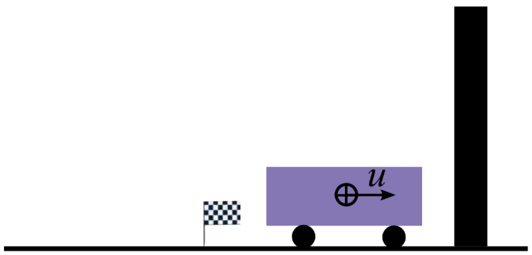
$$P_n^{\mathrm{MIP}} \to 0 \text{ as } n \to \infty.$$



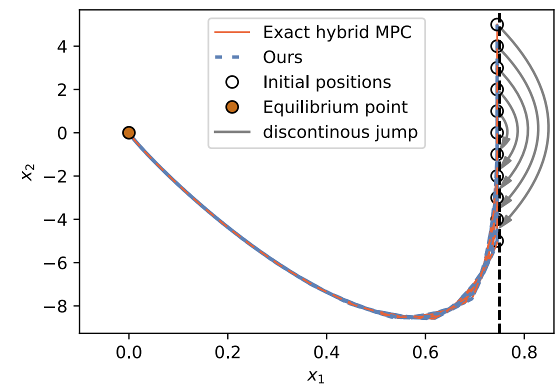



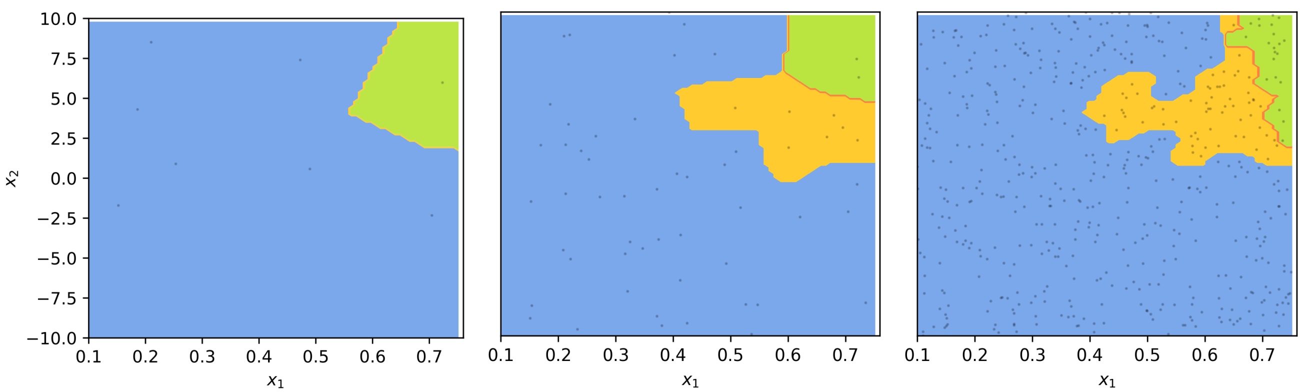


# Running example: cart and wall



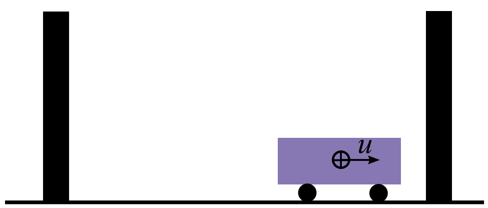

$$\begin{cases} x_1^+ = x_1 + x_2 \ \Delta t \\ x_2^+ = x_2 + \dfrac{\Delta t}{m} \ u, & \text{if } x \in \mathcal{C}_1 \\ x_1^+ = x_1 \\ x_2^+ = -\epsilon x_2, & \text{if } x \in \mathcal{C}_2 \end{cases}$$

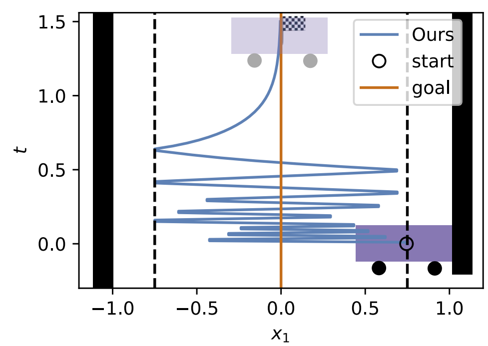


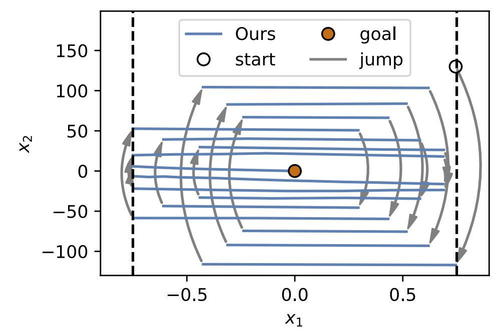



(a) 15 samples (2 region) (b) 100 samples (3 regions) (c) 1000 smpls (3 reg.)

# Control-constrained, high initial velocity

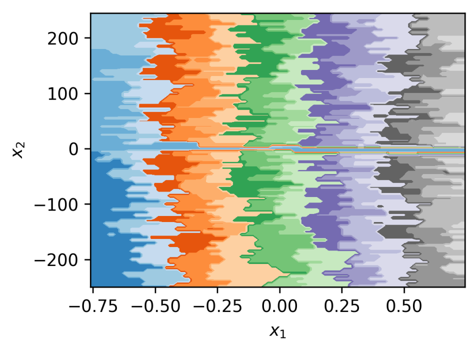


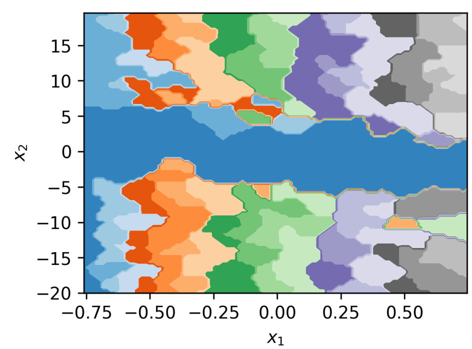




(a) offline, 5000 samples
493 regions

(b) zoomed center of (a)
153 regions

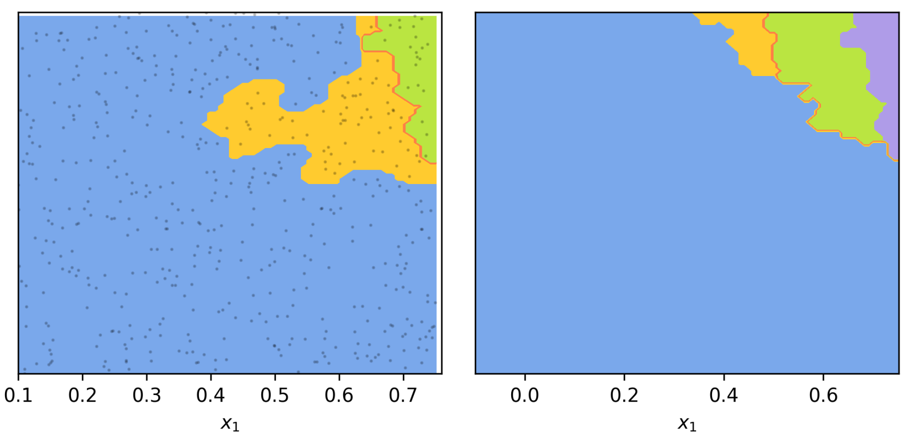


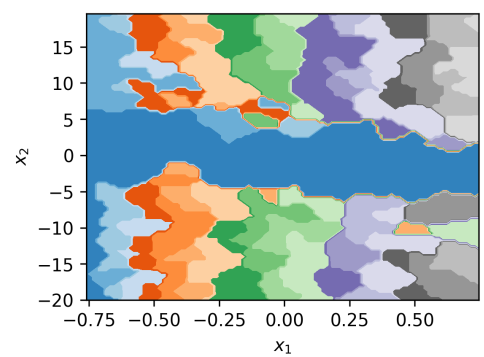
# Improving by warm starting MIP

$$
\begin{aligned}
\underset{u_t,\mu_t,t=0,\ldots,N-1}{\text{minimize}} \quad & \sum_{t=1}^{N-1} x_t^\top Q x_t + u_t^\top R u_t + x_N^\top P x_N \\
\text{subject to} \quad & |x_{t+1} - A^i x_t - B^i u_t| \le (1-\mu_t^i) M \\
& h(x_t, u_t) \le (1-\mu_t^i) M \\
& \sum_i \mu_t^i = 1,\ \mu_t^i \in \{0,1\}, \quad \forall i, t, \\
& x_0 = x_p.
\end{aligned}
$$

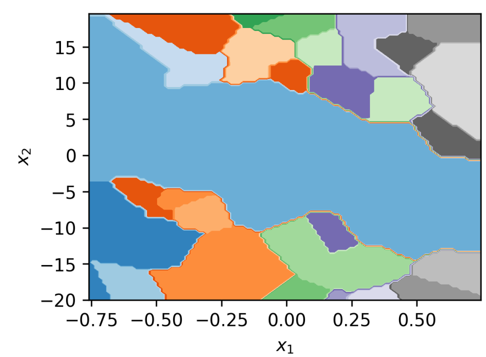
Improve on the resulting controller by "relabeling" the stored samples $\mathcal{D} = \{x_i, \mathcal{M}_i\}$ via warm-starting techniques of MIP.




(b) zoomed center of (a)
153 regions




(c) improved version of (b)
33 regions



# Appliction: used as oracles for Imitation learning

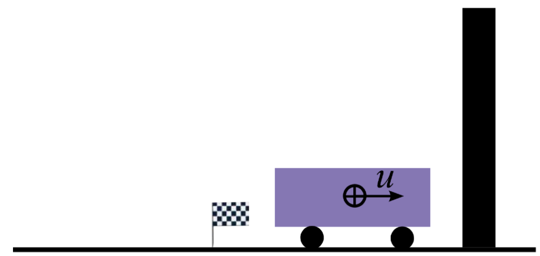
Use the proposed method to gather dataset $\mathcal{D} = \{x_i, u_i\}_{i=1,2,\ldots}$, learn a policy directly

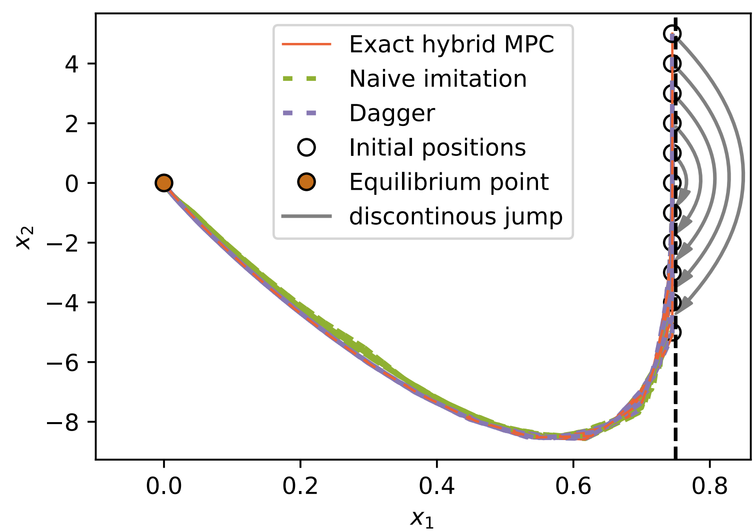
$$\mu : x \mapsto u$$

by, e.g., (parametric) regression

$$\min_{\theta} \Sigma_i \|\mu_\theta(x_i) - u_i\|^2.$$

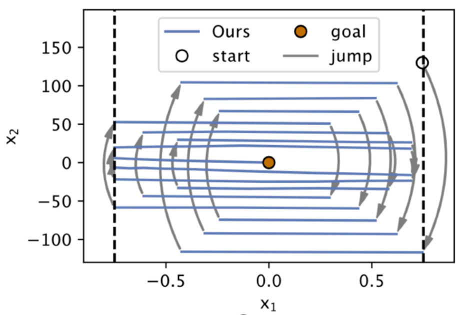
Online policy evaluation is fast!



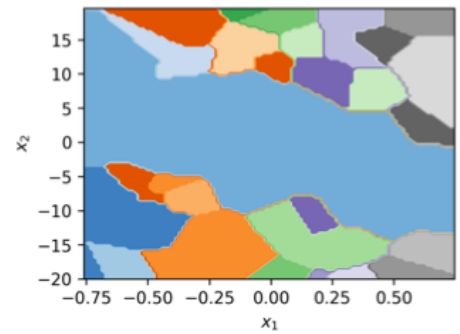



# Summary

- **The problem:** Solving hybrid MPC under the piecewise-affine model involves computationally heavy mixed-integer quadratic programming (MIQP).
- **Main idea:** A simple non-parametric classifier learns predict the switching mode sequences based on sampled states.
- **Insight**: The Voronoi tessellation induced by the nearest neighbor classifier approximates the MIQP control law in hybrid MPC.




A hybrid system with jumps




Proposed learning-control law

Another talk at IFAC 20: Z et al., Kernel Probabilistic Programming for Dynamical Systems