Data-Driven Abstraction of Monotone Systems

Anas Makdesi, Antoine Girard & Laurent Fribourg

CNRS, Laboratoire des Signaux et Systèmes (L2S)
CNRS, Laboratoire Méthodes Formelles (LMF)
Gif-sur-Yvette, France

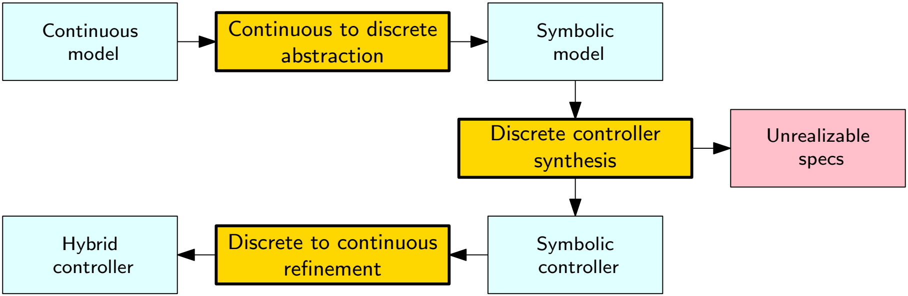
Workshop Automatique & IA
June 10-11, 2021

Abstraction-based control

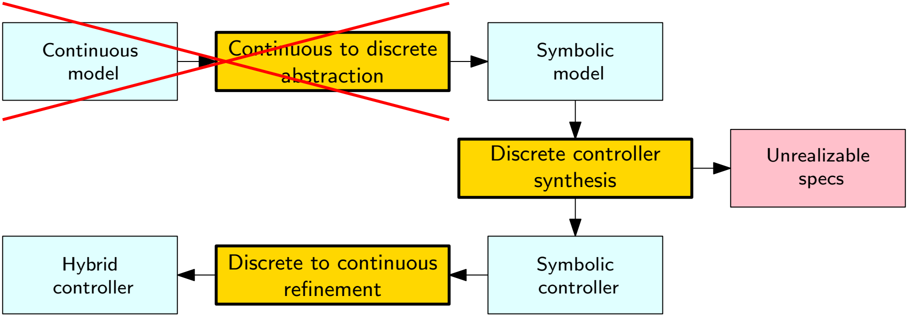
A computational approach to controller design based on finite-state approximations (symbolic models) of continuous dynamics:

- Nonlinear dynamics with state/input constraints and bounded disturbances
- Rich specifications (safety, reachability, stability, temporal logic...)
- Various performance criteria (cumulative, maximum, average...)
- Formal guarantees

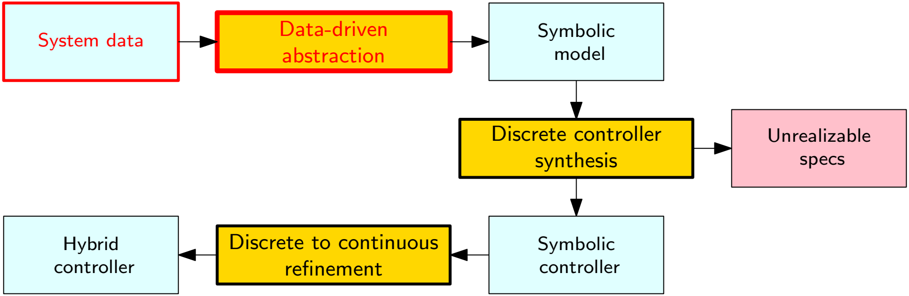
From model-based to data-driven abstraction



From model-based to data-driven abstraction



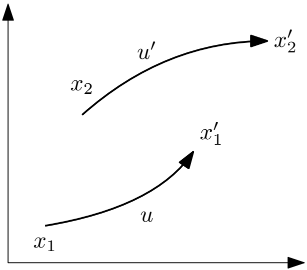
From model-based to data-driven abstraction



Monotone systems

Let f a monotone map, consider the system:

$$x^+ = f(x,u),\ x \in X \subseteq \mathbb{R}^n,\ u \in U \subseteq \mathbb{R}^p.$$

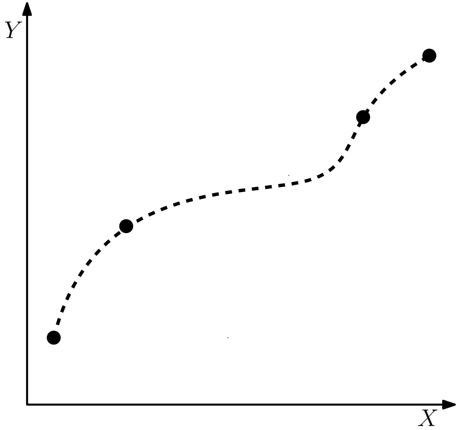


$x_1 \preceq x_2, u \preceq u' \Rightarrow x_1' \preceq x_2'$

- Characterization:
$$\frac{\partial f_i}{\partial x_j} \geq 0,\ \frac{\partial f_i}{\partial u_k} \geq 0,\ \forall i,j,k$$
- Applications: vehicles, energy, biology...

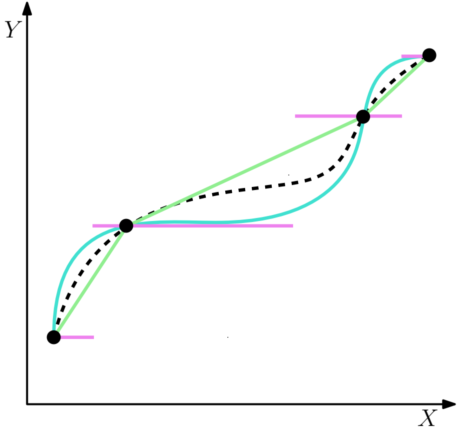
Learning monotone maps from data

Consider a set of samples of an unknown monotone map $f : X \to Y$.



Learning monotone maps from data

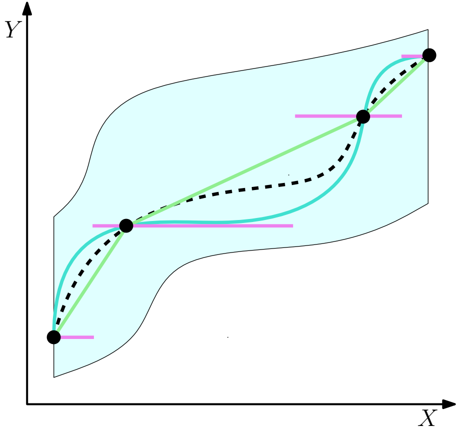
Consider a set of samples of an unknown monotone map $f : X \to Y$.



Consistent (monotone) maps

Learning monotone maps from data

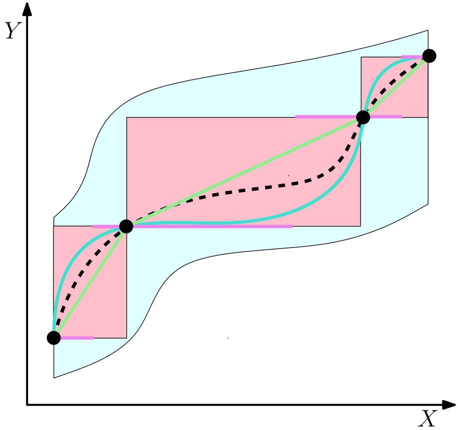
Consider a set of samples of an unknown monotone map $f : X \to Y$.



Simulating (set-valued) map

Learning monotone maps from data

Consider a set of samples of an unknown monotone map $f : X \to Y$.

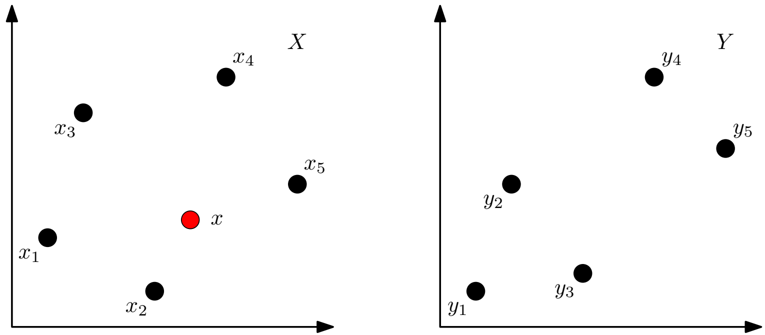


Minimal simulating map

Computing the minimal simulating map

Problem

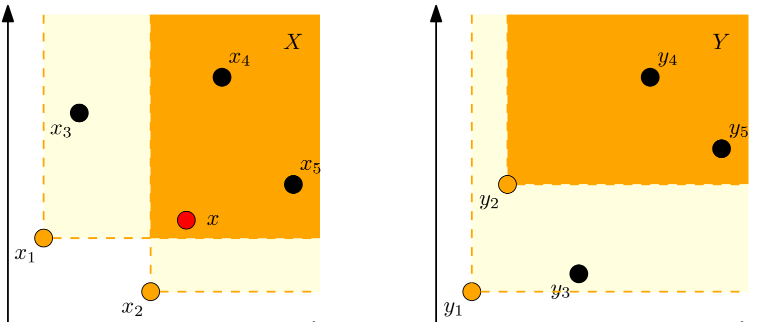
Given data set $\mathcal{D} = \{(x_k, y_k) |\ k = 1, \dots, N\}$, where $y_k = f(x_k)$, compute the minimal simulating map $F_{\mathcal{D}}(x)$.



Computing the minimal simulating map

Problem

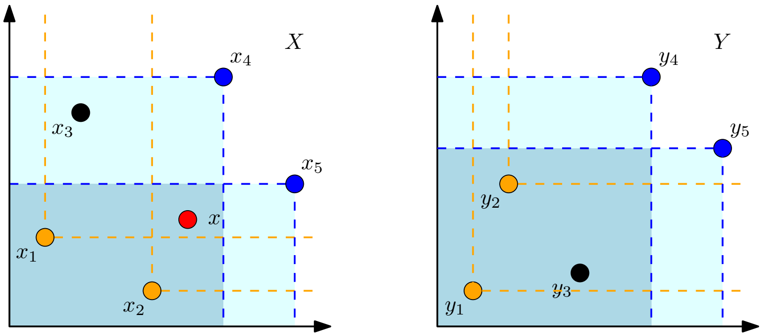
Given data set $\mathcal{D} = \{(x_k, y_k) |\ k = 1, \ldots, N\}$, where $y_k = f(x_k)$, compute the minimal simulating map $F_{\mathcal{D}}(x)$.



Computing the minimal simulating map

Problem

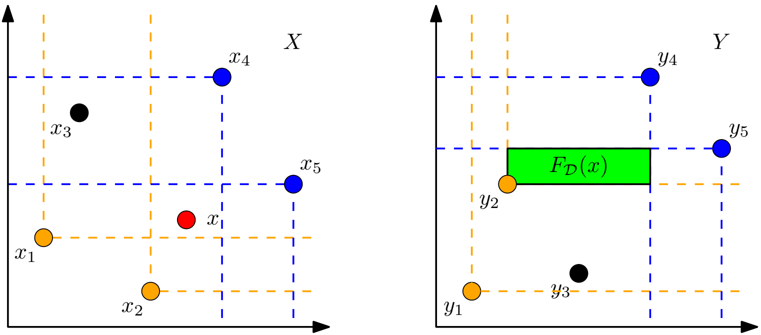
Given data set $\mathcal{D} = \{(x_k, y_k) | \ k = 1, \ldots, N\}$, *where* $y_k = f(x_k)$, *compute the minimal simulating map* $F_{\mathcal{D}}(x)$.



Computing the minimal simulating map

Problem

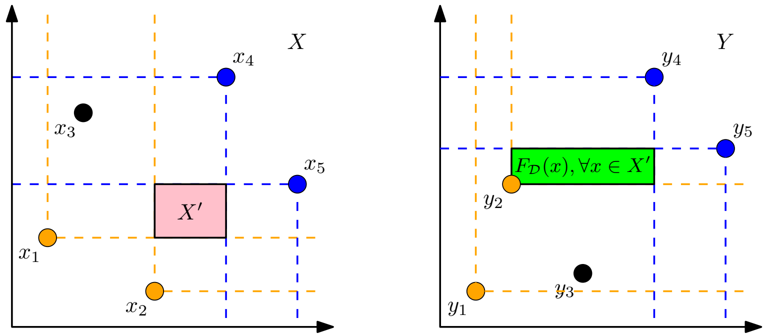
Given data set $\mathcal{D} = \{(x_k, y_k) | \ k = 1, \ldots, N\}$, where $y_k = f(x_k)$, compute the minimal simulating map $F_{\mathcal{D}}(x)$.



Computing the minimal simulating map

Problem

Given data set $\mathcal{D} = \{(x_k, y_k) | \ k = 1, \ldots, N\}$, where $y_k = f(x_k)$, compute the minimal simulating map $F_{\mathcal{D}}(x)$.



Symbolic representation of the minimal simulating map

Theorem

Given data set $\mathcal{D}$, there exist

- *a finite rectangular partition $(X_q)_{q \in Q}$ of X;*
- *rectangular subsets $(Y_q)_{q \in Q}$ of Y;*

such that for (almost) all $x \in X$

$$F_{\mathcal{D}}(x) = Y_q \iff x \in X_q.$$

Proposition

Given data set $\mathcal{D}_1$, $\mathcal{D}_2$, for all $x \in X$

$$F_{\mathcal{D}_1 \cup \mathcal{D}_2}(x) = F_{\mathcal{D}_1}(x) \cap F_{\mathcal{D}_2}(x).$$

Data-driven abstraction of monotone systems

Consider an unknown monotone systems Σ:

$$x_{t+1} = f(x_t, u_t),\ x_t \in X,\ u_t \in U$$

Given a data set $\mathcal{D} = \{(x_k, u_k y_k) |\ k = 1, \ldots, N\}$, where $y_k = f(x_k, u_k)$, the optimal data-driven model with formal guarantees is $\Sigma_{\mathcal{D}}$:

$$x_{t+1} \in F_{\mathcal{D}}(x_t, u_t),\ x_t \in X,\ u_t \in U$$

where $F_{\mathcal{D}}$ is the minimal simulating map.

There exist finite rectangular partitions $(X_q)_{q \in Q}$, $(U_p)_{p \in P}$ of X and U and rectangular subsets $(Y_{q,p})_{(q,p) \in Q \times P}$ such that

$$F_{\mathcal{D}}(x, u) = Y_{q,p} \iff x \in X_q, u \in U_p$$

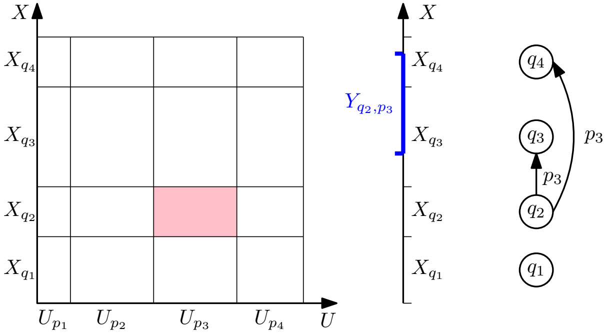
Computation of the symbolic model

Let us define the finite-state symbolic model $\hat{\Sigma}_{\mathcal{D}}$:

$$q_{t+1} \in \hat{F}_{\mathcal{D}}(q_t, p_t),\ q_t \in Q,\ p_t \in P$$

where

$$q^+ \in \hat{F}_{\mathcal{D}}(q, p) \iff Y_{q,p} \cap X_{q^+} \neq \emptyset$$



Properties of the symbolic model

Theorem

The behaviors of the optimal data-driven model $\Sigma_{\mathcal{D}}$ and of the symbolic model $\hat{\Sigma}_{\mathcal{D}}$ are equivalent (in the sense of alternating simulation).

Consequences:

- All realizable closed-loop behaviors of $\hat{\Sigma}_{\mathcal{D}}$ are also realizable for $\Sigma_{\mathcal{D}}$ and conversely.
- Using the symbolic model $\hat{\Sigma}_{\mathcal{D}}$ for controller synthesis is both safe (formal guarantees) and optimal (no conservatism).

Main shortcoming:

- The numbers of symbolic states and inputs grow polynomially with the size of the data set ($|Q| = (N+1)^n$ and $|P| = (N+1)^p$).

Efficient data-driven abstraction

Methods to improve the efficiency of data-driven abstraction:

1. Computational complexity of data-driven abstraction:
   - Naive approach: $\mathcal{O}\big(N \times |Q| \times |P|\big)$
   - By factorizing some computations: $\mathcal{O}\big(N \times \log(|Q| \times |P|) + |Q| \times |P|\big)$
   - With $|Q| \times |P| = (N+1)^{n+p}$, we get a slight improvement:

$$\mathcal{O}\big(N^{n+p+1}\big) \text{ vs } \mathcal{O}\big(N^{n+p}\big)$$

2. Fix the partitions $(X_q)_{q\in Q}$, $(U_p)_{p\in P}$ a priori:
   - Still safe, but introduces some conservatism: optimal in the class of simulating maps piecewise constant on these partitions.
   - The complexity becomes linear in the size of the data set.

Extension to non-monotone systems

The approach can be extended to the following class of (non-monotone) systems

$$x_{t+1} = f(x_t, u_t) + d_t,\ x_t \in X,\ u_t \in U$$

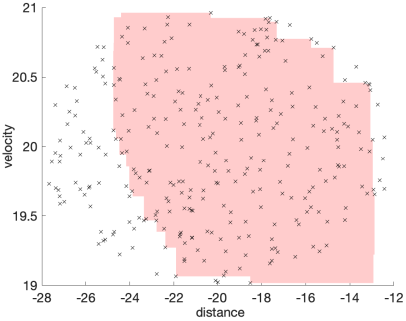
where:

- the partial derivatives of the unknown map f have known lower (or/and upper) bounds;
- d_t is an unknown disturbance with known lower and upper bounds.

If not known, the bounds can be estimated from the data set with probabilistic guarantees.

Example - leader-follower platoon

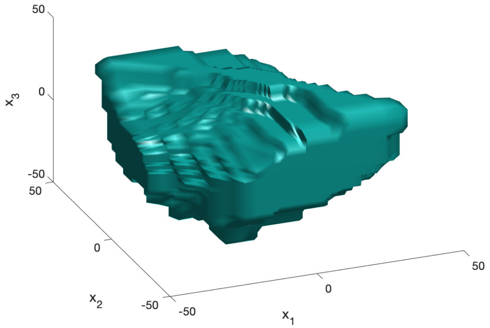
- Monotone system with 2 states, 1 input: velocity v, distance d and torque
- Symbolic model with data-driven partition: $|\mathcal{D}| = 300$, $|Q| = 90601$, $|P| = 301$
- Specification: $v \in [19, 21]$ and $d \in [-28, -12]$
- CPU time: 15 minutes



Maximal controlled invariant

Example - Lorenz system

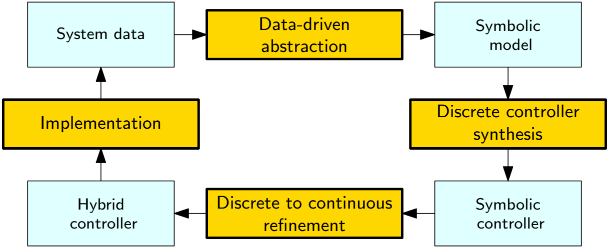
- Nonlinear system with bounded derivatives with 3 states, 1 input
- Symbolic model with fixed partition: $|\mathcal{D}| = 10^7$, $|Q| = 125000$, $|P| = 200$
- Specification: $x \in [-50, 50]^3$
- CPU time: $\sim$ 1h



Maximal controlled invariant

Conclusions

- Data-driven approach to controller synthesis for monotone systems:
  - Use of finite-state symbolic models
  - Safe (formal guarantees) and optimal (no conservatism)
  - Extension to (non-monotone) systems with bounded derivatives and bounded disturbances
- Next step: safe learning using online data collection



References

- Makdesi, Girard & Fribourg, Data-Driven Abstraction of Monotone Systems, *Learning for Dynamics and Control Conference (L4DC)*, 2021.
- Makdesi, Girard & Fribourg, Efficient Data-Driven Abstraction of Monotone Systems with Disturbances, *IFAC Conference on Analysis and Design of Hybrid Systems (ADHS)*, 2021.