Symbolic Control of Nonlinear Systems

Safety, Optimization & Learning

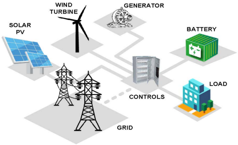
Antoine Girard

Laboratoire des Signaux et Systèmes (L2S)
CNRS, Université Paris-Saclay

School on Safe Autonomous Systems
Université Paris-Saclay, 2023

Cyber-physical systems

Cyber-physical systems (CPS) are physical systems enhanced with computation and communication capabilities: *smart vehicle, smart grid, smart building...*



CPS characteristics:

- Evolve in uncertain and highly dynamic environment
- Are subject to critical safety requirements
- Achieve complex tasks with a high degree of autonomy

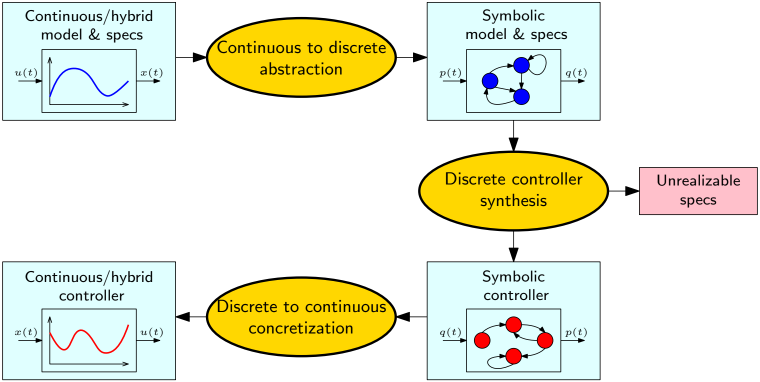
Formal methods for CPS

- Computational approaches for verification or synthesis of systems, based on mathematical formalization and rigorous reasoning, they require
  - Specifications whose semantics is precisely defined mathematically
  - Mathematical models of systems (possibly based on data)
- Suitable for the design of safety critical systems (correctness guarantees)
- Personal reflection fed by several projects CODECSYS (2016-2019) / PROCSYS (2017-2023) / Chair RTE-CentraleSupélec (2017-2027)

The symbolic control approach

Symbolic control is a formal method for controller synthesis:

- based on symbolic (i.e. finite state) abstractions of systems
- applies to nonlinear systems with input/state constraints and bounded uncertainties
- mathematical correctness of synthesized controllers



Outline of the lecture

1. Fundamentals of symbolic control:
   - Discrete controller synthesis
     Safety, reachability and recurrence
   - Symbolic control of nonlinear systems
     System abstraction, controller concretization, robustness
2. Recent advances in symbolic control:
   - Symbolically-guided model predictive control
     High performance controllers with safety guarantees
   - Data-driven symbolic control
     Towards safe learning approaches for nonlinear systems

Outline of the lecture

1. Fundamentals of symbolic control:
   - Discrete controller synthesis
     Safety, reachability and recurrence
   - Symbolic control of nonlinear systems
     System abstraction, controller concretization, robustness
2. Recent advances in symbolic control:
   - Symbolically-guided model predictive control
     High performance controllers with safety guarantees
   - Data-driven symbolic control
     Towards safe learning approaches for nonlinear systems

Transition systems

Definition

A transition system is a tuple $S = (Q, P, F)$ where

- Q is a set of states
- P is a set of inputs
- $F : Q \times P \rightrightarrows Q$ is a (set-valued) transition map

S is said to be finite or symbolic if Q and P are finite.

- The set of enabled inputs at state $q \in Q$ is

$$\text{enab}_F(q) = \{p \in P |\ F(q,p) \neq \emptyset\}$$

- The set of non-blocking states is

$$\text{nbs}_F = \{q \in Q |\ \text{enab}_F(q) \neq \emptyset\}$$

- The transition system is deterministic, if

$$\forall q \in \text{nbs}_F,\ \forall p \in \text{enab}_F(q),\ \text{card}(F(q,p)) = 1$$

Trajectories

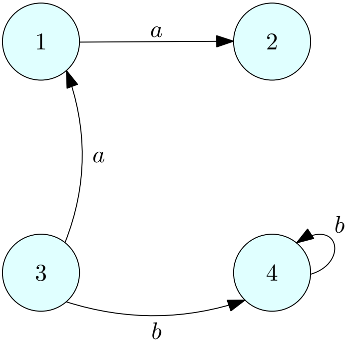
Definition

A trajectory of S is a couple of state and input sequences $(\{q_t\}_{t=0}^{t=T}, \{p_t\}_{t=0}^{t=T-1})$, where $T \in \mathbb{N} \cup \{+\infty\}$ and

$$p_t \in \mathrm{enab}_F(q_t) \text{ and } q_{t+1} \in F(q_t, p_t),\ \forall t = 0, \ldots, T-1.$$

- A trajectory is maximal, if $T = +\infty$ or else if $q_T \notin \mathrm{nbs}_F$
- A trajectory is complete, if $T = +\infty$
- The set of maximal trajectories of S is called the behavior of S, denoted $\mathcal{B}_{\max}(S)$

Symbolic system - example

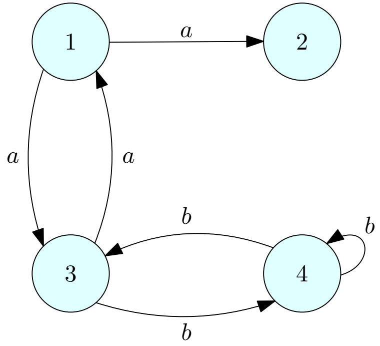


$$\begin{cases} \mathrm{enab}_F(1) & = & \{a\} \\ \mathrm{enab}_F(2) & = & \emptyset \\ \mathrm{enab}_F(3) & = & \{a, b\} \\ \mathrm{enab}_F(4) & = & \{b\} \end{cases}$$

$$\mathrm{nbs}_F = \{1, 3, 4\}$$

S is deterministic

Symbolic system - example



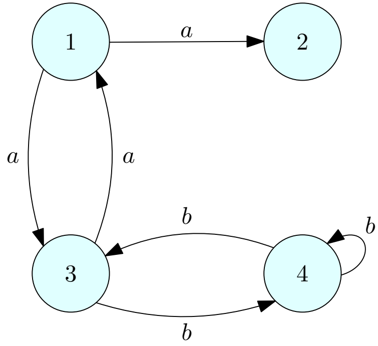
$$\begin{cases} \text{enab}_F(1) & = & \{a\} \\ \text{enab}_F(2) & = & \emptyset \\ \text{enab}_F(3) & = & \{a, b\} \\ \text{enab}_F(4) & = & \{b\} \end{cases}$$

$$\text{nbs}_F = \{1, 3, 4\}$$

$$\text{card}(F(1, a)) = \text{card}(F(4, b)) = 2$$

$$\implies S \text{ is non-deterministic}$$

Symbolic system - example



Trajectories:

$(1, a), (3, b), 4$

maximal:
$(1, a), (3, b), (4, b), (3, a), (1, a), 2$

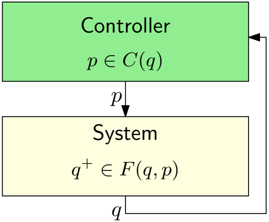
complete:
$(1, a), (3, b), (4, b), (4, b), (4, b), \ldots$

Controllers for transition systems

Definition

A (static state-feedback) controller for transition system $S = (Q, P, F)$ is a set-valued map $C : Q \rightrightarrows P$ such that for all $q \in Q$, $C(q) \subseteq \text{enab}_F(q)$.

The domain of the controller is $\text{dom}(C) = \{q \in Q |\ C(q) \neq \emptyset\}$.



The controlled dynamics is described by transition system $S_C = (Q, P, F_C)$ where

$$q^+ \in F_C(q,p) \iff p \in C(q) \text{ and } q^+ \in F(q,p)$$

Safety controllers

Safety: keep the system state in a set $Q_s \subseteq Q$ forever.

Definition

$C : Q \rightrightarrows P$ is a safety controller if for any initial state $q_0 \in \mathrm{dom}(C)$, all maximal trajectories of S_C $\left(\{q_t\}_{t=0}^{t=T}, \{p_t\}_{t=0}^{t=T-1}\right)$ are complete and satisfy:

$$\forall t \in \mathbb{N},\ q_t \in Q_s$$

Definition

A state q is safety controllable if there exists a safety controller C such that $q \in \mathrm{dom}(C)$.

The set of safety controllable states is denoted s-cont(S, Q_s).

Safety synthesis

Controllable predecessors of a subset $R \subseteq Q$:

$$\mathrm{Pre}(R) = \{q \in \mathrm{nbs}_F \mid \exists p \in \mathrm{enab}_F(q),\ F(q,p) \subseteq R\}.$$

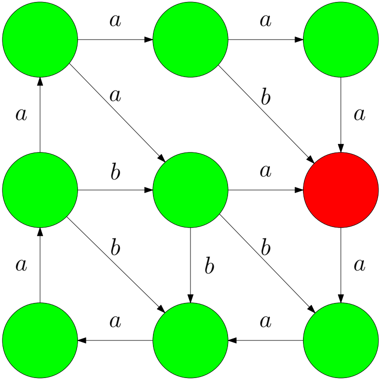
Safety synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cap \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{s-cont}(S, Q_s)$.



Controllable predecessors of a subset $R \subseteq Q$:

$$\mathrm{Pre}(R) = \{q \in \mathrm{nbs}_F \mid \exists p \in \mathrm{enab}_F(q),\ F(q,p) \subseteq R\}.$$

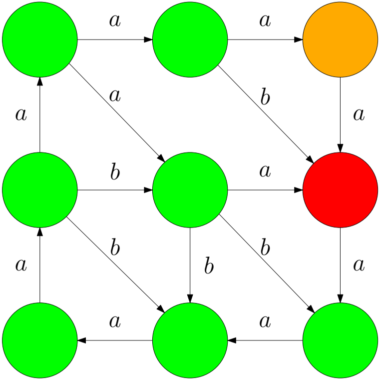
Safety synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cap \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = s\text{-}cont(S, Q_s)$.



Controllable predecessors of a subset $R \subseteq Q$:

$$\mathrm{Pre}(R) = \{q \in \mathrm{nbs}_F \mid \exists p \in \mathrm{enab}_F(q),\ F(q,p) \subseteq R\}.$$

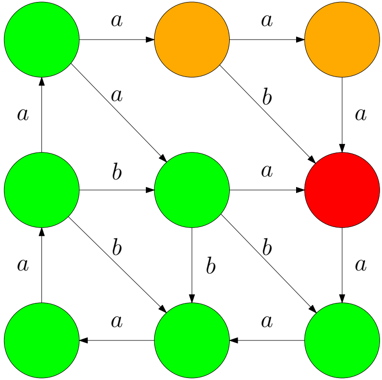
Safety synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cap \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{s-cont}(S, Q_s)$.



Safety synthesis

Controllable predecessors of a subset $R \subseteq Q$:

$$\mathrm{Pre}(R) = \{q \in \mathrm{nbs}_F \mid \exists p \in \mathrm{enab}_F(q),\ F(q,p) \subseteq R\}.$$

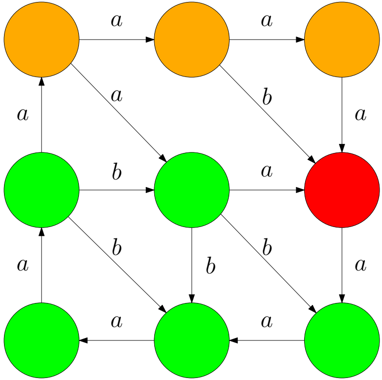
Safety synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cap \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = s\text{-}cont(S, Q_s)$.



Safety synthesis

Controllable predecessors of a subset $R \subseteq Q$:

$$\mathrm{Pre}(R) = \{q \in \mathrm{nbs}_F |\ \exists p \in \mathrm{enab}_F(q),\ F(q,p) \subseteq R\}.$$

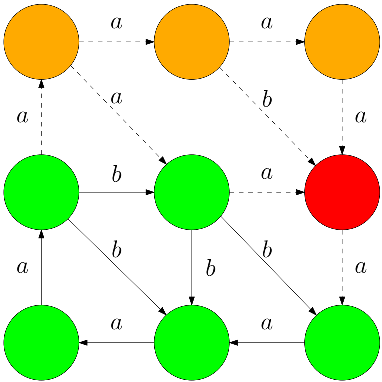
Safety synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cap \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{s-cont}(S, Q_s)$.



Reachability controllers

Reachability: bring the system state in $Q_s \subseteq Q$.

Definition

$C : Q \rightrightarrows P$ is a reachability controller if for any initial state $q_0 \in \mathrm{dom}(C)$, all maximal trajectories of S_C $\left(\{q_t\}_{t=0}^{t=T}, \{p_t\}_{t=0}^{t=T-1}\right)$ satisfy:

$$\exists t \in \mathbb{N},\; q_t \in Q_s.$$

Definition

A state q is reachability controllable if there exists a reachability controller C such that $q \in \mathrm{dom}(C)$.

The set of reachability controllable states is denoted r-cont(S, Q_s).

Safety and reachability synthesis algorithms look similar.

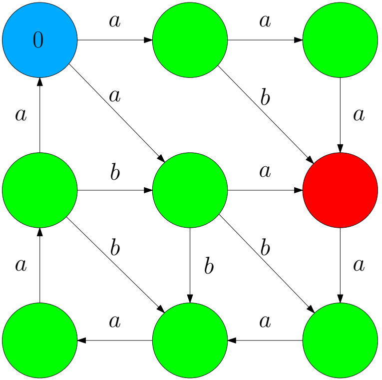
Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

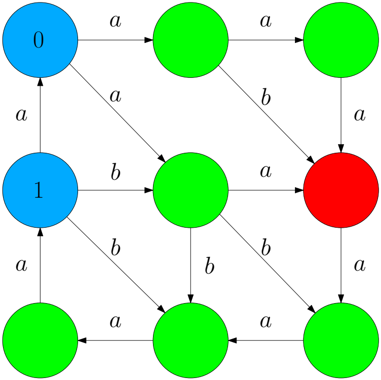
$R^* = \textit{r-cont}(S, Q_s)$.



Safety and reachability synthesis algorithms look similar.

Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$



Termination by finiteness of Q.

Theorem

$R^* = \textit{r-cont}(S, Q_s)$.

Safety and reachability synthesis algorithms look similar.

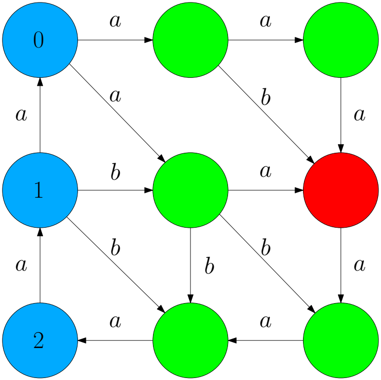
Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{r-cont}(S, Q_s)$.



Safety and reachability synthesis algorithms look similar.

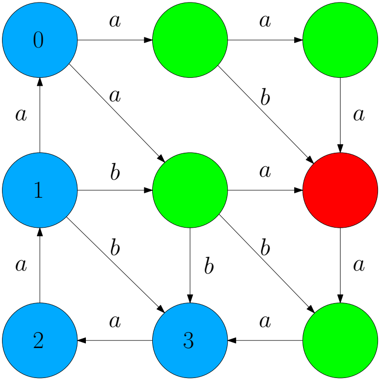
Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{r-cont}(S, Q_s)$.



Reachability synthesis

Safety and reachability synthesis algorithms look similar.

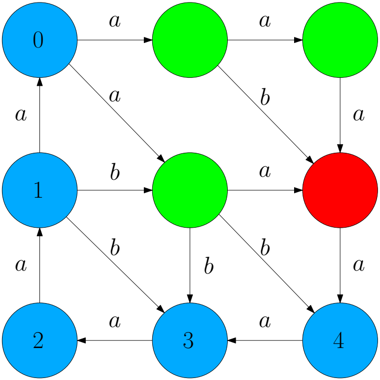
Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{r-cont}(S, Q_s)$.



Reachability synthesis

Safety and reachability synthesis algorithms look similar.

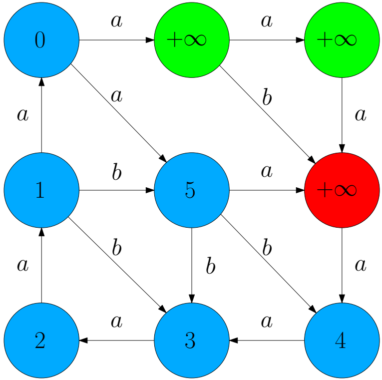
Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{r-cont}(S, Q_s)$.



Safety and reachability synthesis algorithms look similar.

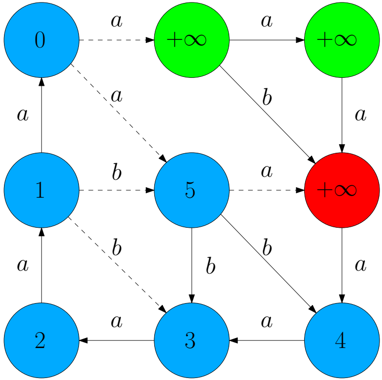
Reachability synthesis

$R_0 = Q_s$
loop
| $R_{k+1} = Q_s \cup \mathrm{Pre}(R_k)$
until $R_{k+1} = R_k$
return $R^* = R_k$

Termination by finiteness of Q.

Theorem

$R^* = \textit{r-cont}(S, Q_s)$.



Recurrence controllers

Recurrence: bring the system state in Q_s infinitely often.

Definition

$C : Q \rightrightarrows P$ is a recurrence controller if for any initial state $q_0 \in \mathrm{dom}(C)$, all maximal trajectories of S_C $\left(\{q_t\}_{t=0}^{t=T}, \{p_t\}_{t=0}^{t=T-1}\right)$ are complete and satisfy:

$$\forall K \in \mathbb{N},\ \exists t \geq K,\ q_t \in Q_s.$$

Definition

A state q is recurrence controllable if there exists a recurrence controller C such that $q \in \mathrm{dom}(C)$.

The set of recurrence controllable states is denoted $\text{rec-cont}(S, Q_s)$.

Recurrence synthesis

Synthesis through nested fixed point computation:

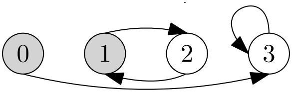
Recurrence synthesis

$R_0 = \text{r-cont}(T, Q_s)$
loop
$|\ R_{k+1} = \text{r-cont}(S, Q_s \cap \mathit{Pre}(R_k))$
until $R_{k+1} = R_k$
return $R^* = R_k$

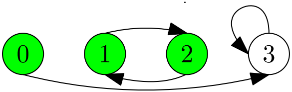
Termination by finiteness of Q.

Theorem

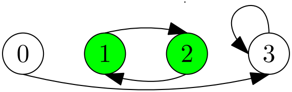
$R^* = \mathit{rec\text{-}cont}(S, Q_s)$.



$R_0 = \text{r-cont}(S, Q_s)$

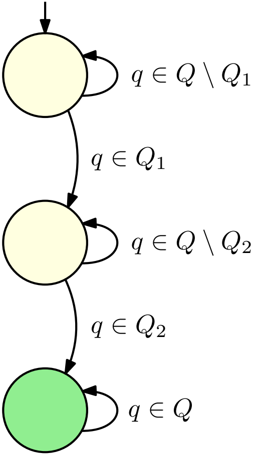


$R_1 = R^* = \text{rec-cont}(S, Q_s)$



Automata-based specifications

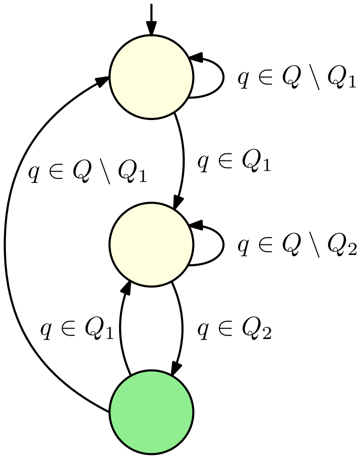
Example 1: go to Q_1, then go to Q_2



- Compute the product of system and specification automaton
- Solve a reachability problem in the product space: *Reach the green state.*

Automata-based specifications

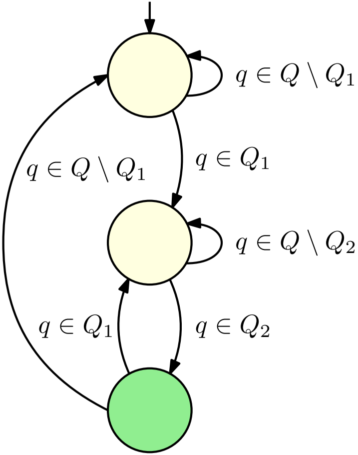
Example 2: go to Q_1, then go to Q_2; repeat this task infinitely often



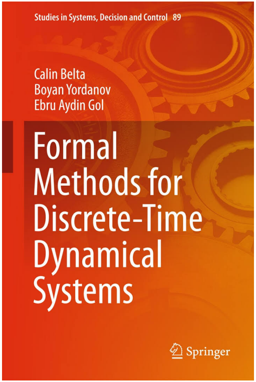
- Compute the product of system and specification automaton
- Solve a recurrence problem in the product space: *Visit the green state infinitely often.*

Wide range of possible specifications
$\rightarrow$ Linear Temporal Logic

Example 2: go to Q_1, then go to Q_2; repeat this task infinitely often



Further reading:



Outline of the lecture

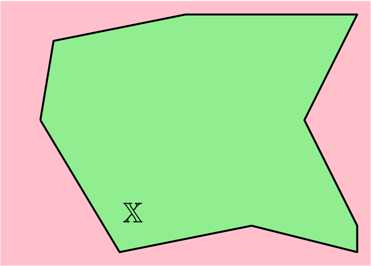
1. Fundamentals of symbolic control:
   - Discrete controller synthesis
     Safety, reachability and recurrence
   - Symbolic control of nonlinear systems
     System abstraction, controller concretization, robustness
2. Recent advances in symbolic control:
   - Symbolically-guided model predictive control
     High performance controllers with safety guarantees
   - Data-driven symbolic control
     Towards safe learning approaches for nonlinear systems

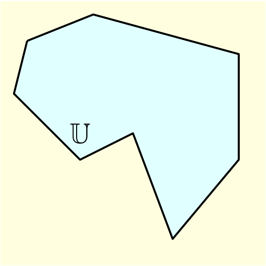
A control problem with safety constraints

We consider a nonlinear system subject to state/input constraints and bounded disturbances:

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

where $\mathbb{X} \subseteq \mathbb{R}^{n_x}$, $\mathbb{U} \subseteq \mathbb{R}^{n_u}$, $\mathbb{W} \subseteq \mathbb{R}^{n_w}$.





Objective: compute a symbolic model that can be used to synthesize controllers with formal guarantees.

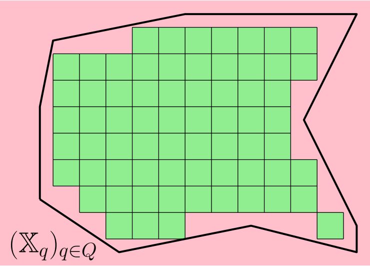
Abstraction: from continuous to discrete

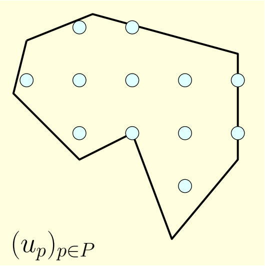
Let us consider:

- A finite "partition" $(\mathbb{X}_q)_{q\in Q}$ of $\mathbb{R}^{n_x}$ such that

$$Q = Q_{\mathbb{X}} \cup \{q_{\text{out}}\} \text{ and } \bigcup_{q\in Q_{\mathbb{X}}} \mathbb{X}_q \subseteq \mathbb{X};$$

- A finite sample $(u_p)_{p\in P}$ of $\mathbb{U}$.





Abstraction: from continuous to discrete

We consider a symbolic transition system $S = (Q, P, F)$:

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

where

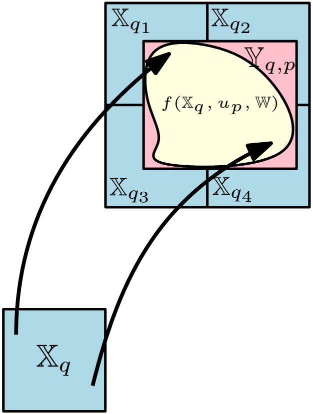
- Q and P are the finite sets of symbolic states and inputs;
- $F : Q \times P \rightrightarrows Q$ is the transition map defined by

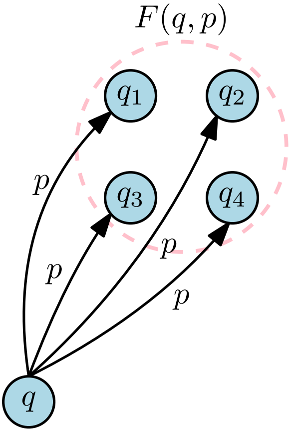
$$F(q,p) = \left\{ q^+ \in Q \,\middle|\, \mathbb{X}_{q^+} \cap \mathbb{Y}_{q,p} \neq \emptyset \right\}$$

where $\mathbb{Y}_{q,p} \subseteq \mathbb{R}^{n_x}$ is an over-approximation of the reachable set:

$$f(\mathbb{X}_q, u_p, \mathbb{W}) \subseteq \mathbb{Y}_{q,p},\ \forall q \in Q,\ \ p \in P.$$

Abstraction: from continuous to discrete





Reachability analysis

Assume $\mathbb{X}_q = [\underline{x}_q, \overline{x}_q]$, $\mathbb{W} = [\underline{w}, \overline{w}]$ and let

$$x_q^c = \frac{\underline{x}_q + \overline{x}_q}{2},\ \delta x_q = \frac{\overline{x}_q - \underline{x}_q}{2},\ w^c = \frac{\underline{w} + \overline{w}}{2},\ \delta w = \frac{\overline{w} - \underline{w}}{2}.$$

- If f is Lipschitz with respect to x and w:

$$\mathbb{Y}_{q,p} = \mathcal{B}\left(f(x_q^c, u_p, w^c), L_x\|\delta x_q\| + L_w\|\delta w\|\right)$$

- If f has uniformly bounded derivatives: $|\frac{\partial f}{\partial x}| \leq D_x$, $|\frac{\partial f}{\partial w}| \leq D_w$

$$\mathbb{Y}_{q,p} = \left[f(x_q^c, u_p, w^c) - \delta y_q, f(x_q^c, u_p, w^c) + \delta y_q\right]$$

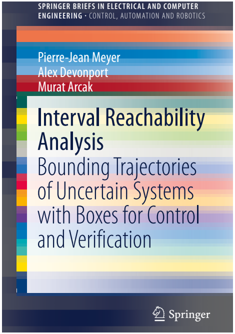
where $\delta y_q = D_x \delta x_q + D_w \delta w$.

Reachability analysis

Assume $\mathbb{X}_q = [\underline{x}_q, \overline{x}_q]$, $\mathbb{W} = [\underline{w}, \overline{w}]$.

- If f is monotone: $\frac{\partial f}{\partial x} \geq 0$, $\frac{\partial f}{\partial w} \geq 0$

$$\mathbb{Y}_{q,p} = \left[f(\underline{x}_q, u_p, \underline{w}), f(\overline{x}_q, u_p, \overline{w})\right]$$



Further reading

Controller concretization: from discrete to continuous

Theorem

Given a symbolic controller $C : Q \rightrightarrows P$, and the quantizer [a] $\theta : \mathbb{R}^n \rightrightarrows Q$, consider the closed-loop system

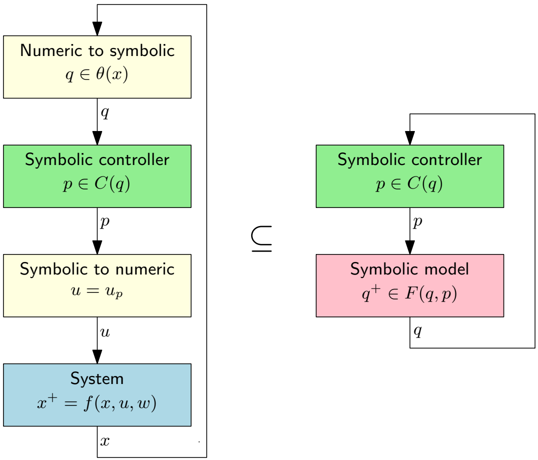
$$\begin{cases} x_{t+1} & = & f(x_t, u_{p_t}, w_t) \\ q_t & \in & \theta(x_t) \\ p_t & \in & C(q_t) \end{cases}$$

Then, $\left(\{q_t\}_{t=0}^{t=T}, \{p_t\}_{t=0}^{t=T-1}\right)$ is a trajectory of S_C.

[a] $q \in \theta(x)$ iff $x \in \mathbb{X}_q$

- Closed loop trajectories of the continuous system are included in those of the symbolic model.
- Extends to more general class of controllers (dynamic, with memory).

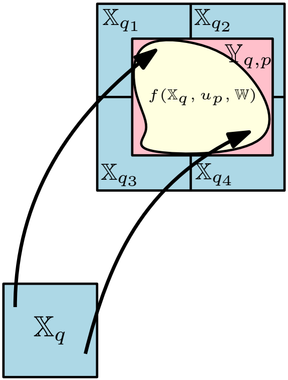
Controller concretization: from discrete to continuous

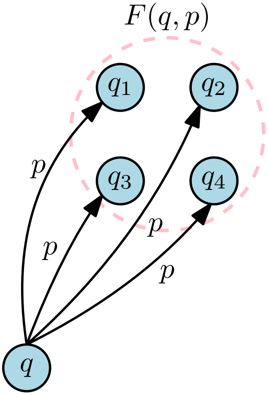


$\Longrightarrow$ We can use the symbolic model to synthesize a controller that provides formal guarantees for the original system.

Robustness margins

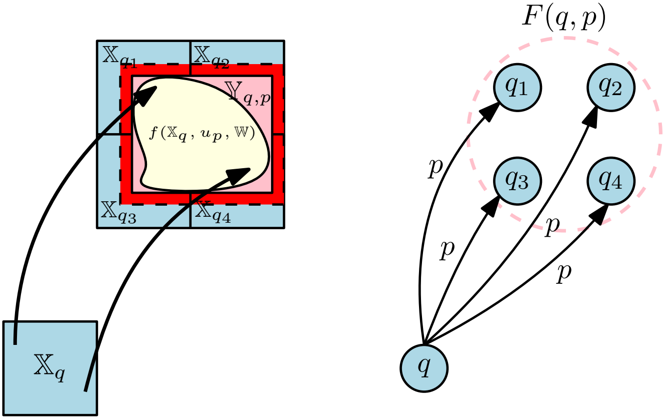
- Symbolic control makes it possible to deal with bounded disturbance





Robustness margins

- Symbolic control makes it possible to deal with bounded disturbance



- We get additional robustness for free !

Robustness margins

Theorem

Let us assume that $\mathbb{X}_q$ is a closed set, for all $q \in Q$ and let us consider the symbolic model S computed for

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

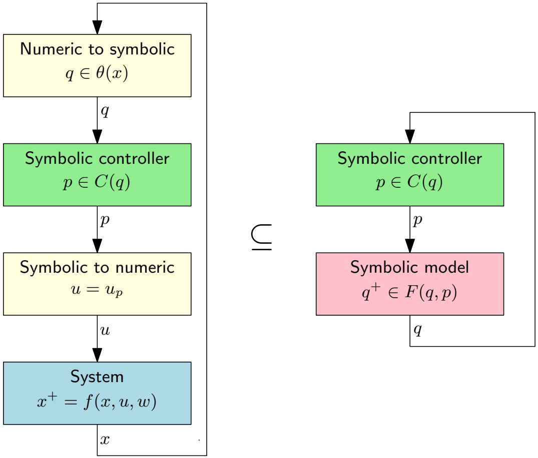
Then, there exists $\varepsilon > 0$ such that all previous results hold for the perturbed system

$$x_{t+1} = f(x_t, u_t, w_t) + w'_t$$

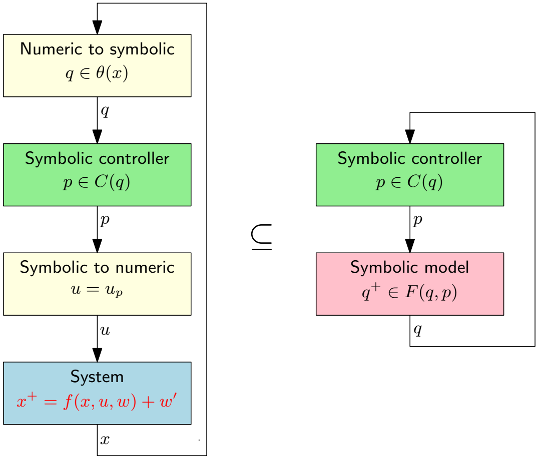
where $w'_t \in \mathcal{B}(0, \varepsilon)$.

Note that the precise value of ε can be effectively computed.

Robustness for free



Robustness for free



Example: safe navigation in complex environments

Consider a mobile robot modeled as a unicycle:

$$\begin{cases} x_1(t+1) & = x_1(t) + u_1(t)\cos(x_3(t)) \\ x_2(t+1) & = x_2(t) + u_1(t)\sin(x_3(t)) \\ x_3(t+1) & = x_3(t) + u_2(t) \end{cases}$$

and subject to state and input constraints:

$$\mathbb{X} = \left\{ x \in \mathbb{R}^3 \;\middle|\; \begin{array}{c} x_1^2 - x_2^2 \leq 4 \\ 4x_2^2 - x_1^2 \leq 16 \end{array} \right\},\ \mathbb{U} = [0.2, 2] \times [-1, 1].$$

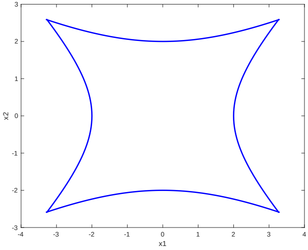
Let us remark that for all $t \in \mathbb{N}$, $u_1(t) \geq 0.2 \implies$ the robot cannot stop.

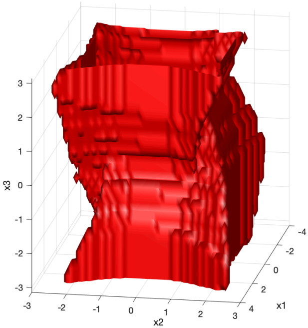
Safety controllable set

Working environment:

$$\mathbb{X} = \left\{ x \in \mathbb{R}^3 \;\middle|\; \begin{array}{c} x_1^2 - x_2^2 \leq 4 \\ 4x_2^2 - x_1^2 \leq 16 \end{array} \right\}$$

Non-convex, sharp corners.



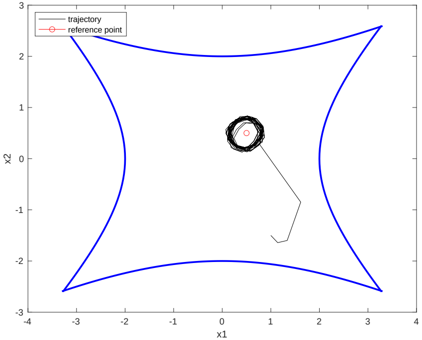


Safety controllable set computed using symbolic control techniques:

- 109200 symbolic states
- 40 symbolic inputs
- CPU time: $\sim$ 2 minutes

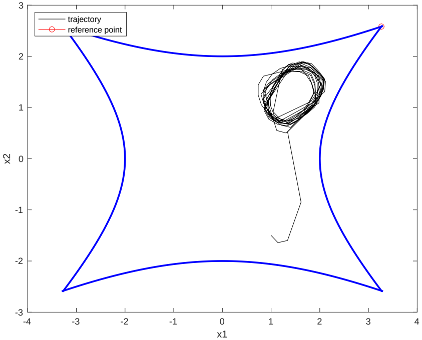
Example: safe navigation in complex environments

- Performance criteria: tracking a constant reference position $p_r \in \mathbb{R}^2$.
- Time horizon: $N = 20$.



Reference position in the interior:

$p_r = (0.5, 0.5)$



Reference position in the corner:

$p_r = (\sqrt{32/3}, \sqrt{20/3})$

Outline of the lecture

1. Fundamentals of symbolic control:
   - Discrete controller synthesis
     Safety, reachability and recurrence
   - Symbolic control of nonlinear systems
     System abstraction, controller concretization, robustness
2. Recent advances in symbolic control:
   - Symbolically-guided model predictive control
     High performance controllers with safety guarantees
   - Data-driven symbolic control
     Towards safe learning approaches for nonlinear systems

Nonlinear model predictive control

Consider a nonlinear system subject to state and input constraints:

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

We want to use a model predictive control scheme to enforce contraints while optimizing some performance criteria, i.e. $u_t = u_{0|t}$ with:

$$\min_{u_{0|t},\dots,u_{N-1|t}} \sum_{k=0}^{N-1} \ell(x_{k|t}, u_{k|t}) + L(x_{N|t})$$

$$\text{subject to } \begin{cases} x_{0|t} = x_t, & \\ x_{k+1|t} = f(x_{k|t}, u_{k|t}), & k = 0, \dots, N-1 \\ x_{k|t} \in \mathbb{X},\ \ u_{k|t} \in \mathbb{U}, & k = 0, \dots, N \end{cases}$$

Recursive feasibility

- For safety critical systems, one needs to guarantee that the optimization problem is feasible at all time.
- One classical solution is to append terminal constraints to the optimization problem:

$$\min_{u_{0|t},\ldots,u_{N-1|t}} \sum_{k=0}^{N-1} \ell(x_{k|t}, u_{k|t}) + L(x_{N|t})$$

$$\text{subject to } \begin{cases} x_{0|t} = x_t, & \\ x_{k+1|t} = f(x_{k|t}, u_{k|t}), & k = 0, \ldots, N-1 \\ x_{k|t} \in \mathbb{X},\ u_{k|t} \in \mathbb{U}, & k = 0, \ldots, N \\ x_{N|t} \in \mathbb{X}_I & \end{cases}$$

where $\mathbb{X}_I \subseteq \mathbb{X}$ is a (maximal) controlled invariant set.

Controlled invariant sets

- (Maximal) controlled invariant sets for nonlinear systems subject to non-convex contraints:
  - can be hard to compute,
  - may not admit simple representations.
- Controlled invariant sets computed using symbolic control are typically unions of many intervals

$$\mathbb{X}_I = \bigcup_{q \in Q_I} \mathbb{X}_q, \text{ where } Q_I = \text{s-cont}(S, Q_s)$$

$\Longrightarrow$ Not suitable for real-time optimization.

Time-varying terminal constraints

Let us consider the following MPC scheme:

$$\min_{u_t, u_{0|t}, \dots, u_{N-1|t}} \sum_{k=0}^{N-1} \ell(x_{k|t}, u_{k|t}) + L(x_{N|t})$$

$$\text{subject to } \begin{cases} x_{0|t} = x_t,\, u_t = u_{0|t} \\ x_{k+1|t} = f(x_{k|t}, u_{k|t}), & k = 0, \dots, N-1 \\ x_{k|t} \in \mathbb{X},\; u_{k|t} \in \mathbb{U}, & k = 0, \dots, N \\ x_{N|t} \in \mathbb{X}_t \end{cases}$$

where $\mathbb{X}_t \subseteq \mathbb{X}$ is a (simple) time-varying terminal constraint.

Objective: propose a design mechanism for time-varying terminal constraints guaranteeing recursive feasibility of the optimization problem.

Symbolically-guided mode predictive control

Let us consider:

- a controlled invariant set $\mathbb{X}_I \subseteq \mathbb{X}$
- an invariance controller κ, i.e. $\forall x \in \mathbb{X}_I,\ f(x, \kappa(x)) \in \mathbb{X}_I$
- an interval-valued map T such that for all $x \in \mathbb{X}_I$

$$f(x, \kappa(x)) \in T(x) \subseteq \mathbb{X}_I$$

Theorem

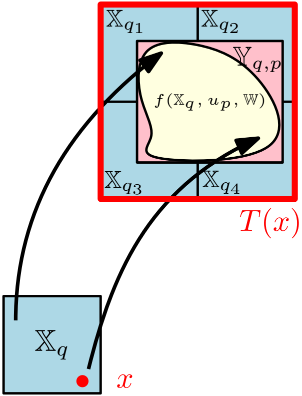
Consider the following sequence of terminal constraints given by

$$\mathbb{X}_{t+1} = T(x_{N|t}), \ \textit{for all } t \in \mathbb{N}$$

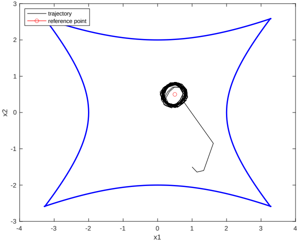
Then, the MPC optimization problem is recursively feasible.

$\mathbb{X}_I$, κ and T can be computed using symbolic control:

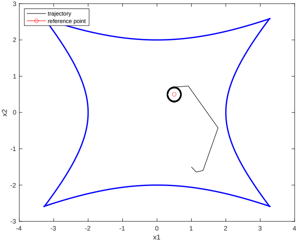
- $\mathbb{X}_I = \bigcup_{q \in Q_I} \mathbb{X}_q$,
  where $Q_I = \text{s-cont}(S, Q_s)$
- $\kappa(x) = u_p$,
  where $p \in C(\theta(x))$
- $T(x) = \theta^{-1}(F(\theta(x), p))$



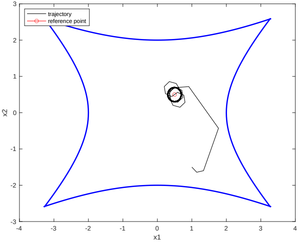
- Reference position inside the environment: $p_r = (0.5, 0.5)$.
- Prediction horizon: 20.



Optimal symbolic control

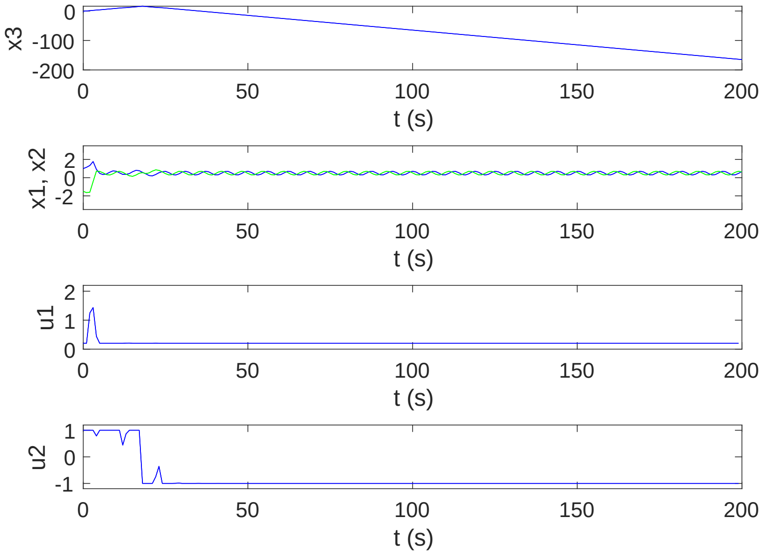


Model predictive control

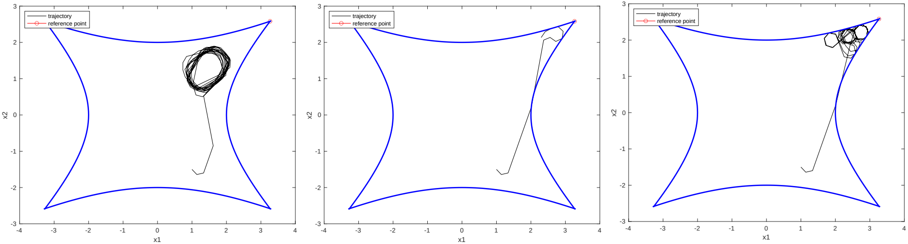


Symbolically-guided model predictive control

Case 1: focus on SgMPC



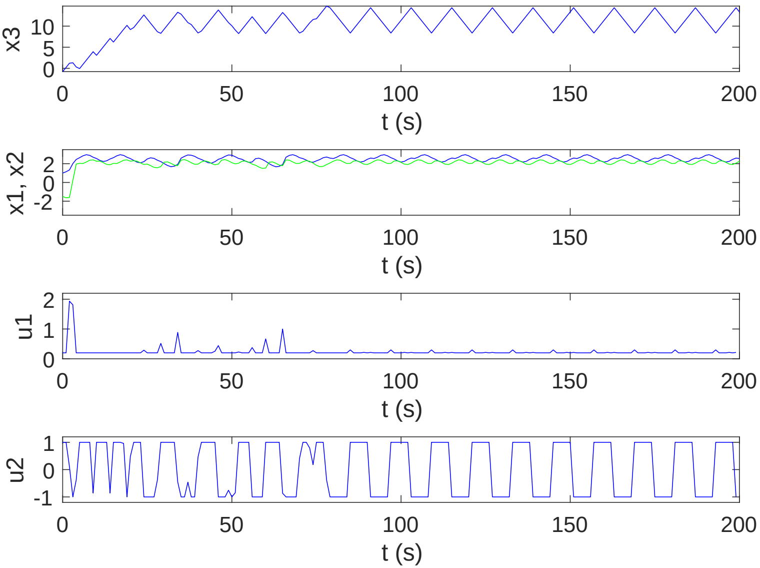
- Reference position in the corner: $p_r = (\sqrt{32/3}, \sqrt{20/3})$.
- Prediction horizon: 20.



Optimal symbolic control | Model predictive control | Symbolically-guided model predictive control

MPC stopped at $t = 13$ because optimization problem becomes infeasible.

Case 2: focus on SgMPC



Outline of the lecture

1. Fundamentals of symbolic control:
   - Discrete controller synthesis
     Safety, reachability and recurrence
   - Symbolic control of nonlinear systems
     System abstraction, controller concretization, robustness
2. Recent advances in symbolic control:
   - Symbolically-guided model predictive control
     High performance controllers with safety guarantees
   - Data-driven symbolic control
     Towards safe learning approaches for nonlinear systems

Data-driven abstraction

We consider an unknown nonlinear system subject to state and input constraints:

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

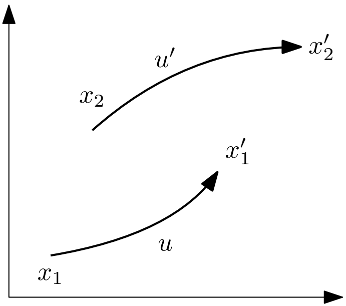
We are given a finite data set

$$\mathcal{D} = \left\{(x_k, u_k, x_k^+)\,|\,k \in \mathbb{K}\right\},\ \text{where } x_k^+ = f(x_k, u_k).$$

- Objective: compute directly from $\mathcal{D}$ a symbolic model providing formal guarantees.
- As for the model-based approach, the question reduces to compute over-approximations of the reachable sets from data:

$$f(\mathbb{X}_q, u_p) \subseteq \mathbb{Y}_{q,p},\ \forall q \in Q,\ p \in P.$$

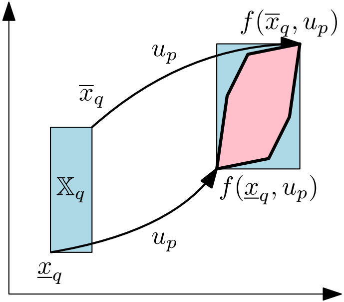
The case of monotone systems



$$x_1 \preceq x_2, u \preceq u' \Rightarrow x'_1 \preceq x'_2$$

- 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...

The case of monotone systems



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

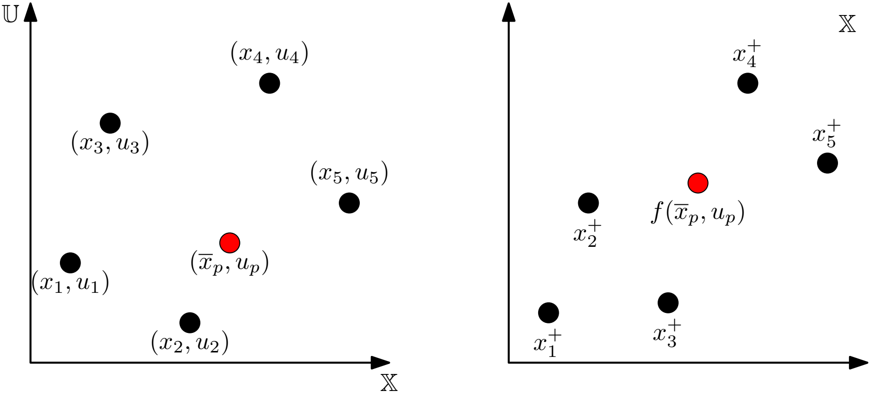
- Characterization:
  $$\frac{\partial f_i}{\partial x_j} \geq 0, \ \frac{\partial f_i}{\partial u_k} \geq 0, \ \forall i, j, k$$
- Applications: vehicles, energy, biology...

Then, assuming $\mathbb{X}_q = [\underline{x}_q, \overline{x}_q]$, it holds

$$f(\mathbb{X}_q, u_p) \subseteq [f(\underline{x}_q, u_p), f(\overline{x}_q, u_p)].$$

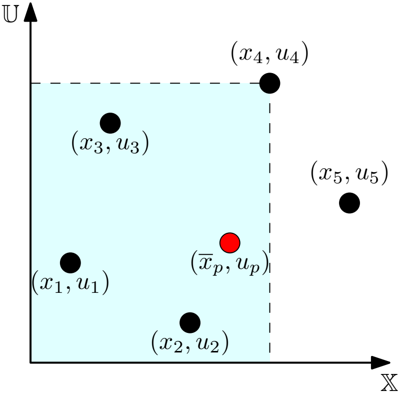
The case of monotone systems

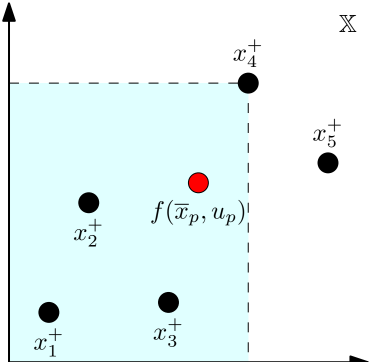
Computing an upper-bound of $f(\overline{x}_q, u_p)$ from data:



The case of monotone systems

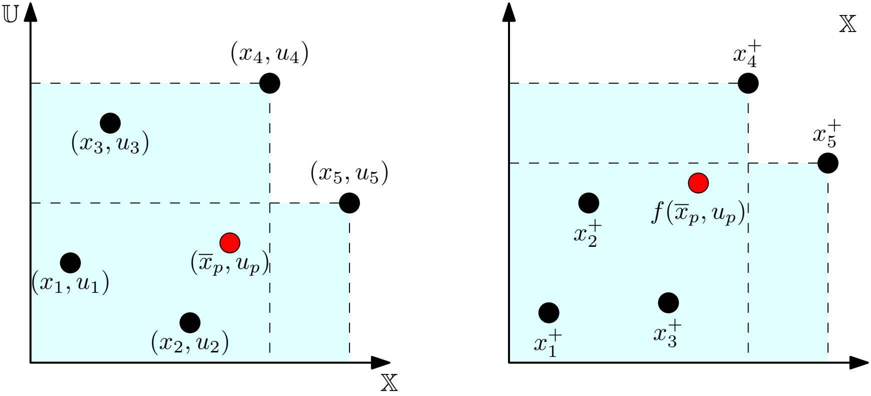
Computing an upper-bound of $f(\overline{x}_q, u_p)$ from data:





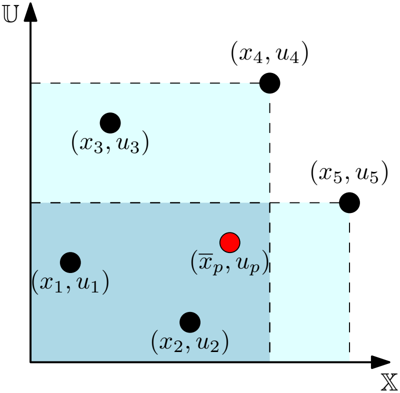
The case of monotone systems

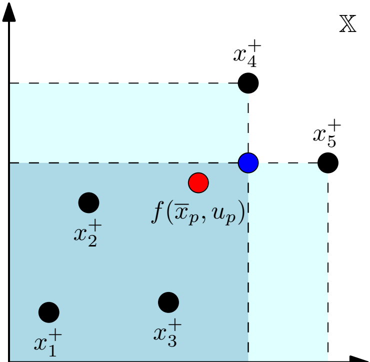
Computing an upper-bound of $f(\overline{x}_q, u_p)$ from data:



The case of monotone systems

Computing an upper-bound of $f(\overline{x}_q, u_p)$ from data:





The case of monotone systems

Theorem

Consider the following set of indices:

$$\mathbb{K}^+(\overline{x}_q, u_p) = \left\{k \in \mathbb{K} \,\middle|\, \overline{x}_q \preceq x_k \text{ and } u_p \preceq u_k \right\}$$

$$\mathbb{K}^-(\underline{x}_q, u_p) = \left\{k \in \mathbb{K} \,\middle|\, x_k \preceq \underline{x}_q \text{ and } u_k \preceq u_p \right\}$$

Then, $f(\mathbb{X}_q, u_p) \subseteq \mathbb{Y}_{q,p}$ *where*

$$\mathbb{Y}_{q,p} = \left(\bigcap_{k \in \mathbb{K}^+(\overline{x}_q, u_p)} \left\{x^+ \,\middle|\, x^+ \preceq x_k^+ \right\} \right) \cap \left(\bigcap_{k \in \mathbb{K}^-(\underline{x}_q, u_p)} \left\{x^+ \,\middle|\, x_k^+ \preceq x^+ \right\} \right).$$

The case of general systems

Assume that we know lower bounds on the partial derivatives of the unknown function f:

$$\frac{\partial f_i}{\partial x_j} \geq a_{ij},\ \frac{\partial f_i}{\partial u_k} \geq b_{ik},\ \forall i, j, k.$$

Consider the matrix A^- and B^- be given by

$$a_{ij}^- = \min(a_{ij}, 0),\ b_{ij}^- = \min(b_{ij}, 0).$$

Then,

$$f(x,u) = A^- x + B^- u + g(x,u)$$

where $g(x,u) = f(x,u) - A^- x - B^- u$ is a monotone function.

The case of general systems

Theorem

We have that $f(\mathbb{X}_q, u_p) \subseteq \mathbb{Y}_{q,p}$ *with*

$$\mathbb{Y}_{q,p} = \left[A^- \overline{x}_q + B^- u_p, A^- \underline{x}_q + B^- u_p\right] + \mathbb{Y}^g_{q,p}$$

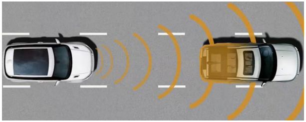
where the over-approximation $\mathbb{Y}^g_{q,p}$ *of the monotone function* g *can be computed from the data set* $\mathcal{D}$.

- Using an efficient implementation, a symbolic model can be computed from data in $\mathcal{O}\left(|\mathcal{D}| \times \log(|\mathbb{Q}| \times |\mathbb{P}|) + |\mathbb{Q}| \times |\mathbb{P}|\right)$.
- If we collect new data, the symbolic model can be updated without restarting from scratch.
- The approach can be also extended to systems with bounded disturbances and/or with partially known dynamics.

Example: adaptive cruise control

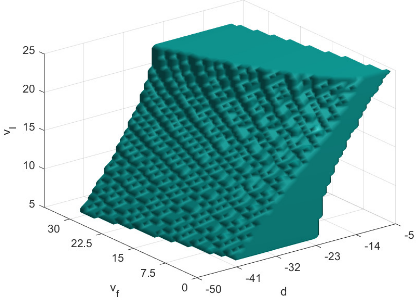
Consider two vehicles (leader and follower):

- Relative distance d;
- Follower velocity v_1;
- Leader velocity v_2;
- Unknown monotone dynamics



Data-driven symbolic model computed from 10^6 data points:

- 125000 symbolic states
- 50 symbolic inputs
- CPU time: < 1s



Conclusion

- Symbolic control is a powerful computational technique for safety-critical control of nonlinear systems with state and input constraints, and robustness guarantees.
- Performances of symbolic controllers are limited but can be drastically improved by combining with MPC, while retaining safety guarantees.
  $\Longrightarrow$ Symbolically-guided Model Predictive Control (SgMPC).
- Symbolic models can be computed from data, opening the way to safe learning-based control of nonlinear systems.
- Current and future work:
  - SgMPC for complex navigation problems (e.g. temporal logics, etc.).
  - Combine SgMPC and data-driven abstraction to design safe learning-based MPC for nonlinear systems.

Recommended reading

Calin Belta, Boyan Yordanov, and Ebru Aydin Gol.
Formal methods for discrete-time dynamical systems, volume 89.
Springer, 2017.

Gunther Reissig, Alexander Weber, and Matthias Rungger.
Feedback refinement relations for the synthesis of symbolic controllers.
IEEE Transactions on Automatic Control, 62(4):1781–1796, 2017.

Pierre-Jean Meyer, Alex Devonport, and Murat Arcak.
Interval reachability analysis: Bounding trajectories of uncertain systems with boxes for control and verification.
Springer Nature, 2021.

Zakeye Azaki, Antoine Girard, and Sorin Olaru.
Predictive and symbolic control: Performance and safety for non-linear systems.
In *IFAC Workshop on Control Applications of Optimization*, 2022.

Anas Makdesi, Antoine Girard, and Laurent Fribourg.
Efficient data-driven abstraction of monotone systems with disturbances.
In *IFAC Conference on Analysis and Design of Hybrid Systems*, 2021.

Anas Makdesi, Antoine Girard, and Laurent Fribourg.
Safe learning-based model predictive control using the compatible models approach.
European Journal of Control, page 100849, 2023.