Synthèse de contrôle garanti pour des systèmes dynamiques spatio-temporels à commutation

Projets Farman SWITCHDESIGN & SWITCHDESIGN2

10 Ans de l'Institut Farman, ENS Paris-Saclay

Adrien Le Coënt [1], Florian De Vuyst [1], Ludovic Chamoin [2], Laurent Fribourg [3]

September 27, 2017

[1] CMLA Centre de Mathématiques et de Leurs Applications
[2] LMT Laboratoire de Mécanique et Technologie
[3] LSV Laboratoire de Spécification et Vérification

Context: control systems

Outline

Switched Systems

A continuous-time switched system

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

is a family of continuous-time dynamical systems with a rule σ that determines at each time which one is active

Switched Systems

A continuous-time switched system

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

is a family of continuous-time dynamical systems with a rule σ that determines at each time which one is active

- state $x \in \mathbb{R}^n$

Switched Systems

A continuous-time switched system

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

is a family of continuous-time dynamical systems with a rule σ that determines at each time which one is active

- state $x \in \mathbb{R}^n$
- bounded perturbation $d \in \mathbb{R}^m$

Switched Systems

A continuous-time switched system

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

is a family of continuous-time dynamical systems with a rule σ that determines at each time which one is active

- state $x \in \mathbb{R}^n$
- bounded perturbation $d \in \mathbb{R}^m$
- switching signal $\sigma(\cdot) : \mathbb{R}^+ \longrightarrow U$ (piecewise constant)

Switched Systems

A continuous-time switched system

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

is a family of continuous-time dynamical systems with a rule σ that determines at each time which one is active

- state $x \in \mathbb{R}^n$
- bounded perturbation $d \in \mathbb{R}^m$
- switching signal $\sigma(\cdot) : \mathbb{R}^+ \longrightarrow U$ (piecewise constant)
- $U = \{1, \dots, N\}$ finite set of modes, associated with the dynamics

$$\dot{x}(t) = f_u(x(t), d(t)), \ u \in U$$

Switched Systems

A continuous-time switched system

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

is a family of continuous-time dynamical systems with a rule σ that determines at each time which one is active

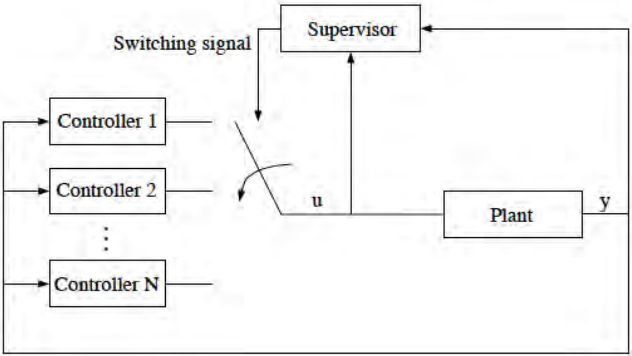
- state $x \in \mathbb{R}^n$
- bounded perturbation $d \in \mathbb{R}^m$
- switching signal $\sigma(\cdot) : \mathbb{R}^+ \longrightarrow U$ (piecewise constant)
- $U = \{1, \dots, N\}$ finite set of modes, associated with the dynamics

$$\dot{x}(t) = f_u(x(t), d(t)), \ u \in U$$

We focus on sampled switched systems: switching instants occur periodically every τ, i.e. σ is constant on $[i\tau, (i+1)\tau)$

Examples of switched systems

Controlled Switched Systems: Schematic View



Control Synthesis Problem

We consider the state-dependent control problem of synthesizing σ:

Control Synthesis Problem

We consider the state-dependent control problem of synthesizing σ:

At each sampling time $k\tau$, find the appropriate switched mode $u \in U$ according to the current value of x, in order to achieve some objectives:

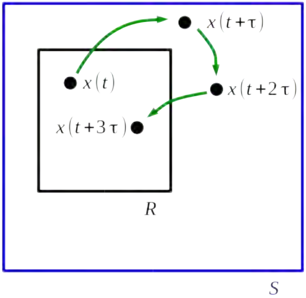
Control Synthesis Problem

We consider the state-dependent control problem of synthesizing σ:

At each sampling time $k\tau$, find the appropriate switched mode $u \in U$ according to the current value of x, in order to achieve some objectives:

Given two sets R, S:

- (R, S)-stability: $x(t)$ returns in R infinitely often, at some multiples of sampling period τ, and always stays in S



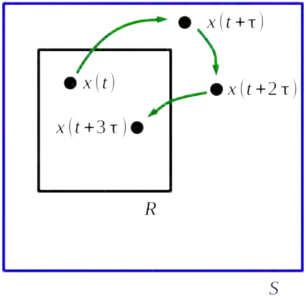
Control Synthesis Problem

We consider the state-dependent control problem of synthesizing σ:

At each sampling time $k\tau$, find the appropriate switched mode $u \in U$ according to the current value of x, in order to achieve some objectives:

Given two sets R, S:

- (R, S)-stability: $x(t)$ returns in R infinitely often, at some multiples of sampling period τ, and always stays in S



NB: classic stabilization impossible here (no common equilibrium pt)
$\rightsquigarrow$ practical stability

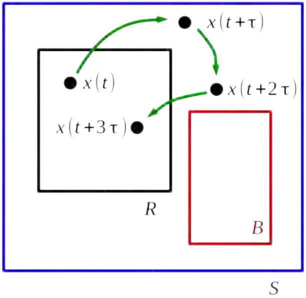
Control Synthesis Problem

We consider the state-dependent control problem of synthesizing σ:

At each sampling time $k\tau$, find the appropriate switched mode $u \in U$ according to the current value of x, in order to achieve some objectives:

Given three sets R, B, S:

- (R, B, S)-avoidance: $x(t)$ returns in R infinitely often, at some multiples of sampling period τ, and always stays in $S \backslash B$



NB: classic stabilization impossible here (no common equilibrium pt)
$\rightsquigarrow$ practical stability

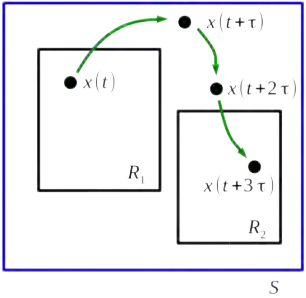
Control Synthesis Problem

We consider the state-dependent control problem of synthesizing σ:

At each sampling time $k\tau$, find the appropriate switched mode $u \in U$ according to the current value of x, in order to achieve some objectives:

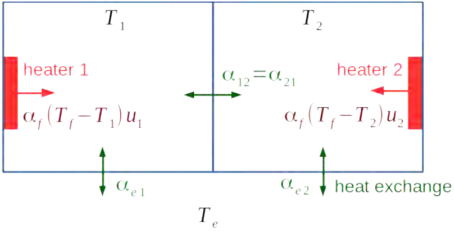
Given three sets R_1, R_2, S:

- (R_1, R_2, S)-reachability: $x(t)$ starting in R_1 reaches R_2 after some multiples of sampling period τ, and always stays in S



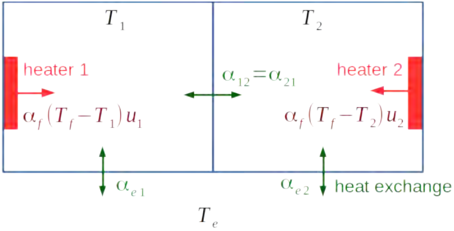
NB: classic stabilization impossible here (no common equilibrium pt)
$\rightsquigarrow$ practical stability

Example: Two-room apartment



$$\begin{pmatrix}\dot{T}_1\\ \dot{T}_2\end{pmatrix} = \begin{pmatrix}-\alpha_{21}-\alpha_{e1}-\alpha_f u_1 & \alpha_{21}\\ \alpha_{12} & -\alpha_{12}-\alpha_{e2}-\alpha_f u_2\end{pmatrix}\begin{pmatrix}T_1\\ T_2\end{pmatrix}+\begin{pmatrix}\alpha_{e1}T_e+\alpha_f T_f u_1\\ \alpha_{e2}T_e+\alpha_f T_f u_2\end{pmatrix}.$$

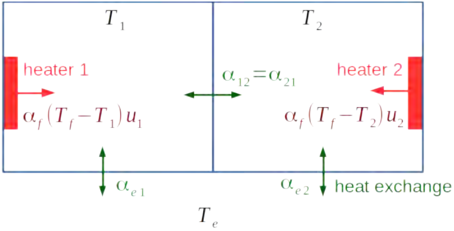
Example: Two-room apartment



$$\begin{pmatrix} \dot{T}_1 \\ \dot{T}_2 \end{pmatrix} = \begin{pmatrix} -\alpha_{21} - \alpha_{e1} - \alpha_f u_1 & \alpha_{21} \\ \alpha_{12} & -\alpha_{12} - \alpha_{e2} - \alpha_f u_2 \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} + \begin{pmatrix} \alpha_{e1} T_e + \alpha_f T_f u_1 \\ \alpha_{e2} T_e + \alpha_f T_f u_2 \end{pmatrix}.$$

- Modes: $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$; sampling period τ

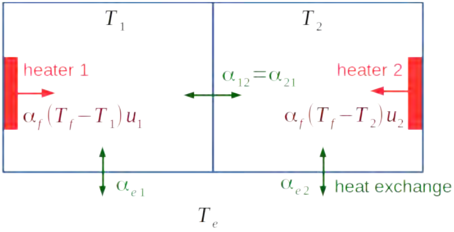
Example: Two-room apartment



$$T_1(t+\tau) = f_1(T_1(t), T_2(t), u_1)$$
$$T_2(t+\tau) = f_2(T_1(t), T_2(t), u_2)$$

- Modes: $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$; sampling period τ

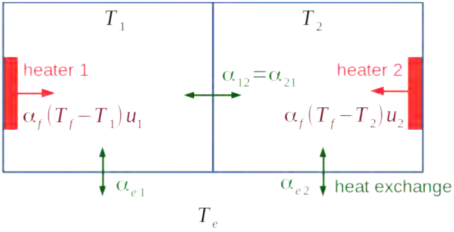
Example: Two-room apartment



$$T_1(t+\tau) = f_1(T_1(t), T_2(t), u_1)$$
$$T_2(t+\tau) = f_2(T_1(t), T_2(t), u_2)$$

- Modes: $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$; sampling period τ
- A pattern π is a finite sequence of modes, e.g. $\left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix}\right)$

Example: Two-room apartment



$$T_1(t+\tau) = f_1(T_1(t), T_2(t), u_1)$$
$$T_2(t+\tau) = f_2(T_1(t), T_2(t), u_2)$$

- Modes: $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$; sampling period τ
- A pattern π is a finite sequence of modes, e.g. $\left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix}\right)$
- A state dependent control consists in selecting at each τ a mode (or a pattern) according to the current value of the state.

(R, S)-stability property for the two-room apartment

Input:

- R, S
- an integer K (maximal length of patterns)

Output: controlled covering of R (each covering set is coupled with a pattern)

Guaranteed properties: (R, S)-stability

(R, S)-stability property for the two-room apartment

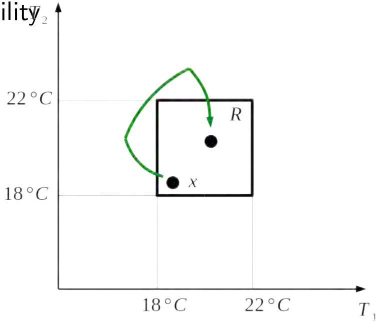
Input:

- R, S
- an integer K (maximal length of patterns)

Output: controlled covering of R (each covering set is coupled with a pattern)

Guaranteed properties: (R, S)-stability

- Recurrence in R: after some ($\leq K$) steps of time, the temperature returns in R



(R, S)-stability property for the two-room apartment

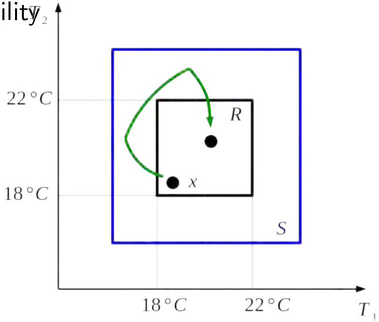
Input:

- R, S
- an integer K (maximal length of patterns)

Output: controlled covering of R (each covering set is coupled with a pattern)

Guaranteed properties: (R, S)-stability

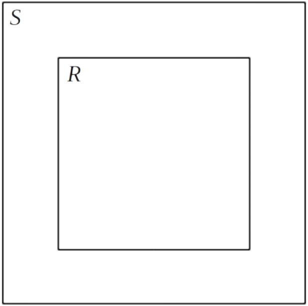
- Recurrence in R: after some ($\leq K$) steps of time, the temperature returns in R
- Safety in S: $x(t)$ always stays in S.



Control tiling procedure

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

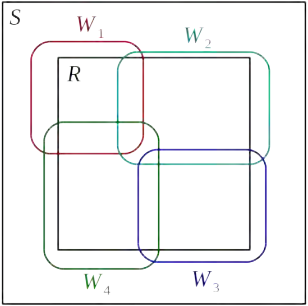
Goal: from any $x \in R$, return in R while always staying in S.



Control tiling procedure

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

Goal: from any $x \in R$, return in R while always staying in S.



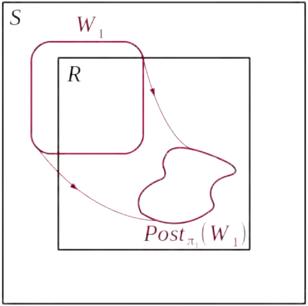
Basic idea:

- Generate a covering of R

Control tiling procedure

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

Goal: from any $x \in R$, return in R while always staying in S.



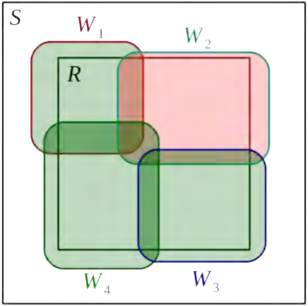
Basic idea:

- Generate a covering of R
- Look for patterns (input sequences) mapping the tiles into R while always staying in S

Control tiling procedure

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

Goal: from any $x \in R$, return in R while always staying in S.



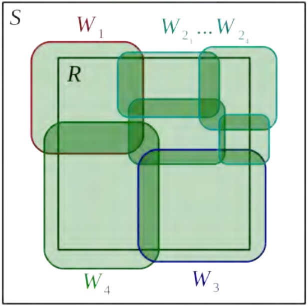
Basic idea:

- Generate a covering of R
- Look for patterns (input sequences) mapping the tiles into R while always staying in S
- If it fails,

Control tiling procedure

$$\dot{x}(t) = f_{\sigma(t)}(x(t), d(t))$$

Goal: from any $x \in R$, return in R while always staying in S.



Basic idea:

- Generate a covering of R
- Look for patterns (input sequences) mapping the tiles into R while always staying in S
- If it fails, generate another covering.

Limits

- Requires the computation of the reachable set
  - unknown in general
  - can be approximated using numerical schemes and/or strong hypotheses
- High computational complexity (curse of dimensionality):
  - m covering sets, patterns of length K, N switched modes $\Rightarrow$ cost in $O(mN^K)$
  - using a bisection heuristics of depth D in dimension n $\Rightarrow$ cost in $O(2^{nD}N^K)$

We propose:

- Handling nonlinear dynamics without strong hypotheses with guaranteed numerical schemes
- Handling higher dimensions using compositionality
- Synthesizing controllers for PDEs using Model Order Reduction

Outline

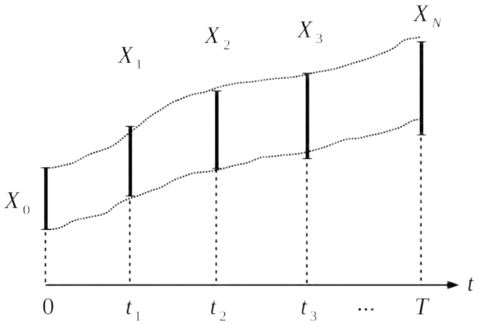
Outline

Validated simulation

DynIBEX [Chapoutot, Alexandre dit Sandretto, 2016]

Runge-Kutta numerical scheme:

- Computation of a sequence of approximations (t_n, X_n) of the solution $X(t; X_0)$
- X_i computed with the previous step: $X_i = h(t_{i-1}, X_{i-1})$



Validated simulation

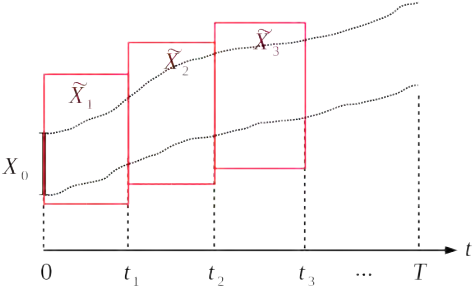
DynIBEX [Chapoutot, Alexandre dit Sandretto, 2016]

Runge-Kutta numerical scheme:

- Computation of a sequence of approximations (t_n, X_n) of the solution $X(t; X_0)$
- X_i computed with the previous step: $X_i = h(t_{i-1}, X_{i-1})$

Making it guaranteed:

- Enclose solutions (using Picard-Linedelöf operator and Banach's theorem) on $[t_{n-1}, t_n]$



Validated simulation

DynIBEX [Chapoutot, Alexandre dit Sandretto, 2016]

Runge-Kutta numerical scheme:

- Computation of a sequence of approximations (t_n, X_n) of the solution $X(t; X_0)$
- X_i computed with the previous step: $X_i = h(t_{i-1}, X_{i-1})$

Making it guaranteed:

- Enclose solutions (using Picard-Linedelöf operator and Banach's theorem) on $[t_{n-1}, t_n]$
- Tighten the error $\|x_n - x(t_n; x_{n-1})\|$

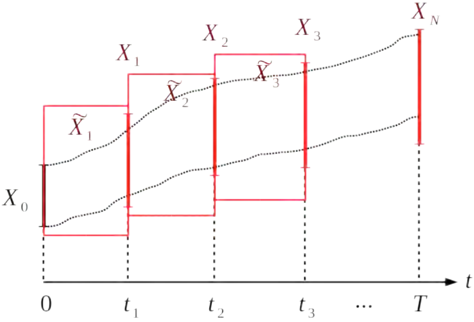
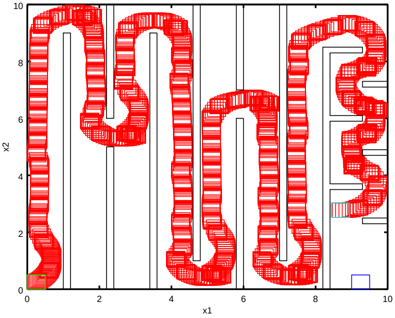
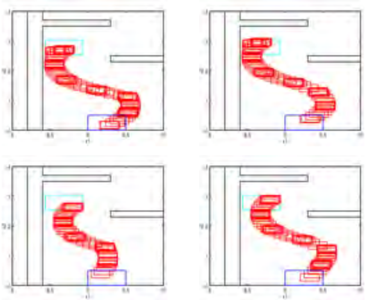


Illustration: a path planning problem
[Aström, Murray, 2010]

$$\dot{x} = v_0 \frac{\cos(\alpha+\theta)}{\cos(\alpha)}$$
$$\dot{y} = v_0 \frac{\sin(\alpha+\theta)}{\cos(\alpha)}$$
$$\dot{\theta} = \frac{v_0}{b} \tan(\delta)$$





Renewing the Euler scheme with the OSL property

$(H0)$ (Lipschitz): for all $j \in U$, there exists a constant $L_j > 0$ such that:

$$\|f_j(y) - f_j(x)\| \leq L_j \|y - x\| \quad \forall x, y \in S.$$

$(H1)$ (One-sided Lipschitz/Strong monotony): for all $j \in U$, there exists a constant $\lambda_j \in \mathbb{R}$ such that

$$\langle f_j(y) - f_j(x), y - x \rangle \leq \lambda_j \|y - x\|^2 \quad \forall x, y \in T,$$

Let us define the constants: C_j for all $j \in U$:

$$C_j = \sup_{x \in S} L_j \|f_j(x)\| \quad \text{for all} \quad j \in U.$$

NB: constants computed by constrained optimization.

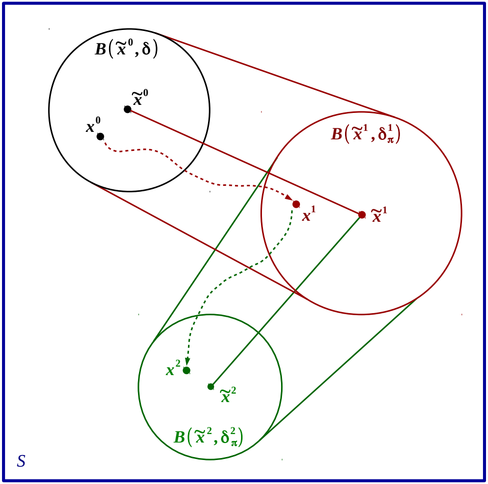
Main result

Theorem

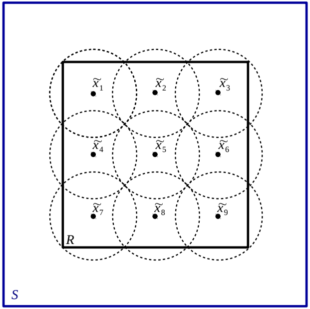
Given a sampled switched system satisfying (H0-H1), consider a point $\tilde{x}^0$ *and a positive real* δ. *We have, for all* $x^0 \in B(\tilde{x}^0, \delta)$, $t \in [0, \tau]$ *and* $j \in U$:
$\phi_j(t; x^0) \in B(\tilde{\phi}_j(t; \tilde{x}^0), \delta_j(t))$.
with

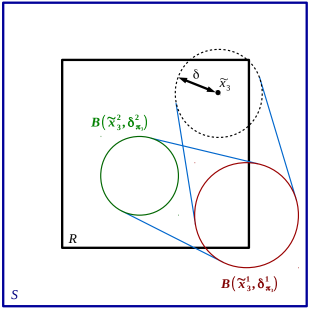
- *if* $\lambda_j < 0$: $\delta_j(t) = \left(\delta^2 e^{\lambda_j t} + \frac{C_j^2}{\lambda_j^2}\left(t^2 + \frac{2t}{\lambda_j} + \frac{2}{\lambda_j^2}\left(1 - e^{\lambda_j t}\right)\right)\right)^{\frac{1}{2}}$
- *if* $\lambda_j = 0$: $\delta_j(t) = \left(\delta^2 e^t + C_j^2(-t^2 - 2t + 2(e^t - 1))\right)^{\frac{1}{2}}$
- *if* $\lambda_j > 0$: $\delta_j(t) = \left(\delta^2 e^{3\lambda_j t} + \frac{C_j^2}{3\lambda_j^2}\left(-t^2 - \frac{2t}{3\lambda_j} + \frac{2}{9\lambda_j^2}\left(e^{3\lambda_j t} - 1\right)\right)\right)^{\frac{1}{2}}$

Application to guaranteed integration

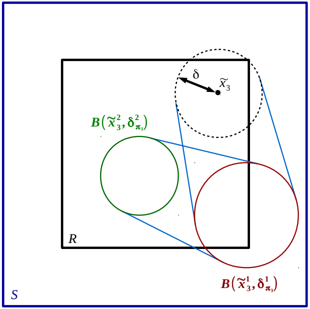


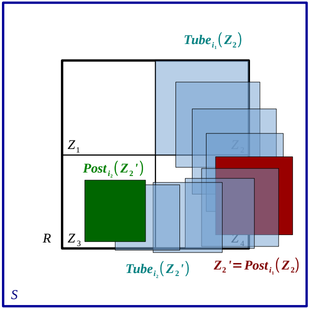
Control synthesis





Validated simulation vs Euler





Building ventilation

[Meyer, Nazarpour, Girard, Witrant, 2014]

Dynamics of a four-room apartment:

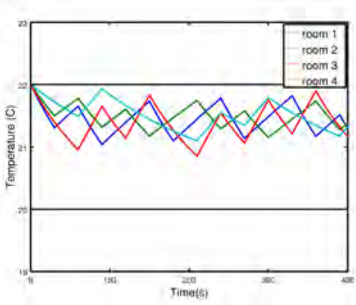
$$\frac{dT_i}{dt} = \sum_{j \in \mathcal{N}^*} a_{ij}(T_j - T_i) + \delta_{s_i} b_i (T_{s_i}^4 - T_i^4) + c_i \max\left(0, \frac{V_i - V_i^*}{\bar{V}_i - V_i^*}\right)(T_u - T_i).$$

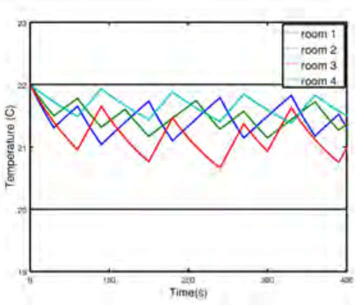
$\mathcal{N}^* = \{1, 2, 3, 4, u, o, c\}$
Control inputs: V_1 and V_4 can take the values 0V or 3.5V, and V_2 and V_3 can take the values 0V or 3V
$\Rightarrow$ 16 switching modes

Building ventilation

	Euler	DynIBEX
R S	$[20, 22]^4$ $[19, 23]^4$	
τ	30	
Complete control	Yes	Yes
$\max_{j=1,\ldots,16} \lambda_j$ $\max_{j=1,\ldots,16} C_j$	-6.30×10^{-3} 4.18×10^{-6}	
Number of balls/tiles Pattern length	4096 1	252 1
CPU time	63 seconds	249 seconds

Building ventilation





Outline

1 Guaranteed control of switched systems

2 The reachable set computation

3 **Distributed synthesis of controllers**
- Distributed synthesis using zonotopes
- Distributed synthesis using Euler's method

4 Control of partial differential equations with model order reduction

Distributed control synthesis

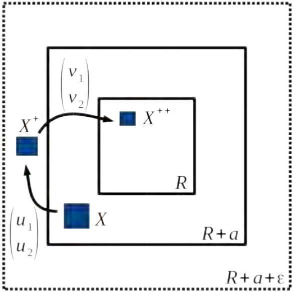
$$x_1(t+1) = f_1(x_1(t), x_2(t), u_1)$$
$$x_2(t+1) = f_2(x_1(t), x_2(t), u_2)$$

Target zone: $R = R_1 \times R_2$

Distributed control synthesis

$$x_1(t+1) = f_1(x_1(t), x_2(t), u_1)$$
$$x_2(t+1) = f_2(x_1(t), x_2(t), u_2)$$

Target zone: $R = R_1 \times R_2$

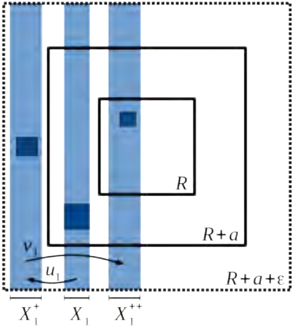


- $X \subset R + a$
- $X^+ = f(X, \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}) \subset R + a + \varepsilon$
- $X^{++} = f(X^+, \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}) \subset R$
- Pattern $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}, \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ depends on X

Distributed control synthesis

$$x_1(t+1) = f_1(x_1(t), x_2(t), u_1)$$
$$x_2(t+1) = f_2(x_1(t), x_2(t), u_2)$$

Target zone: $R = R_1 \times R_2$



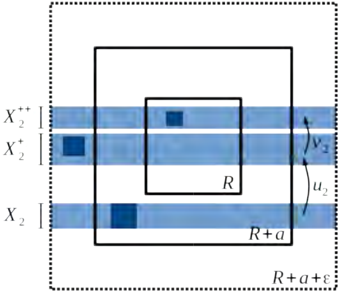
- $X_1 \subset R_1 + a$
- $X_1^+ = f_1(X_1, R_2 + a, u_1) \subset R_1 + a + \varepsilon$
- $X_1^{++} = f_1(X_1^+, R_2 + a + \varepsilon, v_1) \subset R_1$

- Pattern $u_1 \cdot v_1$ depends only on X_1

Distributed control synthesis

$$x_1(t+1) = f_1(x_1(t), x_2(t), u_1)$$
$$x_2(t+1) = f_2(x_1(t), x_2(t), u_2)$$

Target zone: $R = R_1 \times R_2$



- $X_2 \subset R_2 + a$
- $X_2^+ = f_2(R_1 + a, X_2, u_2) \in R_2 + a + \varepsilon$
- $X_2^{++} = f_2(R_1 + a + \varepsilon, X_2^+, v_2) \in R_2$

- Pattern $u_2 \cdot v_2$ depends only on X_2

Seluxit case study

Kim G. Larsen, Marius Mikučionis, Marco Muniz, Jiri Srba, Jakob H. Taankvist. *Online and Compositional Learning of Controllers with Application to Floor Heating*. Tools and Algorithms for Construction and Analysis of Systems 2016.

Seluxit case study

Kim G. Larsen, Marius Mikučionis, Marco Muniz, Jiri Srba, Jakob H. Taankvist. *Online and Compositional Learning of Controllers with Application to Floor Heating*. Tools and Algorithms for Construction and Analysis of Systems 2016.

System dynamics:

$$\frac{d}{dt}T_i(t) = \sum_{j=1}^{n} A_{i,j}^{d}(T_j(t) - T_i(t)) + B_i(T_{env}(t) - T_i(t)) + H_{i,j}.v_j$$

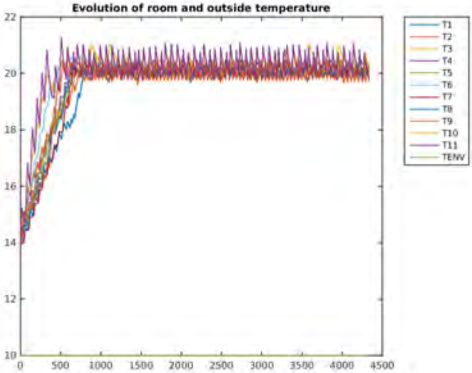
- System of dimension 11
- 2^{11} combinations of v_j (not all admissible, constraint on the number of open valves)
- Pipes heating a room may influence other rooms
- Doors opening and closing (here: average between open and closed)
- Varying external temperature (here: $T_{env} = 10^\circ C$)
- Measures and switching every 15 minutes

Seluxit case study, guaranteed reachability and stability

Decomposition in 5 + 6 rooms (cf. [Larsen et al., TACAS 2016], thanks to the Aalborg team for the simulator)

Input:
$R = [18, 22]^{11}$
$\varepsilon = 0.5$
$T_{env} = 10$

Output:
$a = 4$ in 15 steps
cpu time: 6h



Simulation of the Seluxit case study plotted with time (in min) for $T_{env} = 10^\circ C$.

Perturbed Euler scheme

Additional hypothesis on the dynamics:
(H_W): (Robustly OSL) $\exists\ \lambda_j \in \mathbb{R}$ and $\gamma_j \in \mathbb{R}_{\geq 0}$ s.t.

Perturbed Euler scheme

Additional hypothesis on the dynamics:
(H_W): (Robustly OSL) $\exists\ \lambda_j \in \mathbb{R}$ and $\gamma_j \in \mathbb{R}_{\geq 0}$ s.t.

$$\forall x, x' \in T,\ \forall w, w' \in W$$

$$\langle f_j(x,w) - f_j(x',w'), x - x' \rangle \leq \lambda_j \|x - x'\|^2 + \gamma_j \|x - x'\| \|w - w'\|.$$

NB: λ_j and γ_j can be computed with constrained optimization algorithms.
NB2: This notion is close to incremental input-to-state stability [Angeli].

Outline

1 Guaranteed control of switched systems

2 The reachable set computation

3 Distributed synthesis of controllers

4 **Control of partial differential equations with model order reduction**

- Model Order Reduction for high dimensional ODES
- Control of Partial Differential Equations

A switched system with output

- Described by the differential equation:

$$\begin{cases} \dot{x}(t) & = Ax(t) + Bu(t) \\ y(t) & = Cx(t) \end{cases}$$

A switched system with output

- Described by the differential equation:

$$\begin{cases} \dot{x}(t) & = Ax(t) + Bu(t) \\ y(t) & = Cx(t) \end{cases}$$

  - $x \in \mathbb{R}^n$: state variable
  - $y \in \mathbb{R}^m$: output
  - $u \in \mathbb{R}^p$: control input, takes a finite number of values (modes)
  - A,B,C: matrices of appropriate dimensions

A switched system with output

- Described by the differential equation:

$$\begin{cases} \dot{x}(t) & = Ax(t) + Bu(t) \\ y(t) & = Cx(t) \end{cases}$$

  - $x \in \mathbb{R}^n$: state variable
  - $y \in \mathbb{R}^m$: output
  - $u \in \mathbb{R}^p$: control input, takes a finite number of values (modes)
  - A,B,C: matrices of appropriate dimensions
- Idea: impose the right $u(t)$ such that x and y verify some properties (stability, reachability...)

A switched system with output

- Described by the differential equation:

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{cases}$$

  - $x \in \mathbb{R}^n$: state variable
  - $y \in \mathbb{R}^m$: output
  - $u \in \mathbb{R}^p$: control input, takes a finite number of values (modes)
  - A,B,C: matrices of appropriate dimensions
- Idea: impose the right $u(t)$ such that x and y verify some properties (stability, reachability...)
- Objectives:
  1. *x-stabilization*: make all the state trajectories starting in a compact interest set $R_x \subset \mathbb{R}^n$ return to R_x;
  2. *y-convergence*: send the output of all the trajectories starting in R_x into an objective set $R_y \subset \mathbb{R}^m$;

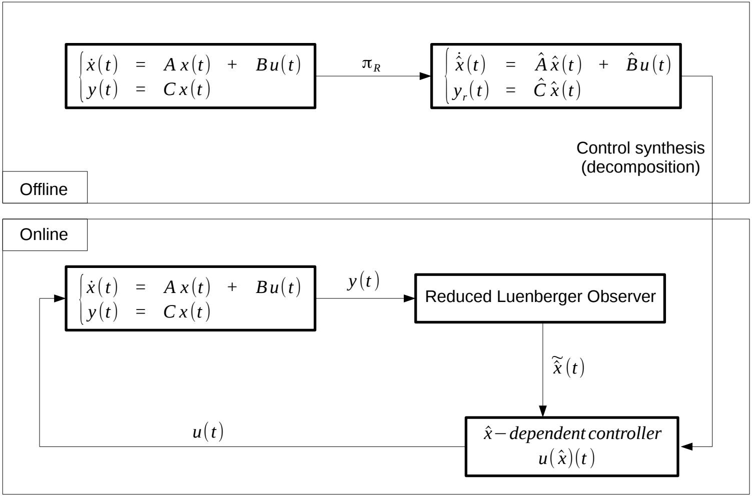
A switched system with output

- Described by the differential equation:

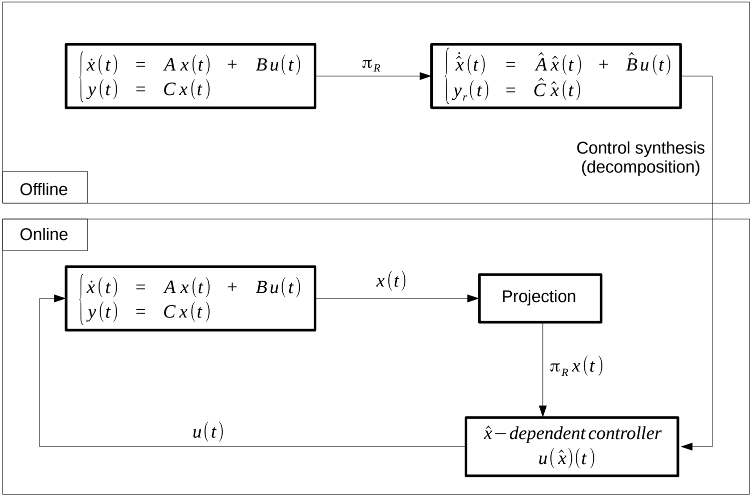
$$\begin{cases} \dot{x}(t) & = Ax(t) + Bu(t) \\ y(t) & = Cx(t) \end{cases}$$

  - $x \in \mathbb{R}^n$: state variable
  - $y \in \mathbb{R}^m$: output
  - $u \in \mathbb{R}^p$: control input, takes a finite number of values (modes)
  - A,B,C: matrices of appropriate dimensions
- Idea: impose the right $u(t)$ such that x and y verify some properties (stability, reachability...)
- Objectives:
  1. *x-stabilization*: make all the state trajectories starting in a compact interest set $R_x \subset \mathbb{R}^n$ return to R_x;
  2. *y-convergence*: send the output of all the trajectories starting in R_x into an objective set $R_y \subset \mathbb{R}^m$;
- Constraint: x of "high" dimension.

Dealing with high dimensionality : model reduction

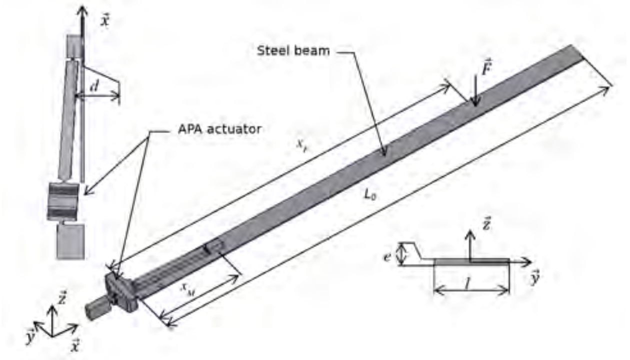


Dealing with high dimensionality : model reduction



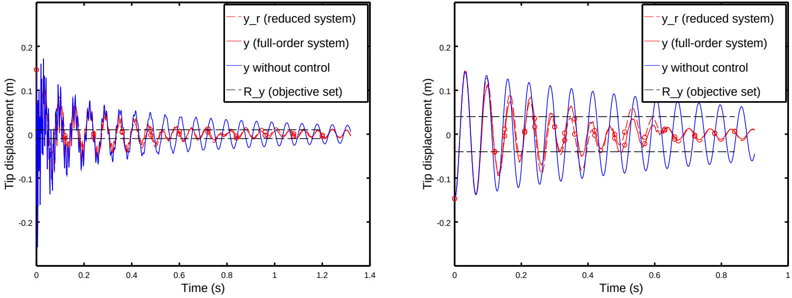
Application

- Vibration (online) control of a cantilever beam: $n = 120$ and $n_r = 4$



Application

- Vibration (online) control of a cantilever beam:
  $n = 120$ and $n_r = 4$



Case of PDE problems

Difficulty:

- The problem becomes infinite-dimensional;
- Even spatially discretized, the *curse of dimensionality* makes the former approaches (bisection, ball overlapping, ...) irrelevant.

$\Longrightarrow$ requires model order reduction (MOR)

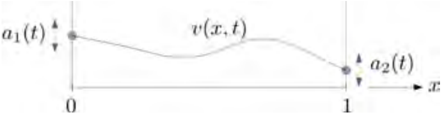
Pb of study: (ODE + 1D heat eq) with boundary control

$$\frac{d\boldsymbol{\xi}}{dt} = A_\sigma \boldsymbol{\xi} + \mathbf{b}_\sigma, \quad t > 0,$$

$$\frac{\partial u}{\partial t} - \nabla \cdot (\kappa(.)\nabla u) = f \quad \text{in } \Omega \times (0, +\infty),$$

$$u(0,t) = \xi_1(t), \quad u(L,t) = \xi_2(t), \quad \text{for all } t > 0,$$

$$u(., t = 0) = u^0$$



Use of 4 constant control modes:

$$\mathbf{b}_1 = (1,1)^T, \ \mathbf{b}_2 = (-1,-1)^T, \ \mathbf{b}_3 = (-1,1)^T, \ \mathbf{b}_4 = (1,-1)^T.$$

Control objective:

$$\boldsymbol{\xi}(t) \in R \quad \text{and} \quad \|u(.,t) - u^\infty\|_{L^2(0,1)} \leq \rho \quad \text{for all } t > 0.$$

Transformation of the problem

Denoting by $u_q = u_q(., t)$ the solution of the quasi-static problem at each time t:

$$-\nabla \cdot (\kappa(.)\nabla u_q) = f + \nabla \cdot (\kappa(.)\nabla u^\infty) \text{ in } \Omega,$$

$$u_q(0, t) = \xi_1(t) - \xi_1^\infty,$$

$$u_q(L, t) = \xi_2(t) - \xi_2^\infty,$$

one can express the solution u as the sum of u^∞, u_q and a function ψ, i.e.

$$u(., t) = u^\infty(.) + u_q(., t) + \psi(., t)$$

where $\psi(., t)$ is solution of the heat problem with homogeneous Dirichlet boundary conditions

Reduced order problem

Look for a low dimensional approximation $\tilde{\psi}$ of ψ:

$$\tilde{\psi}(x,t) = \sum_{k=1}^{K} \tilde{\beta}_k(t)\varphi^k(x)$$

with a reduced basis $\{\varphi^k\}_{k=1,\ldots,K}$ assumed to be orthonormal in $L^2(\Omega)$. Then

$$\|\tilde{\psi}(.,t)\|_{L^2(\Omega)} = \|\tilde{\beta}(t)\|_{2,\mathbb{R}^K}.$$

By the triangular inequality we can write

$$\begin{aligned} \|\psi(.,t)\|_{L^2(\Omega)} &\leq \|\psi(.,t) - \tilde{\psi}(.,t)\|_{L^2(\Omega)} + \|\tilde{\psi}(.,t)\|_{L^2(\Omega)} \\ &\leq \|\psi(.,t) - \tilde{\psi}(.,t)\|_{L^2(\Omega)} + \|\tilde{\beta}(t)\|_2. \end{aligned}$$

Reduced order problem

Additional assumption (can be ensured by a proper construction of the reduced basis):

$$\|\psi(.,t) - \tilde{\psi}(.,t)\|_{L^2(\Omega)} \leq \mu \, \|\psi^0 - \tilde{\psi}^0\|_{L^2(\Omega)} \quad \forall t \in [0,\tau]$$

Then:

$$C\|f + \nabla \cdot (\kappa(.)\nabla u^\infty)\|_{L^2(\Omega)} + L \, \|\boldsymbol{\xi}(t) - \boldsymbol{\xi}^\infty\|_\infty + \|\tilde{\boldsymbol{\beta}}(t)\|_2 + \mu \, \|\psi^0 - \tilde{\psi}^0\|_{L^2(\Omega)} \leq \rho.$$

And finally:

Global stability requirement

$$C\,\|f + \nabla \cdot (\kappa(.)\nabla u^\infty)\|_{L^2(\Omega)} + L \, \|\boldsymbol{\xi}(t) - \boldsymbol{\xi}^\infty\|_\infty + \|\tilde{\boldsymbol{\beta}}(t)\|_2 + \mu \, \|\psi^0 - \pi^K \psi^0\|_{L^2(\Omega)} + \mu \, \|\boldsymbol{\beta}^0 - \tilde{\boldsymbol{\beta}}^0\|_2 \leq \rho.$$

Numerical experiments

$$
\begin{aligned}
&\frac{d\mathbf{a}}{dt} = \mathbf{b}_\sigma, \quad \mathbf{b}_\sigma \in \mathbb{R}^2,\ t > 0, \\
&\alpha\, \partial_t v - \partial^2_{xx} v = 0 \quad \text{in } (0,1) \times (0, +\infty), \\
&v(0,t) = a_1(t), \quad v(1,t) = a_2(t), \quad t > 0, \\
&v(.,0) = v_0
\end{aligned}
$$

- $K = 4$ (reduced-order space of dimension 2+4=6)
- max switching sequence length = 8
- Offline step: Overlapping of the stability domain by $4^6 = 4096$ balls, computed in less than 20 mins on a laptop
- Guaranteed control verified

Numerical experiments (2)

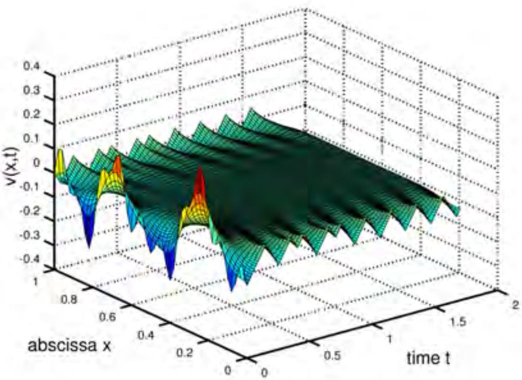


Figure : Controlled discrete solution $t \mapsto v(., t)$.

Conclusions and perspectives

Conclusions:

- Guaranteed control of nonlinear switched systems using guaranteed RK4/Euler
- Renewal of the Euler scheme using OSL property
- Compositional synthesis allowing to handle higher dimensions
- Control of PDEs made possible with Model Order Reduction and proper transformation of the problem

Conclusions and perspectives

Conclusions:

- Guaranteed control of nonlinear switched systems using guaranteed RK4/Euler
- Renewal of the Euler scheme using OSL property
- Compositional synthesis allowing to handle higher dimensions
- Control of PDEs made possible with Model Order Reduction and proper transformation of the problem

Perspectives:

- Stochastic systems using Euler
- Would the OSL property be relevant on other numerical schemes?
- Testing on real PDE case studies
- Coupling of domain decomposition methods and compositional synthesis