Euler's method applied to the control of switched systems

Laurent Fribourg[1]

joint work with: Adrien Le Coënt[2], Florian De Vuyst[2], Ludovic Chamoin[3], Julien Alexandre dit Sandretto[4], Alexandre Chapoutot[4]

June 30, 2017

[1]CNRS - LSV - ENS Paris-Saclay - INRIA
[2]CMLA - ENS Paris-Saclay
[3]LMT - ENS Paris-Saclay
[4]U2IS - ENSTA ParisTech

Plan

1. Switched systems with τ-sampling

Plan

1. Switched systems with τ-sampling

2. General problem of control synthesis for (R,S)-stability
   $\longrightarrow$ basic problem of one-step invariance and set (or symbolic) integration

Plan

1. Switched systems with τ-sampling

2. General problem of control synthesis for (R,S)-stability
   $\longrightarrow$ basic problem of one-step invariance and set (or symbolic) integration

3. Euler's method and error estimation δ (using Dahlquist constant λ)
   $\longrightarrow$ application to set integration and control synthesis

Plan

1. Switched systems with τ-sampling

2. General problem of control synthesis for (R,S)-stability
   $\longrightarrow$ basic problem of one-step invariance and set (or symbolic) integration

3. Euler's method and error estimation δ (using Dahlquist constant λ)
   $\longrightarrow$ application to set integration and control synthesis

4. Euler error in presence of disturbance
   $\longrightarrow$ application to distributed control synthesis

Plan

1. Switched systems with τ-sampling
2. General problem of control synthesis for (R,S)-stability
   $\longrightarrow$ basic problem of one-step invariance and set (or symbolic) integration
3. Euler's method and error estimation δ (using Dahlquist constant λ)
   $\longrightarrow$ application to set integration and control synthesis
4. Euler error in presence of disturbance
   $\longrightarrow$ application to distributed control synthesis
5. Comparison with classical interval-based integration methods

Outline

Switched systems

A continuous switched system

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

- state $x(t) \in \mathbb{R}^n$
- switching rule $\sigma(\cdot) : \mathbb{R}^+ \longrightarrow U$
- finite set of (switched) modes $U = \{1, \ldots, N\}$

Switched systems

A continuous switched system

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

- state $x(t) \in \mathbb{R}^n$
- switching rule $\sigma(\cdot) : \mathbb{R}^+ \longrightarrow U$
- finite set of (switched) modes $U = \{1, \dots, N\}$

We focus on sampled switched systems:
given a sampling period $\tau > 0$, switchings will occur at times τ, 2τ, ...

Switched systems

A continuous switched system

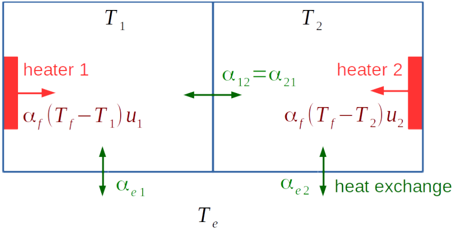
$$\dot{x}(t) = f_{\sigma(t)}(x(t))$$

- state $x(t) \in \mathbb{R}^n$
- switching rule $\sigma(\cdot) : \mathbb{R}^+ \longrightarrow U$
- finite set of (switched) modes $U = \{1, \dots, N\}$

We focus on sampled switched systems:
given a sampling period $\tau > 0$, switchings will occur at times τ, 2τ, ...

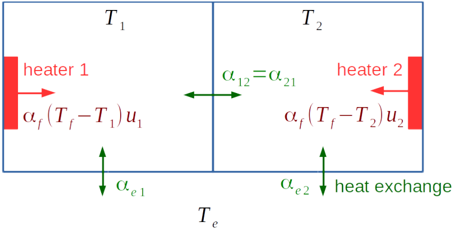
Control Synthesis problem:
Find at each sampling time, the appropriate mode $u \in U$ (in function of the value of $x(t)$) in order to make the system satisfy a certain property.

Example: Two-room apartment



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

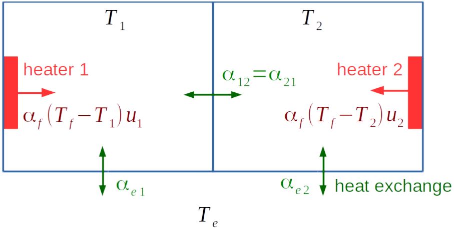
Example: Two-room apartment



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

- 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

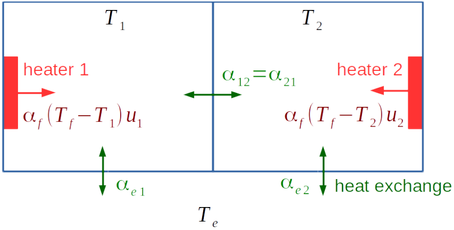
Example: Two-room apartment



$$\dot{T}_1 = f^1_{u_1}(T_1(t), T_2(t))$$
$$\dot{T}_2 = f^2_{u_2}(T_1(t), T_2(t))$$

- 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

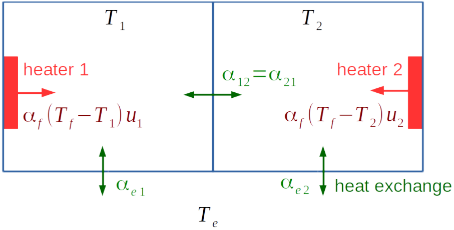
Example: Two-room apartment



$$\dot{T}_1 = f^1_{u_1}(T_1(t), T_2(t))$$
$$\dot{T}_2 = f^2_{u_2}(T_1(t), T_2(t))$$

- 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



$$\dot{T}_1 = f^1_{u_1}(T_1(t), T_2(t))$$
$$\dot{T}_2 = f^2_{u_2}(T_1(t), T_2(t))$$

- 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.

Reachability and Stability Problems

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

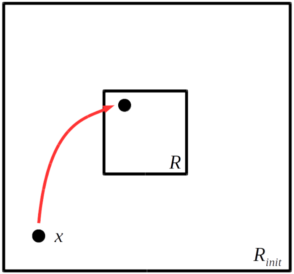
At each sampling time t, find the appropriate switched mode $u \in U$ according to the current value of x, in order to achieve some objectives:

Reachability and Stability Problems

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

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

- reachability (given a target region R, find a control which drives x to R, for any x in R_{init})

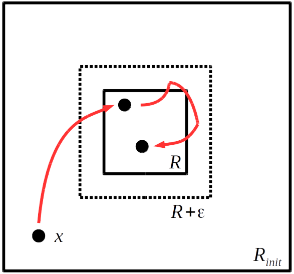


Reachability and Stability Problems

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

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

- reachability (given a target region R, find a control which drives x to R, for any x in R_{init})
- stability (once in R, find a control which always maintain x in a neighborhood $S = R + \varepsilon$ of R)

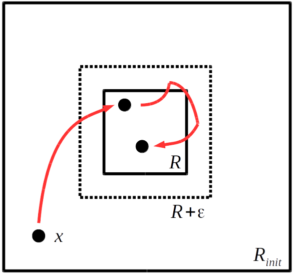


Reachability and Stability Problems

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

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

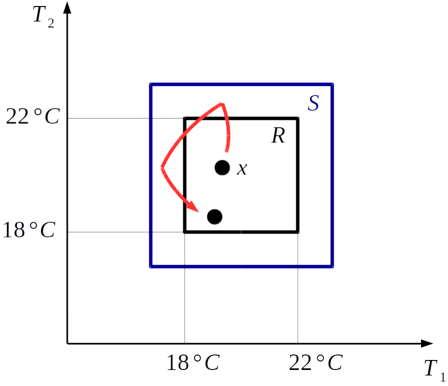
- reachability (given a target region R, find a control which drives x to R, for any x in R_{init})
- stability (once in R, find a control which always maintain x in a neighborhood $S = R + \varepsilon$ of R)



NB: classic stabilization to an equilibrium point, impossible to achieve here
$\leadsto$ practical stability

Outline

Focus on (R, S)-stability



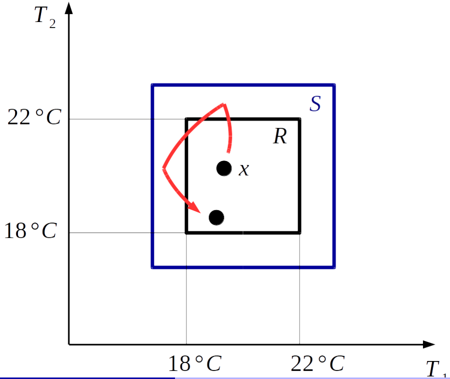
Focus on (R, S)-stability

Being given a recurrence (rectang.) set R and a safety (rectang.) set S, we consider the state-dependent control problem of synthesizing σ:

At each sampling time t, determine the switched mode $u \in U$ in function of the value of $x(t)$, in order to satisfy:

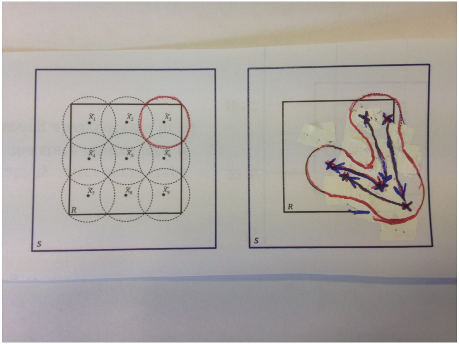
(R,S)-stability:
if $x(0) \in R$, then $x(t)$:

1. returns infinitely often into R, and
2. always stays in S.



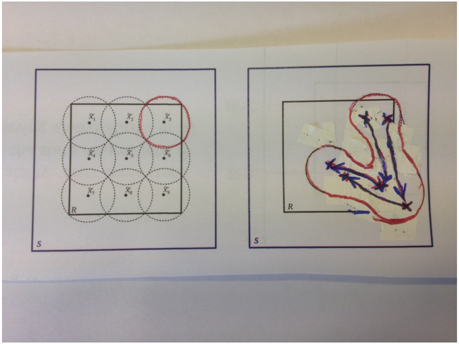
Principle of (R,S)-stability control synthesis (MINIMATOR)

[R. Soulat's PhD, 2013]



Principle of (R,S)-stability control synthesis (MINIMATOR)

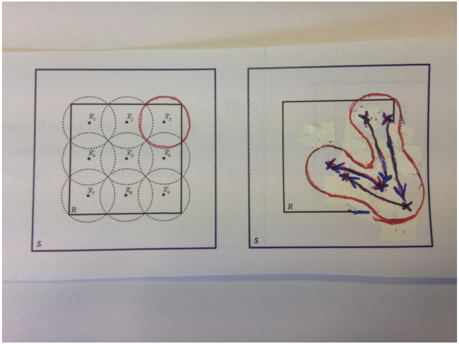
[R. Soulat's PhD, 2013]



1. Cover R with a finite set of balls $B_1^0, B_2^0, \ldots$ all $\subset S$

Principle of (R,S)-stability control synthesis (MINIMATOR)

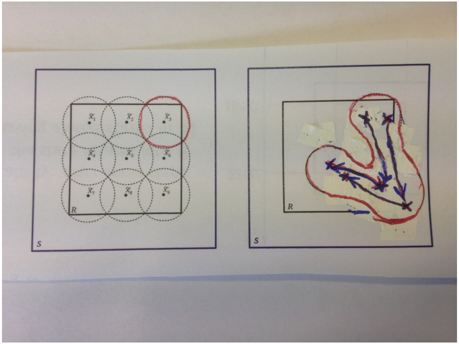
[R. Soulat's PhD, 2013]



1. Cover R with a finite set of balls $B_1^0, B_2^0, \ldots$ all $\subset S$
2. for each ball B^0, find a pattern π of length k s.t. all the controlled traj. $x(t)$ with $x(0) \in B^0$, satisfy:

Principle of (R,S)-stability control synthesis (MINIMATOR)

[R. Soulat's PhD, 2013]

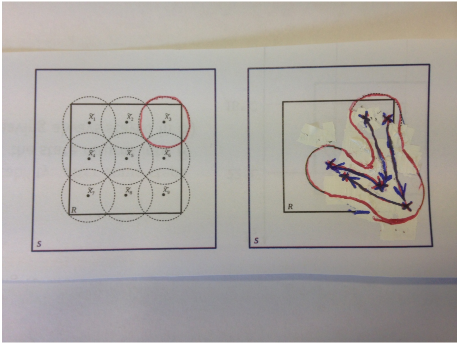


1. Cover R with a finite set of balls $B_1^0, B_2^0, \ldots$ all $\subset S$
2. for each ball B^0, find a pattern π of length k s.t. all the controlled traj. $x(t)$ with $x(0) \in B^0$, satisfy:

$$x(t) \in S \quad \text{for all } t \in [0, k\tau]$$

Principle of (R,S)-stability control synthesis (MINIMATOR)

[R. Soulat's PhD, 2013]



1. Cover R with a finite set of balls $B_1^0, B_2^0, \ldots$ all $\subset S$
2. for each ball B^0, find a pattern π of length k s.t. all the controlled traj. $x(t)$ with $x(0) \in B^0$, satisfy:

$$x(t) \in S \quad \text{for all } t \in [0, k\tau] \quad \wedge \quad x(t) \in R \quad \text{for } t = k\tau$$

Remarks on (R, S)-stability control

1. At each ball B (covering R), is assoc. a "returning pattern" π of lg., say k

[1] $\approx$ Model Predictive Control where the optimal strategy is estimated (online) for the next k_1 steps (but strategy updated there at *each* step, $\neq$ after k_1 steps: "receding prediction horizon"). Note also that, here, control π_1 is computed off line.

Remarks on (R, S)-stability control

1. At each ball B (covering R), is assoc. a "returning pattern" π of lg., say k
2. Once returned in R at $t = t_1$, the sensors give the value of $x(t_1)$, and a control pattern π_1 (corresponding to a ball $B_1 \ni x(t_1)$) is applied; the process iterates at next return time $(t_2 = t_1 + k_1\tau)$.

[1] $\approx$ Model Predictive Control where the optimal strategy is estimated (online) for the next k_1 steps (but strategy updated there at *each* step, $\neq$ after k_1 steps: "receding prediction horizon"). Note also that, here, control π_1 is computed off line.

Remarks on (R, S)-stability control

1. At each ball B (covering R), is assoc. a "returning pattern" π of lg., say k
2. Once returned in R at $t = t_1$, the sensors give the value of $x(t_1)$, and a control pattern π_1 (corresponding to a ball $B_1 \ni x(t_1)$) is applied; the process iterates at next return time $(t_2 = t_1 + k_1\tau)$.
3. Complexity: for n state dimension, N modes, K max. lg. of patterns, 2^{nd} balls (uniform covering, with d bisection depth):

$$2^{nd} N^K \text{ possible tests of patterns}$$

$\rightarrow$ exponential in n, d, K
(note that N can be itself exp. in n, cf. room heating example)

[1] ≈ Model Predictive Control where the optimal strategy is estimated (online) for the next k_1 steps (but strategy updated there at *each* step, $\neq$ after k_1 steps: "receding prediction horizon"). Note also that, here, control π_1 is computed off line.

Remarks on (R, S)-stability control

1. At each ball B (covering R), is assoc. a "returning pattern" π of lg., say k
2. Once returned in R at $t = t_1$, the sensors give the value of $x(t_1)$, and a control pattern π_1 (corresponding to a ball $B_1 \ni x(t_1)$) is applied; the process iterates at next return time $(t_2 = t_1 + k_1\tau)$.
3. Complexity: for n state dimension, N modes, K max. lg. of patterns, 2^{nd} balls (uniform covering, with d bisection depth):

   $$2^{nd}N^K \text{ possible tests of patterns}$$

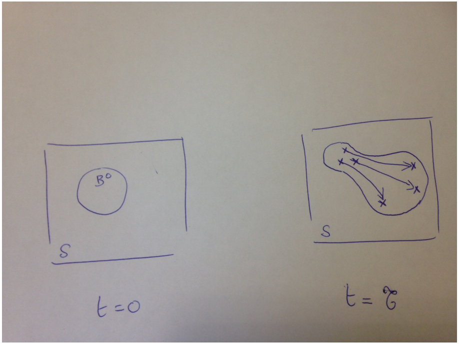
   $\rightarrow$ exponential in n, d, K
   (note that N can be itself exp. in n, cf. room heating example)
4. The length $|\pi_1| = k_1$ can be seen as a time-horizon: the strategy is planned for k_1 steps, then updated after k_1 steps.[1]

[1] $\approx$ Model Predictive Control where the optimal strategy is estimated (online) for the next k_1 steps (but strategy updated there at *each* step, $\neq$ after k_1 steps: "receding prediction horizon"). Note also that, here, control π_1 is computed off line.

Key notion: one-step invariance

Given a ball $B^0 \subset S$, find a mode $u \in U$ s.t. all the u-trajectories $x(t)$ with $x(0) \in B^0$, satisfy:

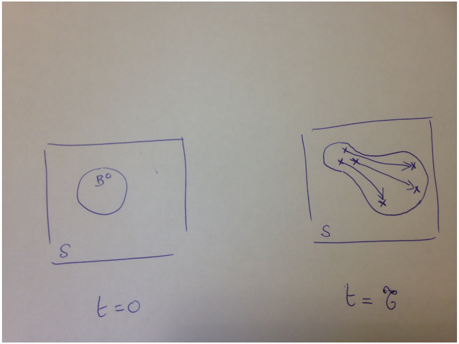
$$x(t) \in S \quad \text{for all } t \in [0, \tau]$$



Key notion: one-step invariance

Given a ball $B^0 \subset S$, find a mode $u \in U$ s.t. all the u-trajectories $x(t)$ with $x(0) \in B^0$, satisfy:

$$x(t) \in S \quad \text{for all } t \in [0, \tau]$$

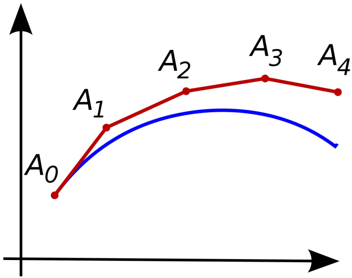


⟶ requires a techn. of set-integration; we will use an Euler-based techn.

Outline

Euler's estimation method of $x(t)$ (with $\dot{x}(t) = f(x(t))$)

$$\tilde{x}(t) = \tilde{x}(t_0) + f(\tilde{x}(t_0))(t - t_0)$$



Suppose that, for the current step size τ (or a sub-sampling size h), the derivative is constant and equal to the derivative at the starting point

Global error estimated with Lipschitz constant L

- The global error at $t = t_0 + kh$ is equal to $\|x(t) - \tilde{x}(t)\|$.

Global error estimated with Lipschitz constant L

- The global error at $t = t_0 + kh$ is equal to $\|x(t) - \tilde{x}(t)\|$.
  In case $n = 1$, if f is Lipschitz cont. ($\|f(y) - f(x)\| \leq L\|y - x\|$), then:

  $$error(t) \leq \frac{hM}{2L}(e^{L(t-t_0)} - 1)$$

  where L is the Lipschitz constant of f (and M an upper bound on f'').

Global error estimated with Lipschitz constant L

- The global error at $t = t_0 + kh$ is equal to $\|x(t) - \tilde{x}(t)\|$.
  In case $n = 1$, if f is Lipschitz cont. ($\|f(y) - f(x)\| \leq L\|y - x\|$), then:
  $$error(t) \leq \frac{hM}{2L}(e^{L(t-t_0)} - 1)$$
  where L is the Lipschitz constant of f (and M an upper bound on f'').

- In case of "stiff" equations, L can be very big.

Global error estimated with Lipschitz constant L

- The global error at $t = t_0 + kh$ is equal to $\|x(t) - \tilde{x}(t)\|$.
  In case $n = 1$, if f is Lipschitz cont. ($\|f(y) - f(x)\| \leq L\|y - x\|$), then:

  $$error(t) \leq \frac{hM}{2L}(e^{L(t-t_0)} - 1)$$

  where L is the Lipschitz constant of f (and M an upper bound on f'').

- In case of "stiff" equations, L can be very big.

We now consider a more appropriate constant λ that leads to sharper estimations of the Euler error.

Dahlquist's constant λ ("one-sided Lipschitz" constant)

- $\lambda \in \mathbb{R}$ is a constant s.t., for all $x, y \in S$:

$$\langle f(y) - f(x), y - x \rangle \leq \lambda \|y - x\|^2$$

where $\langle \cdot, \cdot \rangle$ denote the scalar product of two vectors of $\mathbb{R}^n$

[2] Define $V(x, x') = \|x - x'\|^2$; we have: $\frac{dV}{dt} \leq \lambda V$ (hence $V = V_0 e^{\lambda t}$). So V is an exponentially stable Lyapunov function when $\lambda < 0$.

Dahlquist's constant λ ("one-sided Lipschitz" constant)

- $\lambda \in \mathbb{R}$ is a constant s.t., for all $x, y \in S$:

$$\langle f(y) - f(x), y - x \rangle \leq \lambda \|y - x\|^2$$

  where $\langle \cdot, \cdot \rangle$ denote the scalar product of two vectors of $\mathbb{R}^n$

- λ can be < 0 ($\rightarrow$ contractivity)[2];
  even in case $\lambda > 0$, in practice: $\lambda << L$
  $\longrightarrow$ sharper (λ-exponential based) estimation of Euler error

[2]Define $V(x, x') = \|x - x'\|^2$; we have: $\frac{dV}{dt} \leq \lambda V$ (hence $V = V_0 e^{\lambda t}$). So V is an exponentially stable Lyapunov function when $\lambda < 0$.

Dahlquist's constant λ ("one-sided Lipschitz" constant)

- $\lambda \in \mathbb{R}$ is a constant s.t., for all $x, y \in S$:

$$\langle f(y) - f(x), y - x \rangle \leq \lambda \|y - x\|^2$$

  where $\langle \cdot, \cdot \rangle$ denote the scalar product of two vectors of $\mathbb{R}^n$

- λ can be < 0 ($\rightarrow$ contractivity)[2];
  even in case $\lambda > 0$, in practice: $\lambda << L$
  $\longrightarrow$ sharper (λ-exponential based) estimation of Euler error

- λ can be computed using constraint optimization algorithms

[2] Define $V(x, x') = \|x - x'\|^2$; we have: $\frac{dV}{dt} \leq \lambda V$ (hence $V = V_0 e^{\lambda t}$). So V is an exponentially stable Lyapunov function when $\lambda < 0$.

Local error function $\delta(\cdot)$ estimated using constant λ

Given an initial error δ_0 of $\tilde{x}(t)$ (i.e.: $\|\tilde{x}(0) - x(0)\| \leq \delta_0$),
the local E. error fn $\delta(\cdot)$ (s.t.: $\|x(t) - \tilde{x}(t)\| \leq \delta(t)$, for $t \in [0, \tau]$)
can be defined (for each mode u) by:

Local error function $\delta(\cdot)$ estimated using constant λ

Given an initial error δ_0 of $\tilde{x}(t)$ (i.e.: $\|\tilde{x}(0) - x(0)\| \leq \delta_0$),
the local E. error fn $\delta(\cdot)$ (s.t.: $\|x(t) - \tilde{x}(t)\| \leq \delta(t)$, for $t \in [0, \tau]$)
can be defined (for each mode u) by:

- if $\lambda < 0$:

$$\delta(t) = \left(\delta_0^2 e^{\lambda t} + \frac{C^2}{\lambda^2} \left(t^2 + \frac{2t}{\lambda} + \frac{2}{\lambda^2} \left(1 - e^{\lambda t} \right) \right) \right)^{\frac{1}{2}}$$

- if $\lambda = 0$:

$$\delta(t) = \left(\delta_0^2 e^t + C^2(-t^2 - 2t + 2(e^t - 1)) \right)^{\frac{1}{2}}$$

- if $\lambda > 0$:

$$\delta(t) = \left(\delta_0^2 e^{3\lambda t} + \frac{C^2}{3\lambda^2} \left(-t^2 - \frac{2t}{3\lambda} + \frac{2}{9\lambda^2} \left(e^{3\lambda t} - 1 \right) \right) \right)^{\frac{1}{2}}$$

with $C = \sup_{x \in S} \ L\|f(x)\|$.

Local error function $\delta(\cdot)$ estimated using constant λ

Given an initial error δ_0 of $\tilde{x}(t)$ (i.e.: $\|\tilde{x}(0) - x(0)\| \leq \delta_0$),
the local E. error fn $\delta(\cdot)$ (s.t.: $\|x(t) - \tilde{x}(t)\| \leq \delta(t)$, for $t \in [0, \tau]$)
can be defined (for each mode u) by:

- if $\lambda < 0$:

$$\delta(t) = \left(\delta_0^2 e^{\lambda t} + \frac{C^2}{\lambda^2} \left(t^2 + \frac{2t}{\lambda} + \frac{2}{\lambda^2} \left(1 - e^{\lambda t} \right) \right) \right)^{\frac{1}{2}}$$

- if $\lambda = 0$:

$$\delta(t) = \left(\delta_0^2 e^t + C^2(-t^2 - 2t + 2(e^t - 1)) \right)^{\frac{1}{2}}$$

- if $\lambda > 0$:

$$\delta(t) = \left(\delta_0^2 e^{3\lambda t} + \frac{C^2}{3\lambda^2} \left(-t^2 - \frac{2t}{3\lambda} + \frac{2}{9\lambda^2} \left(e^{3\lambda t} - 1 \right) \right) \right)^{\frac{1}{2}}$$

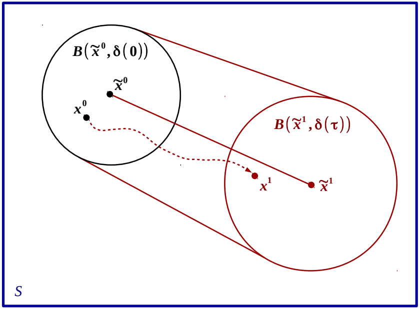
with $C = \sup_{x \in S} \ L\|f(x)\|$.

see [A. Le Coënt's Ph.D Thesis, 2017].

One-step invariance using the E. error fn $\delta(\cdot)$

- Given a ball $B^0 \equiv B(\tilde{x}^0, \delta^0) \subset S$, find a mode u s.t.:

$$x(t) \in S \quad \text{for all } x(0) \in B^0, t \in [0, \tau]$$

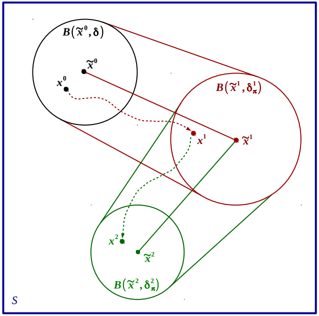


i.e.: $B^1 \equiv B(\tilde{x}^1, \delta^1) \subset S$ with $\tilde{x}^1 = \tilde{x}^0 + f(\tilde{x}^0)\tau$ and $\delta^1 = \delta(\tau)$ (assuming convexity of $\delta(\cdot)$ on $[0, \tau]$).

Finding a control pattern π using $\delta(\cdot)$

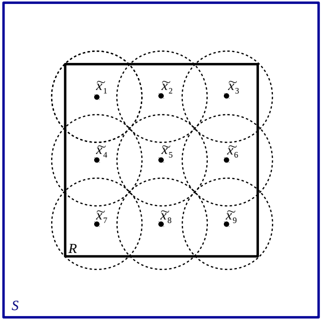
- Given a ball $B^0 \equiv B(\tilde{x}^0, \delta^0) \subset S$, find a pattern π (of length k) s.t.:

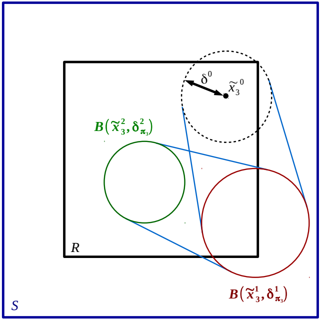
$$x(t) \in S \quad \text{for all } x(0) \in B^0, t \in [0, k\tau]$$



i.e.: $B^1 \equiv B(\tilde{x}^1, \delta^1) \subset S, \quad \ldots, \quad B^k \equiv B(\tilde{x}^k, \delta^k) \subset S$

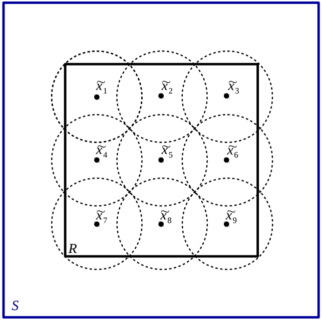
(R,S)-stable control synthesis using E. error fn $\delta(\cdot)$

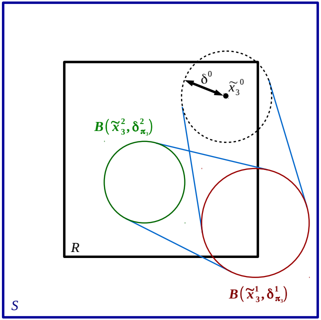




For each ball $B_i^0 \equiv B(\tilde{x}_i^0, \delta_i^0) \subset S$ covering R, find a pattern π_i (of length k_i) s.t.:

(R,S)-stable control synthesis using E. error fn $\delta(\cdot)$

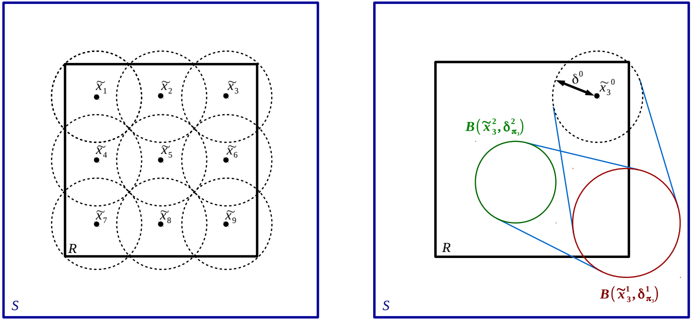




For each ball $B_i^0 \equiv B(\tilde{x}_i^0, \delta_i^0) \subset S$ covering R, find a pattern π_i (of length k_i) s.t.:

- Safety: $B_i^1 \equiv B(\tilde{x}_i^1, \delta_i^1) \subset S, ..., B_i^{k_i-1} \equiv B(\tilde{x}_i^{k_i-1}, \delta_i^{k_i-1}) \subset S$, and

(R,S)-stable control synthesis using E. error fn $\delta(\cdot)$



For each ball $B_i^0 \equiv B(\tilde{x}_i^0, \delta_i^0) \subset S$ covering R, find a pattern π_i (of length k_i) s.t.:

- Safety: $B_i^1 \equiv B(\tilde{x}_i^1, \delta_i^1) \subset S, ..., B_i^{k_i-1} \equiv B(\tilde{x}_i^{k_i-1}, \delta_i^{k_i-1}) \subset S$, and
- Recurrence: $B_i^{k_i} \equiv B(\tilde{x}_i^{k_i}, \delta_i^{k_i}) \subset R$

Outline

incremental Input-to-State Stability (i-ISS) in presence of disturbance $w \in W$

Consider: $\dot{x}(t) = f(x(t), w(t))$ with $w(t) \in W$ for all $t \in [0, \tau]$.

---

[3] In case $\lambda < 0$, (H) expresses (a variant of) the fact that $V(x, x') = \|x - x'\|^2$ is an i-ISS Lyapunov fn (see, e.g., [D. Angeli] [Hespanha et al.]). The constants λ, γ can be numerically computed using constrained optimization algos.

incremental Input-to-State Stability (i-ISS) in presence of disturbance $w \in W$

Consider: $\dot{x}(t) = f(x(t), w(t))$ with $w(t) \in W$ for all $t \in [0, \tau]$.

The eq. $\dot{x} = f(x, w)$ with $w \in W$ is said to satisfy the property of i-ISS w.r.t disturbance set W if
$\exists\ \lambda \in \mathbb{R}$ [3] and $\gamma \in \mathbb{R}_{\geq 0}$ s.t.

[3] In case $\lambda < 0$, (H) expresses (a variant of) the fact that $V(x, x') = \|x - x'\|^2$ is an i-ISS Lyapunov fn (see, e.g., [D. Angeli] [Hespanha et al.]). The constants λ, γ can be numerically computed using constrained optimization algos.

incremental Input-to-State Stability (i-ISS) in presence of disturbance $w \in W$

Consider: $\dot{x}(t) = f(x(t), w(t))$ with $w(t) \in W$ for all $t \in [0, \tau]$.

The eq. $\dot{x} = f(x, w)$ with $w \in W$ is said to satisfy the property of i-ISS w.r.t disturbance set W if

$\exists\ \lambda \in \mathbb{R}$ [3] and $\gamma \in \mathbb{R}_{\geq 0}$ s.t.

(H) $\forall x, x' \in S$, $\forall w, w' \in W$:

$$\langle f(x, w) - f(x', w'), x - x' \rangle \leq \lambda \|x - x'\|^2 + \gamma \|x - x'\| \|w - w'\|.$$

[3] In case $\lambda < 0$, (H) expresses (a variant of) the fact that $V(x, x') = \|x - x'\|^2$ is an i-ISS Lyapunov fn (see, e.g., [D. Angeli] [Hespanha et al.]). The constants λ, γ can be numerically computed using constrained optimization algos.

E. error function $\delta_W(\cdot)$ in presence of disturbance $w \in W$

- Consider the ODE:
  $\dot{x}(t) = f(x(t), w(t)) \quad \text{with} \quad w(t) \in W \quad \text{for all } t \in [0, \tau].$

E. error function $\delta_W(\cdot)$ in presence of disturbance $w \in W$

- The fn $\delta_W(\cdot)$ (s.t: for all $t \in [0,\tau]$, $w(t) \in W$: $\|x(t) - \tilde{x}(t)\| \leq \delta_W(t)$) can now be defined by:
  - if $\lambda < 0$,

$$\delta_W(t) = \left(\frac{(C)^2}{-(\lambda)^4} \left(-(\lambda)^2 t^2 - 2\lambda t + 2e^{\lambda t} - 2 \right) + \frac{1}{(\lambda)^2} \left(\frac{C\gamma|W|}{-\lambda} \left(-\lambda t + e^{\lambda t} - 1 \right) + \lambda \left(\frac{(\gamma)^2 (|W|/2)^2}{-\lambda} (e^{\lambda t} - 1) + \lambda (\delta^0)^2 e^{\lambda t} \right) \right) \right)^{1/2} \quad (1)$$

  - if $\lambda > 0$,

$$\delta_W(t) = \frac{1}{(3\lambda)^{3/2}} \left(\frac{C^2}{\lambda} \left(-9(\lambda)^2 t^2 - 6\lambda t + 2e^{3\lambda t} - 2 \right) + 3\lambda \left(\frac{C\gamma|W|}{\lambda} \left(-3\lambda t + e^{3\lambda t} - 1 \right) + 3\lambda \left(\frac{(\gamma)^2 (|W|/2)^2}{\lambda} (e^{3\lambda t} - 1) + 3\lambda (\delta^0)^2 e^{3\lambda t} \right) \right) \right)^{1/2} \quad (2)$$

  - if $\lambda = 0$,

$$\delta_W(t) = \left((C)^2 \left(-t^2 - 2t + 2e^t - 2 \right) + \left(C\gamma|W| \left(-t + e^t - 1 \right) + ((\gamma)^2 (|W|/2)^2 (e^t - 1) + (\delta^0)^2 e^t) \right) \right)^{1/2}$$

Compositional (R,S)-stability using E. error fns $\delta_{1,S_2}, \delta_{2,S_1}$

$$\dot{x}_1 = f^1(x_1, x_2)$$
$$\dot{x}_2 = f^2(x_1, x_2)$$

Compositional (R,S)-stability using E. error fns $\delta_{1,S_2}, \delta_{2,S_1}$

$$\dot{x}_1 = f^1(x_1, x_2)$$
$$\dot{x}_2 = f^2(x_1, x_2)$$

Suppose:

- (H1) $\dot{x}_1 = f^1(x_1, x_2)$ is i-ISS w.r.t disturbance $x_2 \in S_2$, with λ^1, γ^1.
- (H2) $\dot{x}_2 = f^2(x_1, x_2)$ is i-ISS w.r.t disturbance $x_1 \in S_1$, with λ^2, γ^2.

Compositional (R,S)-stability using E. error fns $\delta_{1,S_2}, \delta_{2,S_1}$

$$\dot{x}_1 = f^1(x_1, x_2)$$
$$\dot{x}_2 = f^2(x_1, x_2)$$

Suppose:

- (H1) $\dot{x}_1 = f^1(x_1, x_2)$ is i-ISS w.r.t disturbance $x_2 \in S_2$, with λ^1, γ^1.
- (H2) $\dot{x}_2 = f^2(x_1, x_2)$ is i-ISS w.r.t disturbance $x_1 \in S_1$, with λ^2, γ^2.

Theorem (compositionality): If

- σ_1 is an (R_1, S_1)-stable control of $x_1(t)$ with S_2 as domain of disturbance, using the E. error fn $\delta_{1,S_2}(t)$ (bounding $\|\tilde{x}_1(t) - x_1(t)\|$ in terms of (λ^1, γ^1))
- σ_2 is an (R_2, S_2)-stable control of $x_2(t)$ using the E. error fn $\delta_{2,S_1}(t)$.

Compositional (R,S)-stability using E. error fns $\delta_{1,S_2}, \delta_{2,S_1}$

$$\dot{x}_1 = f^1(x_1, x_2)$$
$$\dot{x}_2 = f^2(x_1, x_2)$$

Suppose:

- (H1) $\dot{x}_1 = f^1(x_1, x_2)$ is i-ISS w.r.t disturbance $x_2 \in S_2$, with λ^1, γ^1.
- (H2) $\dot{x}_2 = f^2(x_1, x_2)$ is i-ISS w.r.t disturbance $x_1 \in S_1$, with λ^2, γ^2.

Theorem (compositionality): If

- σ_1 is an (R_1, S_1)-stable control of $x_1(t)$ with S_2 as domain of disturbance, using the E. error fn $\delta_{1,S_2}(t)$ (bounding $\|\tilde{x}_1(t) - x_1(t)\|$ in terms of (λ^1, γ^1))
- σ_2 is an (R_2, S_2)-stable control of $x_2(t)$ using the E. error fn $\delta_{2,S_1}(t)$.

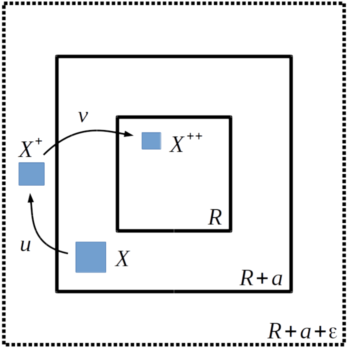
Then: $\sigma = \sigma_1 | \sigma_2$ is an $(R_1 \times R_2, S_1 \times S_2)$-stable control of $x(t) = (x_1(t), x_2(t))$.

Illustration of Distributed vs. Centralized Control

Centralized control synthesis

$$\dot{x}(t) = f_u(x(t))$$

Example of a validated pattern of length 2 mapping the "ball" X into R with $S = R + a + \varepsilon$ as safety box:



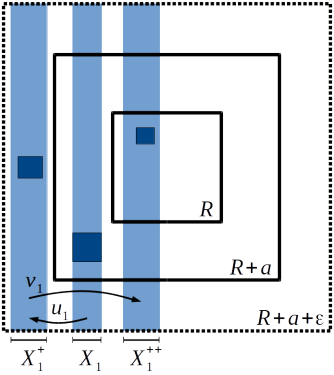
- $X \subset R$
- $X^+ = f_u(X) \subset S$
- $X^{++} = f_v(X^+) \subset R$

- Pattern $u \cdot v$ depends on X

Distrib. Control Synth. (of x_1 using S_2 as approx. of x_2)

$$\dot{x}_1(t) = f^1_{u_1}(x_1(t), x_2(t))$$
$$\dot{x}_2(t) = f^2_{u_2}(x_1(t), x_2(t))$$

Target zone: $R = R_1 \times R_2$



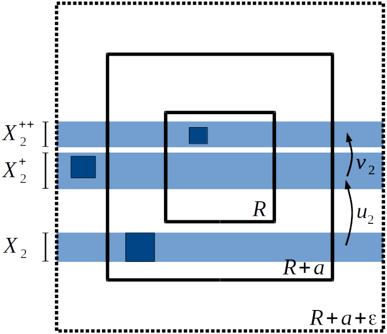
- $X_1 \subset R_1$
- $X_1^+ = f^1_{u_1}(X_1, S_2) \subset S_1$
- $X_1^{++} = f^1_{v_1}(X_1^+, S_2) \subset R_1$

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

Distrib. Control Synth. (of x_2 using S_1 as approx. of x_1)

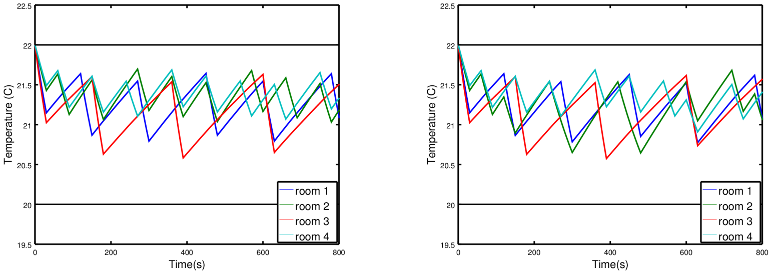
$$\dot{x}_1(t) = f^1_{u_1}(x_1(t), x_2(t))$$
$$\dot{x}_2(t) = f^2_{u_2}(x_1(t), x_2(t))$$

Target zone: $R = R_1 \times R_2$



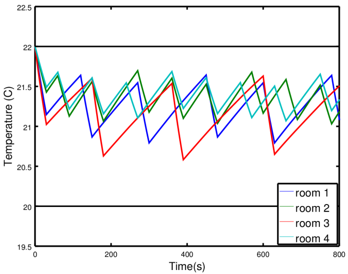
- $X_2 \subset R_2$
- $X_2^+ = f^2_{u_2}(S_1, X_2) \in S_2$
- $X_2^{++} = f^2_{v_2}(S_1, X_2^+) \in R_2$
- Pattern $u_2 \cdot v_2$ depends only on X_2

Application to distributed control of switched systems



Simulations of centralized control (left) and distributed (right) for the 4-rooms problem [P.-J. Meyer's Ph.D., 2015]

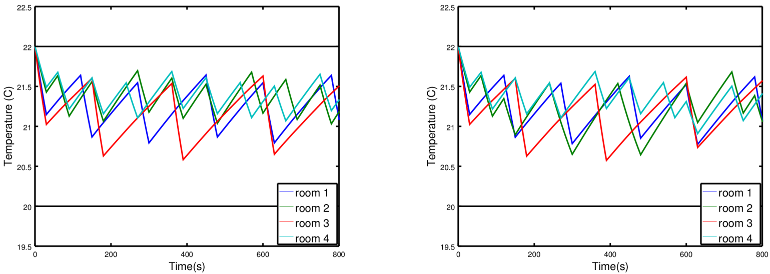
Application to distributed control of switched systems



Simulations of centralized control (left) and distributed (right) for the 4-rooms problem [P.-J. Meyer's Ph.D., 2015]

- centralized synthesis ($|\pi| = 2$): sub-sampling $h = \frac{\tau}{20}$,
  2^4 modes, 256 balls $\rightarrow$ 48 s. of CPU time.

Application to distributed control of switched systems



Simulations of centralized control (left) and distributed (right) for the 4-rooms problem [P.-J. Meyer's Ph.D., 2015]

- centralized synthesis ($|\pi| = 2$): sub-sampling $h = \frac{\tau}{20}$,
  2^4 modes, 256 balls $\quad\rightarrow\quad$ $48\ s$. of CPU time.

- distributed synthesis ($|\pi| = 2$): sub-sampling $h = \frac{\tau}{10} \mid h = \frac{\tau}{1}$,
  $2^2 \mid 2^2$ modes, $16 \mid 16$ balls $\quad\rightarrow\quad$ $< 1\ s$. of CPU time.

Final remarks

Final remarks

1. Very simple method

Final remarks

1. Very simple method
2. Easy to implement (a few hundreds of lines of Octave)

Final remarks

1. Very simple method
2. Easy to implement (a few hundreds of lines of Octave)
3. Fast, but may lack precision w.r.t. sophisticated refinements of interval-based methods (even in the context of control synthesis)

Final remarks

1. Very simple method
2. Easy to implement (a few hundreds of lines of Octave)
3. Fast, but may lack precision w.r.t. sophisticated refinements of interval-based methods (even in the context of control synthesis)
4. Method can be adapted to guarantee reachability (instead of stability)

Final remarks

1. Very simple method
2. Easy to implement (a few hundreds of lines of Octave)
3. Fast, but may lack precision w.r.t. sophisticated refinements of interval-based methods (even in the context of control synthesis)
4. Method can be adapted to guarantee reachability (instead of stability)
5. Replacement of forward Euler's method by better numerical schemes
   (e.g.: backward Euler, Runge-Kutta of order 4)
   does not seem to yield significant gain (due to periodical tracking).

Final remarks

1. Very simple method
2. Easy to implement (a few hundreds of lines of Octave)
3. Fast, but may lack precision w.r.t. sophisticated refinements of interval-based methods (even in the context of control synthesis)
4. Method can be adapted to guarantee reachability (instead of stability)
5. Replacement of forward Euler's method by better numerical schemes (e.g.: backward Euler, Runge-Kutta of order 4) does not seem to yield significant gain (due to periodical tracking).
6. Several examples for which E.-based control synthesis
   - beats state-of-art interval-based control methods (e.g.: 4-room building ventilation)
   - fails/ is beaten by standard interval-based control methods (e.g.: DC-DC Boost converter)

Final remarks

1. Very simple method
2. Easy to implement (a few hundreds of lines of Octave)
3. Fast, but may lack precision w.r.t. sophisticated refinements of interval-based methods (even in the context of control synthesis)
4. Method can be adapted to guarantee reachability (instead of stability)
5. Replacement of forward Euler's method by better numerical schemes (e.g.: backward Euler, Runge-Kutta of order 4) does not seem to yield significant gain (due to periodical tracking).
6. Several examples for which E.-based control synthesis
   - beats state-of-art interval-based control methods (e.g.: 4-room building ventilation)
   - fails/ is beaten by standard interval-based control methods (e.g.: DC-DC Boost converter)
7. Deserves further experimentations...

Thanks!