Compositional Synthesis of State-Dependent Switching Control

Adrien Le Coënt[a], Laurent Fribourg[b], Nicolas Markey[b,c], Florian De Vuyst[a], Ludovic Chamoin[d]

[a] *CMLA, ENS Paris-Saclay, CNRS, Université Paris-Saclay, France*
[b] *LSV, ENS Paris-Saclay, CNRS, Inria, Université Paris-Saclay, France*
[c] *Univ. Rennes, CNRS, Inria, France*
[d] *LMT, ENS Paris-Saclay, CNRS, Université Paris-Saclay, France*

Abstract

We present a correct-by-design method of state-dependent control synthesis for sampled switching systems. Given a target region R of the state space, our method builds a capture set S and a control that steers any element of S into R. The method works by iterated backward *reachability* from R. The method is also used to synthesize a *recurrence control* that makes any state of R return to R infinitely often. We explain how the synthesis method can be performed in a *compositional* manner, and apply it to the synthesis of a compositional control of a concrete floor-heating system with 11 rooms and up to $2^{11} = 2048$ switching modes.

1. Introduction

Control of switching systems. The importance of switching systems has grown up considerably over the last years because of their ease of implementation for controlling cyber-physical systems. A switching system is a family of subsystems, each having its own dynamics, characterized by a parameter u whose values range over a finite set U (see [19]). However, when composing the subsystems together, the number of modes of the global system grows exponentially, and the dynamics may become very complex. It is therefore essential to design *compositional* analysis techniques in order to obtain effective control methods for switching systems with formal correctness guarantees.

In this paper, we give a symbolic compositional method that allows to synthesize a control of sampled switching systems that is guaranteed to satisfy *reachability*. It can also be applied to achieve *recurrence* property: any trajectory originating from R is controlled in order to make it return to R in bounded

This work was supported by ERC project EQualIS (308087), by European project Casting (601148), and by Institut Farman (CNRS FR 3311).

time. The method starts from a (rectangular) target region R of the state space. It then generates an increasing sequence of extended rectangles $\{R^{(i)}\}_{i\geq 0}$ such that any trajectory issued from $R^{(i)}$ can be controlled to reach $R^{(i-1)}$ in bounded time. This approach relies on simple operations (*extension* and *bisection*) over rectangular sets.

We explain how this basic method can be used in a *compositional* manner. The compositional procedure requires the system definition to be given under a *separate* form, and involves a way of *over-approximating* the state components, in order to make the control of the sub-systems independent of each other. Our compositional synthesis method is generally able to find a guaranteed control in a much shorter time than the classical centralized method. Sometimes, as exemplified here on the 11-rooms example that we develop, our compositional method is able to synthesize a guaranteed control while all known centralized approach fails by exhausting computational resources.

Related Work. In symbolic analysis and control-synthesis methods for hybrid systems, the method of backward reachability and the use of polyhedral symbolic states, as used here, is classical (see e.g. [6, 11]). The use of partitioning the state-space using bisection is also classical (see e.g. [14, 13]). The main original contribution of this paper is to give a simple technique of over-approximation, which allows one component to estimate the symbolic states of the other components, in presence of partial information. This is similar in spirit to an *assume-guarantee* (or *contract-based*) reasoning, where the controller synthesis for each sub-system assumes that some safety properties are satisfied by the other sub-systems [5, 7, 8, 9, 15, 21, 25, 26]. The present work is a continuation of [9]. In contrast to [9], we do not need, for the mode selection of a sub-system, to blindly explore all the possible modes selected by the other subsystem. This yields a drastic improvement in efficiency. This approach allows us to treat a real case study, which is intractable using a centralized approach. This case study comes from [18], and we use the same decomposition of the system into two parts. In contrast to the work of [18] which uses an on-line and heuristic approach with limited guarantees, we use here an off-line formal method which guarantees reachability and recurrence properties.

Implementation. In the discrete-time setting, with linear (or affine) mappings, the methods of control synthesis both in the centralized and in the compositional contexts have been integrated to the tool MINIMATOR [17, 10], written in Octave [24]. In the continuous-time setting (which also allows nonlinear flows), the methods have been integrated to the tool DynIBEX [2, 3], written in C++. All the computation times given in the paper have been performed on a 2.80GHz Intel Core i7-4810MQ CPU with 8GB of memory.

Plan. The structure of this paper is as follows. The centralized control approach is given in Section 2. The compositional approach is described in Section 3. The compositional approach is the applied in Section 4 on a real case study of temperature control in a building with 11 rooms and $2^{11} = 2048$ switching

modes of control. The method is extended in the continuous-time framework in Section 5.

2. Centralized Switching Control

We first consider the *centralized control* in the *discrete-time* setting. The time t then takes its values in $\mathbb{N}$.

2.1. Control modes and control patterns

We consider a discrete-time system under the form:

$$x(t+1) = f(x(t), u)$$

where t is in $\mathbb{N}$, x is a vector state variable, taking its values in $\mathbb{R}^n$, and u takes its value in the finite set $U = \{1, \ldots, N\}$ (called the set of *modes*).

It is often easier and/more efficient to slightly relax the control objective, by only checking the objective after several applications of f in a row. We are thus led to the notion of "control pattern": A *(control) pattern π of length k* is a sequence of size k of modes. The set of patterns of length k (resp. at most k) is denoted by Π^k (resp. $\Pi^{\leq k}$). For length $k = 1$, we have $\Pi^1 = U$.

For a system defined by $x(t+1) = f(x(t), u)$ and a pattern π of length k, one can recursively define $x(t+k) = f(x(t), \pi)$ with $\pi \in \Pi^k$, by:

1. $f(x(t), \pi) = f(x(t), u)$, if π is a pattern of length $k = 1$ of the form $u \in U$,
2. $f(x(t), \pi) = f(f(x(t), \pi'), u)$, if $\pi = u \cdot \pi'$ is a pattern of length $k \geq 2$ with $u \in U$ and $\pi' \in \Pi^{k-1}$.

In the following, we fix an upper bound $K \in \mathbb{N}$ on the length of patterns. The value of K can be seen as a maximum number of time steps, for which we compute the future behaviour of the system ("horizon").

We will achieve a desired global property P by using the following *state-dependent* control-synthesis method: at initial time $t_0 = 0$, select some pattern $\pi \in \Pi^{k_1}$ for some $1 \leq k_1 \leq K$, depending on the value of $x(t_0)$; at time $t_1 = t_0 + k_1$, select a new pattern in Π^{k_2} for some $1 \leq k_2 \leq K$, depending on $x(t_1)$, and so on indefinitely. In the centralized approach, the synthesis method consists in covering (part of) state space $\mathbb{R}^n$ with a set of sub-rectangles; for each sub-rectangle, a specific pattern is selected in order to achieve property P *locally* from any point in that sub-rectangle.

2.2. Bisection and extension of rectangles

A closed interval will be denoted by $[b]$ in order to denote the interval $[\underline{b}, \overline{b}]$, where $\underline{b}$ and $\overline{b}$ are two reals with $\underline{b} \leq \overline{b}$. An n-dimensional rectangle R is thus a Cartesian product of n intervals of the form $[b_1] \times [b_2] \times \cdots \times [b_n]$.

Given an interval $[b] = [\underline{b}, \overline{b}]$, the bisection of $[b]$ (of order 1), denotes the set of intervals $\{[\underline{b}, \frac{\underline{b}+\overline{b}}{2}], [\frac{\underline{b}+\overline{b}}{2}, \overline{b}]\}$. Given a rectangle $R = [b_1] \times \cdots \times [b_n]$, the bisection of r (of order 1) denotes the set $B_1 \cup \cdots \cup B_n$ where B_i denotes the

bisection (of order 1) of $[b_i]$, for $1 \leq i \leq n$. The bisection of the n-dimensional rectangle R is thus a set of 2^n n-dimensional rectangles of identical size.

By bisection of these 2^n n-dimensional rectangles, we obtain a bisection of R of order 2 which contains 2^{2n} n-dimensional rectangles. More generally, the bisection of R of order d is the set of 2^{dn} n-dimensional rectangles of identical size obtained by iterating the operation of bisection successively d times. Note that the union of the elements of a bisection (of order d with $d \geq 1$) is equal to R.

In the following, we fix an upper bound D for the bisection order d. The 2^{dn} rectangles of identical size resulting from the bisection of R at order d, are called "sub-rectangles" of R. They are denoted by r_i with $i \in I(n,d) = [1; 2^{dn}]$. We have $R = \bigcup_{i \in I(n,d)} r_i$.

Given a non-negative real $a \in \mathbb{R}_{\geq 0}$ and an n-dimensional rectangle $R = [b_1] \times [b_2] \times \cdots \times [b_n]$, the expression $R \pm a$ denotes the "extended" rectangle $[\underline{b}_1 - a, \overline{b}_1 + a] \times \cdots \times [\underline{b}_n - a, \overline{b}_n + a]$.[1] Given an extended rectangle $R \pm a$, the 2^{dn} sub-rectangles of $R \pm a$ obtained by bisection of $R \pm a$ at order d, will be denoted by r_i^a with $i \in I(n,d)$. We have $R \pm a = \bigcup_{i \in I(n,d)} r_i^a$.

2.3. Reachability procedure

The properties that we consider are essentially *reachability* properties: given a set S and a set R, we look for a control which steers any point x of S into R in a bounded number of steps. We also consider *recurrence* properties, requiring that once the state x of the system is in R at time t, the control will make it return to R (within a finite number of steps) infinitely often. Actually, we consider here a state set R given under the form of a rectangle, and we suppose that S is given under the extended form $R \pm a$ for some positive real $a > 0$. A typical value for a is $|R|/10$ or $|R|/100$, where $|R|$ denotes the length of the smallest side of the rectangle R. The method consists in finding a bisection of $R \pm a$ of order d such that all sub-rectangle of the form r_i^a (with $i \in I(n,d)$) can be mapped to R via some pattern π_i of length no more than K. Our problem is thus to find a control that makes any trajectory originating from a point of $R \pm a$ reach R within at most K steps. Given an n-dimensional rectangle R, and two positive reals a and ε, we formulate the (centralized) control synthesis problem as follows:

> Find $d \leq D$ (as small as possible) such that the property $Q(R, a, \varepsilon, d)$ defined below holds: for any $i \in I(n,d)$, there is a pattern $\pi_i \in \Pi^{\leq K}$
>
> 1. $f(r_i^a, \pi_i) \subseteq R$, and
> 2. $f(r_i^a, \pi_i') \subseteq R \pm (a + \varepsilon)$, for all nonempty prefix π_i' of π_i.

[1] Our technique could handle extended rectangles of R with *different* values of a for each dimension, and different values for the lower and upper extensions. For the sake of simplicity, we consider here extensions $R \pm a$ with the same value a of extension for all the dimensions of R.

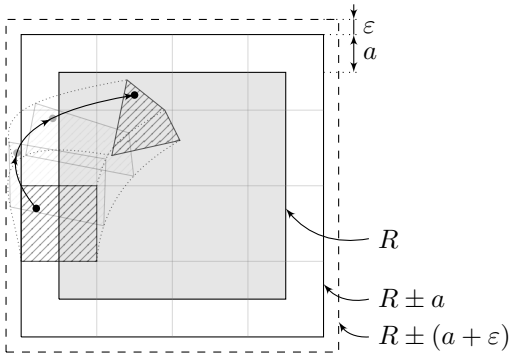

Figure 1: Schematic representation of Property $Q(R, a, \varepsilon, d)$

The first item requires that every trajectory originating from $r_i \pm a$ reaches R within K steps. The second item specifies that the intermediate states of the trajectory lie in $R \pm (a + \varepsilon)$. Intuitively, ε represents the width of an additional margin, around $R \pm a$, by which all the intermediate states are allowed to overflow along a pattern of length $\ell \leq K$ before reaching R. The property $Q(R, a, \varepsilon, d)$ is illustrated on Figure 1

Note that the patterns π_i can have different lengths. A stronger version of the problem consists in finding a set of patterns $\{\pi_i\}_{i \in I(n,d)}$ with the additional property that they all have the *same length* $\ell \leq K$. This is called the *synchronous control synthesis* problem. It is stated as follows:

> Find $d \leq D$ (as small as possible) and $0 < \ell \leq K$ (as small as possible) satisfying the property $P(R, a, \varepsilon, d, \ell)$ defined by: for any $i \in I(n, d)$, there is a pattern $\pi_i \in \Pi^\ell$ such that
>
> 1. $f(r_i^a, \pi_i) \subseteq R$, and
> 2. $f(r_i^a, \pi_i') \subseteq R \pm (a + \varepsilon)$, for all nonempty prefix π_i' of π_i.

In the following, we focus on the synchronous synthesis problem[2]; we solve it by designing a *generate-and-test* procedure $\mathsf{PROC}(R, a, \varepsilon)$, with fixed parameters D and K, which terminates with *failure* if $P(R, a, \varepsilon, d, \ell)$ fails to hold for and $0 \leq d \leq D$ and any $0 < l \leq K$, and terminates with *success* otherwise, with output (d^*, ℓ^*) defined by:

- $d^* = \min\{0 \leq d \leq D \mid \exists 0 < \ell \leq K.\ P(R, a, \varepsilon, d, \ell)\}$;
- $\ell^* = \min\{0 < \ell \leq K \mid P(R, a, \varepsilon, d^*, \ell)\}$.

2.4. Recurrence as a special case of reachability

The recurrence problem consists in finding a control that forces all trajectory starting from a given rectangle R to return to R within a finite number of steps.

[2]The "asynchronous" problem can be solved similarly, see [10].

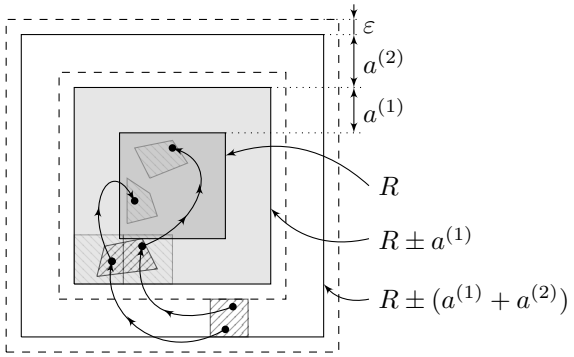

Figure 2: Iterated control of $R^{(1)} = R \pm a^{(1)}$ towards R, and $R^{(2)} = R^{(1)} \pm a^{(2)}$ towards $R^{(1)}$.

The problem can be solved by applying the procedure $\mathsf{PROC}(R, a, \varepsilon)$ in the limit case where $a = 0$: The procedure fails if, for all $0 \leq d \leq D$ and all $0 < \ell \leq K$, Property $P(R, 0, \varepsilon, d, \ell)$ does not hold; it succeeds if $P(R, 0, \varepsilon, d, \ell)$ holds for some d and ℓ. Property $P(R, 0, \varepsilon, d, \ell)$ expresses the fact that a controlled trajectory starting at any point $x \in R$ will return to R within at most K steps, without going too far away from R.

In the following, we focus on the "pure" reachability problem, and assume $a > 0$.

2.5. Iterated control synthesis

Given a rectangle R, and two positive reals $a^{(1)}$ and ε, suppose that the procedure $\mathsf{PROC}(R, a^{(1)}, \varepsilon)$ described in Section 2.3 succeeds with $(d^{(1)}, \ell^{(1)})$ as output. The control synthesis[3] $\mathsf{PROC}(R^1, a^{(2)}, \varepsilon)$ can then be applied with $R^{(1)} = R \pm a^{(1)}$ as input argument, and some positive constant $a^{(2)}$. By iteration, the application of $\mathsf{PROC}(R^{(i)}, a^{(i+1)}, \varepsilon)$, with $R^{(i)} = R^{(i-1)} \pm a^{(i)}$ thus induces at each step a control that steers $R^{(i)}$ to $R^{(i-1)}$, with $R^{(i)} = R + \Sigma_{j \leq i} a^{(j)}$. In the end, this synthesizes a control that steers $R^{(i)}$ to R via a sequence of i patterns $(\pi_j)_{1 \leq j \leq i}$. This is illustrated in Figure 2, for $i = 2$. The process can be iterated until $\mathsf{PROC}(R^{(i)}, a^{(i+1)}, \varepsilon)$ fails.

2.6. Complexity

For each sub-rectangle r_i of R and each $\pi \in \Pi^{\leq K}$, we have to solve the inclusion test $f(r_i, \pi) \subseteq R$. The resolution of such a test is a major objective of the method of *interval arithmetic* [23, 14]. When f is *affine*, the inclusion test can be done using the data structure of *zonotopes* [16, 12, 4] in $O(n^3)$ (cf. [10]). Since I may contain 2^{nD} elements (where D is the maximal depth of

[3] We assume that the procedure is re-applied with the same positive value for parameter ε for the sake of simplicity.

bisection), and $\Pi^{\leq K}$ contains $O(N^K)$ elements, the generate-and-test procedure $\mathsf{PROC}(R, a, \varepsilon)$ is in $O(2^{nD} \cdot N^K)$.

Example 1. *Let us consider the example of a two-room apartment, heated by one heater in each room (adapted from [13]). In this example, the objective is to control the temperature of both rooms. There is heat exchange between both rooms and with the environment. The* continuous *dynamics of the system is given by the equation:*

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

Here T_1 and T_2 are the temperatures of the rooms, and the state of the system corresponds to $T = (T_1, T_2)$. The control mode variable u_1 (respectively u_2) can take the values 0 or 1, depending whether the heater in room 1 (respectively room 2) is switched off or on (hence $U_1 = U_2 = \{0, 1\}$). Hence, here $n = 2$ and $N = 4$. The temperature T_e corresponds to the temperature of the environment, and T_f to the temperature of the heaters when turned on. The values of the parameters are as follows: $\alpha_{12} = 5 \times 10^{-2}$, $\alpha_{21} = 5 \times 10^{-2}$, $\alpha_{e1} = 5 \times 10^{-3}$, $\alpha_{e2} = 5 \times 10^{-3}$, $\alpha_f = 8.3 \times 10^{-3}$, $T_e = 10$ and $T_f = 35$.

We suppose that the heaters can be switched periodically at sampling instants τ, 2τ, ... (here, $\tau = 5$ minutes). By integration of the continuous dynamics between t and $t + \tau$, the system can be easily put under the discrete-time *form:*

$$\begin{pmatrix} T_1 \\ T_2 \end{pmatrix}(t+1) = f\left(\begin{pmatrix} T_1 \\ T_2 \end{pmatrix}(t), \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \right).$$

where f is an affine function.

Given a target rectangle for $T = (T_1, T_2)$ of the form $R = [18.5, 22] \times [18.5, 22]$, the control-synthesis problem is to find a rectangular capture set S (as large as possible) such that one can steer any state T of S to R ("reachability"), and then make any T of R return to R infinitely often ("recurrence").

Let $D = 1$ (the depth of bisection is at most 1), and $K = 4$ (the maximum length of patterns is 4). Let us look for a centralized controller which will steer all the trajectories starting at the rectangle $S = [18.5 - A, 22] \times [18.5 - A, 22]$ to R, then makes the trajectories return to R infinitely often. The procedure $\mathsf{PROC}(R^{(i)}, a^{(i)}, \varepsilon)$ is iterated successfully 15 times with the following inputs: $\varepsilon = 0.5$, and $a^{(i)}$ is set to the maximum value in $\{0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4\}$ for which $\mathsf{PROC}(R^{(i)}, a^{(i)}, \varepsilon)$ succeeds. In the end, we find $S = R \pm A$ with $A = \Sigma_{i=0}^{15} a^{(i)} = 53$, i.e. $S = [-35, 22] \times [-35, 22]$. Any element of S can thus be driven to R within 15 control patterns of length (at most) 4, i.e., within $15 \times 4 = 60$ units of time. Since each unit of time is of duration $\tau = 5$ minutes, any trajectory starting from S reaches R within $60 \times 5 = 300$ minutes. Furthermore, the recurrence property holds for R: Once the trajectory $x(t)$ has entered R, it returns to R under control patterns of length (at most) 4, i.e., every $4 \times 5 = 20$ minutes. Using our implementation, the iteration computation of PROC takes 4.14s of CPU time.

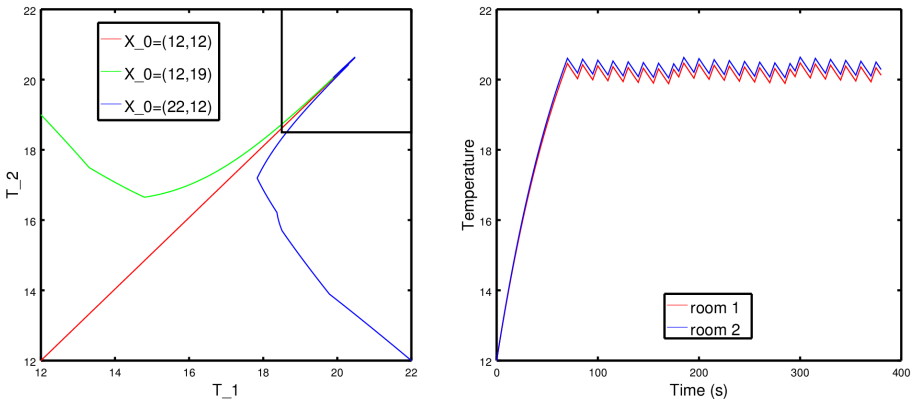

Figure 3: Simulations of the centralized reachability controller for three different initial conditions plotted in the state space plane (left); simulation of the centralized reachability controller for the initial condition $(12, 12)$ plotted within time (right).

These results are consistent with the simulation given in Figure 3 for the time evolution of (T_1, T_2) *starting from* $(12, 12)$*. Simulations of the control, starting from* $(T_1, T_2) = (12, 19)$ *and* $(T_1, T_2) = (22, 12)$ *are also given in the state space plane in Figure 3.*

3. Compositional Control

3.1. Separate systems

From now on, we suppose that the discrete-time system:

$$x(t+1) = f(x(t), u),$$

where x is a vector state variable of dimension n and u is a mode in $U = \{1, \ldots, N\}$, can be expressed under the *separate form*:

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

where x_1 (resp. x_2) is the first (resp. second) component of the state vector x, and takes its values in $\mathbb{R}^{n_1}$ (resp. $\mathbb{R}^{n_2}$), and where u_1 (resp. u_2) is the first (resp. second) component of the control *mode*, and takes its values in the *finite* set U_1 (resp. U_2). We will often write x for (x_1, x_2), u for (u_1, u_2), and n for $n_1 + n_2$. We will also abbreviate the set $U_1 \times U_2$ as U. Let N_1 (resp. N_2) by the cardinality of U_1 (resp. U_2), so that $N = N_1 \cdot N_2$ is the cardinality of U.

For $k \geq 1$, we denote by Π_1^k (resp. Π_2^k) the set of sequences of k elements of U_1 (resp. U_2). Given two positive integers K_1 and K_2, we denote by $\Pi_1^{\leq K_1}$ (resp. $\Pi_2^{\leq K}$) the set $\bigcup_{1 \leq k \leq K_1} \Pi_1^k$ (resp. $\bigcup_{1 \leq k \leq K_2} \Pi_2^k$). A pattern $u \in \Pi^k$ is thus of the form $(u_1^1, u_2^1) \cdots (u_1^k, u_2^k)$ with $(u_1^1 \cdots u_1^k) \in \Pi_1^k$ and $(u_2^1 \cdots u_2^k) \in \Pi_2^k$.

We denote with $(f(x,\pi))_{|1} \in \mathbb{R}^{n_1}$ and $(f(x,\pi))_{|2} \in \mathbb{R}^{n_2}$ the first and second components of $f(x,\pi) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} = \mathbb{R}^n$ respectively. We have: $f(x,\pi) = ((f(x,\pi))_{|1}, (f(x,\pi))_{|2})$.

Finally, we suppose that the target rectangle R is of the form $R_1 \times R_2$: R is a product of n closed intervals of reals, and R_1 (resp. R_2) is a product of n_1 (resp. n_2) closed intervals of reals. Then $R_1 \pm a$ (resp. $R_2 \pm a$) denotes the product of n_1 (resp. n_2) extended intervals (and we have $R \pm a = (R_1 \pm a) \times (R_2 \pm a)$).

3.2. Compositional synthesis

We consider a discrete-time system

$$x(t+1) = f(x(t), u)$$

that can be written in separate form as

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

Our basic idea for synthesizing a control in a *compositional manner* is to find a procedure of the form $\mathsf{PROC}_1(R_1, a, \varepsilon)$ which outputs a pair (d_1, ℓ_1), and a procedure of the form $\mathsf{PROC}_2(R_2, a, \varepsilon)$ which outputs a pair (d_2, ℓ_2), such that

1. for all $i_1 \in I(n_1, d_1)$, there exists $\pi_1 \in \Pi_1^{\ell_1}$ such that for all $x_2 \in R_2 \pm a$, there exists $\pi_2 \in \Pi_2^{\ell_1}$ such that for all prefix π_1' (resp. π_2') of π_1 (resp. π_2),

$$(f(r_{i_1}^a \times \{x_2\}, (\pi_1', \pi_2')))_{|1} \subseteq R_1 \pm (a + \varepsilon)$$
$$(f(r_{i_1}^a \times \{x_2\}, (\pi_1, \pi_2)))_{|1} \subseteq R_1.$$

2. for all $i_2 \in I(n_2, d_2)$, there exists $\pi_2 \in \Pi_2^{\ell_2}$ such that for all $x_1 \in R_1 \pm a$, there exists $\pi_1 \in \Pi_1^{\ell_2}$ such that for all prefix π_1' (resp. π_2') of π_1 (resp. π_2),

$$(f(\{x_1\} \times r_{i_2}^a, (\pi_1', \pi_2')))_{|2} \subseteq R_2 \pm (a + \varepsilon)$$
$$(f(\{x_1\} \times r_{i_2}^a, (\pi_1, \pi_2)))_{|2} \subseteq R_2.$$

where $(r_{i_k}^a)_{i_k \in I(n_k, d_k)}$ (for $k \in \{1, 2\}$) are the sub-rectangles resulting from the bisection of $R_k \pm a$ at order d_k, and the sets of the form $(f(r_{i_1}^a \times \{x_2\}, (\pi_1', \pi_2')))_{|1}$ stand for the corresponding sets $\{(f((x_1, x_2), (\pi_1', \pi_2')))_{|1} \mid x_1 \in r_{i_1} \pm a\}$.

The partition $I(n_1, d_1)$ (resp. $I(n_2, d_2)$) thus induces a control that ensures that x_1 (resp. x_2) will reach R_1 (resp. R_2) every ℓ_1 (resp. ℓ_2) steps. Hence the repeated application of the concurrent control on R_1 and R_2 ensures that (x_1, x_2) will reach $R_1 \times R_2$ within at most ℓ steps, where ℓ is the least common multiple of ℓ_1 and ℓ_2. We have:

$$\forall (x_1, x_2) \in (R_1 \pm a) \times (R_2 \pm a).\ \exists \pi \in \Pi^{\leq K_1 \times K_2}.$$
$$f(x_1, x_2, \pi') \subseteq (R_1 \pm (a + \varepsilon)) \times (R_2 \pm (a + \varepsilon)) \qquad \forall \pi' \text{ prefix of } \pi.$$
$$f(x_1, x_2, \pi) \subseteq R_1 \times R_2.$$

The concurrent application of the controls induced by PROC_1 and PROC_2 is thus analogous to the direct application of a *centralized* procedure PROC. The advantage of the compositional approach is that the complexity of PROC_1 and PROC_2 is much lower than the complexity of the centralized procedure PROC (see Section 2.6). The existence of such a compositional approach however requires that the first and second components x_1 and x_2 be "weakly interdependent": the existence of pattern π_1 has to depend on i_1 only (regardless the value of i_2), and, symmetrically the existence of pattern π_2 has to only depend on i_2 (regardless the value of i_1). In order to formalize this notion of "weak interdependence", we define an *approximation* $X_{i_1}(a, \pi_1)$ of the first component of $f(r^a_{i_1} \times r^a_{i_2}, (\pi_1, \pi_2))$ depending on i_1 and π_1 only, and symmetrically, an *approximation* $X_{i_2}(a, \pi_2)$ of the second component of $f(r^a_{i_1} \times r^a_{i_2}, (\pi_1, \pi_2))$ depending on i_2 and π_2 only.

Let π_1^k (resp.π_2^k) denote the prefix of length k of π_1 (resp. π_2), and $\pi_1(k)$ (resp. $\pi_2(k)$) the k-th element of pattern π_1 (resp. π_2).

Definition 1. *Consider an index $i_1 \in I(n_1, d_1)$ (resp. $i_2 \in I(n_2, d_2)$) and a pattern $\pi_1 \in \Pi_1^{\ell_1}$ (resp. $\pi_2 \in \Pi_2^{\ell_2}$). The* approximate first-component (resp. second-component) sequence $(X^k_{i_1}(a, \pi_1))_{0 \le k \le \ell_1}$ *(resp.* $(X^k_{i_2}(a, \pi_2))_{0 \le k \le \ell_2}$*) is defined as follows:*

- $X^0_{i_1}(a, \pi_1) = r^a_{i_1}$ *(resp.* $X^0_{i_2}(a, \pi_2) = r^a_{i_2}$*) and*
- $X^k_{i_1}(a, \pi_1) = f_1(X^{k-1}_{i_1}(a, \pi_1), R_2 \pm (a + \varepsilon), \pi_1(k))$ *for* $1 \le k \le \ell_1$ *(resp.* $X^k_{i_2}(a, \pi_2) = f_2(R_1 \pm (a + \varepsilon), X^{k-1}_{i_2}(a, \pi_2), \pi_2(k))$ *for* $1 \le k \le \ell_2$*).*

Definition 2. *We refine Property $P(R, a, \varepsilon, d, \ell)$ as follows: for $0 \le d_1 \le D_1$ and $0 < \ell_1 \le K_1$, we define $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ as: for any $i_1 \in I(n_1, d_1)$, there exists a pattern $\pi_1 \in \Pi_1^{\ell_1}$ such that*

1. $X^{\ell_1}_{i_1}(a, \pi_1) \subseteq R_1$*, and*
2. $X^k_{i_1}(a, \pi_1) \subseteq R_1 \pm (a + \varepsilon)$ *for* $1 \le k \le \ell_1 - 1$*.*

Similarly, we define $P_2(R_2, a, \varepsilon, d_2, \ell_2)$ by: for all $i_2 \in I(n_2, d_2)$, there exists $\pi_2 \in \Pi_2^{\ell_2}$ such that

1. $X^{\ell_2}_{i_2}(a, \pi_2) \subseteq R_2$*, and*
2. $X^k_{i_2}(a, \pi_2) \subseteq R_2 \pm (a + \varepsilon)$ *for* $1 \le k \le \ell_2 - 1$*.*

In the sequel, we write I_1 and I_2 for the sets $I(n_1, d_1)$ and $I(n_2, d_2)$, respectively. When $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ (resp. $P_2(R_2, a, \varepsilon, d_2, \ell_2)$) holds, we write $\pi^1_{i_k}$ (resp. $\pi^2_{i_k}$) for an element $\pi_1 \in \Pi^{\ell_1}$ (resp. $\pi_2 \in \Pi^{\ell_2}$) satisfying both conditions of $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ (resp. $P_2(R_2, a, \varepsilon, d_2, \ell_2)$).

Figure 4 illustrates property $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ for $\ell_1 = 2$, with a pattern $\pi_1 = u_1 \cdot v_1$, and a given sub-rectangle $r^a_{i_1}$. Property $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ holds here because $X^1_{i_1}(a, \pi_1) \subseteq R_1 \pm (a + \varepsilon)$ and $X^2_{i_1}(a, \pi_1) \subseteq R_1$.

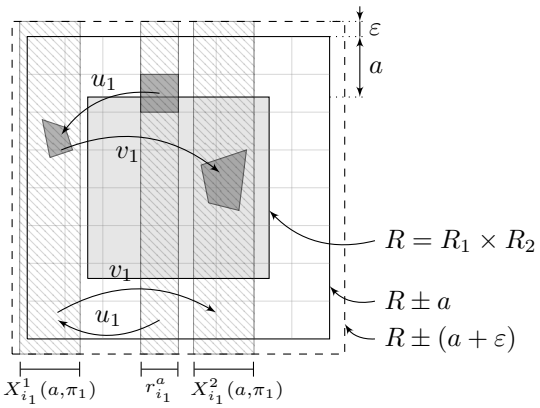

Figure 4: Illustration of $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ with $\ell_1 = 2$, and $\pi_1 = u_1 \cdot v_1$. The dark grey squares correspond to the centralized case, where both dimensions are controlled. The hatched strips represent the compositional case, where only one dimension is controlled, and all possible behaviours are considered for the second dimension.

Lemma 3. *Suppose that $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ and $P_2(R_2, a, \varepsilon, d_2, \ell_2)$ both hold. Then for all $i_1 \in I_1$ and $i_2 \in I_2$, and all $1 \leq k \leq \min\{\ell_1, \ell_2\}$, it holds*

$$(f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2})))_{|1} \subseteq X^k_{i_1}(a, \pi_{i_1})$$
$$(f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2})))_{|2} \subseteq X^k_{i_2}(a, \pi_{i_2})$$

Proof. For $m \in \{1, 2\}$, write $P^m_{i_1,i_2}(k)$ for the property that is always true when $k > \min\{\ell_1, \ell_2\}$, and that is defined as

$$(f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2})))_{|m} \subseteq X^k_{i_m}(a, \pi_{i_m})$$

for $k \leq \min\{\ell_1, \ell_2\}$.

Assuming that both $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ and $P_2(R_2, a, \varepsilon, d_2, \ell_2)$ hold, we show by induction on k that

$$\forall i_1 \in I_1.\ \forall i_2 \in I_2.\ P^1_{i_1,i_2}(k)\ \wedge\ P^2_{i_1,i_2}(k).$$

When $k = 1$, by definition of the separate form, we have

$$(f((r^a_{i_1} \times r^a_{i_2}), (\pi^1_{i_1}, \pi^1_{i_2})))_{|1} = f_1(r^a_{i_1}, r^a_{i_2}, \pi_{i_1}(1)),$$

and by definition of $X^1_{i_1}(a, \pi_{i_1})$,

$$\begin{aligned} X^1_{i_1}(a, \pi_{i_1}) &= f_1(X^0_{i_1}(a, \pi_{i_1}), R_2 \pm (a + \varepsilon), \pi_{i_1}(1)) \\ &= f_1(r^a_{i_1}, R_2 \pm (a + \varepsilon), \pi_{i_1}(1)). \end{aligned}$$

Since $r^a_{i_2} \subseteq R_2 \pm (a + \varepsilon)$, we have $(f((r^a_{i_1} \times r^a_{i_2}), (\pi^1_{i_1}, \pi^1_{i_2})))_{|1} \subseteq X^1_{i_1}(a, \pi_{i_1})$, for any $i_1 \in I_1$ and $i_2 \in I_2$. The proof that $(f((r^a_{i_1} \times r^a_{i_2}), (\pi^1_{i_1}, \pi^1_{i_2})))_{|2} \subseteq X^1_{i_2}(a, \pi_{i_2})$ is similar.

Assuming that our induction hypothesis holds for some integer $k \geq 1$, we prove it at step $k+1$. First, if $k+1 > \min\{\ell_1, \ell_2\}$, the result is trivial. Now, assume $k+1 \leq \min\{\ell_1, \ell_2\}$ (hence also $k \leq \min\{\ell_1, \ell_2\}$). We have

$$\begin{aligned}(f((r^a_{i_1} \times r^a_{i_2}), (\pi^{k+1}_{i_1}, \pi^{k+1}_{i_2})))_{|1} &= \big(f(f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2})), \\ &\qquad (\pi_{i_1}(k+1), \pi_{i_2}(k+1)))\big)_{|1} \\ &= f_1\big([f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|1}, \\ &\qquad [f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|2}, \pi_{i_1}(k+1)\big) \\ &\subseteq X^{k+1}_{i_1}(a, \pi_{i_1}).\end{aligned}$$

The last inclusion can be shown as follows: first note that $[f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|1} \subseteq X^k_{i_1}(a, \pi_{i_1})$ by induction hypothesis. Moreover, we have:

$$[f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|2} \subseteq R_2 \pm (a+\varepsilon)$$

because $[f(r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|2} \subseteq X^k_{i_2}(a, \pi_{i_1})$ (which follows from our induction hypothesis), and $X^k_{i_2}(a, \pi_{i_1}) \subseteq R_2 \pm (a+\varepsilon)$ (by assumption $P_2(R_2, a, \varepsilon, d_2, \ell_2)$).

Hence

$$\begin{aligned}f_1([f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|1}, [f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))]_{|2}, \pi_{i_1}(k+1)) \\ \subseteq f_1(X^{k-1}_{i_1}(a, \pi_{i_1}), R_2 \pm (a+\varepsilon), \pi_{i_1}(k+1)) = X^{k+1}_{i_1}(a, \pi_{i_1})\end{aligned}$$

The proof for $P^2_{i_1,i_2}(k+1)$ is similar. □

Theorem 4. *Suppose that $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ and $P_2(R_2, a, \varepsilon, d_2, \ell_2)$ both hold, and that $\ell_1 \leq \ell_2$ (the other case being symmetric). Then for any $x_1 \in R_1 \pm a$, there exists $\pi_1 \in \Pi_1^{\ell_2}$ such that for any $x_2 \in R_2 \pm a$, there exists $\pi_2 \in \Pi_2^{\ell_2}$ such that*

$$\begin{aligned}&(f(x_1, x_2, (\pi^k_1, \pi^k_2)))_{|1} \in R_1 \pm (a+\varepsilon) && \text{for all } 1 \leq k \leq \ell_2 \\ &(f(x_1, x_2, (\pi^k_1, \pi^k_2)))_{|1} \in R_1 && \text{for all } k \in \ell_1 \cdot \mathbb{N},\ 1 \leq k \leq \ell_2 \\ &(f(x_1, x_2, (\pi^k_1, \pi^k_2)))_{|2} \in R_2 \pm (a+\varepsilon) && \text{for all } 1 \leq k \leq \ell_2 \\ &(f(x_1, x_2, (\pi_1, \pi_2)))_{|2} \in R_2\end{aligned}$$

Proof. By application of Lemma 3 and Property $P_1(R_1, a, \varepsilon, d_1, \ell_1)$, we have:

$$\begin{aligned}x_1(k) \in (f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2})))_{|1} \subseteq R_1 \pm (a+\varepsilon) \qquad \text{for } k \leq \ell_1 - 1 \\ x_1(\ell_1) \in (f((r^a_{i_1} \times r^a_{i_2}), (\pi^{\ell_1}_{i_1}, \pi^{\ell_1}_{i_2})))_{|1} \subseteq R_1\end{aligned}$$

In case $\ell_1 = \ell_2$, Lemma 3 and $P_2(R_2, a, \varepsilon, d_2, \ell_2)$ give the symmetric inclusions on the other dimension, and the theorem holds. If $\ell_2 > \ell_1$, we write $\ell_2 = \alpha \cdot \ell_1 + \beta$ with $0 \leq \beta < \ell_1$. Lemma 3 and $P_2(R_2, a, \varepsilon, d_2, \ell_2)$ give

$$x_2(k) \in (f((r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2})))_{|2} \subseteq R_2 \pm (a+\varepsilon) \qquad \text{for } k \leq \ell_1$$

In order to prove the last two properties, we have to prove $x_2(k) \in R_2 \pm (a+\varepsilon)$ for $\ell_1 + 1 \leq k \leq \ell_2 - 1$, and $x_2(k) \in R_2$ for $k = \ell_2$.

Since $x_1(\ell_1) \in R_1$, we have $x_1(\ell_1) \in r_{i'_1}$ for some $i'_1 \in I_1$. Let us apply $\pi_{i'_1}$ to $x_1(\ell_1)$. Since $x_2(\ell_1) \in R_2 \pm (a+\varepsilon)$, we have $x_1(\ell_1+1) \in X^1_{i'_1}(a, \pi_{i'_1})$; moreover, $X^1_{i'_1}(a, \pi_{i'_1}) \subseteq R_1 \pm (a+\varepsilon)$ by $P_1(R_1, a, \varepsilon, d_1, \ell_1)$. It follows: $x_1(\ell_1+1) \in R_1 \pm (a+\varepsilon)$. This implies $x_2(\ell_1+1) \in X^{\ell_1+1}_{i_1}(a, \pi_{i'_1})$. Now, since $X^{\ell_1+1}_{i_1}(a, \pi_{i'_1}) \subseteq R_2 \pm (a+\varepsilon)$ (by $P_2(R_2, a, \varepsilon, d_2, \ell_2)$), we have: $x_2(\ell_1+1) \in R_2 \pm (a+\varepsilon)$.

By iterating this argument $k \leq \ell_1$ times (as long as $\ell_1 + k \leq \ell_2$), we have: $x_1(\ell_1 + k) \in R_1 \pm (a+\varepsilon)$ if $k \leq \ell_1 - 1$, and $x_1(2\ell_1) \in R_1$ on the one hand; $x_2(\ell_1 + k) \in R_2 \pm (a+\varepsilon)$ for $1 \leq k \leq \ell_1$ on the other hand.

By iterating the same reasoning α times, we have: $x_1(k) \in R_1 \pm (a+\varepsilon)$ for $1 \leq k \leq \alpha\ell_1 - 1$, and $x_1(\alpha\ell_1) \in R_1$ on the one hand; $x_2(\ell_1 + k) \in R_2 \pm (a+\varepsilon)$ for $1 \leq k \leq \alpha\ell_1$.

Since $x(\alpha\ell_1) \in R_1$, we have $x(\alpha\ell_1) \in r_{i^\alpha_1}$ for some $i^\alpha_1 \in I_1$. By application of (the prefix of) $\pi_{i^\alpha_1}$, we have: $x_1(k) \in R_1 \pm (a+\varepsilon)$ and $x_2(k) \in R_2 \pm (a+\varepsilon)$ for $\alpha\ell_1 + 1 \leq k \leq \alpha\ell_1 + \beta = \ell_2$, and $x_2(\alpha\ell_1 + \beta) = x_2(\ell_2) \in R_2$. □

As in the centralized case (see Section 2), it is easy to construct a *generate-and-test* procedure $\mathsf{PROC}_1(R_1, a, \varepsilon)$ that terminates with *failure* if:

$$\forall 0 \leq d_1 \leq D.\ \forall 1 \leq \ell_1 \leq K_1.\ \neg P_1(R_1, a, \varepsilon, d_1, \ell_1),$$

and terminates with *success* otherwise, then outputing (d^*_1, ℓ^*_1) defined by:

$$d^*_1 = \min\{d_1 \in \{0, 1, \ldots, D\} \mid \exists \ell_1 \in \{1, \ldots, K_1\}.\ P_1(R_1, a, \varepsilon, d_1, \ell_1)\}$$
$$\ell^*_1 = \min\{\ell_1 \in \{1, \ldots, K_1\}.\ P_1(R_1, a, \varepsilon, d^*_1, \ell_1)\}.$$

Likewise, we can construct a *generate-and-test* procedure $\mathsf{PROC}_2(R_2, a, \varepsilon)$ which terminates with *failure* if

$$\forall 0 \leq d_2 \leq D.\ \forall 1 \leq \ell_2 \leq K_2.\ \neg P_2(R_2, a, \varepsilon, d_2, \ell_2)$$

and terminates with *success* otherwise, with an output (d^*_2, ℓ^*_2) defined similarly.

Using the same reasoning as in the centralized case (see Section 2), one can see that the complexity of $\mathsf{PROC}_1(R_1, a, \varepsilon)$ and $\mathsf{PROC}_2(R_2, a, \varepsilon)$ are in $O(2^{n_1 D} N_1^{K_1})$ and $O(2^{n_2 D} N_2^{K_2})$ respectively. This generally yields a drastic cut down with respect to the complexity of the centralized approach, which is in $O(2^{nD} N^K)$ with $n = n_1 + n_2$, $N = N_1 N_2$ and $K \geq \max\{K_1, K_2\}$.

Theorem 4 allows us to implement the method as far as we are able to compute the results of applying mappings f_1 and f_2 to symbolic states represented by rectangles. When f_1 and f_2 are affine, the results can be easily computed using the data structure of "zonotopes" [12]. The method has been implemented in the case of affine mappings, using the system MINIMATOR [17, 10].

Remark 1. *In the compositional context, the selection of an appropriate value for ε is for the moment performed by hand, and is the result of a compromise: if ε is too small, then $f_1(r_{i_1}, R_2, \pi_1(1)) \subseteq R_1 \pm \varepsilon$ for no $\pi_1 \in \Pi^{\ell_1}$; if ε is too large, then $f_1(X^{\ell_1}_{i_1}, R_2 \pm \varepsilon, \pi_1(\ell_1)) \subseteq R_1$ for no $\pi_1 \in \Pi^{\ell_1}$.*

Example 2. *Consider again the specification of a two-room apartment given in Example 1. Notice that it can be written in separate form as*

$$T_1(t+1) = f_1(T_1(t), T_2(t), u_1) \qquad T_2(t+1) = f_2(T_1(t), T_2(t), u_2)$$

We consider the compositional control synthesis problem where the first (resp. second) state component corresponds to the temperature of the first (resp. second) room T_1 (resp. T_2), and the first (resp. second) control mode component corresponds to the heater u_1 (resp. u_2) of the the first (resp. second) room.

Set $R = R_1 \times R_2 = [18.5, 22] \times [18.5, 22]$. Let $D = 3$ (the depth of bisection is at most 3), and $K_1 = K_2 = 10$ (the maximum length of patterns is 10). The parameter ε is set to value $1.5°C$. We look for a compositional controller which steers any temperature state in $S = S_1 \times S_2 = [18.5 - a, 22] \times [18.5 - a, 22]$ to R, then makes the trajectories return to R infinitely often.

Using our implementation, the computation of the control synthesis takes 220s of CPU time. The method iterates 8 times the control synthesis procedure $\mathsf{PROC}(R, a, \varepsilon)$ with $\varepsilon = 1.5$, and $a \in \{0.5, 1\}$. We find $S = [18.5 - A, 22] \times [18.5 - A, 22]$ with $A = \Sigma_i a^{(i)} = 6.5$, i.e. $S = [12, 22] \times [12, 22]$. This means that any element of S can be driven to R within 8 control patterns of length (at most) 10, i.e., within $8 \times 10 = 80$ units of time. Since each unit of time is of duration $\tau = 5$ minutes, any trajectory starting from S reaches R within $80 \times 5 = 400$ minutes. The trajectory is then guaranteed to always stay (at each discrete time t) in $R \pm \varepsilon = [17, 23.5] \times [17, 23.5]$.

These results are consistent with the simulation given in Figure 5 showing the time evolution of (T_1, T_2) starting from $(12, 12)$. Simulations of the control are also given in the state-space plane, in Figure 5, for initial states $(T_1, T_2) = (12, 12)$, $(T_1, T_2) = (12, 19)$ and $(T_1, T_2) = (22, 12)$.

Not surprisingly, the performance guaranteed by the compositional approach ($a = 6.5$, reachability of R in 400 minutes) are worse than those guaranteed by the centralized approach of Example 1 ($a = 53.5$, reachability of R in 300 minutes). On the other hand, the CPU computation time in the compositional approach (220s) is here worse than the CPU time of the centralized approach (4.14s). This relative inefficiency is due to the small size of the example, and the stronger properties of decentralized control.

4. Case Study

This case study, proposed by the Danish company Seluxit, aims at controlling the temperature of an eleven rooms house, heated by geothermal energy. The *continuous* dynamics of the system is the following:

$$\frac{d}{dt}T_i(t) = \sum_{j=1}^{n} A^d_{i,j}(T_j(t) - T_i(t)) + B_i(T_{env}(t) - T_i(t)) + H^v_{i,j}.v_j \qquad (1)$$

Variables T_i represent the temperatures of the rooms. The matrix A^d contains the heat transfer coefficients between the rooms, while matrix B contains

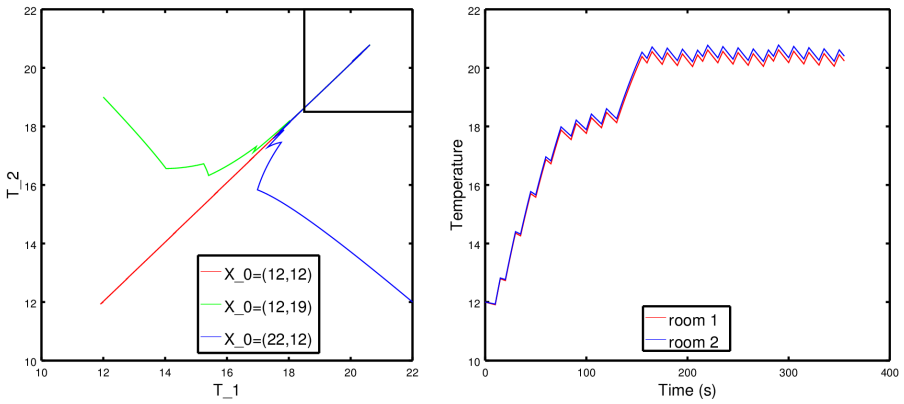

Figure 5: Simulations of the compositional reachability controller for three different initial conditions plotted in the state space plane (left); simulation of the compositional reachability controller for the initial condition $(12, 12)$ plotted within time (right).

the heat transfer coefficients between the rooms and the external temperature (this temperature is set to $T_{env} = 10°C$ for the computations). The control matrix H^v contains the effects of the control on the room temperatures, and the control variable is here denoted by v_j. We have $v_j = 1$ (resp. $v_j = 0$) if the heater in room j is turned on (resp. turned off). We thus have $n = 11$ and $N = 2^{11} = 2048$ switching modes.

Note that the matrix A^d is parameterized by the open of closed state of the doors in the house. In our case, the average between closed and open matrices was taken for the computations. The exact values of the coefficients are given in [18]. The controller has to select which heater(s) to turn on in the eleven rooms. Due to a limitation of the capacity supplied by the geothermal device, the 11 heaters cannot be running at the same time. In our case, we limit to 4 the number of heaters that can be running at the same time.

We consider the compositional control synthesis problem where the first (resp. second) state component corresponds to the temperatures of rooms 1 to 5 (resp. 6 to 11), and the first (resp. second) control mode component corresponds to the heaters of rooms 1 to 5 (resp. 6 to 11). Hence $n_1 = 5$, $n_2 = 6$, $N_1 = 2^5$, and $N_2 = 2^6$. We impose that at most two heaters are switched on at the same time in the first sub-system, and at most two in the second sub-system.

Let $D = 1$ (the bisection depth is at most 1), and $K_1 = K_2 = 4$ (the maximum length of patterns is 4). The parameter ε is set to value $0.5°C$. The sampling time is $\tau = 15$ minutes.

We look for a compositional controller which steers any temperature state in the rectangle $S = [18 - a, 22]^{11}$ to $R = [18, 22]^{11}$, with a as large as possible, ad then makes the temperature trajectories return to R infinitely often. Using our implementation, the computation of the control synthesis takes around 20 hours of CPU time. The method iterates the control synthesis procedure $\mathsf{PROC}(R, a, \varepsilon)$ 15 times, with $a \in \{0.1, 0.2, 0.3\}$ and $\varepsilon = 0.5$. We find

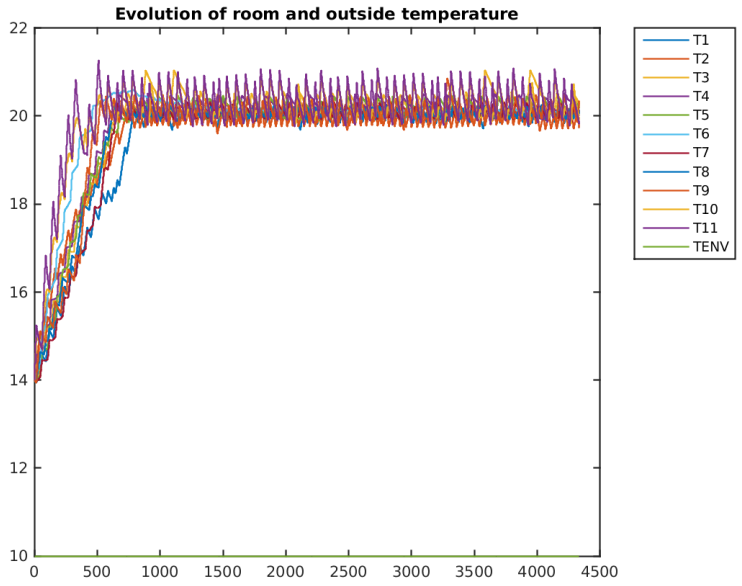

Figure 6: Simulation of the Seluxit case study plotted with time (in min) for $T_{env} = 10°C$.

$S = [18 - A, 22]^{11}$ with $A = \Sigma_i a^{(i)} = 4.2$, i.e. $S = [13.8, 22]^{11}$. This means that any element of S can be driven into R within 15 control patterns of length (at most) 4, i.e., within $15 \times 4 = 60$ units of time. Since each time unit is of duration $\tau = 15$ minutes, any trajectory starting from S reaches R within $60 \times 15 = 900$ minutes. The trajectory is then guaranteed to stay in $R \pm \varepsilon = [17.5, 22.5]^{11}$. These results are consistent with the simulation given in Figure 6 showing the time evolution of the temperature of the rooms, starting from 14^{11}.

4.1. Robustness Experiments

We performed the same simulations as in Figure 6, except that the environment temperature is not fixed at 10°C but follows scenarios of soft winter (Figure 7) and spring (Figure 8). The environment temperature is plotted in green in the figures. The spring scenario is taken from [18], and the soft winter scenario is the winter scenario of [18] increased by 5 degrees. We see that our controller, which is designed for $T_{env} = 10°C$ still satisfies the properties of reachability and recurrence. These simulations are very close those obtained in [18].

5. Continuous-time case

In this section, we consider the case of continuous-time differential equations. The time t now takes value in $\mathbb{R}_{\geq 0}$.

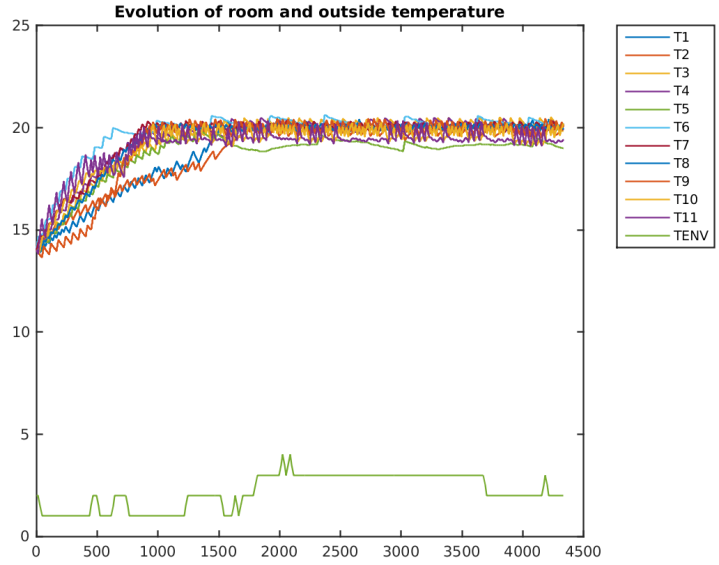

Figure 7: Simulation of the Seluxit case study in the soft winter scenario.

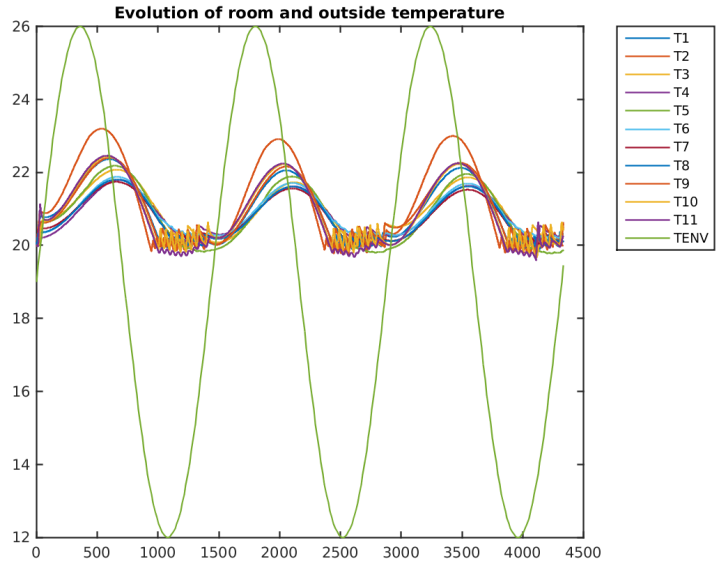

Figure 8: Simulation of the Seluxit case study in the spring scenario.

5.1. Reachability in continuous time

Consider the continuous-time system with *finite control*:

$$\dot{x}_1(t) = f_1(x_1(t), x_2(t), u_1) \tag{2}$$
$$\dot{x}_2(t) = f_2(x_1(t), x_2(t), u_2) \tag{3}$$

where x_1 (resp. x_2) is the first (resp. second) component of the state vector variable, taking its values in $\mathbb{R}^{n_1}$ (resp. $\mathbb{R}^{n_2}$), and where u_1 (resp. u_2) is the first (resp. second) component of the control *mode*, taking its values in the *finite* set U_1 (resp. U_2). We still use notations of the previous sections, writing e.g. x for (x_1, x_2) and u for (u_1, u_2). We abbreviate the continuous-time system under the form:

$$\dot{x}(t) = f(x(t), u) \tag{4}$$

where x is a vector state variable taking its values in $\mathbb{R}^n = \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$, and where u is of the form (u_1, u_2), with u_1 taking its values in U_1 and u_2 in U_2. We assume that, given an initial value x_0, Equation (4) has a solution (e.g., assuming that the vector field f (resp. f_1, f_2) is Lipschtiz).

We define the reachable set of (4) from a set of initial states X_0, at time t ($0 \leq t \leq \tau$) under control mode u:

$$\mathit{Reach}_f(t, X_0, u) = \{\Phi(t, x_0, u) \mid x_0 \in X_0\}.$$

where $\Phi(t, x, u)$ denotes the state $x(t)$ reached at time t ($0 \leq t \leq \tau$) starting from the initial state x, under control mode $u \in U$.

We define the reachable set of (2) from a set of initial states $X_1 \subseteq \mathbb{R}^{n_1}$, at time t ($0 \leq t \leq \tau$) under control mode $u_1 \in U_1$ and perturbation $X_2 \subseteq \mathbb{R}^{n_2}$:

$$\mathit{Reach}_{f_1}(t, X_1, X_2, u_1) = \{\Phi_1(t, x_1, X_2, u_1) \mid x_1 \in X_1\}.$$

where $\Phi_1(t, x_1, X_2, u_1)$ is the set of states $x_1(t)$ reached at time t ($t \geq 0$) from the initial state x_1, under control mode u_1 and perturbation X_2.

Symmetrically, we define the reachable set of (3) from a set of initial states $X_2 \subseteq \mathbb{R}^{n_2}$, at time t ($0 \leq t \leq \tau$) under control mode $u_2 \in U_2$ and perturbation $X_1 \subseteq \mathbb{R}^{n_1}$: x

$$\mathit{Reach}_{f_2}(t, X_1, X_2, u_2) = \{\Phi_2(t, X_1, x_2, u_2) \mid x_2 \in X_2\}.$$

where $\Phi_2(t, X_1, x_2, u_2)$ is the set of states $x_2(t)$ reached at time $t \geq 0$ from the initial state x_2, under control mode u_2 and perturbation X_1.

All the notions of reachable sets for modes are extended in the natural manner to the notions of reachable sets for *patterns*. For example, for the pattern $\pi = u \cdot v$ of length 2, and for $0 \leq t \leq \tau$, we define:

$$\begin{aligned} \mathit{Reach}_f(t, X_0, \pi) &= \mathit{Reach}_f(t, X_0, u) \\ \mathit{Reach}_f(\tau + t, X_0, \pi) &= \mathit{Reach}_f(t, X_1, v) \qquad \text{with } X_1 = \mathit{Reach}_f(\tau, X_0, u). \end{aligned}$$

5.2. Compositional control

Recall that π_1^k (resp. π_2^k) denotes the prefix of length k of π_1 (resp.π_2), and $\pi_1(k)$ (resp. $\pi_2(k)$) the k-th element of sequence π_1 (resp. π_2). We now give the counterpart of Definition 1.

Definition 5. *Consider an index $i_1 \in I(n_1, d_1)$ (resp. $i_2 \in I(n_2, d_2)$) and a pattern $\pi_1 \in \Pi_1^{\ell_1}$ (resp. $\pi_2 \in \Pi_2^{\ell_2}$). The* approximate first-component sequence $\{Y_{i_1}^k(a, \pi_1)\}_{0 \le k \le \ell_1}$ *is defined as follows:*

$$Y_{i_1}^0(a, \pi_1) = r_{i_1}^a$$
$$Y_{i_1}^k(a, \pi_1) = \bigcup_{0 \le t \le \tau} Reach_{f_1}(t, Y_{i_1}^{k-1}(a, \pi_1), R_2 \pm (a + \varepsilon), \pi_1(k)) \quad \textit{for } 1 \le k \le \ell_1.$$

Similarly, the approximate second-component sequence $\{Y_{i_2}^k(a, \pi_2)\}_{0 \le k \le \ell_2}$ *is defined by*

$$Y_{i_2}^0(a, \pi_2) = r_{i_2}^a$$
$$Y_{i_2}^k(a, \pi_2) = \bigcup_{0 \le t \le \tau} Reach_{f_2}(t, R_1 \pm (a + \varepsilon), Y_{i_2}^{k-1}(a, \pi_2), \pi_2(k)) \quad \textit{for } 1 \le k \le \ell_2.$$

We define the property $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ as: for all $i_1 \in I(n_1, d_1)$, there exists $\pi_1 \in \Pi_1^{\ell_1}$ such that

$$Y_{i_1}^k(a, \pi_1) \subseteq R_1 \pm (a + \varepsilon) \qquad \text{for } 1 \le k \le \ell_1$$
$$Reach_{f_1}(\ell_1 \tau, r_{i_1}^a, R_2 \pm (a + \varepsilon), \pi_1) \subseteq R_1.$$

Likewise, we define the property $P_1(R_1, a, \varepsilon, d_1, \ell_1)$ as: for all $i_1 \in I(n_1, d_1)$, there exists $\pi_1 \in \Pi_1^{\ell_1}$ such that

$$Y_{i_2}^k(a, \pi_2) \subseteq R_2 \pm (a + \varepsilon) \qquad \text{for } 1 \le k \le \ell_2$$
$$Reach_{f_2}(\ell_2 \tau, R_1 \pm (a + \varepsilon), r_{i_2}^a, \pi_2) \subseteq R_2.$$

Procedures $\mathsf{PROC}_1(R_1, a, \varepsilon)$, $\mathsf{PROC}_2(R_2, a, \varepsilon)$ and expressions I_1, I_2, π_{i_1}, π_{i_2} are defined exactly as in Section 3. We now give the counterpart of Lemma 3 (the proof being similar).

Lemma 6. *Assume that $\mathsf{PROC}_1(R_1, a, \varepsilon)$ and $\mathsf{PROC}_2(R_2, a, \varepsilon)$ both terminate with* success*, with respective outputs (d_1, ℓ_1) and (d_2, ℓ_2).*

Then we have:

- *in case $\ell_1 \le \ell_2$, for all $t \in [(k-1)\tau, k\tau]$ $(1 \le k \le \ell_1)$:*

$$Reach_f(t, (r_{i_1}^a \times r_{i_2}^a), (\pi_{i_1}^k, \pi_{i_2}^k))_{|1} \subseteq Y_{i_1}^k(a, \pi_{i_1}) \subseteq R_1 \pm (a + \varepsilon)$$
$$Reach_f(t, (r_{i_1}^a \times r_{i_2}^a), (\pi_{i_1}^k, \pi_{i_2}^k))_{|2} \subseteq Y_{i_2}^k(a, \pi_{i_2}) \subseteq R_2 \pm (a + \varepsilon)$$
$$Reach_f(\ell_1 \tau, (r_{i_1}^a \times r_{i_2}^a), (\pi_{i_1}^{\ell_1}, \pi_{i_2}^{\ell_1}))_{|1} \subseteq R_1.$$

- *in case* $\ell_2 \leq \ell_1$, *for all* $t \in [(k-1)\tau, k\tau]$ $(1 \leq k \leq \ell_2)$*:*

$$Reach_f(t, (r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))_{|1} \subseteq Y^k_{i_1}(a, \pi_{i_1}) \subseteq R_1 \pm (a + \varepsilon)$$
$$Reach_f(t, (r^a_{i_1} \times r^a_{i_2}), (\pi^k_{i_1}, \pi^k_{i_2}))_{|2} \subseteq Y^k_{i_2}(a, \pi_{i_2}) \subseteq R_2 \pm (a + \varepsilon)$$
$$Reach_f(\ell_2\tau, (r^a_{i_1} \times r^a_{i_2}), (\pi^{\ell_2}_{i_1}, \pi^{\ell_2}_{i_2}))_{|2} \subseteq R_2.$$

Mutatis mutandis, Theorem 4 still holds in the continuous-time context (the proof is similar to the proof in the discrete-time context).

This allows us to implement the method along the same lines as in the discrete-time case, except that we apply the operator $Reach_{f_1}$ and $Reach_{f_2}$ on continuous time intervals of the form $[k, (k+1)\tau]$ instead of the mappings f_1 and f_2 at times $k\tau$. We implemented the method using the system *DynIBEX* [2, 3] which makes use of interval arithmetic [23] and Runge-Kutta methods to compute (an over-approximation of) the application results of $Reach_{f_1}$ and $Reach_{f_2}$.

5.3. Application

We demonstrate the feasibility of our approach on a building ventilation application adapted from [20]. The system is a four-room apartment subject to heat transfer between the rooms, with the external environment, with the underfloor, and with human beings. The dynamics of the system is given by the following equation:

$$\frac{dT_i}{dt} = \sum_{j \in \mathcal{N}^* \setminus \{i\}} a_{ij}(T_j - T_i) + \delta_{s_i} b_i (T^4_{s_i} - T^4_i) + c_i \max\left(0, \frac{V_i - V^*_i}{\overline{V_i} - V^*_i}\right)(T_u - T_i). \quad (5)$$

The state of the system is given by the temperatures in the rooms T_i, for $i \in \mathcal{N} = \{1, \ldots, 4\}$. Room i is subject to heat exchange with different entities stated by the indices $\mathcal{N}^* = \{1, 2, 3, 4, u, o, c\}$. The heat transfer between the rooms is given by the coefficients a_{ij} for $i, j \in \mathcal{N}^2$, and the different perturbations are the following:

- The external environment: it has an effect on room i with the coefficient a_{io} and the outside temperature T_o, varying between $27°C$ and $30°C$.
- The heat transfer through the ceiling: it has an effect on room i with the coefficient a_{ic} and the ceiling temperature T_c, varying between $27°C$ and $30°C$.
- The heat transfer with the underfloor: it is given by the coefficient a_{iu} and the underfloor temperature T_u, set to $17°C$ (T_u is constant, regulated by a PID controller).

- The perturbation induced by the presence of humans: it is given in room i by the term $\delta_{s_i} b_i (T_{s_i}^4 - T_i^4)$, the parameter δ_{s_i} is equal to 1 when someone is present in room i, 0 otherwise, and T_{s_i} is a given identified parameter.

The control V_i, $i \in \mathcal{N}$, is applied through the term $c_i \cdot \max(0, \frac{V_i - V_i^*}{\overline{V}_i - V_i^*})(T_u - T_i)$. A voltage V_i is applied to force ventilation from the underfloor to room i, and the command of an underfloor fan is subject to a dry friction. Because we work in a switching-control framework, V_i can take only discrete values, which removes the problem of dealing with a "max" function in interval analysis. In the experiment, V_1 and V_4 can take the values 0V or 3.5V, while V_2 and V_3 can take the values 0V or 3V. This leads to a system of the form (4) with $u(t) \in U = \{1, \ldots, 16\}$, the 16 switching modes corresponding to the different possible combinations of voltages V_i. The system can be decomposed in sub-systems of the form (2)-(3). The sampling period is $\tau = 10$s.

The parameters T_{s_i}, V_i^*, $\overline{V}_i$, a_{ij}, b_i, c_i are given in [20] and have been identified with a proper identification procedure detailed in [22]. Note that here we have neglected the term $\sum_{j \in \mathcal{N}} \delta_{d_{ij}} c_{i,j} \cdot h(T_j - T_i)$ of [20], representing the perturbation induced by the open or closed state of the doors between the rooms. Taking a "max" function into account with interval analysis is actually still a difficult task. However, this term could have been taken into account with a proper regularization (smoothing).

The main difficulty of this example is the large number of modes in the switching system, which induces a combinatorial issue. The centralized controller was obtained with 704 sub-rectangles in 29 minutes, the compositional controller was obtained with $16+16$ sub-rectangles in 20 seconds. In both cases, patterns of length 1 are used. The perturbation due to human beings has been taken into account by setting the parameters δ_{s_i} equal to the whole interval $[0, 1]$ for the decomposition, and the imposed perturbation for the simulation is given Figure 9. The temperatures T_o and T_c have been set to the interval $[27, 30]$ for the decomposition, and are set to 30°C for the simulation. A simulation of the controller obtained with the state-space bisection procedure is given in Figure 10, where the control objective is to "stabilize" the temperature $[20, 22]^2 \times [22, 24]^2$ while never going out of $[19, 23]^2 \times [21, 25]^2$.

6. Final Remarks

In this paper, we have proposed a compositional approach for control synthesis of sampled switching systems in the discrete-time framework and applied it to a real floor heating system. To our knowledge, this is the first time that reachability and recurrence properties are guaranteed for a case study of this size. We have also explained how the method extends to the continuous-time framework. The method can be extended to take into account obstacles and safety constraints.

Note that it is essential in our method that the components are *sampled* with the *same* sampling period τ, and that their clocks are synchronized. It would be interesting to investigate how the approach behaves when clocks are badly synchronized or when they have different periods (see e.g., [1]).

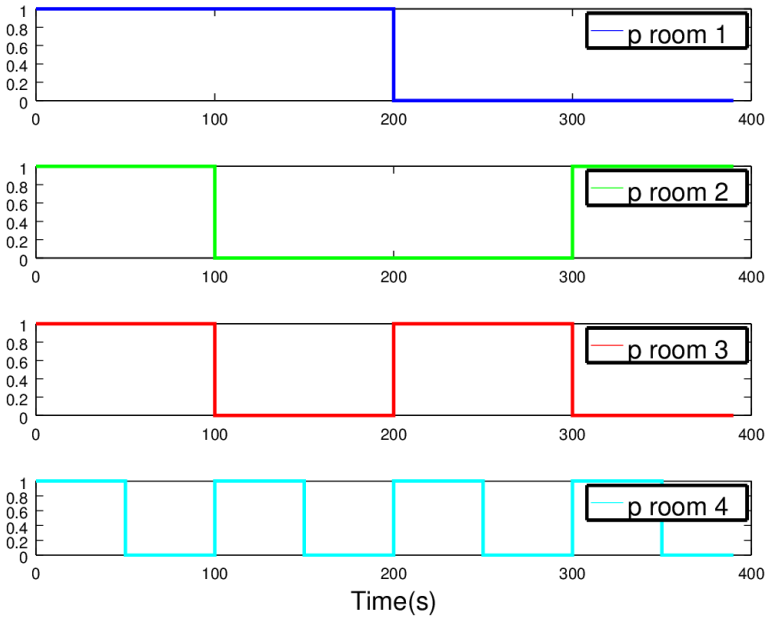

Figure 9: Perturbation (presence of humans) imposed within time in the different rooms.

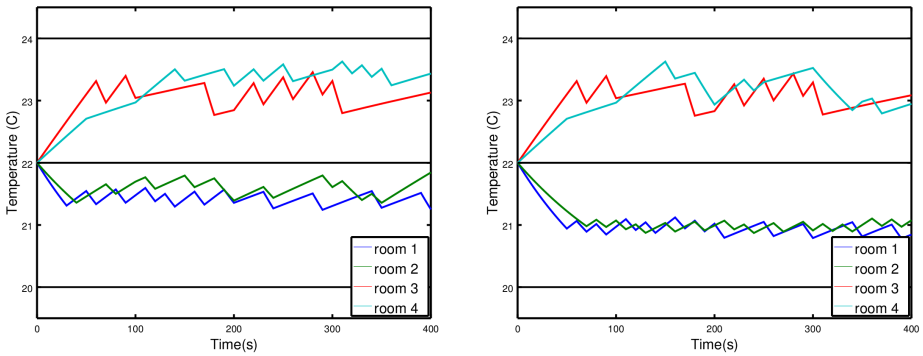

Figure 10: Simulation of the centralized (left) and compositional (right) controllers from the initial condition $(22, 22, 22, 22)$.

References

[1] Mohammad Al Khatib, Antoine Girard, and Thao Dang. Verification and synthesis of timing contracts for embedded controllers. In *Proceedings of the 19th International Conference on Hybrid Systems: Computation and Control (HSCC'16)*, pages 115–124, 2016.

[2] Julien Alexandre dit Sandretto and Alexandre Chapoutot. Dynibex library, 2015. `http://perso.ensta-paristech.fr/~chapoutot/dynibex/`.

[3] Julien Alexandre dit Sandretto and Alexandre Chapoutot. Validated explicit and implicit Runge-Kutta methods. *Reliable Computing*, 22(1):79–103, 2016.

[4] Matthias Althoff, Olaf Stursberg, and Martin Buss. Verification of uncertain embedded systems by computing reachable sets based on zonotopes. *IFAC Proceedings Volume*, 41(2):5125–5130, 2008.

[5] Rajeev Alur and Thomas A Henzinger. Reactive modules. *Formal Methods in System Design*, 15(1):7–48, 1999.

[6] Eugene Asarin, Olivier Bournez, Thao Dang, Oded Maler, and Amir Pnueli. Effective synthesis of switching controllers for linear systems. *Proceedings of the IEEE*, 88(7):1011–1025, 2000.

[7] Sergiy Bogomolov, Goran Frehse, Marius Greitschus, Radu Grosu, Corina S. Pasareanu, Andreas Podelski, and Thomas Strump. Assume-guarantee abstraction refinement meets hybrid systems. In *Hardware and Software: Verification and Testing – Proceesings of the 10th International Haifa Verification Conference (HVC'14)*, pages 116–131, 2014.

[8] Eric Dallal and Paulo Tabuada. On compositional symbolic controller synthesis inspired by small-gain theorems. In *Proceedings of the 54th IEEE Conference on Decision and Control (CDC'15)*, pages 6133–6138. IEEE, 2015.

[9] Laurent Fribourg, Ulrich Kühne, and Nicolas Markey. Game-based synthesis of distributed controllers for sampled switched systems. In *2nd International Workshop on Synthesis of Complex Parameters (SynCoP'15)*, volume 44 of *OASIcs*, pages 48–62, Dagstuhl, Germany, 2015.

[10] Laurent Fribourg, Ulrich Kühne, and Romain Soulat. Finite controlled invariants for sampled switched systems. *Formal Methods in System Design*, 45(3):303–329, 2014.

[11] Jeremy H. Gillula, Gabriel M. Hoffmann, Haomiao Huang, Michael P. Vitus, and Claire J. Tomlin. Applications of hybrid reachability analysis to robotic aerial vehicles. *The International Journal of Robotics Research*, 30(3):335354, 2011.

[12] Antoine Girard. Reachability of uncertain linear systems using zonotopes. In *Proceedings of the 8th International Workshop on Hybrid Systems: Computation and Control (HSCC'05)*, volume 3414 of *LNCS*, pages 291–305, 2005.

[13] Antoine Girard. Low-complexity switching controllers for safety using symbolic models. In *Proceedings of 4th IFAC Conference on Analysis and Design of Hybrid Systems (ADHS'12)*, pages 82–87. IFAC, 2012.

[14] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Éric Walter. *Applied Interval Analysis*. Springer, 2001.

[15] Eric S. Kim, Murat Arcak, and Sanjit A. Seshia. Compositional controller synthesis for vehicular traffic networks. In *Proceedings of the 54th IEEE Conference on Decision and Control (CDC'15)*, pages 6165–6171. IEEE, 2015.

[16] Wolfgang Kühn. Zonotope dynamics in numerical quality control. In Hans-Christian Hege and Konrad Polthier, editors, *Mathematical Visualization*, pages 125–134. Springer, 1998.

[17] Ulrich Kühne and Romain Soulat. Minimator 1.0. `https://bitbucket.org/ukuehne/minimator/overview`, 2015.

[18] Kim G Larsen, Marius Mikučionis, Marco Muñiz, Jiří Srba, and Jakob Haahr Taankvist. Online and compositional learning of controllers with application to floor heating. In *Proceedings of the 22nd International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'16)*, volume 9636 of *LNCS*, pages 244–259. Springer, 2016.

[19] Daniel Liberzon. *Switching in systems and control*. Springer Science+Business Media, 2012.

[20] Pierre-Jean Meyer. *Invariance and symbolic control of cooperative systems for temperature regulation in intelligent buildings*. PhD thesis, Université Grenoble Alpes, September 2015.

[21] Pierre-Jean Meyer, Antoine Girard, and Emmanuel Witrant. Safety control with performance guarantees of cooperative systems using compositional abstractions. *IFAC-PapersOnLine*, 48(27):317–322, 2015.

[22] Pierre-Jean Meyer, Hosein Nazarpour, Antoine Girard, and Emmanuel Witrant. Experimental implementation of UFAD regulation based on robust controlled invariance. In *Proceedings of the 13th European Control Conference (ECC'14)*, pages 1468–1473. IEEE, 2014.

[23] Ramon E. Moore. *Interval Analysis*. Prentice Hall, 1966.

[24] Octave Web page. `http://www.gnu.org/software/octave/`.

[25] Alberto L. Sangiovanni-Vincentelli, Werner Damm, and Roberto Passerone. Taming dr. frankenstein: Contract-based design for cyber-physical systems. *European Journal of Control*, 18(3):217–238, 2012.

[26] Stanley W. Smith, Petter Nilsson, and Necmiye Ozay. Interdependence quantification for compositional control synthesis with an application in vehicle safety systems. In *Proceedings of the 55th IEEE Conference on Decision and Control (CDC'16)*, pages 5700–5707. IEEE, 2016.