HAL
open science

Contract-based Design of Symbolic Controllers for Safety in Distributed Multiperiodic Sampled-Data Systems

Adnane Saoud, Antoine Girard, Laurent Fribourg

To cite this version:

Adnane Saoud, Antoine Girard, Laurent Fribourg. Contract-based Design of Symbolic Controllers for Safety in Distributed Multiperiodic Sampled-Data Systems. IEEE Transactions on Automatic Control, 2021, 66 (3), pp.1055-1070. 10.1109/tac.2020.2992446 . hal-02132070v2

HAL Id: hal-02132070

https://hal.science/hal-02132070v2

Submitted on 2 May 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Contract-based Design of Symbolic Controllers for Safety in Distributed Multiperiodic Sampled-Data Systems

Adnane Saoud, *Student Member, IEEE,* Antoine Girard, *Senior Member, IEEE,* and Laurent Fribourg

***Abstract*—This paper presents a symbolic control approach to the design of distributed safety controllers for a class of continuous-time nonlinear systems. More precisely, we consider systems made of components where each component is equipped with a sampled-data controller with its own sampling period, resulting globally in a distributed multiperiodic sampled-data system. Moreover, controllers receive partial information on the state of the other components. We propose a component-based approach to controller synthesis, which relies on the use of abstractions and continuous-time assume-guarantee contracts. The abstractions describe the dynamics of the system from the point of view of each component based on the information structure, while assume-guarantee contracts specify guarantees that a component must satisfy if assumptions on the other components are met. We show that our approach makes it possible to decompose a global safety control problem into local ones that can be solved independently. We then show how symbolic control techniques can be used to synthesize controllers that enforce the local control objectives. Illustrative applications in building automation and vehicle platooning are shown.**

***Index Terms*—Distributed control, multiperiodic sampling, component-based design, abstraction, assume-guarantee contracts, symbolic control, safety specifications.**

I. Introduction

The use of symbolic models for the control of continuous and hybrid systems has attracted considerable attention in the past decade (see e.g. [1], [2] and the references therein). A symbolic model (also called discrete abstraction) is a dynamical system with a finite number of states and inputs and related to the original system by some formal behavioral relationship, which makes it possible to refine a symbolic controller, designed for the abstraction, into a concrete one that can be used for the original system. Symbolic controllers can be synthesized using techniques developed in the areas of supervisory control of discrete event systems [3] and algorithmic game theory [4]. Symbolic models are often obtained through discretization of the state-space and of the time (if the original system is continuous-time), see e.g. [5], [6], [7], [8].

Due to discretization of the state-space, these abstraction techniques suffer from the curse of dimensionality (the number of symbolic states increases exponentially with respect to the state-space dimension). Several approaches have been proposed in the literature to improve the scalability of symbolic control techniques. In [9], [10], [11], symbolic models were computed using adaptive multi-resolution or multi-scale state-space discretization. In [12], [13], state-space discretization is not required since symbolic states are given by input sequences. In [14], [15] optimal abstraction parameters are derived to minimize the size of symbolic models.

For large systems made of interconnected components, a way to tackle scalability issues is to develop compositional approaches for the construction of symbolic abstractions. In such approaches, starting from a system made of interconnected components, a symbolic model is constructed for each of them, then the composition of the local symbolic models is shown to be an abstraction of the global system. However, in that approaches, the controller synthesis is done monolithically on the computed abstraction. Different approaches have been proposed in this direction. The authors in [16] proposed a compositional abstraction approach based on the notion of interconnection-compatible approximate bisimulation. Other approaches have been proposed recently using small-gain (or relaxed small-gain) like conditions [17], [18], [19], [20], [21] and dissipativity property [22], [23], [24]. The authors in [25], [26] presented a compositional abstraction and controller synthesis approach to the class of cascade interconnected systems. In [27], the notion of approximate disturbance simulation was used for compositional abstraction of continuous-time systems, where the states of other components were modeled as disturbance signals.

On the other hand, different approaches have been proposed for the compositional synthesis of symbolic controllers [28], [29], [30], [31] using assume-guarantee reasoning and contract-based design [32]. In those approaches, the controller synthesis is done compositionally, at the level of the components.

In all approaches mentioned above, it is implicitly assumed that sampling occurs synchronously in all components, which essentially makes it possible to reason in discrete time. In this paper, we consider that the components are equipped with sampled-data controllers with possibly different sampling periods, resulting globally in a distributed multiperiodic sampled-data system. To be able to handle multiperiodicity, we develop a compositional approach to safety synthesis based on continuous-time assume-guarantee contracts. Assume-guarantee contracts specify guarantees that a component must satisfy if assumptions on the other components are met. We rely on a notion of strong satisfaction introduced in [33], and develop a new composition result which allows us to deal with arbitrary interconnections of components.

In the proposed setup, the controller of a component can

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 725144). This research was partially supported by Labex DigiCosme (project ANR-11-LABEX-0045-DIGICOSME) operated by ANR as part of the program "Investissement d'Avenir" Idex Paris Saclay (ANR-11-IDEX-0003-02).

A. Saoud and A. Girard are with Laboratoire des Signaux et Systèmes, CNRS, CentraleSupélec, Université Paris-Sud, Université Paris-Saclay, 3, rue Joliot-Curie, 91192 Gif-sur-Yvette, cedex, France.

A. Saoud and L. Fribourg are with Laboratoire Spécification et Vérification, CNRS, ENS Paris-Saclay, 61, avenue du Président Wilson, 94235 Cachan Cedex, France.

receive partial information on the state of other components, when existing approaches, except [31], assume that only the state of the component is available to the controller. We then use abstractions that include, in addition to the dynamics of the component, a partial description of the dynamics of the other components, which reflects the available information. Intuitively, these abstractions describe the behavior of the system from the point of view of a component. We show that the combined use of assume-guarantee contracts and of abstractions makes it possible to decompose the global safety control problem into local ones that can be solved independently. We then show how symbolic control techniques can be used to synthesize controllers that enforce the local control objectives.

Our approach has similarities with [31], but differs significantly by considering continuous-time assume-guarantee contracts to deal with multiperiodicity, by introducing continuous abstractions and by using a different construction of symbolic models, which allows us to enforce an assume-guarantee contract either by enforcing the guarantee or by falsifying the assumption, while [31] does not exploit this second possibility. The current paper also extends the preliminary results presented in [34] where a controller has only information on the state of the associated component, which allows to work directly with components and does not require the use of abstractions giving a partial description of the dynamics of the system. Moreover, [34] only deals with cascade and feedback interconnections while in the current work arbitrary interconnection of components is allowed. Let us mention that by "arbitrary" interconnections, we mean known and fixed interconnections, where the state of a component may depend on all the other components.

The paper is organized as follows. In Section II, we introduce the class of systems considered throughout the paper and formulate the control problem under consideration. In Section III, we present our compositional framework based on abstractions and continuous-time assume-guarantee contracts. Section IV shows how the resulting local control problems can be solved using symbolic control techniques. Finally, in Section V, we apply the theoretical framework to illustrative applications in building automation and vehicle platooning.

Notations: $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_0^+$, $\mathbb{R}^+$ denote the set of nonnegative integers, of reals, of nonnegative reals and of positive reals, respectively. For $x \in \mathbb{R}^n$, $\|x\|$ is the Euclidean norm of x. For $\varepsilon \in \mathbb{R}^+$ and $A \subseteq \mathbb{R}^n$, the ε-expansion of A is the set $\mathcal{B}_\varepsilon(A) = \{y \in \mathbb{R}^n \mid \exists x \in A, \|x-y\| \leq \varepsilon\}$. The set of continuous-time domains is $\mathbb{I}(\mathbb{R}_0^+) = \{[0,a],\ a \in \mathbb{R}_0^+\} \cup \{[0,a),\ a \in \mathbb{R}^+\} \cup \{\mathbb{R}_0^+\}$. For $I \in \mathbb{I}(\mathbb{R}_0^+)$ and a metric space X, $C(I,X)$ denotes the set of continuous functions from I to X. Given two sets A and B, a set-valued map $f : A \rightrightarrows B$ is a map from A to the set of subsets of B, its domain is $\mathrm{dom}(f) = \{a \in A |\ f(a) \neq \emptyset\}$.

II. Problem formulation

We consider a system modeled by a differential inclusion:

$$\dot{x}(t) \in f(x(t), u(t)),\ x(t) \in X,\ u(t) \in U \tag{1}$$

where $x(t)$ and $u(t)$ denote the state and the control input, $X \subseteq \mathbb{R}^n$, $U \subseteq \mathbb{R}^p$ and $f : \mathbb{R}^n \times \mathbb{R}^p \rightrightarrows \mathbb{R}^n$. For the differential inclusion (1), we consider solutions in the Caratheodory sense, conditions for existence of solutions can be found for example in [35], [36].

The problem considered in the paper can be roughly formulated as follows: given $S \subseteq X$ a subset of safe states, synthesize a controller for (1) such that all controlled trajectories satisfy for all $t \in \mathbb{R}_0^+$, $x(t) \in S$.

A. Components

We consider systems that consist of N *components*, $N \geq 2$. For $i \in I = \{1, \ldots, N\}$, $x_i(t) \in \mathbb{R}^{n_i}$ and $u_i(t) \in \mathbb{R}^{p_i}$ denote the state and control input of component i. Then, $x(t) = (x_1(t), \ldots, x_N(t))$ and $u(t) = (u_1(t), \ldots, u_N(t))$. We do not make any specific assumption on the structure of vector field f so arbitrary interconnections of components can be considered. However, we assume that there is no static coupling between control inputs imposed by the set U as stated below:

Assumption 1: $U = \prod_{i \in I} U_i$ where $U_i \subseteq \mathbb{R}^{p_i}$, $i \in I$.

Assumption 1 implies that if $u_i \in U_i$, for all $i \in I$, then $u \in U$, which means that control inputs may be chosen independently. For controller synthesis, the considered setup is the following. Each component is equipped with a sampled-data controller, with possibly different sampling periods. Moreover, controllers receive partial information on the state of the system, as specified by some information structure. Hence, the sampled-data system under consideration is distributed, multiperiodic and with partial information.

B. Information structure

For $i \in I$, let us define the linear maps $\pi_{i,0} : \mathbb{R}^n \to \mathbb{R}^{n_i}$ such that for all $x = (x_1, \ldots, x_N) \in \mathbb{R}^n$, $\pi_{i,0}(x) = x_i$.

The *information structure* of the system reflects the knowledge that the controller of each component has on the state of the system. Formally, the information structure is defined by linear maps $\pi_{i,1} : \mathbb{R}^n \to \mathbb{R}^{m_i}$, $i \in I$ such that for all $i \in I$, the map $x \mapsto (\pi_{i,0}(x), \pi_{i,1}(x))$ is surjective. Then, let

$$z_i(t) = \pi_{i,1}(x(t)),\ i \in I. \tag{2}$$

There exist linear maps $\pi_{i,2} : \mathbb{R}^n \to \mathbb{R}^{n-n_i-m_i}$, $i \in I$, such that for all $i \in I$, $\pi_i : \mathbb{R}^n \to \mathbb{R}^n$ given by $\pi_i(x) = (\pi_{i,0}(x), \pi_{i,1}(x), \pi_{i,2}(x))$ is a bijection. Then, let us define

$$w_i(t) = \pi_{i,2}(x(t)),\ i \in I.$$

While $x_i(t)$ is the state of component i, $z_i(t)$ and $w_i(t)$ contains the information on the state of other components that constitute the system. The controller of component i has access to the state of the component $x_i(t)$ and to a portion of the state of the system $z_i(t)$, it has no information on the value of $w_i(t)$. In the following, we will denote $X_i = \pi_{i,0}(X)$, $Z_i = \pi_{i,1}(X)$ and $W_i = \pi_{i,2}(X)$.

Remark 1: When $m_i = n - n_i$, the component has full information on the state of the system. When $m_i = 0$, the controller of component i has only information on the state

of the component $x_i(t)$, and we recover the case considered in [34].

For $i \in I$, we also define $\nu_{i,0} : \mathbb{R}^p \to \mathbb{R}^{p_i}$ such that for all $u = (u_1, \ldots, u_N) \in \mathbb{R}^p$, $\nu_{i,0}(u) = u_i$. Under Assumption 1, we have $U_i = \nu_{i,0}(U)$. Then, there exist linear maps $\nu_{i,1} : \mathbb{R}^p \to \mathbb{R}^{p-p_i}$, $i \in I$, such that for all $i \in I$, $\nu_i : \mathbb{R}^p \to \mathbb{R}^p$ given by $\nu_i(u) = (\nu_{i,0}(u), \nu_{i,1}(u))$ is a bijection. We assume that the controller of component i has no information on the input values of other components (i.e. on $\nu_{i,1}(u(t))$).

C. Sampled-data controllers

For $i \in I$, the *sampled-data controller* of component i is defined by a set-valued map $g_i : X_i \times Z_i \rightrightarrows U_i$ associated to a sampling period $\tau_i \in \mathbb{R}^+$. Let us remark that the control map depend on the state of the component $x_i(t) \in X_i$ and on the known portion of the state of the system $z_i(t) \in Z_i$, as specified by the information structure of the system. The sequence of sampling instants $(\tau_{i,k})_{k\in\mathbb{N}}$ is given by $\tau_{i,k} = k\tau_i$, for $k \in \mathbb{N}$. The initial sampling instant $\tau_{i,0}$ coincides with the initial time 0.

Remark 2: In this paper, it is assumed that all controllers have the same initial sampling time $\tau_{i,0} = 0$. This restriction is made for the sake of simplicity and the following results could be generalized to controllers with an initial clock offset.

Remark 3: Since we are dealing with components equipped with sampled-data controllers that have different sampling periods, existing compositional controller synthesis techniques [28], [30], [31], [27] (which assume that controllers of different components have the same sampling period and reason on discrete-time) would fail. Indeed, continuous-time reasoning is crucial to reason compositionally when sampling occurs asynchronously in the components.

D. Trajectories

The notion of trajectory is defined below:

Definition 1: A *trajectory* of the system is an absolutely continuous map $x : E \to X$ defined on a time domain $E \in \mathbb{I}(\mathbb{R}_0^+)$, with $x = (x_1, \ldots, x_N)$ and such that there exists a piecewise constant function $u : E \to U$, with $u = (u_1, \ldots, u_N)$ such that:

- for almost all $t \in E$, (1) is satisfied;
- for all $i \in I$, for all $k \in \mathbb{N}$ with $\tau_{i,k} \in E$,

$$\begin{cases} u_i(t) = u_{i,k}, \quad \forall t \in E \cap [\tau_{i,k}, \tau_{i,k+1}), \\ \text{where } u_{i,k} \in g_i(x_i(\tau_{i,k}), z_i(\tau_{i,k})), \end{cases} \tag{3}$$

  and z_i is given by (2).

We denote by Σ the multiperiodic distributed sampled-data system with partial information defined by (1), (2) and (3), and we denote by $\mathcal{T}(\Sigma)$ its set of trajectories. A pictural representation of Σ for $N = 2$ is shown in Figure 1.

Given two trajectories of Σ, $x : E \to X$ and $x' : E' \to X$, x' is said to be a *prefix* of x if $E' \subseteq E$ and for all $t \in E'$, $x(t) = x'(t)$. A trajectory $x \in \mathcal{T}(\Sigma)$ is said to be *maximal* if there does not exist any trajectory $x' \in \mathcal{T}(\Sigma)$ such that $x' \neq x$ and x is a prefix of x'. A trajectory of Σ, $x : E \to X$, is said to be *complete* if $E = \mathbb{R}_0^+$.

In the rest of the paper, we make the following technical assumption on the system Σ:

Assumption 2: Let $\tau = \min(\tau_1, \ldots, \tau_N)$, for all initial conditions $x_0 \in X$, for all $u_0 \in U$, any solution[1] $x : E \to X$ to differential inclusion (1), defined on $E = [0, s)$ with $s \in (0, \tau]$, or on $E = [0, s]$ with $s \in [0, \tau)$, such that $x(0) = x_0$ and $u(t) = u_0$ for all $t \in E$, can be extended to a solution defined on $[0, \tau]$, with $u(t) = u_0$ for all $t \in [0, \tau]$.

Assumption 2 guarantees that the trajectories of Σ are well-defined between two successive sampling instants. More precisely, from the previous assumptions, we can establish the following instrumental lemma, whose proof is given in appendix.

Lemma 1: Under Assumptions 1 and 2, a maximal trajectory of Σ, $x : E \to X$, is not complete if and only if there exist $i \in I$ and $k \in \mathbb{N}$ such that $E = [0, \tau_{i,k+1})$ and $(x_i(\tau_{i,k+1}^-), z_i(\tau_{i,k+1}^-)) \notin \mathrm{dom}(g_i)$[2].

Now, we can give a formal statement of the problem considered in the paper:

Problem 1: Given a system with X, U and f satisfying Assumptions 1 and 2, given a subset of safe states $S \subseteq X$, given an information structure $\pi_{i,1}$ and sampling periods τ_i, for $i \in I$; synthesize control maps g_i, for $i \in I$, such that any maximal trajectory of Σ, x, is complete and satisfies $x(t) \in S$, for all $t \in \mathbb{R}_0^+$.

III. Component-based design

In this section, we present a component-based solution to Problem 1. We first introduce abstractions of the system Σ from the point of view of each component based on the information structure. Then, we present the notion of assume-guarantee contract and state the main result of the section, which claims that if each abstraction satisfies an assume-guarantee contract and fulfills a completeness condition, then the control objective defined in Problem 1 is achieved.

A. Abstraction

Based on the information structure, we construct an abstraction that represents the point of view of component $i \in I$ on the system Σ. The abstraction is denoted $\hat{\Sigma}_i$ and given by the following differential inclusion together with control law (3):

$$\begin{cases} \dot{x}_i(t) \in \hat{f}_{i,0}(x_i(t), z_i(t), w_i(t), u_i(t)), \\ \dot{z}_i(t) \in \hat{f}_{i,1}(x_i(t), z_i(t), w_i(t), u_i(t)), \\ x_i(t) \in X_i,\ z_i(t) \in Z_i,\ w_i(t) \in W_i,\ u_i(t) \in U_i, \end{cases} \tag{4}$$

where $\hat{f}_{i,j}$, are defined for $j = 0, 1$ by

$$\hat{f}_{i,j}(x_i, z_i, w_i, u_i) = \pi_{i,j}\big(f(\pi_i^{-1}(x_i, z_i, w_i), \nu_i^{-1}(u_i, \nu_{i,1}(U))\big).$$

A pictural representation of the abstraction $\hat{\Sigma}_i$ is shown in Figure 2. The abstraction of the system Σ from the point of

[1] A solution to differential inclusion (1), $x : E \to X$, is an absolutely continuous map such that for almost all $t \in E$, (1) is satisfied.

[2] $x_i(\tau_{i,k+1}^-) = \lim_{t \searrow \tau_{i,k+1}}$ and represents the limit from the left.

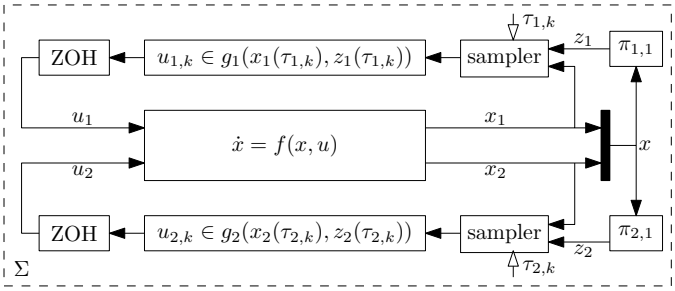

Fig. 1: Architecture of the multiperiodic distributed sampled-data system with partial information Σ with $N = 2$, defined by (1), (2) and (3).

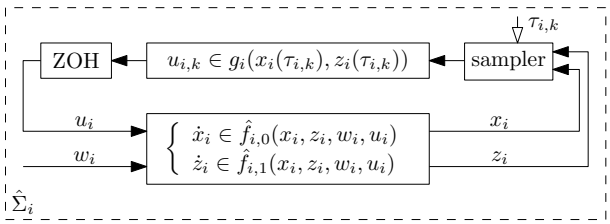

Fig. 2: Abstraction $\hat{\Sigma}_i$ of the system Σ from the point of view of a component $i \in I$, defined by (4) and (3).

view of component i includes a model of the component, but also a partial description of the dynamics of the rest of the system. Indeed, in the abstraction $\hat{\Sigma}_i$, the evolutions of the state of the component $x_i(t)$ and of the known portion of the state of the system $z_i(t)$ are modeled. The dynamics of unknown states $w_i(t)$ as well as inputs of other components are abstracted.

Example 1: Let the system Σ made of three components and described by:

$$\Sigma : \begin{cases} \dot{x}_1(t) \in f_1(x_1(t), x_2(t), x_3(t), u_1(t)), \\ \dot{x}_2(t) \in f_2(x_1(t), x_2(t), x_3(t), u_2(t)), \\ \dot{x}_3(t) \in f_3(x_1(t), x_2(t), x_3(t), u_3(t)), \\ x_i(t) \in X_i,\ u_i(t) \in U_i,\ i \in I = \{1, 2, 3\}. \end{cases}$$

Given the information structure defined by $\pi_{1,1}(x(t)) = (x_2(t), x_3(t))$, $\pi_{2,1}(x(t)) = (x_2(t))$ and $\pi_{3,1}(x(t)) = \emptyset$. The abstractions of the system Σ from the point of view of the three components are computed as follows:

- $\pi_{1,1}(x(t)) = (x_2(t), x_3(t))$ which means that the controller of the first component g_1 can receive informations on the states of the second and third components ($z_1 = (x_2, x_3)$). Hence, in the abstraction of the system Σ from the point of view of the first component, we describe the evolution of x_1, x_2 and x_3, while the inputs to the second and third components are abstracted and considered as disturbances. The abstraction $\hat{\Sigma}_1$ is given by:

$$\hat{\Sigma}_1 : \begin{cases} \dot{x}_1(t) \in f_1(x_1(t), x_2(t), x_3(t), u_1(t)), \\ \dot{x}_2(t) \in f_2(x_1(t), x_2(t), x_3(t), U_2), \\ \dot{x}_3(t) \in f_3(x_1(t), x_2(t), x_3(t), U_3) \end{cases}$$

- $\pi_{2,1}(x(t)) = x_1(t)$ which means that the controller of the second component g_2 can receive only informations on the states of the first component ($z_2 = x_1$ and $w_2 = x_3$). Hence, in the abstraction of the system Σ from the point of view of the second component, we describe only the evolution of x_1 and x_2, while the dynamics of the state of the third component x_3 is abstracted and considered as a disturbance. Moreover, the inputs of the first and third components are abstracted. The abstraction $\hat{\Sigma}_2$ is given by:

$$\hat{\Sigma}_2 : \begin{cases} \dot{x}_1(t) \in f_1(x_1(t), x_2(t), x_3(t), U_1), \\ \dot{x}_2(t) \in f_2(x_1(t), x_2(t), x_3(t), u_2(t)) \end{cases}$$

- $\pi_{3,1}(x(t)) = \emptyset$ which means that the controller of the third component g_3 cannot receive any information on the states of the other components ($w_3 = (x_1, x_2)$). Hence, in the abstraction of the system Σ from the point of view of the third component, we describe only the evolution of the local state x_3, while the dynamics of the states of other components (x_1 and x_2) are abstracted and considered as a disturbance. Moreover, the inputs of the first and second components are abstracted. The abstraction $\hat{\Sigma}_3$ is given by:

$$\hat{\Sigma}_3 : \left\{ \ \dot{x}_3(t) \in f_3(x_1(t), x_2(t), x_3(t), u_3(t)) \right.$$

Definition 2: A *trajectory* of the abstraction $\hat{\Sigma}_i$ is a triple of maps $(x_i, z_i, w_i) : E \to X_i \times Z_i \times W_i$ defined on a time domain $E \in \mathbb{I}(\mathbb{R}_0^+)$, where x_i and z_i are absolutely continuous and w_i is continuous and such that there exists a piecewise constant function $u_i : E \to U$, such that:

- for almost all $t \in E$, (4) is satisfied;
- for all $k \in \mathbb{N}$ with $\tau_{i,k} \in E$, (3) is satisfied.

We use $\mathcal{T}(\hat{\Sigma}_i)$ to denote the set of trajectories of the abstraction $\hat{\Sigma}_i$. The notion of prefix, maximal and complete trajectories are defined as for the system Σ.

Remark 4: While Σ is a multiperiodic distibuted sampled-data system with partial information, for any $i \in I$, the abstraction $\hat{\Sigma}_i$ is a periodic sampled-data system of period τ_i with a single control law defined by the map g_i, and which has full information on the state of the differential inclusion (4). Moreover, the dimension of the differential inclusions (1) and (4) are n and $n_i + m_i$, respectively. In typical situations, n is much larger than $n_i + m_i$. All together, these facts make it much easier to work on the abstraction $\hat{\Sigma}_i$ than on Σ.

Similar to Assumption 2, we will make the following assumption:

Assumption 3: For all initial conditions $(x_{i,0}, z_{i,0}) \in X_i \times Z_i$, for all $u_{i,0} \in U_i$, for all $w_i \in C([0, \tau_i], W_i)$, any solution[3] $(x_i, z_i) : E \to X_i \times Z_i$ to differential inclusion (4), defined on $E = [0, s)$ with $s \in (0, \tau_i]$, or on $E = [0, s]$ with $s \in [0, \tau_i)$, such that $(x_i(0), z_i(0)) = (x_{i,0}, z_{i,0})$ and $u_i(t) = u_{i,0}$ for all $t \in E$ can be extended to a solution defined on $[0, \tau_i]$, with $u_i(t) = u_{i,0}$ for all $t \in [0, \tau_i]$.

[3]A solution to differential inclusion (4), $(x_i, z_i) : E \to X_i \times Z_i$, is a pair of absolutely continuous maps such that for almost all $t \in E$, (4) is satisfied.

We have the following result, whose proof is similar to that of Lemma 1 and therefore ommitted:

Lemma 2: Under Assumption 3, a maximal trajectory of $\hat{\Sigma}_i$, $(x_i, z_i, w_i) : E \to X_i \times Z_i \times W_i$, is not complete if and only if there exists $k \in \mathbb{N}$ such that $E = [0, \tau_{i,k+1})$ and $(x_i(\tau^-_{i,k+1}), z_i(\tau^-_{i,k+1})) \notin \text{dom}(g_i)$.

Remark 5: Let us mention that in practice, Assumption 3 needs only to be satisfied for all initial conditions $(x_{i,0}, z_{i,0}) \in \text{dom}(g_i)$ and for all $u_{i,0} \in g_i(x_{i,0}, z_{i,0})$, as shown in Proposition 3.

In order to relate the trajectories of the system Σ to the trajectories of its abstractions, the following result shows that for all $i \in \mathcal{I}$, any trajectory of Σ is a trajectory of $\hat{\Sigma}_i$.

Proposition 1: If $x : E \to X$ is a trajectory of Σ, then for all $i \in I$, $\pi_i(x) = (x_i, z_i, w_i) : E \to X_i \times Z_i \times W_i$ is a trajectory of $\hat{\Sigma}_i$.

Proof: Let us consider $x : E \to X$ and $u : E \to U$ such that differential inclusion (1) is satisfied and let $\pi_i(x) = (x_i, z_i, w_i) : E \to X_i \times Z_i \times W_i$ and $u_i = \nu_{i,0}(u) : E \to U_i$. Then, one can check that by construction, differential inclusion (4) is satisfied. Then, the result stated in the proposition follows directly from the Definitions 1 and 2 of trajectories of Σ and $\hat{\Sigma}_i$. ■

B. Assume-guarantee contracts and compositional reasoning

Contracts make it possible to reason about the properties of a system based on properties of its components [32]. In this paper, we consider the following type of contracts adapted from [33]:

Definition 3: Let $i \in I$, an *assume-guarantee contract* for $\hat{\Sigma}_i$ is a tuple $\mathcal{C}_i = (A_{i,1}, A_{i,2}, G_i)$ where:

- $A_{i,1} \subseteq Z_i$ and $A_{i,2} \subseteq W_i$ are sets of assumptions;
- $G_i \subseteq X_i$ is a set of guarantees, where G_i is closed.

We say that $\hat{\Sigma}_i$ *strongly satisfies* contract $\mathcal{C}_i$, denoted $\hat{\Sigma}_i \models_s \mathcal{C}_i$ if for all trajectories of $\hat{\Sigma}_i$, $(x_i, z_i, w_i) : E \to X_i \times Z_i \times W_i$:

- $x_i(0) \in G_i$;
- for all $t \in E$, such that for all $s \in [0, t]$, $z_i(s) \in A_{i,1}$ and $w_i(s) \in A_{i,2}$, there exists $\delta > 0$, such that for all $s \in [0, t + \delta] \cap E$, $x_i(s) \in G_i$.

Strong satisfaction of an assume-guarantee contract states that if the states of the other components, z_i, w_i, belong to the specified sets of assumptions, $A_{i,1}$, $A_{i,2}$, up to an arbitrary time instant t, then the state of the component, x_i belongs to the specified set of guarantees G_i at least until $t+\delta$ with $\delta > 0$. it is noteworthy to mention that, in general, the value of δ may depend on the trajectory (x_i, z_i, w_i) and on the value of the time instant $t \in E$. In [33], a notion of weak satisfaction is also defined where δ needs only be non-negative.

We now provide a result allowing us to reason about the behavior of the system Σ from the properties satisfied by the abstractions $\hat{\Sigma}_i$, $i \in I$.

Proposition 2: Under Assumptions 1 and 2, for $i \in I$, let $\mathcal{C}_i = (A_{i,1}, A_{i,2}, G_i)$ be an assume-guarantee contract for $\hat{\Sigma}_i$ and let $G = \prod_{i \in I} G_i$. Let us assume that for all $i \in I$, $\hat{\Sigma}_i \models_s \mathcal{C}_i$, $\pi_{i,1}(G) \subseteq A_{i,1}$ and $\pi_{i,2}(G) \subseteq A_{i,2}$. Then, for any trajectory of Σ, $x : E \to X$, we have $x(t) \in G$ for all $t \in E$.

Proof: It is sufficient to show that the conclusion holds for maximal trajectories of Σ. Then, let $x : E \to X$ be a maximal trajectory of Σ, from Lemma 1 it follows that $E = [0, a)$ with $a \in \mathbb{R}^+ \cup \{+\infty\}$. Let us define

$$T = \sup\{t \in E | \ \forall s \in [0, t], \ x(s) \in G\}. \tag{5}$$

From Proposition 1, $\pi_i(x) = (x_i, z_i, w_i) : E \to X_i \times Z_i \times W_i$ is a trajectory of $\hat{\Sigma}_i$, for all $i \in I$. Strong satisfaction of $\mathcal{C}_i$ gives that $x_i(0) \in G_i$ for all $i \in I$, and thus $x(0) \in G$. Then, $T \in \mathbb{R}^+_0 \cup \{+\infty\}$ and for all $s \in [0, T)$, $x(s) \in G$. Let us consider two different cases.

Case 1 - $T < a$:

Using the continuity of x and since G is closed, we have for all $s \in [0, T]$, $x(s) \in G$. Then, for all $i \in I$, for all $s \in [0, T]$ $z_i(s) \in \pi_{i,1}(G) \subseteq A_{i,1}$ and $w_i(s) \in \pi_{i,2}(G) \subseteq A_{i,2}$. Since $\hat{\Sigma}_i$ strongly satisfies $\mathcal{C}_i$, there exists $\delta_i \in (0, a - T)$ such that for all $s \in [0, T + \delta_i]$, $x_i(s) \in G_i$. Then, for $\delta = \min_{i \in I} \delta_i$, we have for all $s \in [0, T + \delta]$, $x(s) \in G$. This contradicts the definition of T given by (5), which shows that this case is actually impossible.

Case 2 - $T = a$:

Then, we directly get that for all $s \in E = [0, T)$, $x(s) \in G$. ■

Remark 6: It is important to note that strong satisfaction of the assume-guarantee contracts is crucial for the proof of Proposition 2 and that with weak satisfaction ($\delta = 0$), it would not hold without additional assumptions on the system (See [33] for an example illustrating a case where strong satisfaction is necessary to achieve a global safety control objective). It is also important to note that the compositional approach presented in the section is quite general, since it allows to reason on any type of interconnections (when only cascade and feedback interconnections are considered in [33], [34]).

C. Completeness condition

Completeness of maximal trajectories of Σ is necessary to achieve the control objective defined in Problem 1. Assume-guarantee contracts only ensure the satisfaction of the safety specification on the time interval E on which the trajectories of the system Σ are defined, as shown in Proposition 2, and do not ensure the completeness of those trajectories ($E = +\infty$). For this reason, we need a supplementary condition on the abstractions $\hat{\Sigma}_i$ to ensure completeness of the maximal trajectories of the system Σ. In this part, we provide a sufficient condition (called completeness condition) on the abstractions $\hat{\Sigma}_i$ to ensure the existence of complete trajectories for the system Σ. Then, we present the main result of the section, which states that if all the abstractions strongly satisfy their contracts and satisfy the completeness condition, then the control objective defined in Problem 1 is achieved. First, we introduce the completeness condition.

Definition 4: Let $i \in I$, let $\mathcal{C}_i = (A_{i,1}, A_{i,2}, G_i)$ be an assume-guarantee contract for $\hat{\Sigma}_i$. Under Assumption 3, we say that $\hat{\Sigma}_i$ satisfies *the completeness condition*, denoted (CC), if for all initial conditions $(x_{i,0}, z_{i,0}) \in \text{dom}(g_i)$, for all $u_{i,0} \in g_i(x_{i,0}, z_{i,0})$, for all $w_i \in C([0, \tau_i], W_i)$, any solution $(x_i, z_i) : [0, \tau_i] \to X_i \times Z_i$ to differential inclusion (4) with $u_i(t) = u_{i,0}$ for all $t \in [0, \tau_i]$ satisfies:

$$\begin{aligned}(\forall t \in [0, \tau_i],\ z_i(t) \in A_{i,1} \text{ and } w_i(t) \in A_{i,2}) \\ \implies ((x_i(\tau_i), z_i(\tau_i)) \in \text{dom}(g_i)).\end{aligned} \tag{6}$$

Intuitively, the completeness condition states that if $z_i(t)$ and $w_i(t)$ remain in the specified sets of assumptions at all times, the trajectories of $\hat{\Sigma}_i$ remain in $dom(g_i)$ at sampling instants, and according to Lemma 2 are complete. The idea here is to remove blocking (non complete) trajectories by removing trajectories that do not belong to the domain of the controller at some sampling instant. We can now state the main result of the section:

Theorem 1: Under Assumptions 1, 2 and 3, for $i \in I$, let $\mathcal{C}_i = (A_{i,1}, A_{i,2}, G_i)$ be an assume-guarantee contract for $\hat{\Sigma}_i$ and let $G = \prod_{i \in I} G_i$. Let us assume that $G \subseteq S$, and for all $i \in I$, $\hat{\Sigma}_i \models_s \mathcal{C}_i$, $\pi_{i,1}(G) \subseteq A_{i,1}$, $\pi_{i,2}(G) \subseteq A_{i,2}$ and $\hat{\Sigma}_i$ satisfies (CC). Then, any maximal trajectory of Σ, x, is complete and satisfies $x(t) \in S$, for all $t \in \mathbb{R}_0^+$.

Proof: In view of Proposition 2 it can be seen that for any trajectory of Σ, $x : E \to X$, we have $x(t) \in G \subseteq S$ for all $t \in E$. Let us now prove that any maximal trajectory of Σ is complete. Let us consider a maximal trajectory of Σ, $x : E \to X$, and let us assume that x is not complete. Then from Lemma 1, there exists $i \in I$ and $k \in \mathbb{N}$ such that $E = [0, \tau_{i,k+1})$ and $(x_i(\tau_{i,k+1}^-), z_i(\tau_{i,k+1}^-)) \notin \text{dom}(g_i)$. Let $(x_i, z_i, w_i) = \pi_i(x)$, from Proposition 1, (x_i, z_i, w_i) is a trajectory of $\hat{\Sigma}_i$. Moreover, since, for all $t \in E$, $x(t) \in G$, we get that for all $t \in E$, $z_i(t) \in \pi_{i,1}(G) \subseteq A_{i,1}$ and $w_i(t) \in \pi_{i,2}(G) \subseteq A_{i,2}$. Then, since $\hat{\Sigma}_i$ satisfies (CC) and by continuity of x_i and z_i, we get that $(x_i(\tau_{i,k+1}^-), z_i(\tau_{i,k+1}^-)) \in \text{dom}(g_i)$; which yields a contradiction. Hence, x is necessarily complete. ■

Theorem 1 provides a mean to tackle Problem 1 using a component-based approach. If the set of safe states is given by $S = \prod_{i \in I} S_i$ where $S \subseteq X$ and $S_i \subseteq X_i$, for all $i \in I$, a natural assignment of assume-guarantee contracts for abstractions in order to enforce the safety specification S is to define $\mathcal{C}_i = (\pi_{i,1}(S), \pi_{i,2}(S), S_i)$, (cf.examples).

IV. Local controller design

In view of Theorem 1, a solution to Problem 1 can be found by considering local control problems for the abstractions $\hat{\Sigma}_i$, $i \in I$. These control problems can be solved independently. For this reason and to improve readability, the index $i \in I$ is dropped in the following. Hence, the local control problem under consideration in this section is the following:

Problem 2: Given an abstraction $\hat{\Sigma}$ defined by (4), (3) and satisfying Assumption 3, given an assume-guarantee contract $\mathcal{C} = (A_1, A_2, G)$ for $\hat{\Sigma}$; synthesize a control map g, such that $\hat{\Sigma} \models_s \mathcal{C}$ and $\hat{\Sigma}$ satisfies (CC).

In this section, we first develop sufficient conditions for strong satisfaction of assume-guarantee contracts. Then, we present a solution to Problem 2 based on the symbolic control approach [1], [2]. Finally, we analyse the influence of the information structure on the feasibility of Problem 2.

Remark 7: Let us point out that the compositional framework presented in the previous section is quite general. In the following, we propose an approach based on symbolic control, however one can use any synthesis technique to ensure the strong satisfaction of assume-guarantee contracts and enforce the completeness condition (CC). Actually, one can even decide to use different techniques for different components.

A. Sufficient conditions for assume-guarantee contracts

In this part, we establish sufficient conditions for the strong satisfaction of an assume-guarantee contract. This criterion is more practical than Definition 3 since it makes it possible to reason between two successive sampling instants rather than on the whole time domain of the trajectory.

First, we introduce the following auxiliary result.

Lemma 3: Let $\mathcal{C} = (A_1, A_2, G)$ be an assume-guarantee contract for $\hat{\Sigma}$ and let us assume that there exists $\varepsilon > 0$ such that for any trajectory of $\hat{\Sigma}$, $(x, z, w) : E \to X \times Z \times W$, we have $x(0) \in G$, and for all $t \in E$:

$$\begin{aligned}(\forall s \in [0, t], \quad z(s) \in \mathcal{B}_\varepsilon(A_1) \text{ and } w(s) \in \mathcal{B}_\varepsilon(A_2)) \\ \implies (\forall s \in [0, t],\ x(s) \in G).\end{aligned} \tag{7}$$

Then, $\hat{\Sigma} \models_s \mathcal{C}$.

Proof: Let $(x, z, w) : E \to X \times Z \times W$ be a trajectory of $\hat{\Sigma}$. We have $x(0) \in G$ and the first condition for the strong satisfaction of the assume-guarantee contract is satisfied. Let $t \in E$, such that for all $s \in [0, t]$, $z(s) \in A_1$ and $w(s) \in A_2$. From the continuity of z and w and for $\varepsilon > 0$, there exists $\delta > 0$ such that for all $s \in [0, t + \delta] \cap E$, $z(s) \in \mathcal{B}_\varepsilon(A_1)$ and $w(s) \in \mathcal{B}_\varepsilon(A_2)$. Then, from (7) we have for all $s \in [0, t + \delta] \cap E$, $x(s) \in G$. Hence, $\hat{\Sigma} \models_s \mathcal{C}$. ■

Lemma 3 essentially states that strong satisfaction of $\mathcal{C}$ is ensured if one can prove the weak satisfaction of a similar assume-guarantee contract with relaxed assumptions $\mathcal{C}_\varepsilon = (\mathcal{B}_\varepsilon(A_1), \mathcal{B}_\varepsilon(A_2), G)$ where $\varepsilon > 0$ can be arbitrarily small.

In the following result, we give a simple criterion for the abstraction $\hat{\Sigma}$ to strongly satisfy an assume-guarantee contract. This criterion benefits from the nature of the controller (sampled-data controller) and allows us to reason between two successive sampling instants.

Proposition 3: Under Assumption 3, let $\mathcal{C} = (A_1, A_2, G)$ be an assume-guarantee contract for $\hat{\Sigma}$. Let us assume that $\text{dom}(g) \subseteq G \times Z$ and that there exists $\varepsilon > 0$ such that for all initial conditions $(x_0, z_0) \in \text{dom}(g)$, for all $u_0 \in g(x_0, z_0)$, for all $w \in C([0, \tau], W)$, any solution $(x, z) : [0, \tau] \to X \times Z$ to differential inclusion (4) with $(x(0), z(0)) = (x_0, z_0)$ and $u(t) = u_0$ for all $t \in [0, \tau]$ satisfies:

$$\begin{aligned}(\forall s \in [0, \tau],\ w(s) \in \mathcal{B}_\varepsilon(A_2)) \implies \\ \big((\forall s \in [0, \tau],\ x(s) \in G) \vee [\exists s' \in [0, \tau], \\ (z(s') \notin \mathcal{B}_\varepsilon(A_1)) \wedge (\forall s \in [0, s'],\ x(s) \in G)]\big).\end{aligned} \tag{8}$$

Then, $\hat{\Sigma} \models_s \mathcal{C}$.

Proof: We prove the strong satisfaction of the contract using Lemma 3. Let $(x, z, w) : E \to X \times Z \times W$ be a trajectory of $\hat{\Sigma}$. We have $(x(0), z(0)) \in \text{dom}(g) \subseteq G \times Z$, then $x(0) \in G$.

Now let us prove that the logical implication (7) is satisfied. Let $t \in E$, such that for all $s \in [0, t]$, $z(s) \in \mathcal{B}_\varepsilon(A_1)$ and $w(s) \in \mathcal{B}_\varepsilon(A_2)$, and let $m \in \mathbb{N}$ such that $\tau_m \leq t < \tau_{m+1}$.

For $k \in \{0, \ldots, m-1\}$, $(x(\tau_k), z(\tau_k)) \in \text{dom}(g)$ and there exists $u_k \in g(x(\tau_k), z(\tau_k))$ such that $u(s) = u_k$ for all $s \in [\tau_k, \tau_{k+1})$. Then, by (8), since for all $s \in [\tau_k, \tau_{k+1}]$, $z(s) \in \mathcal{B}_\varepsilon(A_1)$ and $w(s) \in \mathcal{B}_\varepsilon(A_2)$, we have for all $s \in [\tau_k, \tau_{k+1}]$, $x(s) \in G$. Hence, we have $x(s) \in G$, for all $s \in [0, \tau_m]$.

If $t = \tau_m$, then from above, we obtain directly that $x(s) \in G$, for all $s \in [0, t]$.

If $t > \tau_m$, then $(x(\tau_m), z(\tau_m)) \in \text{dom}(g)$ and there exists $u_m \in g(x(\tau_m), z(\tau_m))$ such that $u(t) = u_m$ for all $s \in [\tau_k, t]$. Let $\bar{w} : [\tau_m, \tau_{m+1}] \to W$ such that $\bar{w}(s) = w(s)$ for $s \in [\tau_m, t]$ and $\bar{w}(s) = w(t)$ for $s \in [t, \tau_{m+1}]$. Clearly, $\bar{w}$ is continuous. Then, from Assumption 3, there exists $(\bar{x}, \bar{z}) : [\tau_m, \tau_{m+1}] \to X \times Z$, solution to differential inclusion (4) with inputs $\bar{w}(s)$ and $u(s) = u_m$ for all $s \in [\tau_m, \tau_{m+1}]$, and such that for all $s \in [\tau_m, t]$, $(\bar{x}(s), \bar{z}(s)) = (x(s), z(s))$. Since for all $s \in [\tau_m, t]$, $w(s) \in \mathcal{B}_\varepsilon(A_2)$, we have for all $s \in [\tau_m, \tau_{m+1}]$, $\bar{w}(s) \in \mathcal{B}_\varepsilon(A_2)$. Moreover, for all $s \in [\tau_m, t]$, $\bar{z}(s) = z(s) \in \mathcal{B}_\varepsilon(A_1)$. It follows from (8) that $\bar{x}(s) \in G$, for all $s \in [\tau_m, t]$. Then, using the fact that for all $s \in [\tau_m, t]$, $\bar{x}(s) = x(s)$. We get that $x(s) \in G$, for all $s \in [0, t]$. ■

Intuitively, Proposition 3 states that there are essentially two ways to satisfy the assume-guarantee contract between two successive sampling instants. The first one is to enforce the guarantee on x on the whole sampling period, the second is to falsify the assumption on z between the sampling instants, while enforcing the guarantee until the falsification time.

Proposition 3 and Definition 4 provide sufficient conditions that the controller g has to satisfy in order to provide a solution to Problem 2. The main advantage of these conditions is that they make it possible to focus on the behavior of $\hat{\Sigma}$ over a sampling period.

B. Synthesis using the symbolic control approach

In this section, we design a control law $g : X \times Z \rightrightarrows U$, which is a solution to Problem 2, based on the conditions given in Proposition 3 and Definition 4. For that purpose, we use the symbolic control approach [1], [2] that relies on the use of symbolic models, which are discrete abstractions of the continuous dynamics given by differential inclusion (4).

1) Symbolic model: While classical symbolic approaches proposed in the literature [1], [2] allow to achieve specifications such as safety, reachability or LTL formulas, the novelty here is to deal with assume-guarantee contracts. Moreover, while in classical approaches, specifications need to be satisfied by the trajectory at sampling instants, in this paper we also deal with the intersampling behaviour by requiring contract satisfaction in continuous-time. In this part, we show how to design a symbolic model that guarantees by construction the fulfilment of the conditions of Proposition 3 for strong satisfaction of the assume-guarantee contract.

The *symbolic model* is given by a transition system $\mathcal{A} = (Q, V, \Delta)$ where Q and V are finite sets of symbolic states and inputs and $\Delta : Q \times V \rightrightarrows Q$ is a transition relation. In the following, we define formally each of these elements.

For a state $q \in Q$ of the symbolic model, the set of *enabled* inputs is $\text{enab}_\Delta(q) = \{v \in V |\ \Delta(q, v) \neq \emptyset\}$. The set of *non-blocking* states is $\text{nb}_\Delta = \{q \in Q |\ \text{enab}_\Delta(q) \neq \emptyset\}$.

a) Discretization: Our approach is based on a discretization of the sets of states and inputs:

- We start by constructing a finite partition of the continuous state-space $G \times A_1$. An element of the partition is denoted Y_q, $q \in Q_0$, where Q_0 is the index set of the partition ($\bigcup_{q \in Q_0} Y_q = G \times A_1$). Moreover, we are also using a special state q_{sink}. Hence, the set of symbolic states is given by $Q = Q_0 \cup \{q_{\text{sink}}\}$;
- The set of symbolic inputs V is a finite subset of U.

Intuitively, the special symbol q_{sink} is used to encode that the assumption on $z(t)$ has been falsified. As long as the assumption on $z(t)$ is verified, the symbolic state $q \in Q_0$ corresponds to states of $\hat{\Sigma}$ belonging to Y_q.

We define a quantizer $\lfloor\ .\ \rfloor_{Q_0} : G \times A_1 \to Q_0$ associated to the partition of $G \times A_1$:

$$\forall (x, z) \in G \times A_1,\ \big(\lfloor (x, z) \rfloor_{Q_0} = q \iff (x, z) \in Y_q\big).$$

b) Transition relation: To define the transition relation Δ, we rely on reachability analysis. We define the reachable set of differential inclusion (4) at time $s \in [0, \tau]$, from a set of initial states $Y_0 \subseteq X \times Z$, under the constant control input $u_0 \in U$, and input $w \in C([0, \tau], W^*)$ where $W^* \subseteq W$ as:

$$R_s(Y_0, u_0, W^*) = \left\{ (x(s), z(s)) \left| \begin{array}{c} (x,z):[0,\tau] \to X \times Z \\ \text{is a solution of (4) with} \\ (x(0),z(0)) \in Y_0, \\ u(t) = u_0,\ t \in [0,\tau], \\ w \in C([0,\tau], W^*) \end{array} \right. \right\}.$$

Similarly the reachable set of (4) on the time interval $[0, s]$, $s \in [0, \tau]$ is defined by

$$R_{[0,s]}(Y_0, u_0, W^*) = \bigcup_{t \in [0,s]} R_t(Y_0, u_0, W^*).$$

In the following, we assume that we are able to compute an over-approximation of the reachable sets denoted $\hat{R}_s(Y_0, u_0, W^*)$ and $\hat{R}_{[0,s]}(Y_0, u_0, W^*)$ for all $s \in [0, \tau]$. Several methods exist for the computation of such over-approximations, see e.g. [37] for linear systems, [38] for monotone systems or [39] for general nonlinear systems.

First, we give an intuitive explanation on which symbolic inputs should be enabled from a symbolic state $q \in Q_0$, in order to guarantee the strong satisfaction of the assume-guarantee contract. Let $\varepsilon > 0$, implication (8) is satisfied by any solution $(x, z) : [0, \tau] \to X \times Z$ of differential inclusion (4) with initial state in Y_q and with the constant control input value $v \in V$, if one of the following conditions is satisfied:

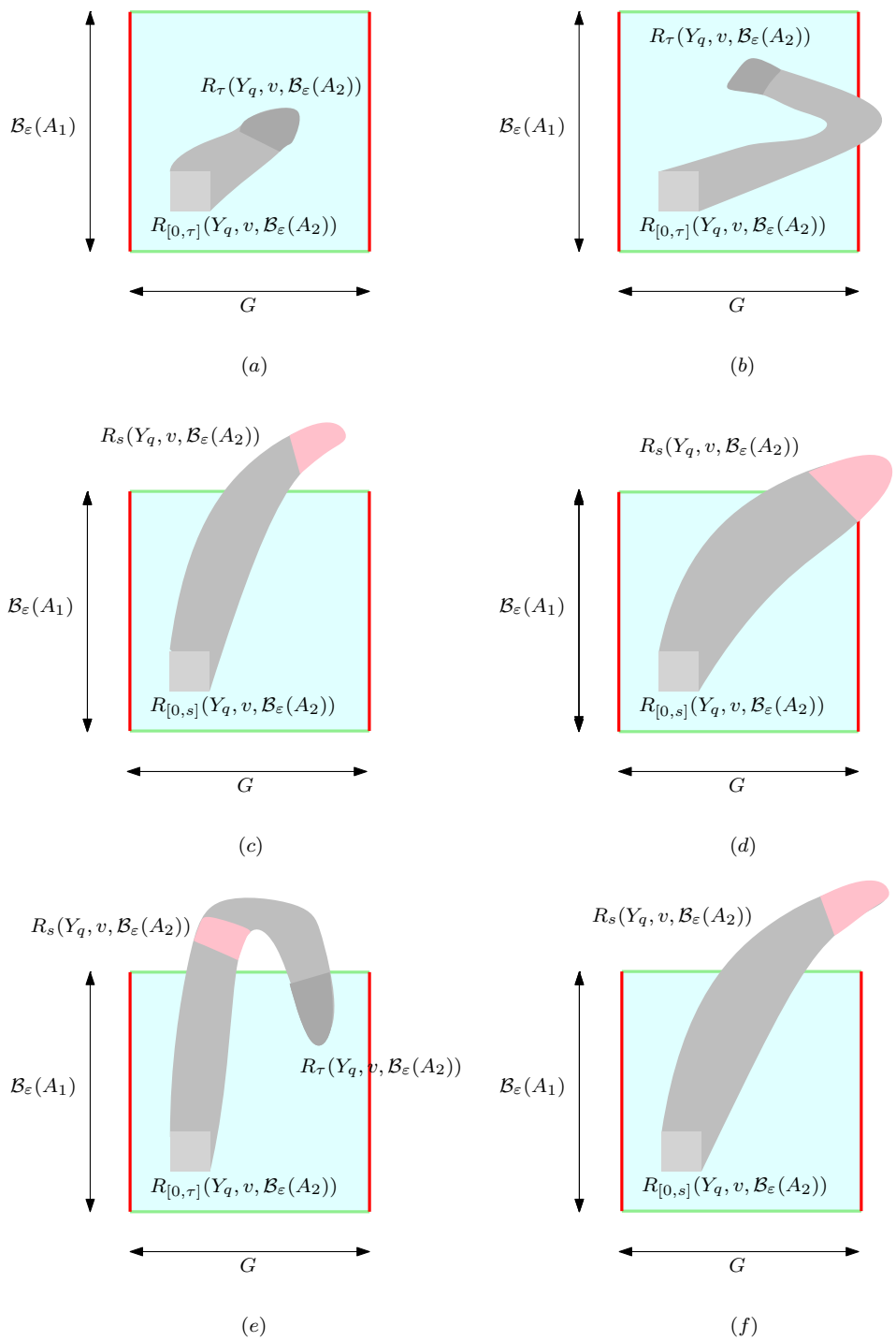

Fig. 3: Illustration of the possible transitions starting from a state $q \in Q$ under the input $v \in V$ while assuming that unknown inputs belong to $\mathcal{B}_\varepsilon(A_2)$. The set of initial states Y_q (light gray), the reachable set between between 0 and τ (or between 0 and s) (gray), the reachable set at time s (pink) and the reachable set at time τ (dark gray). (a) transitions to $q' \in Q_0$, (c) and (e) transitions to q_{sink}, (b), (d) and (f) no transition is created.

- $\hat{R}_{[0,\tau]}(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z$, in this case we are enforcing the guarantee on $[0, \tau]$ for all trajectories initiating in Y_q (see cases (a) and (e) in Figure 3);
- There exists $s \in [0, \tau]$ such that $\hat{R}_s(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \cap (X \times \mathcal{B}_\varepsilon(A_1)) = \emptyset$ and $\hat{R}_{[0,s]}(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z$; in this case, the assumption is falsified at time s, while the

guarantee is enforced on $[0, s]$ for all trajectories initiating in Y_q (see cases (c) and (e) in Figure 3).

Then, by enabling only such inputs, we ensure by Proposition 3 that the assume-guarantee contract will be strongly satisfied. Note that the two conditions are not mutually exclusive and when both conditions are satisfied (case (e) in Figure 3) we give priority to the second condition since the completeness condition (CC) is automatically satisfied in that case, the left part of implication (6) being falsified. There also exists cases where none of the conditions is satisfied (case (b), (d) and (f) in Figure 3) and in this case the symbolic input v should not be enabled from symbolic state q.

Hence, we formally define the transition relation $\Delta : Q \times V \rightrightarrows Q$ as follows:

- for $q \in Q$ and $v \in V$, $q_{\text{sink}} \in \Delta(q, v)$ if $q = q_{\text{sink}}$ or if there exists $s \in [0, \tau]$ such that
$$\begin{aligned} &\hat{R}_s(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \cap (X \times \mathcal{B}_\varepsilon(A_1)) = \emptyset \\ &\text{and } \hat{R}_{[0,s]}(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z, \end{aligned} \tag{9}$$
- for $q, q' \in Q_0$ and $v \in V$, $q' \in \Delta(q, v)$ if $q_{\text{sink}} \notin \Delta(q, v)$ and
$$\begin{aligned} &\hat{R}_{[0,\tau]}(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z \\ &\text{and } Y_{q'} \cap \hat{R}_\tau(Y_q, v, A_2) \neq \emptyset. \end{aligned} \tag{10}$$

The parameter $\varepsilon > 0$ used in the construction of the symbolic model is critical to ensure the strong satisfaction of the assume-guarantee contract using the criterion of Proposition 3. Interestingly, ε can be chosen to be arbitrarily small. Let us also mention that there exist systems for which contracts are satisfied with $\varepsilon = 0$ but not with any $\varepsilon > 0$, which is the case for the system described by: $\dot{x}(t) = -x(t) + w(t) + z(t)$ and the contract $\mathcal{C} = (\{0\}; [0, 1]; [0, 1])$.

Remark 8: Our construction of the transition relation differs from the one proposed in [31]. In that work, only transitions of type (10) are enabled. Then, assume-guarantee are satisfied only by enforcing the guarantee and the possibility of falsifying the assumption is not considered. This is done in the current work by enabling transitions of type (9).

Remark 9: Let us point out that in the construction of the transition relation (equations (9) and (10)), we are requiring the existence of a uniform s for all the trajectories initiating in Y_q such that the assumption is falsified at s and the guarantee is enforced on $[0, s]$, which is stronger than the requirement of the original contract, where different trajectories initiating in Y_q may falsify the assumption at different times. Checking the original contract would require advances reachability computations that are beyond the elementary ones used in the symbolic control community.

The following lemma establishes the formal behavioral relationship between the dynamics of $\mathcal{A}$ and $\hat{\Sigma}$:

Lemma 4: Under Assumption 3, let $\mathcal{C} = (A_1, A_2, G)$ be an assume-guarantee contract for $\hat{\Sigma}$ and let $\mathcal{A} = (Q, V, \Delta)$ be the associated symbolic model. Let $q \in Q_0 \cap \text{nb}_\Delta$, $v \in \text{enab}_\Delta(q)$, $w \in C([0, \tau], W)$ such that for all $t \in [0, \tau]$, $w(t) \in A_2$. Then for any solution $(x, z) : [0, \tau] \to X \times Z$ to differential inclusion (4) with $(x(0), z(0)) \in Y_q$, $u(t) = v$ for all $t \in [0, \tau]$ and such that $z(t) \in A_1$ for all $t \in [0, \tau]$, there exists $q' \in \Delta(q, u)$ such that $q' \in Q_0$ and $(x(\tau), z(\tau)) \in Y_{q'}$.

Proof: Since $z(t) \in A_1 \subseteq \mathcal{B}_\varepsilon(A_1)$ and $w(t) \in A_2 \subseteq \mathcal{B}_\varepsilon(A_2)$ for all $t \in [0, \tau]$, then $\hat{R}_s(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \cap (X \times \mathcal{B}_\varepsilon(A_1)) \neq \emptyset$ for all $s \in [0, \tau]$. Then, from the definition of Δ and since $v \in \text{enab}_\Delta(q)$, we get that (10) holds and thus $(x(\tau), z(\tau)) \in \hat{R}_{[0,\tau]}(Y_q, v, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z$. Moreover, using the fact that $z(t) \in A_1$ for all $t \in [0, \tau]$, we have that $(x(\tau), z(\tau)) \in G \times A_1$. Then, there exists $q' \in Q_0$ such that $(x(\tau), z(\tau)) \in Y_{q'}$. Moreover, since $(x(\tau), z(\tau)) \in \hat{R}_\tau(Y_q, v, A_2)$ we have $Y_{q'} \cap \hat{R}_\tau(Y_q, u_p, A_2) \neq \emptyset$ and thus by (10), $q' \in \Delta(q, p)$. ■

Intuitively, the previous Lemma shows that $\mathcal{A}$ is formally related to the uncontrolled (i.e. with $g(x) = U$ for all $x \in X$) dynamics at sampling times of $\hat{\Sigma}$ by some type of alternating simulation relation [1]. The main difference with usual alternating simulation relations is that the current relation is conditioned by the fact that the states $z(t)$ and $w(t)$ must belong for all time to sets of assumptions A_1 and A_2.

Finally, the next proposition provides a simple condition relating the control map g to be designed to the symbolic abstraction $\mathcal{A}$, which guarantees the strong satisfaction of assume-guarantee contracts:

Proposition 4: Under Assumption 3, if the control map $g : X \times Z \rightrightarrows U$ satisfies:
$$\begin{aligned} &\text{dom}(g) \subseteq G \times A_1, \\ &\forall (x, z) \in G \times A_1,\ g(x, z) \subseteq \text{enab}_\Delta\big(\lfloor (x, z) \rfloor_{Q_0}\big), \end{aligned} \tag{11}$$
then, $\hat{\Sigma} \models_s \mathcal{C}$.

Proof: We prove the strong satisfaction of the contract using Proposition 3. First, we have that $\text{dom}(g) \subseteq G \times A_1 \subseteq G \times Z$. Then, let $(x_0, z_0) \in \text{dom}(g)$, $u_0 \in g(x_0, z_0)$ and $w \in C([0, \tau], W)$ such that for all $t \in [0, \tau]$, $w(t) \in \mathcal{B}_\varepsilon(A_2)$. Let us consider a solution $(x, z) : [0, \tau] \to X \times Z$ to differential inclusion (4) with $(x(0), z(0)) = (x_0, z_0)$ and $u(t) = u_0$ for all $t \in [0, \tau]$. By (11), we have that $u_0 \in \text{enab}_\Delta(q_0)$ where $q_0 = \lfloor (x_0, z_0) \rfloor_{Q_0}$. Using the definition of the transition relation Δ, if condition (10) is satisfied, then $\hat{R}_{[0,\tau]}(Y_{q_0}, u_0, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z$ and we have for all $t \in [0, \tau]$, $x(t) \in G$. Else, if condition (9) is satisfied, then there exists $s \in [0, \tau]$ with $\hat{R}_s(Y_{q_0}, u_0, \mathcal{B}_\varepsilon(A_2)) \cap (X \times \mathcal{B}_\varepsilon(A_1)) = \emptyset$ and such that $\hat{R}_{[0,s]}(Y_{q_0}, u_0, \mathcal{B}_\varepsilon(A_2)) \subseteq G \times Z$. This implies the existence of $s' \in [0, s]$ such that $z(s') \notin \mathcal{B}_\varepsilon(A_1)$ and such that for all $t \in [0, s']$, $x(t) \in G$. Hence, implication (8) holds and we can conclude that $\hat{\Sigma} \models_s \mathcal{C}$. ■

2) Symbolic controller synthesis: In this part, we show how to design the control map g, solving Problem 2. For that purpose, we constrain the controller g to be designed to satisfy
$$\begin{aligned} &\text{dom}(g) \subseteq G \times A_1, \\ &\forall (x, z) \in G \times A_1,\ g(x, z) = \Theta\big(\lfloor (x, z) \rfloor_{Q_0}\big), \end{aligned} \tag{12}$$
where $\Theta : Q \rightrightarrows V$ is a symbolic controller to be synthesized for the abstraction $\mathcal{A}$.

We state the main result of this section:

Theorem 2: Under Assumption 3, let the symbolic controller $\Theta : Q \rightrightarrows V$ for the abstraction $\mathcal{A}$ satisfy:

$$\forall q \in Q,\ \Theta(q) \subseteq \text{enab}_\Delta(q), \tag{13}$$

$$\forall q \in \text{dom}(\Theta),\ \forall v \in \Theta(q),\ \Delta(q,v) \subseteq \text{dom}(\Theta). \tag{14}$$

Let the control map $g : X \times Z \rightrightarrows U$ be given by (12). Then, $\hat{\Sigma} \models_s \mathcal{C}$ and $\hat{\Sigma}$ satisfies (CC).

Proof: Let us remark that (12) and (13) imply that g satisfies the condition (11). Then, $\hat{\Sigma} \models_s \mathcal{C}$. To prove the second part of the theorem, we show that condition (6) holds. Let $(x_0, z_0) \in \text{dom}(g)$, $u_0 \in g(x_0, z_0)$ and $w \in C([0,\tau], W)$. Let us consider a solution $(x, z) : [0,\tau] \to X \times Z$ to differential inclusion (4) with $(x(0), z(0)) = (x_0, z_0)$ and $u(t) = u_0$ for all $t \in [0,\tau]$. By (12), $u_0 \in \Theta(q_0)$ where $q_0 = \lfloor (x_0, z_0) \rfloor_{Q_0}$. By (13), we get $u_0 \in \text{enab}_\Delta(q_0)$. Let us assume that for all $t \in [0,\tau]$, $z(t) \in A_1$ and for all $w(t) \in A_2$. Then, from Lemma 4, there exists $q' \in \Delta(q_0, u_0)$ such that $q' = \lfloor (x(\tau), z(\tau)) \rfloor_{Q_0}$. By, (14) we also get that $q' \in \text{dom}(\Theta)$, which in turn implies by (12) that $(x(\tau), z(\tau)) \in \text{dom}(g)$. Hence, the completeness condition (CC) is satisfied. ■

The previous result establishes conditions that the set-valued map $\Theta : Q \rightrightarrows V$ has to satisfy in order to solve Problem 2. Indeed these conditions actually state that Θ is a discrete safety controller for the abstraction $\mathcal{A}$ keeping the trajectories of $\mathcal{A}$ in nb_Δ. Thus, Θ can be synthesized by computing the maximal controlled invariant of $\mathcal{A}$ in nb_Δ, which can be done using maximal fixed point computation (see e.g. [1]).

C. Influence of the information structure

In this section, we investigate the influence of the information structure on the feasibility of Problem 2. We provide theoretical comparisons between different system abstractions $\hat{\Sigma}$ and $\hat{\Sigma}'$ obtained from different information structures given by maps π_0, π_1, π_2 and π'_0, π'_1 and π'_2 respectively. We assume that $\pi_0 = \pi'_0$, which means $\hat{\Sigma}$ and $\hat{\Sigma}'$ represent the system from the point of view of the same component. We also assume that there exists a bijective linear map α such that $\alpha = (\alpha_1, \alpha_2)$ and such that:

$$\forall x \in \mathbb{R}^n,\ \pi'_1(x) = \alpha_1(\pi_1(x)),\ \pi'_2(x) = (\pi_2(x), \alpha_2(\pi_1(x))).$$

This essentially means that the information on the state of the system received by the controller in $\hat{\Sigma}'$ is a subset of that received by the controller in $\hat{\Sigma}$. Let us remark that we have $X' = X$, $Z' = \alpha_1(Z)$ and $W' \subseteq W \times \alpha_2(Z)$. The following result relates the trajectories of $\hat{\Sigma}$ and $\hat{\Sigma}'$:

Lemma 5: Let g' be a control map for $\hat{\Sigma}'$ and let g be a control map for $\hat{\Sigma}$ given by

$$\forall (x,z) \in X \times Z,\ g(x,z) = g'(x, \alpha_1(z)). \tag{15}$$

Let us assume that the following equality holds:

$$W \times \alpha_2(Z) = W', \tag{16}$$

Then for any trajectory $(x, z, w) : E \to X \times Z \times W$ of $\hat{\Sigma}$, the trajectory $(x', z', w') : E \to X' \times Z' \times W'$ defined by $x' = x$, $z' = \alpha_1(z)$ and $w' = (w, \alpha_2(z))$ is a trajectory of $\hat{\Sigma}'$.

Proof: Let $u : E \to U$ be the control input of $\hat{\Sigma}$ associated to the trajectory (x, z, w). We have that for all $t \in E$, $x(t) \in X$ gives $x'(t) \in X' = X$, $z(t) \in Z$ gives $z'(t) \in Z' = \alpha_1(Z)$ and by (16), $w(t) \in W$ and $z(t) \in Z$ gives $w'(t) \in W \times \alpha_2(Z) = W'$. Then it is easy to check that if (x, z, w) satisfies differential inclusion (4), then (x', z', w') satisfies also (4), for the same control input u. Then, by (15), we have that for all $k \in \mathbb{N}$, with $\tau_k \in E$, $g(x(\tau_k), z(\tau_k)) = g'(x(\tau_k), \alpha_1(z(\tau_k))) = g'(x'(\tau_k), z'(\tau_k))$, it follows that u is also a control input for $\hat{\Sigma}'$ associated to (x', z', w'). Thus, (x', z', w') is a trajectory of $\hat{\Sigma}'$. ■

It should be noted that the technical condition given by (16) is needed to prove the result above. Intuitively, this condition states that by providing more information to $\hat{\Sigma}$, we do not remove some implicit information contained in $\hat{\Sigma}'$ about existing coupling between variables that would be induced by the constraint set W'.

Now, let $\mathcal{C} = (A_1, A_2, G)$ and $\mathcal{C}' = (A'_1, A'_2, G')$ be assume-guarantee contracts for $\hat{\Sigma}$ and $\hat{\Sigma}'$. In the following, we establish conditions on $\mathcal{C}$ and $\mathcal{C}'$ such that the feasibility of Problem 2 for $\hat{\Sigma}'$ and $\mathcal{C}'$ implies the feasibility of Problem 2 for $\hat{\Sigma}$ and $\mathcal{C}$.

Proposition 5: Let g' be a control map for $\hat{\Sigma}'$ and let g be a control map for $\hat{\Sigma}$ given by (15). Let us assume that (16) and the following inclusions hold:

$$\alpha_1(A_1) \subseteq A'_1,\ (A_2 \times \alpha_2(A_1)) \subseteq A'_2,\ G' \subseteq G. \tag{17}$$

Then, the following statements hold:

- If $\hat{\Sigma}' \models \mathcal{C}'$ then $\hat{\Sigma} \models \mathcal{C}$;
- If $\hat{\Sigma}'$ satisfies (CC) then so does $\hat{\Sigma}$.

Proof: Let us prove the first item. Let $(x, z, w) : E \to X \times Z \times W$ be a trajectory of $\hat{\Sigma}$ and let $(x', z', w') : E \to X' \times Z' \times W'$ be given by $x' = x$, $z' = \alpha_1(z)$ and $w' = (w, \alpha_2(z))$. By Lemma 5, (x', z', w') is trajectory if $\hat{\Sigma}'$. Then $\hat{\Sigma}' \models \mathcal{C}'$ gives that $x'(0) \in G'$ and by (17) $x(0) \in G$. Let $t \in E$ and let us assume that for all $s \in [0,t]$, $z(s) \in A_1$ and $w(s) \in A_2$. Then, by (17), for all $s \in [0,t]$, $z'(s) \in A'_1$ and $w'(s) \in A'_2$. Thus, $\hat{\Sigma}' \models \mathcal{C}'$, gives that there exists $\delta > 0$ such that $x'(s) \in G'$ for all $s \in [0, t+\delta] \cap E$. By (17), $x(s) \in G$, for all $s \in [0, t+\delta] \cap E$.

We now prove the second item. Let $(x_0, z_0) \in \text{dom}(g)$, $u_0 \in g(x_0, z_0)$ and $w \in C([0,\tau], W)$, let $(x, z) : [0,\tau] \to X \times Z$ be a solution to differential inclusion (4) with $u(t) = u_0$ for all $t \in [0,\tau]$. By the proof of Lemma 5, we have that (x', z') given by $x' = x$, $z' = \alpha_1(z)$ is a solution to differential inclusion (4) with $w' = (w, \alpha_2(z))$ and the same control input u. Let us assume that $z(t) \in A_1$ and $w(t) \in A_2$ for all $t \in [0,\tau]$ then by (17), for all $t \in [0,\tau]$, $z'(t) \in A'_1$ and $w'(t) \in A'_2$. Since $\hat{\Sigma}'$ satisfies (CC), then $(x(\tau), z(\tau)) \in \text{dom}(g)$ which by (15) gives $(x(\tau), \alpha_1(z(\tau))) \in \text{dom}(g')$ and therefore $(x'(\tau), z'(\tau)) \in \text{dom}(g')$. Thus, $\hat{\Sigma}$ satisfies (CC). ■

Proposition 5 explains how one should modify the abstraction and the contracts to reduce conservatism by providing more informations on the states of other components. Let us remark that by reducing the conservatism, we are increasing the dimension of differential inclusion (4) which renders the solution of the problem more complex.

V. Examples

In this section, we demonstrate the practicality of our approach on two control problems, a temperature regulation system and a vehicle platooning problem. The objective of the first example is to show the effect of the information structure in terms of conservatism and computational complexity, the construction of the symbolic model is based on a uniform partition of the state space (as in standard examples of symbolic control area). In the second example, we show how the proposed framework can be applied to a more complex example, for which standard uniform partitioning technique fails to find a solution. Moreover, we will also explore the effect of the multiperiodicity on vehicle platoons, which shows how the proposed approach is able to deal with heterogeneous components with different sampling periods. In the following, the numerical implementations has been done in MATLAB, Processor 2.7 GHz Intel Core i5, Memory 8 GB 1867 MHz DDR3.

A. Temperature regulation

In this part, we consider the problem of regulating the temperature in a circular building of $m \geq 3$ rooms, each one is equipped with a heater. The dynamics of room $i \in \{1, \ldots, m\}$ is given by the following differential equation:

$$\dot{T}_i = \alpha(T_{i+1} + T_{i-1} - 2T_i) + \beta(T_e - T_i) + \gamma(T_h - T_i)u_i \quad (18)$$

where T_{i+1} and T_{i-1} are the temperature of the neighbour rooms (here $T_0 = T_m$ and $T_{m+1} = T_m$), T_e is the external temperature, T_h is the temperature of the heater, u_i is the control input to room i and α, β and γ are the conduction factors. The numerical parameters are taken from [31] and shown in Table I:

TABLE I: Room parameters

Parameter	Value	Unit
T_e	-1	C
T_h	50	C
u_i	$[0, 0.6]$	.
α	0.45	.
β	0.045	.
γ	0.09	.

Given a safe set $S = S_1 \times S_2 \times \ldots \times S_m \subseteq \mathbb{R}^m$, it can be seen that requirements of Assumption 1 are satisfied. To compare the effect of the information structure on the conservatism, we consider 2 possible information structures:

- Totally decentralized case (TD): for $i \in \{1, \ldots, m\}$ and $T = (T_1, \ldots, T_m)$, $\pi_{i,1}(T) = \{0\}$ and $\pi_{i,2}(T) = (T_1, \ldots, T_{i-1}, T_{i+1}, \ldots, T_m)$. In this case, a room has no knowledge about the temperatures of the other rooms.
- Partially decentralized case (PD): for $i \in \{1, \ldots, m\}$, and $T = (T_1, \ldots, T_m)$, $\pi_{i,1}(T) = (T_{i-1}, T_{i+1})$ and $\pi_{i,2}(T) = (T_1, \ldots, T_{i-2}, T_{i+2}, \ldots, T_m)$. In this case, the temperatures of the neighbouring rooms are accessible from the room i.

For each room $i \in \{1, \ldots, m\}$, we construct an abstraction $\hat{\Sigma}_i$ as presented in Section III-A to which we assign an assume-guarantee contract $\mathcal{C}_i = (A_{i,1}, A_{i,2}, G_i)$, where $A_{i,1} = \pi_{i,1}(S)$, $A_{i,2} = \pi_{i,2}(S)$ and $G_i = S_i$. We use the symbolic approach presented in Section IV-B1 to construct a symbolic model $\mathcal{A}$ of $\hat{\Sigma}_i$ which guarantees by design the strong satisfaction of the contract $\mathcal{C}_i$. Then, a controller Θ is synthesized for $\mathcal{A}$, using the approach presented in IV-B2. The controller Θ is then refined into a controller $g_i : X_i \times Z_i \rightrightarrows U_i$ for the abstraction $\hat{\Sigma}_i$ ensuring the satisfaction of the (CC) condition. Then, using Theorem 1 one can ensure that the whole safety objective for the interconnected system is achieved. In the following we report numerical results for $m = 4$ and $S = [17, 22] \times [19, 22] \times [19, 23] \times [19, 24]$. The parameter of the construction of the transitions is $\varepsilon = 0.1$, the sampling period $\tau = 1s$ and supposed to be the same for all rooms and the values of the symbolic model parameters are $n_u = 3$, $n_d = 5$ per dimension (the number of states for a symbolic model in the totally decentralized case is 5, while in the partially decentralized case, the number of states is 5^3 since we are modelling the neighbouring rooms). Table II reports the percentage of controllable states and the computation time for generating the symbolic model and synthesizing the controller. The comparisons are also done with the centralized case (C).

TABLE II: Percentage of controllable states and computation time

	% of controllable states	Computation time(s)
C	99	15000
PD	98	40
TD	0	2

Table II shows that partially decentralized approach is a compromise in terms of conservatism and computational complexity between the centralized and the totally decentralized approach. Particularly, it can be seen that when the totally decentralized approach fails to find a controller, the partially decentralized case is able to find one, which is compatible with the theoretical results presented in Section IV-C. Another interesting remark is that the domain of the controller in the partially decentralized case, is almost the same as in the centralized one with a significant reduction of the computation time.

B. Vehicle platooning

Vehicle platoons are groups of autonomous vehicles traveling closely. Platooning makes it possible to reduce traffic congestion while increasing safety and fuel efficiency. Symbolic control techniques have previously been applied to the design of autonomous vehicles. In [40], [41], symbolic controllers have been designed for adaptive cruise control of a single vehicle. The paper [42] deals with distributed symbolic controller synthesis for a vehicle platoon. However, it is assumed in that work that: all vehicles are identical; sampling in all vehicles is synchronous; the vehicle platoon is on a straight road (the case of circular roads is not considered).

1) Model description: In the following, each vehicle in the platoon is modeled as a nonlinear and nonsmooth control system. We shall adapt the model from [43]:

$$M\dot{v} = \alpha(F, v) = \begin{cases} F - f_0 - f_1 v - f_2 v^2 & \text{if} \quad v > 0 \\ \max(F - f_0, 0) & \text{if} \quad v = 0 \end{cases} \quad (19)$$

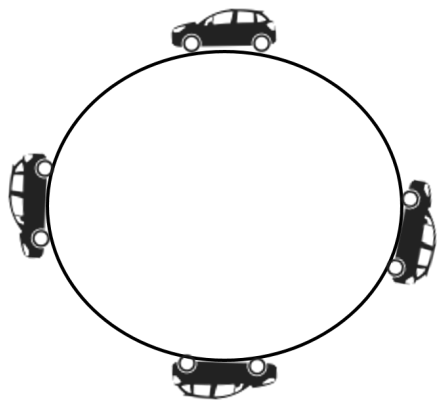

Fig. 4: A platoon of 4 vehicles on a circular road.

where $M > 0$ represent the mass of the vehicle, v its velocity, F is the net engine torque applied to the wheels and the term $f_0+f_1v+f_2v^2$ include the rolling resistance and aerodynamics ($f_0, f_1, f_2 \in \mathbb{R}^+$). In this equation, F is the control input and satisfies $F \in [F_{\min}, F_{\max}]$, where $F_{\min} < 0 < F_{\max}$.

Contrarily to [43], we have added the second equation to eliminate the unrealistic behaviour where the vehicle is moving backward (i.e $v(t) \geq 0$ for all $t \in \mathbb{R}_0^+$).

In this paper, we deal with a platoon of vehicles in a circular road (see Figure 4). In a platoon of m vehicles on a circular road, the dynamic of each vehicle $i \in \{1, \ldots, m\}$ is given by:

$$\begin{cases} \dot{d}_i &= v_{i-1} - v_i \\ M\dot{v}_i &= \alpha(F_i, v_i). \end{cases} \tag{20}$$

with the convention that $v_0 = v_m$, where $d_i \geq 0$ represents the relative distance between vehicle i and the preceding vehicle $i-1$, v_i its velocity and F_i its control input.

Remark 10: We assume that all vehicles are identical only to keep notations simple. However, our approach can be extended directly to heterogeneous vehicles with α_i depending on the vehicle parameters.

2) Problem formulation and solution strategy: Our goal is to synthesize controllers, giving values of input F_i, for all vehicles of a platoon such that the velocity of each vehicle remains between 0 and $v_{\max}$, and the relative distance between two vehicles remains larger than $d_{\min} \geq 0$.

$$\forall i \in \{1, \ldots, m\},\ \forall t \in \mathbb{R}_0^+,\ v_i(t) \in [0, v_{\max}] \\ \text{and}\ \ d_i(t) \in [d_{\min}, +\infty) \tag{21}$$

Let the safe set $S = S_1 \times \ldots \times S_m$, where $S_i = [d_{\min}, +\infty) \times [0, v_{\max}]$. In this example, we only show the results for the partially decentralized case, where each vehicle knows the velocity of its preceding one (for $i \in \{1, \ldots, m\}$ and $x = (d, v) = (d_1, v_1, \ldots, d_m, v_m)$, $\pi_{i,1}(x) = v_{i-1}$ and $\pi_{i,2}(x) = (d_1, v_1, \ldots, d_{i-1}, d_{i+1}, v_{i+1}, \ldots, d_m, v_m)$). For each vehicle we construct an abstraction $\hat{\Sigma}_i$ as presented in Section III-A to which we assign an assume-guarantee contract $\mathcal{C}_i = (A_{i,1}, A_{i,2}, G_i)$, where $A_{i,1} = \pi_{i,1}(S)$, $A_{i,2} = \pi_{i,2}(S)$ and $G_i = S_i$. We use the symbolic approach presented in Section IV-B1 to construct a symbolic model $\mathcal{A}$ of $\hat{\Sigma}_i$ which guarantees by design the strong satisfaction of the contract $\mathcal{C}_i$. Then, a controller Θ is synthesized for $\mathcal{A}$, using the approach presented in IV-B2. The controller Θ is then refined into a controller $g_i : X_i \times Z_i \rightrightarrows U_i$ for the system Σ_i ensuring the satisfaction of the (CC) condition. Then, using Theorem 1 one can ensure that the whole safety objective for the vehicle platoon is achieved. First we explain the partitioning technique used for this problem. To improve readability, the index $i \in I$ is dropped in the following.

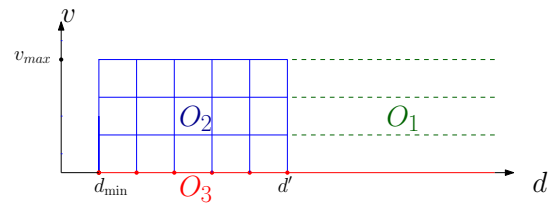

Fig. 5: Partition of $G_X = [d_{\min}, +\infty) \times [0, v_{\max}]$ with $n_d = 5$ and $n_v = 3$.

3) Symbolic model: Given the state space $G \times Z = [d_{\min}, +\infty) \times [0, v_{\max}] \times [0, v_{\max}]$. For the sake of simplicity, we explain the construction of the symbolic model on the set $S = [d_{\min}, +\infty) \times [0, v_{\max}]$ and for a fixed value $w \in [0, v_{\max}]$ of the velocity for the preceding vehicle. However, the same reasoning applied to the velocity v of the controlled vehicle is applied to w when constructing the symbolic model. Let $d' > d_{\min}$, we have that $S = O_1 \cup O_2 \cup O_3$, where: $O_1 = [d', +\infty) \times (0, v_{\max}]$, $O_2 = [d_{\min}, d'] \times (0, v_{\max}]$ and $O_3 = [d_{\min}, +\infty) \times \{0\}$, as shown in figure 5. Using n_v and n_d as abstraction parameters for velocity and distance axis respectively, partitions of O_1, O_2 and O_3 are constructed as follows:

- We use unbounded regions for the partition of the set O_1. Let us remark that this is necessary to cover the unbounded state space S with a finite number of subsets;
- We construct a partition of O_2 using a uniform grid;
- We use regions with empty interior (flat symbols) for the set O_3. This is necessary to discriminate the case when the velocity is 0 from the case when it belongs to $(0, v_{\max}]$. For instance, if the leading vehicle stops and remains motionless, it is necessary to stop the following vehicle. Not being able to discriminate the case when the velocity is 0 from the case when it is (even slightly) positive would result in uncontrollable symbolic abstraction. Moreover, the partition of the set O_3 contains an unbounded region corresponding to $[d', +\infty) \times \{0\}$.

Using similar ideas, we can construct the abstraction of the whole state space $G \times Z = [d_{\min}, +\infty) \times [0, v_{\max}] \times [0, v_{\max}]$ by using the following 6 regions: $O_1 = [d', +\infty) \times (0, v_{\max}] \times (0, v_{\max}]$, $O_2 = [d', +\infty) \times (0, v_{\max}] \times \{0\}$, $O_3 = [d_{\min}, d'] \times (0, v_{\max}] \times (0, v_{\max}]$, $O_4 = [d_{\min}, d'] \times (0, v_{\max}] \times \{0\}$, $O_5 = [d_{\min}, +\infty) \times \{0\} \times (0, v_{\max}]$ and $O_6 = [d_{\min}, +\infty) \times \{0\} \times \{0\}$.

Remark 11: We can see that our partition differs from the

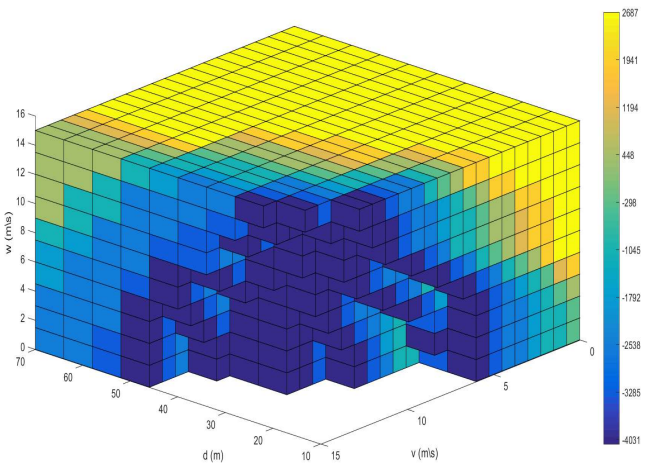

Fig. 6: Synthesized control map g.

classical partitions used in the literature. Indeed the problem cannot be solved using a uniform partition for two reasons: First, the state space is unbounded, and second because we have to discriminate the case for which $v = 0$ from the case where $v > 0$.

The input space $U = [F_{\min}, F_{\max}]$ is uniformly discretized into $n_u = 10$ values. The transition relation is constructed based on (9) and (10) where we used the monotonicity of the system to construct an overapproximation of the reachable set.

4) Numerical results: In this section, we illustrate our results using numerical simulations. We use the numerical values from [40] for the vehicle parameters. These values as well as the safety parameters are shown in Table III.

TABLE III: Vehicle and safety parameters

Parameter	Value	Unit
M	1370	Kg
f_0	51.0709	N
f_1	0.3494	Ns/m
f_2	0.4161	Ns^2/m^2
F_{min}	-4031.9	mKg/s^2
F_{max}	2687.9	mKg/s^2
d_{max}	-10	m
v_{max}	15	m/s

We compute the symbolic abstraction $\mathcal{A}$ using the approach described in Section IV-B1, with the partition technique presented in Section V-B3. For discrete controller synthesis, the maximal fixed point computation allows us to determine the most permissive safety controller. The controller Θ is obtained after determinization of the most permissive safety controller by selecting the maximal safe input. Intuitively, it means that the vehicles drive as fast as possible while guaranteeing satisfaction of assume-guarantee contracts.

Figure 6 represents the resulting controller g for sampling period $\tau = 2$, parameter of the construction of the transition relation $\varepsilon = \frac{v_{\max}}{1000}$ and the following values of abstraction parameters: $n_u = 10$, $d' = 70$, $n_d = 10$, $n_v = 20$ and $n_{v'} = 10$. The computation time for generating the symbolic abstraction and synthesizing the controller is about 5 minutes.

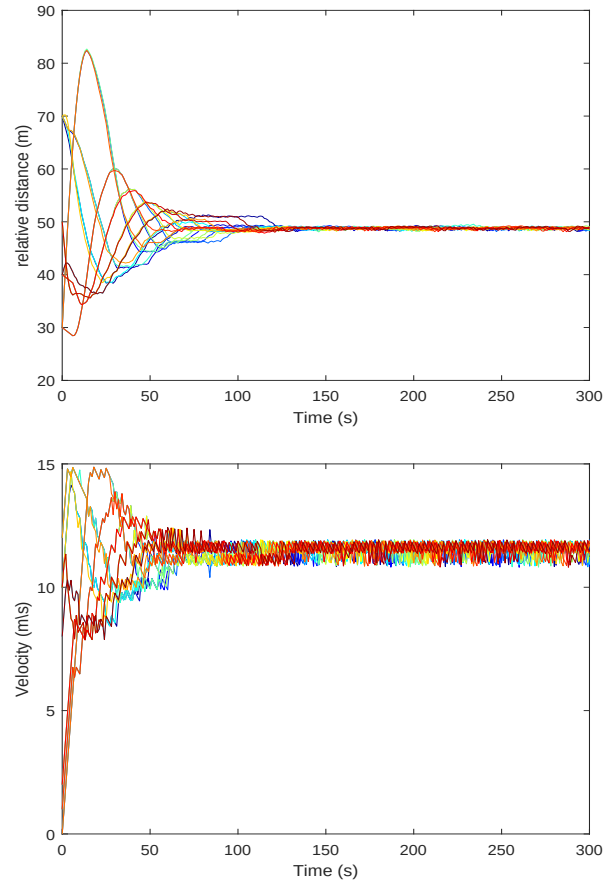

Fig. 7: Simulation results of a platoon of 20 vehicles on a circular road with the same sampling period: inter-vehicle distance (top), velocities (bottom).

The choice of the abstraction parameters is important, of course the larger n_u, n_d, n_v and $n_{v'}$, the more accurate the abstraction. Conversely, small values of these parameters may lead to uncontrollable abstractions (i.e. the maximal controlled invariant of $\mathcal{A}$ is empty).

For numerical simulations, we consider a platoon of 20 vehicles. We consider identical vehicles, with parameters given by Table III, to emphasize the effect of the sampling periods. However the same approach can be applied even if we have heterogeneous vehicles.

a) Periodic sampling: We consider that all the vehicles have the same sampling period and abstraction parameters. Note that these parameters are the same as the ones used for computing the controller shown on Figure 6.

Figure 7 shows the simulation results for given initial conditions. One can check that distances between vehicles are always greater than 10 m and that velocities remain between 0 and 15 m/s at all time, so the overall objective is satisfied. It is interesting to remark that after a transient period, the vehicles distribute themselves uniformly on the road (i.e. the distances between vehicles are all equal) and drive at almost

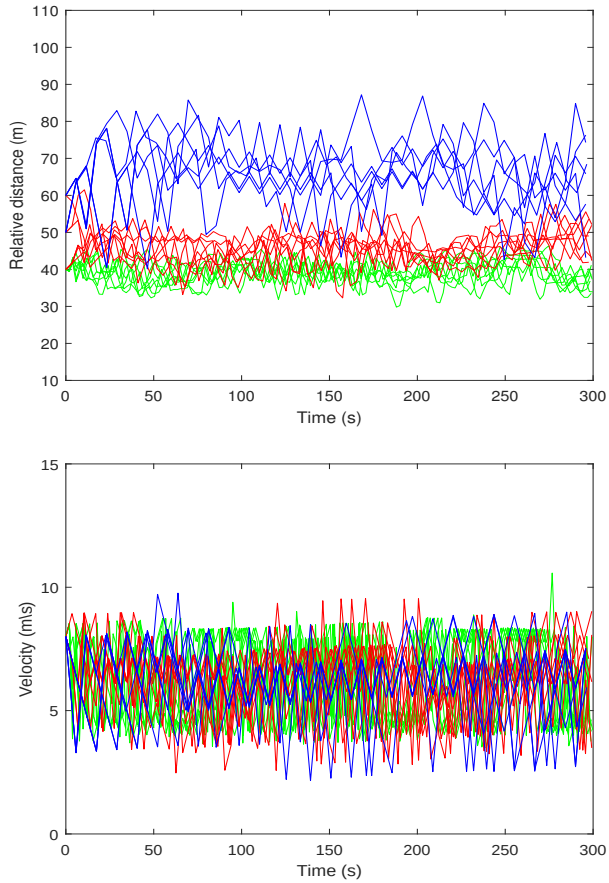

Fig. 8: Simulation results of a platoon of 20 vehicles on a circular road with different sampling periods: inter-vehicle distance (top), velocities (bottom)(green: vehicles with different sampling periods in $[1.9, 2.02]$, red: vehicles with different sampling periods in $[3.4, 3.52]$, blue: vehicles with different sampling periods in $[5.7, 5.8]$).

constant speed.

b) Multiperiodic sampling: We consider 20 vehicles with different sampling periods, where 7 vehicles have the sampling periods in $[1.9, 2.02]$, 7 vehicles have the sampling periods in $[3.4, 3.52]$ and 6 vehicles have their sampling periods in $[5.7, 5.8]$.

Figure 8 shows the simulation results. One can check that distances between vehicles are always greater than $10\ m$ and that velocities remain between 0 and $15\ m/s$ at all time, so the overall objective is satisfied despite multiperiodic sampling. Similar to the periodic sampling case, we remark that after a transient period, the vehicles drive at almost constant speed. However, it is interesting to note that the final speed is smaller than in the periodic sampling case. An even more significant difference is seen on the inter-vehicle distances. Indeed, the vehicles do not distribute uniformly on the road. On this simulation, one can see that the vehicles with larger sampling period need to keep a larger distance to the front vehicle, which can be explained by the fact, that they need more time to react.

VI. Conclusion

In this paper, we have presented a compositional approach to the design of distributed safety controllers for continuous-time nonlinear systems, based on a notion of continuous-time assume-guarantee contracts. This approach makes it possible to decompose a global safety control problem into local ones that can be solved independently. Symbolic control techniques are then used to synthesize controllers enforcing the local control objectives. The proposed approach makes it possible to deal with heterogeneous components where controllers have different sampling periods and receive partial information on the state of other components. The overall picture of the paper is depicted in Figure 9. Illustrative applications in building automation and vehicle platooning are shown. In future work we will develop more general contracts allowing to extend the approach to other types of specifications, such as reachability, stability or more general properties described by temporal logic formula.

Appendix

Proof of Lemma 1

Proof: Let $x : E \to X$ be a maximal trajectory of Σ. Let us assume that x is not complete. We consider three distinct cases.

Case 1 - $E = [0, a]$ with $a \geq 0$:
Then, let

$$\overline{\tau} = \min\{\tau_{i,k} > a |\ i \in I,\ k \in \mathbb{N}\}.$$

Intuitively, $\overline{\tau}$ is the first sampling instant after a. We have $\overline{\tau} - a \leq \tau$, then it follows from Assumption 2 that x can be extended on $[0, \overline{\tau})$ with $u(t) = u(a)$ for all $t \in [a, \overline{\tau})$.

Case 2 - $E = [0, a)$ with $a > 0$ and $a \neq \tau_{i,k+1}$, for all $i \in I$ and $k \in \mathbb{N}$:
Then, let us define

$$\underline{\tau} = \max\{\tau_{i,k} < a |\ i \in I,\ k \in \mathbb{N}\},$$
$$\overline{\tau} = \min\{\tau_{i,k} > a |\ i \in I,\ k \in \mathbb{N}\}.$$

Intuitively, $\underline{\tau}$ and $\overline{\tau}$ are the last sampling instant before a and the first sampling instant after a, respectively. We have $\overline{\tau} - \underline{\tau} \leq \tau$, then it follows from Assumption 2 that x can be extended on $[0, \overline{\tau})$ with $u(t) = u(\underline{\tau})$ for all $t \in [\underline{\tau}, \overline{\tau})$.

Case 3 - $E = [0, a)$ with $a > 0$ and there exists $i \in I$ and $k \in \mathbb{N}$, such that $a = \tau_{i,k+1}$:

It follows from Assumption 2, that the limit $x(a^-)$ exists and belongs to X. Also, $u(a^-)$ exists and belongs to U because u is piecewise constant. Then, let us assume that for all $i \in I$ and $k \in \mathbb{N}$, such that $a = \tau_{i,k+1}$, we have $(x_i(\tau^-_{i,k+1}), z_i(\tau^-_{i,k+1})) \in \mathrm{dom}(g_i)$. Let us show that x and u can be extended to $[0, a]$. Firstly, x can be extended by continuity $x(a) = x(a^-)$. Then, for all $i \in I$ and $k \in \mathbb{N}$, such that $a = \tau_{i,k+1}$, let $u_{i,k} \in g_i(x_i(\tau^-_{i,k+1}), z_i(\tau^-_{i,k+1}))$, and $u_i(a) = u_{i,k}$. For $i \in I$ such that $a \neq \tau_{i,k+1}$, for all $k \in \mathbb{N}$, let $u_i(a) = u_i(a^-)$. Then, for all $i \in I$, $u_i(a) \in U_i$, which by Assumption 1 gives $u(a) \in U$. One can then check that the extended map x defined on $[0, a]$ satisfies Definition 1 for the input function u defined on $[0, a]$.

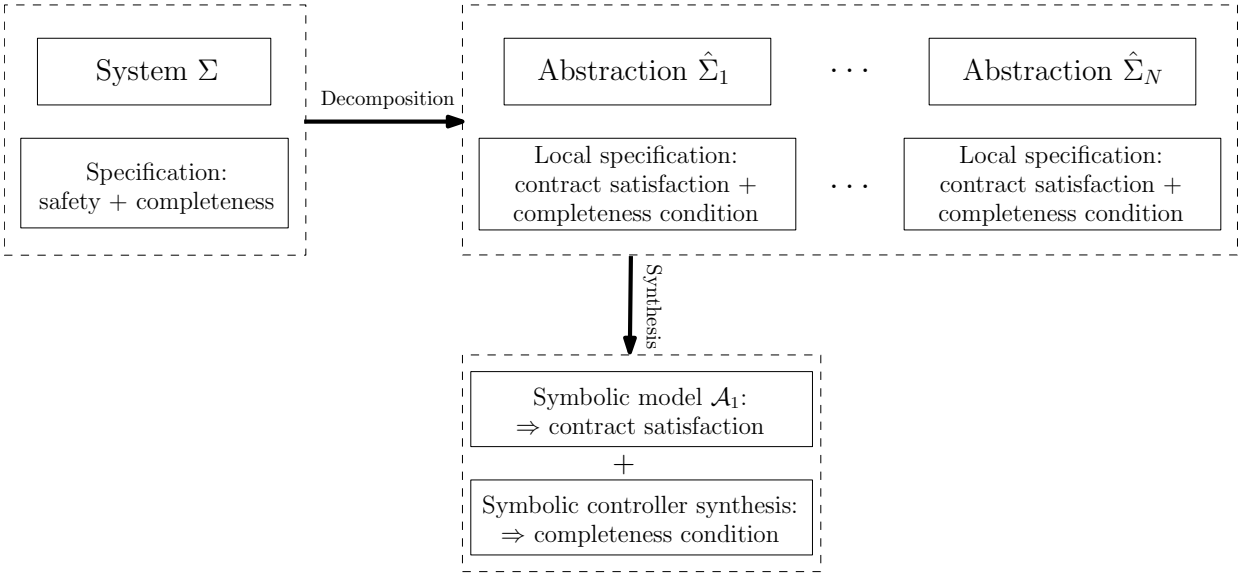

Fig. 9: Schematic description of the compositional synthesis approach presented in the paper.

Hence, the first two cases lead to a contradiction of the maximality of x. The third case also leads to a contradiction unless there exists $i \in I$ and $k \in \mathbb{N}$, such that $a = \tau_{i,k+1}$, and $(x_i(\tau^-_{i,k+1}), z_i(\tau^-_{i,k+1})) \notin \mathrm{dom}(g_i)$. ■

References

[1] P. Tabuada, *Verification and control of hybrid systems: a symbolic approach*. Springer Science & Business Media, 2009.

[2] C. Belta, B. Yordanov, and E. Gol, *Formal methods for discrete-time dynamical systems*. Springer, 2017.

[3] C. Cassandras and S. Lafortune, *Introduction to discrete event systems*. Springer Science & Business Media, 2009.

[4] R. Bloem, B. Jobstmann, N. Piterman, A. Pnueli, and Y. Sa'ar, "Synthesis of reactive (1) designs," *Journal of Computer and System Sciences*, vol. 78, no. 3, pp. 911–938, 2012.

[5] A. Girard, G. Pola, and P. Tabuada, "Approximately bisimilar symbolic models for incrementally stable switched systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 116–126, 2010.

[6] G. Reißig, "Computing abstractions of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2583–2598, 2011.

[7] M. Zamani, G. Pola, M. Mazo, and P. Tabuada, "Symbolic models for nonlinear control systems without stability assumptions," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1804–1809, 2012.

[8] S. Coogan and M. Arcak, "Finite abstraction of mixed monotone systems with discrete and continuous inputs," *Nonlinear Analysis: Hybrid Systems*, vol. 23, pp. 254–271, 2017.

[9] Y. Tazaki and J.-i. Imura, "Discrete-state abstractions of nonlinear systems using multi-resolution quantizer," pp. 351–365, 2009.

[10] A. Girard, G. Gössler, and S. Mouelhi, "Safety controller synthesis for incrementally stable switched systems using multiscale symbolic models," *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1537–1549, 2016.

[11] K. Hsu, R. Majumdar, K. Mallik, and A.-K. Schmuck, "Multi-layered abstraction-based controller synthesis for continuous-time systems," in *Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week)*. ACM, 2018, pp. 120–129.

[12] E. Le Corronc, A. Girard, and G. Goessler, "Mode sequences as symbolic states in abstractions of incrementally stable switched systems," in *IEEE Conference on Decision and Control (CDC)*, 2013, pp. 3225–3230.

[13] M. Zamani, A. Abate, and A. Girard, "Symbolic models for stochastic switched systems: A discretization and a discretization-free approach," *Automatica*, vol. 55, pp. 183–196, 2015.

[14] A. Weber, M. Rungger, and G. Reissig, "Optimized state space grids for abstractions," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5816–5821, 2017.

[15] A. Saoud and A. Girard, "Multirate symbolic models for incrementally stable switched systems," in *Proceedings of the* 20th *IFAC World Congress, Toulouse, France*, 2017.

[16] Y. Tazaki and J.-i. Imura, "Bisimilar finite abstractions of interconnected systems," *Hybrid Systems: Computation and Control*, pp. 514–527, 2008.

[17] M. Rungger and M. Zamani, "Compositional construction of approximate abstractions of interconnected control systems," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 116–127, 2018.

[18] G. Pola, P. Pepe, and M. Di Benedetto, "Symbolic models for networks of control systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3663–3668, November 2016.

[19] N. Noroozi, F. R. Wirth, and M. Zamani, "Compositional construction of abstractions via relaxed small-gain conditions part i: continuous case," in *European Control Conference (ECC)*, 2018, pp. 76–81.

[20] N. Noroozi, A. Swikir, F. R. Wirth, and M. Zamani, "Compositional construction of abstractions via relaxed small-gain conditions part ii: discrete case," in *European Control Conference (ECC)*, 2018, pp. 1–4.

[21] A. Swikir and M. Zamani, "Compositional synthesis of finite abstractions for networks of systems: A small-gain approach," *Automatica*, vol. 107, pp. 551–561, 2019.

[22] M. Zamani and M. Arcak, "Compositional abstraction for networks of control systems: A dissipativity approach," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1003–1015, 2017.

[23] A. U. Awan and M. Zamani, "Compositional abstraction of interconnected control systems under dynamic interconnection topology," in *IEEE Conference on Decision and Control (CDC)*, 2017, pp. 3543–3550.

[24] A. Swikir, A. Girard, and M. Zamani, "From dissipativity theory to compositional synthesis of symbolic models," in *Indian Control Conference (ICC)*, 2018, pp. 30–35.

[25] O. Hussien, A. Ames, and P. Tabuada, "Abstracting partially feedback linearizable systems compositionally," *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 227–232, Oct 2017.

[26] A. Saoud, P. Jagtap, M. Zamani, and A. Girard, "Compositional abstraction-based synthesis for cascade discrete-time control systems," *6th IFAC Conference on Analysis and Design of Hybrid Systems ADHS*, vol. 51, no. 16, pp. 13 – 18, 2018.

[27] K. Mallik, A.-K. Schmuck, S. Soudjani, and R. Majumdar, "Compositional synthesis of finite state abstractions," *IEEE Transactions on Automatic Control*, vol. 64, no. 6, pp. 2629–2636, 2018.

[28] E. Dallal and P. Tabuada, "On compositional symbolic controller synthesis inspired by small-gain theorems," in *IEEE Conference on Decision and Control (CDC)*, 2015, pp. 6133–6138.

[29] E. Kim, M. Arcak, and S. Seshia, "Compositional controller synthesis for vehicular traffic networks," in *IEEE Conference on Decision and Control (CDC)*, 2015, pp. 6165–6171.

[30] A. Le Coënt, L. Fribourg, N. Markey, F. De Vuyst, and L. Chamoin, "Distributed synthesis of state-dependent switching control," in *International Workshop on Reachability Problems*, 2016, pp. 119–133.

[31] P.-J. Meyer, A. Girard, and E. Witrant, "Compositional abstraction and safety synthesis using overlapping symbolic models," *IEEE Transactions on Automatic Control*, vol. 63, no. 6, pp. 1835–1841, 2018.

[32] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. Henzinger, and K. Larsen, "Contracts for systems design: Theory," Inria Rennes Bretagne Atlantique; INRIA, Tech. Rep., 2015.

[33] A. Saoud, A. Girard, and L. Fribourg, "On the composition of discrete and continuous-time assume-guarantee contracts for invariance," in *2018 European Control Conference (ECC)*, June 2018, pp. 435–440.

[34] ——, "Contract based design of symbolic controllers for interconnected multiperiodic sampled-data systems," in *IEEE Conference on Decision and Control (CDC)*, Dec 2018, pp. 773–779.

[35] J.-P. Aubin and A. Cellina, *Differential inclusions: set-valued maps and viability theory*. Springer Science & Business Media, 2012, vol. 264.

[36] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. New Jersey: Princeton University Press, 2012.

[37] C. Le Guernic and A. Girard, "Reachability analysis of linear systems using support functions," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 250–262, 2010.

[38] N. Ramdani, N. Meslem, and Y. Candau, "Computing reachable sets for uncertain nonlinear monotone systems," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 263–278, 2010.

[39] G. Reissig, A. Weber, and M. Rungger, "Feedback refinement relations for the synthesis of symbolic controllers," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1781–1796, 2017.

[40] P. Nilsson, O. Hussien, Y. Chen, A. Balkan, M. Rungger, A. Ames, J. Grizzle, N. Ozay, H. Peng, and P. Tabuada, "Preliminary results on correct-by-construction control software synthesis for adaptive cruise control," pp. 816–823, 2014.

[41] P. Nilsson, O. Hussien, A. Balkan, Y. Chen, A. Ames, J. Grizzle, N. Ozay, H. Peng, and P. Tabuada, "Correct-by-construction adaptive cruise control: Two approaches," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1294–1307, 2016.

[42] A. Borri, D. Dimarogonas, K. Johansson, M. Di Benedetto, and G. Pola, "Decentralized symbolic control of interconnected systems with application to vehicle platooning," *IFAC Proceedings Volumes*, vol. 46, no. 27, pp. 285–292, 2013.

[43] P. Ioannou and C.-C. Chien, "Autonomous intelligent cruise control," *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 657–672, 1993.

PLACE PHOTO HERE

Adnane Saoud Adnane Saoud is a Postdoctoral Research Fellow in the Electrical and Computer Engineering Department at the University of California, Santa Cruz Since 2020. He received the Ph.D. degree in Control from CentraleSupelec, France, in 2019. During his Ph.D. studies, he was selected as one of the three finalists for the Best student paper award at the European Control Conference, ECC, 2018. He obtained the M.Sc. degree in control from University Paris-Saclay, France, in 2016, and Electrical Engineering degree from Mohammadia School of Engineers, Rabat, Morocco, in 2014. His current research interests include formal methods for cyber–physical systems and compositional analysis and synthesis of interconnected systems.

PLACE PHOTO HERE

Antoine Girard Antoine Girard is a Senior Researcher at CNRS and a member of the Laboratory of Signals and Systems. He received the Ph.D. degree in applied mathematics from Grenoble Institute of Technology, in 2004. From 2004 to 2006, he held postdoctoral positions at University of Pennsylvania and Université Grenoble-Alpes. From 2006 to 2015, he was an Assistant/Associate Professor at the Université Grenoble-Alpes. His main research interests deal with analysis and control of hybrid systems with an emphasis on computational approaches, formal methods and applications to cyber-physical systems. He received the George S. Axelby Outstanding Paper Award from the IEEE Control Systems Society in 2009. In 2014, he was awarded the CNRS Bronze Medal. In 2015, he was appointed as a junior member of the Institut Universitaire de France (IUF). In 2016, he was awarded an ERC Consolidator Grant. In 2018, he received the first HSCC Test of Time Award and the European Control Award.

PLACE PHOTO HERE

Laurent Fribourg Laurent Fribourg received a civil engineering degree in 1980 from Aeronautics School Sup'Aero/Toulouse, and a PhD in computer science in 1982 from University Paris 7. He was appointed by CNRS in 1984 as a full-time researcher. He cofounded the CNRS laboratory of computer science LSV at ENS Cachan (now ENS Paris-Saclay) in 1997, and became head of the CNRS interdisciplinary "Institut Farman" at ENS Paris-Saclay in 2018. He has written more than 100 papers in the area of formal methods, several of them in collaboration with industrial partners.