Automatica 45 (2009) 566–571

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Hierarchical control system design using approximate simulation[☆]

Antoine Girard[a,*], George J. Pappas[b]

[a] Laboratoire Jean Kuntzmann, Université de Grenoble, B.P. 53, 38041 Grenoble Cedex 9, France

[b] Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA

ARTICLE INFO

Article history:
Received 3 December 2007
Received in revised form
30 April 2008
Accepted 3 September 2008
Available online 20 December 2008

Keywords:
Hierarchical control
Abstraction
Approximate simulation
Simulation functions

ABSTRACT

In this paper, we present a new approach for hierarchical control based on the recent notions of approximate simulation and simulation functions, a quantitative version of the simulation relations. Given a complex system that needs to be controlled and a simpler abstraction, we show how the knowledge of a simulation function allows us to synthesize hierarchical control laws by first controlling the abstraction and then lifting the abstract control law to the complex system using an interface. For the class of linear control systems, we give an effective characterization of the simulation functions and of the associated interfaces. This characterization allows us to use algorithmic procedures for their computation. We show how to choose an abstraction for a linear control system such that our hierarchical control approach can be used. Finally, we show the effectiveness of our approach on an example.

1. Introduction

Controlling complex (e.g. nonlinear and/or high-order) systems in order to achieve sophisticated tasks constitutes one of the great challenges of modern engineering. Handling at once both complexities of the dynamics and of the specification often leads to untractable problems and therefore a hierarchical approach to control synthesis is highly desirable. A hierarchical control architecture has (at least) two layers. The first layer consists of a precise (and complex) model of the plant that needs to be controlled and is usually referred to as the *concrete system*. The second layer consists of a coarse (and simple) model of the plant that is used for control synthesis and is referred to as the *abstract system* or *abstraction*. The main challenge of such approaches is the refinement of control laws designed for the abstract system in order to control the concrete system. The older techniques, such as aggregation (Aoki, 1968) or the inclusion principle (Ikeda, Siljak, & White, 1984) impose the condition that the concrete and abstract systems are driven by the same inputs. In the past decade, approaches based on the notion of simulation relations, have been proposed for computing consistent abstractions driven by refinable abstract inputs (Pappas, Lafferriere, & Sastry, 2000; Tabuada & Pappas, 2005).

Recently, the notion of simulation function, a quantitative version of the simulation relation, has been introduced in Girard and Pappas (2007). This generalization is more appropriate for systems whose state space is equipped with a natural topology such as continuous systems. In this framework, it is not required that the trajectories of the system and of its abstraction match exactly but only approximately. This relaxation allows us to consider simpler abstractions thus simplifying the design of high level control tasks.

In this paper, we present a hierarchical control framework based on this notion of approximate simulation. Given a complex system that needs to be controlled and a simpler abstraction, we show how the knowledge of a simulation function allows us to synthesize hierarchical control laws. For the class of linear control systems, we give an effective characterization of the simulation functions allowing us to use algorithmic procedures for their computation. We show how to choose an abstraction for a linear control system so that our hierarchical control approach is possible. Finally, we show the effectiveness of our approach on an example.

Preliminary versions of the results presented in this paper appeared in Girard and Pappas (2006) and Girard and Pappas (2007). An application of our hierarchical control framework to motion planning problems with temporal logic specifications has been discussed in Fainekos, Girard, and Pappas (2007). An approach with similar flavors has been proposed in Cavarischia and Lanari (2007a,b) for hierarchical stabilization and tracking control. In this paper, we consider continuous abstractions of

☆ This paper was not presented at any IFAC meeting. This research is partially supported by the ANR SETIN project VAL-AMS and by the NSF ITR Grant 0121431. This paper was recommended for publication in revised form by Associate Editor Zhihua Qu under the direction of Editor Hassan K. Khalil.

* Corresponding author. Tel.: +33 476514342; fax: +33 476631263.
E-mail addresses: Antoine.Girard@imag.fr (A. Girard), pappasg@seas.upenn.edu (G.J. Pappas).

0005-1098/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2008.09.016

continuous systems; however, a similar philosophy can be used for hierarchical control using discrete abstractions (Pola, Girard, & Tabuada, 2007; Tabuada, 2007; Tazaki & Imura, 2008).

2. A framework for hierarchical control

Let us consider the control system given by:

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = g(\mathbf{x}(t)) \end{cases}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^p$, $\mathbf{y}(t) \in \mathbb{R}^k$. We assume that $f : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^k$ are continuous maps. An *admissible input* for Σ is a locally measurable map $\mathbf{u} : \mathcal{I} \to \mathbb{R}^p$ where the interval $\mathcal{I} \subseteq \mathbb{R}^+$ contains 0. A *state trajectory* of Σ is an absolutely continuous map $\mathbf{x} : \mathcal{I} \to \mathbb{R}^n$ such that there exists an admissible input $\mathbf{u} : \mathcal{I} \to \mathbb{R}^p$ satisfying: $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$ for almost every $t \in \mathcal{I}$. We assume that the vector field f is such that for any admissible input, for any initial condition in $\mathbb{R}^n$, this differential equation has a unique solution (see e.g. Sontag (1998) for sufficient conditions). An *output trajectory* of Σ is a map $\mathbf{y} : \mathcal{I} \to \mathbb{R}^k$ such that there exists a state trajectory $\mathbf{x} : \mathcal{I} \to \mathbb{R}^n$ satisfying $\mathbf{y}(t) = g(\mathbf{x}(t))$ for all $t \in \mathcal{I}$.

Throughout the paper, we will refer to Σ as the *concrete system*, that is the (complex) system that we actually want to control. Control will be synthesized hierarchically, using an *abstract system*, that is a system giving a simpler, though less precise, description of the dynamics of Σ. This abstract system will be denoted by Σ' defined by:

$$\Sigma' : \begin{cases} \dot{\mathbf{z}}(t) = h(\mathbf{z}(t), \mathbf{v}(t)) \\ \mathbf{w}(t) = k(\mathbf{z}(t)) \end{cases}$$

where $\mathbf{z}(t) \in \mathbb{R}^m$, $\mathbf{v}(t) \in \mathbb{R}^q$, $\mathbf{w}(t) \in \mathbb{R}^k$. We make the same assumptions on h and k than on f and g. Note that systems Σ and Σ' have the same observations space (i.e. $\mathbb{R}^k$), but may have different state and input spaces.

Simulation functions have been introduced in Girard and Pappas (2007) as a quantitative generalization of the notion of simulation relations. Initially developed for discrete systems (Milner, 1989), simulation relations have been extended in the latest decade to continuous systems (Pappas et al., 2000; Tabuada & Pappas, 2005). Intuitively, a simulation relation of Σ' by Σ is a relation over their state spaces explaining how a state trajectory of Σ' can be transformed into a state trajectory of Σ with identical output trajectories. Similarly, a simulation function of Σ' by Σ is a function over their state spaces explaining how a state trajectory of Σ' can be transformed into a state trajectory of Σ such that the distance between the associated output trajectories remains within some computable bounds. Formally, a simulation function is defined as follows:

Definition 1. Let $\mathcal{V} : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^+$ be a smooth function and $u_{\mathcal{V}} : \mathbb{R}^q \times \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^p$ be a continuous function. $\mathcal{V}$ is a simulation function of Σ' by Σ and $u_{\mathcal{V}}$ is an associated interface if there exists a $\mathcal{K}$ function[1] γ such that for all $(z, x) \in \mathbb{R}^m \times \mathbb{R}^n$,

$$\mathcal{V}(z, x) \geq \|k(z) - g(x)\| \tag{1}$$

and for all $v \in \mathbb{R}^q$, satisfying $\gamma(\|v\|) < \mathcal{V}(z, x)$,

$$\frac{\partial \mathcal{V}(z, x)}{\partial z} \cdot h(z, v) + \frac{\partial \mathcal{V}(z, x)}{\partial x} \cdot f(x, u_{\mathcal{V}}(v, z, x)) < 0. \tag{2}$$

[1] A function $\gamma : \mathbb{R}^+ \to \mathbb{R}^+$ is a $\mathcal{K}$ function if it is continuous, strictly increasing and satisfies $\gamma(0) = 0$.

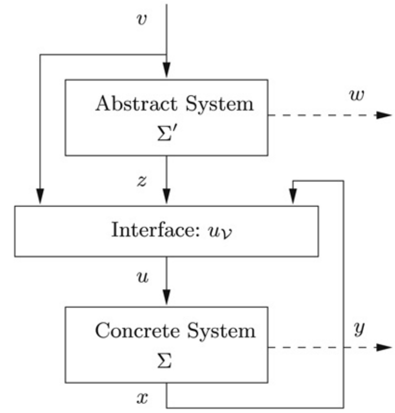

Fig. 1. Hierarchical control system architecture.

A simulation function allows us to bound the distance between output trajectories of Σ' and Σ.

Theorem 1. *Let $\mathcal{V}$ be a simulation function of Σ' by Σ and $u_{\mathcal{V}}$ an associated interface. Let $\mathbf{v} : \mathcal{I} \to \mathbb{R}^+$ be an admissible input of Σ', let $\mathbf{z}$ and $\mathbf{w}$ be the associated state and output trajectories of Σ'. Let $\mathbf{x}$ be a state trajectory of Σ satisfying the differential equation*

$$\dot{\mathbf{x}}(t) = f\left(\mathbf{x}(t), u_{\mathcal{V}}(\mathbf{v}(t), \mathbf{z}(t), \mathbf{x}(t))\right) \tag{3}$$

and let $\mathbf{y}$ be the associated output trajectory. Then, for all $t \in \mathcal{I}$,

$$\|\mathbf{w}(t) - \mathbf{y}(t)\| \leq \max\left\{\mathcal{V}(\mathbf{z}(0), \mathbf{x}(0)), \gamma(\|\mathbf{v}\|_\infty)\right\}.$$

Before giving the proof of Theorem 1, we should point out that Eq. (3) involves a feedback composition. In the following, we shall assume that this composition is well defined: for any initial state there exists a unique solution to differential equation (3) defined on the interval $\mathcal{I}$.

Proof. Let $\delta = \max\{\mathcal{V}(\mathbf{z}(0), \mathbf{x}(0)), \gamma(\|\mathbf{v}\|_\infty)\}$, we shall prove that for all $t \in \mathcal{I}$, $\mathcal{V}(\mathbf{z}(t), \mathbf{x}(t)) \leq \delta$. Initially, it is clear that $\mathcal{V}(\mathbf{z}(0), \mathbf{x}(0)) \leq \delta$. Assume there exists $\tau \in \mathcal{I}$, $\tau > 0$ such that $\mathcal{V}(\mathbf{z}(\tau), \mathbf{x}(\tau)) > \delta$. Then, there exists $0 \leq \tau' < \tau$ such that $\mathcal{V}(\mathbf{z}(\tau'), \mathbf{x}(\tau')) = \delta$ and for all $t \in (\tau', \tau]$, $\mathcal{V}(\mathbf{z}(t), \mathbf{x}(t)) > \delta$. Then, for all $t \in (\tau', \tau]$, $\gamma(\|\mathbf{v}(t)\|) \leq \gamma(\|\mathbf{v}\|_\infty) \leq \delta < \mathcal{V}(\mathbf{z}(t), \mathbf{x}(t))$. It follows from Eq. (2) that

$$\forall t \in (\tau', \tau], \quad \frac{\mathrm{d}\mathcal{V}(\mathbf{z}(t), \mathbf{x}(t))}{\mathrm{d}t} < 0.$$

Therefore,

$$\mathcal{V}(\mathbf{z}(\tau), \mathbf{x}(\tau)) - \mathcal{V}(\mathbf{z}(\tau'), \mathbf{x}(\tau')) = \int_{\tau'}^{\tau} \frac{\mathrm{d}\mathcal{V}(\mathbf{z}(t), \mathbf{x}(t))}{\mathrm{d}t} < 0.$$

This contradicts $\mathcal{V}(\mathbf{z}(\tau), \mathbf{x}(\tau)) > \delta = \mathcal{V}(\mathbf{z}(\tau'), \mathbf{x}(\tau'))$. Therefore, we proved by contradiction that for all $t \in \mathcal{I}$, $\mathcal{V}(\mathbf{z}(t), \mathbf{x}(t)) \leq \delta$. We conclude by remarking, from Eq. (1), that for all $t \in \mathcal{I}$,

$$\|\mathbf{w}(t) - \mathbf{y}(t)\| = \|k(\mathbf{z}(t)) - g(\mathbf{x}(t))\| \leq \mathcal{V}(\mathbf{z}(t), \mathbf{x}(t)) \leq \delta. \quad \blacksquare$$

The control architecture, allowing us to refine the inputs the abstract system through the interface $u_{\mathcal{V}}$ is shown on Fig. 1.

Definition 2. If there exists a simulation function of Σ' by Σ, then we say that Σ approximately simulates Σ'.

Let us remark that Σ approximately simulating Σ' is equivalent to the composite system

$$\Sigma_e : \begin{cases} \dot{\mathbf{z}}(t) = h(\mathbf{z}(t), \mathbf{v}(t)) \\ \dot{\mathbf{x}}(t) = f\left(\mathbf{x}(t), u_{\mathcal{V}}(\mathbf{v}(t), \mathbf{z}(t), \mathbf{x}(t))\right) \\ \mathbf{e}(t) = k(\mathbf{z}(t)) - g(\mathbf{x}(t)) \end{cases}$$

being input to output stable (where $\mathbf{v}$ is the input and $\mathbf{e}$ is the output). The simulation function $\mathcal{V}$ is an input to output stability Lyapunov function for that system (Sontag & Wang, 2000). Synthesizing the interface $u_{\mathcal{V}}$ can be viewed as finding a controller rendering the composite system input to output stable. However, in such problems, one generally regards the input $\mathbf{v}$ as an unknown disturbance. On the contrary, in our setting, $\mathbf{v}$ is a control input which can be used by the controller. There are also similarities with control problems such as the regulator problem (Wonham, 1979) or the more general asymptotic model matching problem (Di Benedetto & Grizzle, 1994). However, asymptotic model matching requires the distance between $\mathbf{y}$ and $\mathbf{w}$ to converge asymptotically to 0 which is equivalent to the composite system Σ_e being output globally asymptotically stable (Ingalls, Sontag, & Wang, 2002). In our setting, it is only needed that the distance between $\mathbf{y}$ and $\mathbf{w}$ remains within some computable bounds. Therefore, asymptotic model matching implies Σ approximately simulating Σ', however the converse is not true.

The applicability of our approach relies on our capability of computing a simulation function and an associated interface. In the following section, we give an effective characterization of simulation functions for linear control systems allowing us to design algorithmic procedures for their computation.

3. Simulation functions for linear systems

In the following, we assume that the concrete and abstract systems are linear control systems:

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) = C\mathbf{x}(t) \end{cases}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^p$, $\mathbf{y}(t) \in \mathbb{R}^k$ and

$$\Sigma' : \begin{cases} \dot{\mathbf{z}}(t) = F\mathbf{z}(t) + G\mathbf{v}(t) \\ \mathbf{w}(t) = H\mathbf{z}(t) \end{cases}$$

where $\mathbf{z}(t) \in \mathbb{R}^m$, $\mathbf{v}(t) \in \mathbb{R}^q$, $\mathbf{w}(t) \in \mathbb{R}^k$. We assume, without loss of generality, that $\text{rank}(B) = p$ and $\text{rank}(C) = k$. Since Σ' is typically simpler than Σ, we assume that $m \leq n$.

We shall further assume that the concrete system Σ is stabilizable. Then, there exists a $p \times n$ matrix K such that the matrix $A+BK$ is Hurwitz. The proof of the following lemma is omitted here but can be found in Girard and Pappas (2007).

Lemma 1. *There exists a positive definite symmetric matrix M and a strictly positive scalar number λ such that the following matrix inequalities hold:*

$$M \geq C^{\mathrm{T}} C, \tag{4}$$

$$(A + BK)^{\mathrm{T}} M + M(A + BK) \leq -2\lambda M. \tag{5}$$

The stabilizing gain K and the matrix M can be computed jointly using semidefinite programming. Indeed, denoting $\bar{K} = KM^{-1}$, $\bar{M} = M^{-1}$ and using Schur complements, (4) and (5) are equivalent to the following linear matrix inequalities:

$$\begin{bmatrix} \bar{M} & \bar{M}C^{\mathrm{T}} \\ C\bar{M} & I_k \end{bmatrix} \geq 0 \tag{6}$$

where I_k denotes the $k \times k$ identity matrix, and

$$\bar{M}A^{\mathrm{T}} + A\bar{M} + \bar{K}^{\mathrm{T}} B^{\mathrm{T}} + B\bar{K} \leq -2\lambda \bar{M}. \tag{7}$$

We now give an effective characterization of simulation functions and of the associated interfaces.

Theorem 2. *Let us assume there exists an $n \times m$ matrix P and a $p \times m$ matrix Q such that the following linear matrix equations hold:*

$$PF = AP + BQ, \tag{8}$$

$$H = CP. \tag{9}$$

Then, the function defined by

$$\mathcal{V}(z, x) = \sqrt{(Pz - x)^{\mathrm{T}} M (Pz - x)}$$

is a simulation function of Σ' by Σ and an associated interface is given by

$$u_{\mathcal{V}}(v, z, x) = Rv + Qz + K(x - Pz).$$

where R is an arbitrary $p \times q$ matrix.

Proof. From Eqs. (4) and (9), we have that

$$\mathcal{V}(z, x) \geq \sqrt{(x - Pz)^{\mathrm{T}} C^{\mathrm{T}} C (x - Pz)} = \|Cx - Hz\|.$$

Then, Eq. (1) holds. Now, let us show that Eq. (2) holds as well. We have

$$\frac{\partial \mathcal{V}(z,x)}{\partial z}(Fz + Gv) + \frac{\partial \mathcal{V}(z,x)}{\partial x}(Ax + Bu_{\mathcal{V}}) = \frac{(x - Pz)^{\mathrm{T}} M (Ax + B(Rv + Qz + K(x - Pz)) - P(Fz + Gv))}{\sqrt{(x - Pz)^{\mathrm{T}} M (x - Pz)}}.$$

From Eq. (8), it follows that

$$\frac{\partial \mathcal{V}(z,x)}{\partial z}(Fz + Gv) + \frac{\partial \mathcal{V}(z,x)}{\partial x}(Ax + Bu_{\mathcal{V}}) = \frac{(x - Pz)^{\mathrm{T}} M ((A + BK)(x - Pz) + (BR - PG)v)}{\sqrt{(x - Pz)^{\mathrm{T}} M (x - Pz)}}.$$

Using Eq. (5), we can show that

$$\frac{\partial \mathcal{V}(z,x)}{\partial z}(Fz + Gv) + \frac{\partial \mathcal{V}(z,x)}{\partial x}(Ax + Bu_{\mathcal{V}}) \leq \frac{-\lambda(x - Pz)^{\mathrm{T}} M (x - Pz) + (x - Pz)^{\mathrm{T}} M (BR - PG)v}{\sqrt{(x - Pz)^{\mathrm{T}} M (x - Pz)}} \leq -\lambda \mathcal{V}(z, x) + \|\sqrt{M}(BR - PG)v\|.$$

Let γ be the $\mathcal{K}$ function defined by

$$\gamma(\nu) = \frac{\left\|\sqrt{M}(BR - PG)\right\|}{\lambda} \nu. \tag{10}$$

Then,

$$\frac{\partial \mathcal{V}(z,x)}{\partial z}(Fz + Gv) + \frac{\partial \mathcal{V}(z,x)}{\partial x}(Ax + Bu_{\mathcal{V}}) \leq \lambda\left(-\mathcal{V}(z, x) + \gamma(\|v\|)\right).$$

Therefore, for all $v \in \mathbb{R}^q$ such that $\gamma(\|v\|) < \mathcal{V}(z, x)$, Eq. (2) holds. Then, $\mathcal{V}$ is a simulation function of Σ' by Σ, $u_{\mathcal{V}}$ is an associated interface. ■

Let us remark that if the abstract system Σ' is fed with a zero input (i.e. $\mathbf{v} = 0$), then our approach solves the regulator problem (Wonham, 1979) and the simulation function $\mathcal{V}$ provides a Lyapunov function proving asymptotic stability of the output $\mathbf{e} = \mathbf{w} - \mathbf{y}$. It is to be noted that the linear matrix equations (8) and (9) are key ingredients in the resolution of the regulator problem. Conditions on the matrices A, B, C, F and H that guarantee the existence of matrices P and Q such that Eqs. (8) and (9) hold can be found in Hautus (1983).

The matrix R can be chosen arbitrarily and thus constitutes a free design parameter of the interface. It can be used, for instance, to minimize the $\mathcal{K}$ function γ. The minimization of γ is important for the following reason. From Theorem 1, it is clear that the smaller the function γ, the smaller we can expect the distance between the output trajectories of Σ and Σ'.

Proposition 1. *The function γ defined by Eq. (10) is minimal for $R = (B^{\mathrm{T}}MB)^{-1}B^{\mathrm{T}}MPG$.*

Proof. The minimization of the function γ over R is equivalent to

$$\min_{R\in\mathbb{R}^{p\times q}} \left\| \sqrt{M}(BR - PG) \right\|.$$

This is clearly a least squares approximation problem. Then, since $\operatorname{rank}(B) = p$ and M is definite positive, it follows that the matrix $B^{\mathrm{T}}MB$ is invertible and the minimal γ is reached for $R = (B^{\mathrm{T}}MB)^{-1}B^{\mathrm{T}}MPG$. ■

Remark 1. The minimization of the function γ over all the design variables (i.e. K, M, λ, P, Q and R) is a complex, non-convex optimization problem that is out of the scope of this paper.

For practical implementation of the hierarchical control architecture, it is desirable that a uniformly bounded input $\mathbf{v}$ of Σ' results in a uniformly bounded input $\mathbf{u}$ of Σ. A sufficient condition is given in the following proposition:

Proposition 2. *Let Eqs. (8) and (9) hold with $Q = 0$, let $\mathcal{V}$ and $u_{\mathcal{V}}$ be given as in Theorem 2. Let $x_0 \in \mathbb{R}^n$, $z_0 \in \mathbb{R}^m$ be initial states of Σ and Σ' and $\nu \in \mathbb{R}^+$. Then, for all inputs $\mathbf{v}$ of Σ' such that, $\|\mathbf{v}\|_\infty \leq \nu$; the associated input $\mathbf{u}$ of Σ given by the interface $u_{\mathcal{V}}$ satisfies:*

$$\|\mathbf{u}\|_\infty \leq \|R\|\nu + \|K\sqrt{M^{-1}}\| \max\{\mathcal{V}(z_0, x_0), \gamma(\nu)\}.$$

Proof. Since $Q = 0$, $\mathbf{u}(t) = R\mathbf{v}(t) + K(\mathbf{x}(t) - P\mathbf{z}(t))$. Therefore,

$$\|\mathbf{u}(t)\| \leq \|R\|\|\mathbf{v}(t)\| + \|K\sqrt{M^{-1}}\|\mathcal{V}(\mathbf{z}(t), \mathbf{x}(t)).$$

From the proof of Theorem 1, we have the inequality $V(\mathbf{z}(t), \mathbf{x}(t)) \leq \max\{\mathcal{V}(z_0, x_0), \gamma(\nu)\}$ which allows us to conclude. ■

Remark 2. The assumption $Q = 0$ in the previous proposition essentially means that the matrix F is similar to the restriction of A on an A-invariant subspace given by $\operatorname{im}(P)$. This condition can be enforced when the choice of the abstract system Σ' is free.

4. Abstraction design for linear systems

In the previous sections, we assumed that the abstract system Σ' was given a priori. It is clear that this is generally not the case. In this section, we give a procedure to compute an abstract system enabling the hierarchical control approach described previously.

4.1. Π-related systems

A "good" abstraction should preserve all the behaviors of the concrete system; for that purpose we use the notion of Π-related systems introduced in Pappas et al. (2000).

Definition 3. Let Π be a $m \times n$ matrix, Σ' is Π-related to Σ if the following hold: for all $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, there exists $v \in \mathbb{R}^q$ such that

$$\Pi(Ax + Bu) = F\Pi x + Gv, \tag{11}$$

and

$$C = H\Pi. \tag{12}$$

In Pappas et al. (2000), it is shown that (11) holds if and only if for any state trajectory $\mathbf{x}$ of Σ, there exists a state trajectory $\mathbf{z}$ of Σ' such that for all t, $\mathbf{z}(t) = \Pi\mathbf{x}(t)$. Together with Eq. (12), it implies the following result:

Theorem 3. *If Σ' is Π-related to Σ, then for any output trajectory $\mathbf{y}$ of Σ, $\mathbf{w} = \mathbf{y}$ is an output trajectory of Σ'.*

In our abstraction procedure, we want to determine an injective abstraction map P and then a system Σ' such that the linear matrix equations (8) and (9) hold for some matrix Q, in order to use Σ' in our hierarchical control approach. Further, we want that Σ' is Π-related to Σ for some matrix Π so that we do not disregard controllable behaviors of Σ when working on Σ'.

Remark 3. Our abstraction procedure uses two maps, P and Π. It is simpler to derive a characterization of the map P. Π is essentially a pseudo-inverse of P, though not necessarily the Moore–Penrose pseudo-inverse.

4.2. Characterization of abstraction maps

In the following, I_d denotes the $d \times d$ identity matrix. We start by remarking that Eq. (8) is equivalent to $AP = PF - BQ$. Then, the following result is straightforward.

Lemma 2. *Let P be an injective map, there exists matrices F and Q satisfying the linear matrix equation (8) if and only if*

$$\operatorname{im}(AP) \subseteq \operatorname{im}(P) + \operatorname{im}(B). \tag{13}$$

Equivalently, this means that $\operatorname{im}(P)$ needs to be an (A, B)-controlled invariant subspace (Wonham, 1979). Eq. (9) gives us the matrix $H = CP$. Eq. (12) states that there must exist a matrix Π such that $C = H\Pi$.

Lemma 3. *Let P be an injective map, let $H = CP$. Then, there exists Π such that $C = H\Pi$ if and only if*

$$\operatorname{im}(P) + \ker(C) = \mathbb{R}^n. \tag{14}$$

Moreover, Π can be chosen such that $\Pi P = I_m$.

Proof. Let us assume that there exists Π such that $C = H\Pi$. Then, since $H = CP$, we have $C = CP\Pi$ and $C(I_n - P\Pi) = 0$. For all $x \in \mathbb{R}^n$, we have $x = P\Pi x + (I_n - P\Pi)x$. Since $P\Pi x \in \operatorname{im}(P)$ and $C(I_n - P\Pi)x = 0$, we have $\operatorname{im}(P) + \ker(C) = \mathbb{R}^n$. Conversely, let us assume that $\operatorname{im}(P) + \ker(C) = \mathbb{R}^n$. Then, there exists a matrix D such that

$$\operatorname{im}(D) \subseteq \ker(C), \text{ and } \operatorname{im}(P) \oplus \operatorname{im}(D) = \mathbb{R}^n. \tag{15}$$

Then, there exists matrices Π and E such that

$$I_n = P\Pi + DE. \tag{16}$$

It follows that $C = CP\Pi + CDE = CP\Pi = H\Pi$ because $\operatorname{im}(D) \subseteq \ker(C)$. It remains to show that $\Pi P = I_m$. Since $I_n = P\Pi + DE$, it follows that $P = P\Pi P + DEP$. Then, $P(I_m - \Pi P) = DEP$. Since $\operatorname{im}(P) \cap \operatorname{im}(D) = \{0\}$, $P(I_m - \Pi P) = DEP = 0$. Finally, $P(I_m - \Pi P) = 0$ implies $(I_m - \Pi P) = 0$ because P is injective. ■

Lemmas 2 and 3 give necessary and sufficient conditions for a suitable abstraction map P: $\operatorname{im}(P)$ must be an (A, B)-controlled invariant subspace complementing $\ker(C)$.

4.3. Computation of the abstract system

Given an abstraction map P, a suitable abstract system Σ' can be computed as follows:

Theorem 4. *Let P be an injective map such that*

$$\mathrm{im}(AP) \subseteq \mathrm{im}(P) + \mathrm{im}(B) \text{ and } \mathrm{im}(P) + \ker(C) = \mathbb{R}^n.$$

Let D and Π be maps satisfying Eqs. (15) and (16). Let F and Q be matrices such that $AP = PF - BQ$, let $H = CP$ and $G = [\Pi B \;\; \Pi AD]$. Then, Eqs. (8) and (9) hold and Σ' is Π-related to Σ.

Proof. By Lemma 2 and by construction of H, we have that Eqs. (8) and (9) hold. Let us verify that Σ' is Π-related to Σ. By Lemma 3, Eq. (12) holds. Let $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, then

$$\begin{aligned}\Pi(Ax + Bu) &= \Pi AP\Pi x + \Pi A(I_m - P\Pi)x + \Pi Bu \\ &= \Pi AP\Pi x + \Pi ADEx + \Pi Bu\end{aligned}$$

where E is as in Eq. (16). Since Eq. (8) holds, we have $AP = PF - BQ$. Then,

$$\begin{aligned}\Pi(Ax + Bu) &= \Pi(PF - BQ)\Pi x + \Pi ADEx + \Pi Bu \\ &= \Pi PF\Pi x + \Pi B(u - BQ\Pi x) + \Pi ADEx.\end{aligned}$$

From Lemma 3, $\Pi P = I_m$, then

$$\Pi(Ax + Bu) = F\Pi x + Gv \quad \text{with } v = \begin{bmatrix} u - BQ\Pi x \\ Ex \end{bmatrix}.$$

Therefore, Eq. (11) holds as well and Σ' is Π-related to Σ. ■

If we impose the condition that $\mathrm{im}(P)$ is A-invariant:

$$\mathrm{im}(AP) \subseteq \mathrm{im}(P)$$

instead of (A, B)-controlled invariant, then the assumptions of Proposition 2 hold. This allows us to use a hierarchical control architecture where uniformly bounded inputs $\mathbf{v}$ for the abstract system Σ' result in uniformly bounded inputs $\mathbf{u}$ for the concrete system Σ. Finding such a matrix P is easy: if (C, A) is observable, then the intersection of any A-invariant subspace with $\ker(C)$ is the null subspace. Then, it is sufficient to choose P as a matrix spanning an arbitrary A-invariant subspace of dimension greater than or equal to k. The A-invariant subspaces are easily determined using the Jordan normal form of the matrix A.

5. Example

In this section, we use our approach to design abstractions for the following class of systems:

$$\Sigma : \mathbf{y}^{(r)}(t) + \sum_{i=0}^{r-1} \alpha_i \mathbf{y}^{(i)}(t) = \mathbf{u}(t),$$

where $\mathbf{u}(t) \in \mathbb{R}^k$ and $\mathbf{y}(t) \in \mathbb{R}^k$, $r \geq 1$ and $\mathbf{y}^{(r)}$ denotes the rth-order derivative of the output trajectory $\mathbf{y}$. With the notations of the previous sections, we have:

$$A = \begin{bmatrix} 0_k & I_k & 0_k & \dots & 0_k \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0_k \\ 0_k & \dots & \dots & 0_k & I_k \\ -\alpha_0 I_k & \dots & \dots & \dots & -\alpha_{r-1} I_k \end{bmatrix},$$

$$B^{\mathrm{T}} = \begin{bmatrix} 0_k & \dots & 0_k & I_k \end{bmatrix}, \qquad C = \begin{bmatrix} I_k & 0_k & \dots & 0_k \end{bmatrix},$$

where I_k denotes the $k \times k$ identity matrix, 0_k denotes the $k \times k$ zero matrix. For simplicity, we assume that the polynomial $s^r + \alpha_{r-1}s^{r-1} + \dots + \alpha_1 s + \alpha_0$ has at least one real root denoted λ (note that this is always true if r is odd). In the following, we compute a first-order abstraction of Σ.

Let us consider the abstraction map P given by:

$$P^{\mathrm{T}} = \begin{bmatrix} I_k & \lambda I_k & \dots & \lambda^{r-1} I_k \end{bmatrix}.$$

It is easy to check that $im(P) \oplus \ker(C) = \mathbb{R}^n$ and that $AP = \lambda P$. Thus P is a suitable abstraction map. Let us compute the associated abstract system given by Theorem 4.

We can choose the following matrices D and Π

$$D = \begin{bmatrix} 0_{k,n-k} \\ I_{n-k} \end{bmatrix}, \qquad \Pi = \begin{bmatrix} I_k & 0_k & \dots & 0_k \end{bmatrix}$$

where $0_{k,n-k}$ denotes the $k \times (n-k)$ zero matrix. Eq. (8) holds with $F = \lambda I_k$ and $Q = 0$. Then, $H = CP = I_k$ and

$$G = \begin{bmatrix} \Pi B & \Pi AD \end{bmatrix} = \begin{bmatrix} 0_k & I_k & 0_k \dots 0_k \end{bmatrix}$$

which can equivalently be replaced by $G' = I_k$ by changing the dimension of the input of the abstract system. Then, the abstract system is

$$\Sigma' : \dot{\mathbf{w}}(t) = \lambda \mathbf{w}(t) + \mathbf{v}(t),$$

where $\mathbf{v}(t) \in \mathbb{R}^k$ and $\mathbf{w}(t) \in \mathbb{R}^k$. In order to use our hierarchical control approach, it remains to find a stabilizing gain K and a positive definite symmetric matrix M such that Eqs. (4) and (5) hold. This can be done for instance by solving the linear matrix inequalities (6) and (7).

We now apply our approach for the following particular system:

$$\Sigma : \mathbf{y}^{(3)}(t) = \mathbf{u}(t)$$

where $\mathbf{u}(t) \in \mathbb{R}^2$ and $\mathbf{y}(t) \in \mathbb{R}^2$ Following our approach, with the abstraction map $P^{\mathrm{T}} = [I_2 \; 0_2 \; 0_2]$, the following abstract system is computed:

$$\Sigma' : \dot{\mathbf{w}}(t) = \mathbf{v}(t)$$

where $\mathbf{v}(t) \in \mathbb{R}^2$ and $\mathbf{w}(t) \in \mathbb{R}^2$. We designed a simulation function $\mathcal{V}$ of Σ' by Σ. The associated interface is given by:

$$u_{\mathcal{V}}(v, w, y, \dot{y}, \ddot{y}) = 26.0v - 13.3\ddot{y} - 52.3\dot{y} - 52.0(y - w) \qquad (17)$$

and the associated function γ is $\gamma(\nu) = 2.128\nu$. Assume that Σ describes the (high-order) dynamics of a mobile planar robot. The output $\mathbf{y}(t) \in \mathbb{R}^2$ gives the position of the robot. Initially, $\dot{\mathbf{y}}(0) = 0$ and $\ddot{\mathbf{y}}(0) = 0$. Particularly, if we set the initial state of the abstract system to $\mathbf{w}(0) = \mathbf{y}(0)$, then the initial value of the simulation function $\mathcal{V}$ is 0.

We consider the problem of driving the robot in the environment shown in Fig. 2. It consists of a corridor of width 1. At the end of the corridor, there is a room with an obstacle. The goal of the motion planning problem consists in reaching a target which is a circle of diameter 1, behind the obstacle. Since the abstract system Σ' is fully actuated, it is easy to synthesize a path for this system. This path is represented by the dashed line in Fig. 2. It is clear that any trajectory remaining within distance 0.5 from this path satisfies the problem specification. We thus choose a bound ν for the inputs of the abstraction Σ' such that $\gamma(\nu) \leq 0.5$. Then, from Theorem 1, we know that the output trajectory $\mathbf{y}$ remains within distance 0.5 from $\mathbf{w}$, and thus satisfies the specification of the motion planning problem.

We can choose $\nu = 0.235$ ($\gamma(\nu) = 0.5$). The output trajectory $\mathbf{y}$ obtained by connecting the abstract system and the concrete system through the interface given by Eq. (17) is represented by the full line in Fig. 2. It is clear that it satisfies the specification of the motion planning problem. In Fig. 3, we represented the evolution of $\|\mathbf{y}(t) - \mathbf{w}(t)\|$ for the trajectories of Σ and Σ' presented in Fig. 2. We can check that it remains bounded by $\gamma(\nu) = 0.5$ which is expected from Theorem 1. Moreover, we can see that this bound is quite tight.

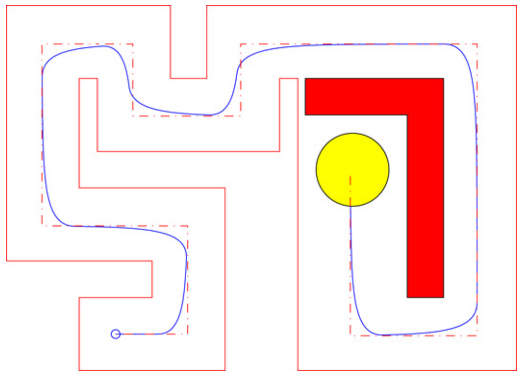

Fig. 2. Output trajectory $\mathbf{y}$ of the concrete system Σ (plain, blue) and output trajectory $\mathbf{w}$ of the abstract system Σ' (dashed, red). $\mathbf{y}$ satisfies the specification of the problem. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

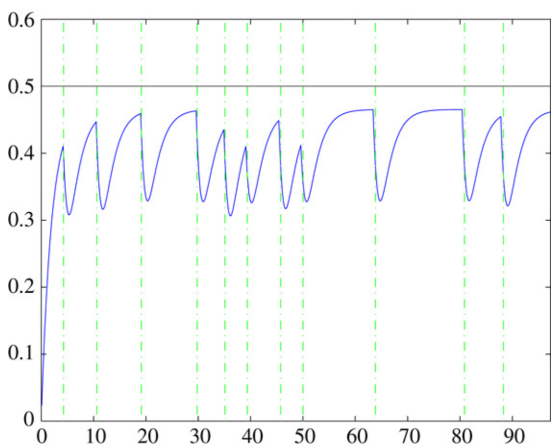

Fig. 3. Value of $\|\mathbf{y}(t) - \mathbf{w}(t)\|$ for the trajectories of Σ and Σ' presented in Fig. 2. We can see that it is bounded by $\gamma(\nu) = 0.5$ (horizontal line). The vertical lines correspond to the times at which the direction of the trajectory of the abstract system Σ' changes.

6. Conclusion

In this paper, we presented a theoretical framework for hierarchical control based on the recently introduced notion of simulation function. For the class of linear systems we gave an effective characterization of simulation functions and of the associated interfaces. We derived a method to compute abstractions of linear control systems allowing us to design hierarchical control systems. An example of application was shown, more complex examples for a second-order model of a robot can be found in Fainekos et al. (2007). Future work includes the extension of our approach to synthesize output feedback interfaces and the extension of the framework to networked control systems.

References

Aoki, M. (1968). Control of large-scale dynamic systems by aggregation. *IEEE Transaction on Automatic Control*, *13*(3), 246–253.
Cavarischia, L., & Lanari, L. (2007a). Hierarchical control implementation. In *Proc. mediterranean control conf*.
Cavarischia, L., & Lanari, L. (2007b). Hierarchical tracking implementation. In *proc. IEEE conf. decision and control*.
Di Benedetto, M. D., & Grizzle, J. W. (1994). Asymptotic model matching for nonlinear systems. *IEEE Transaction on Automatic Control*, *39*(8), 1539–1549.
Fainekos, G. E., Girard, A., & Pappas, G. J. (2007). Hierarchical synthesis of hybrid controllers from temporal logic specification. In *LNCS*: *Vol. 4416*. *Hybrid systems: computation and control* (pp. 203–216). Springer.
Girard, A., & Pappas, G. J. (2006). Hierarchical control using approximate simulation relations. In *proc. IEEE conf. decision and control*.
Girard, A., & Pappas, G. J. (2007). Approximate bisimulation relations for constrained linear systems. *Automatica*, *43*(8), 1307–1317.
Girard, A., & Pappas, G. J. (2007). Approximate hierarchies of linear control systems. In *proc. IEEE conf. decision and control*.
Girard, A., & Pappas, G. J. (2007). Approximation metrics for discrete and continuous systems. *IEEE Transaction on Automatic Control*, *52*(5), 782–798.
Hautus, M. (1983). Linear matrix equations with applications to the regulator problem. In I. D. Landau (Ed.), *Outils et modèles mathématiques pour l'Automatique* (pp. 399–412). Editions du CNRS.
Ikeda, M., Siljak, D. D., & White, D. E. (1984). An inclusion principle for dynamic systems. *IEEE Transaction on Automatic Control*, *29*(3), 244–249.
Ingalls, B., Sontag, E. D., & Wang, Y. (2002). A relaxation theorem for differential inclusions with applications to stability properties. In *proc. mathematical theory of networks and systems*.
Milner, R. (1989). *Communication and concurrency*. Prentice-Hall.
Pappas, G. J., Lafferriere, G., & Sastry, S. (2000). Hierarchically consistent control systems. *IEEE Transaction on Automatic Control*, *45*(6), 1144–1160.
Pola, G., Girard, A., & Tabuada, P. (2007). Symbolic models for nonlinear control systems using approximate bisimulation. In *proc. IEEE conf. decision and control*.
Sontag, E. D. (1998). *Mathematical control theory: Deterministic finite dimensional systems*. Springer.
Sontag, E. D., & Wang, Y. (2000). Lyapunov characterizations of input to output stability. *SIAM Journal on Control and Optimization*, *39*(1), 226–249.
Tabuada, P. (2007). Approximate simulation relations and finite abstractions of quantized control systems. In *LNCS*: *4416*. *Hybrid systems: computation and control* (pp. 529–542). Springer.
Tabuada, P., & Pappas, G. J. (2005). Hierarchical trajectory generation for a class of nonlinear systems. *Automatica*, *41*(4), 701–708.
Tazaki, Y., & Imura, J. (2008). Bisimilar finite abstractions of discrete-time linear systems and its application to optimal control. In *IFAC world congress*.
Wonham, W. M. (1979). *Linear multivariable control: A geometric approach*. Springer.

Antoine Girard received the Dip.Ing. from the Ecole Nationale Supérieure d'Informatique et de Mathématiques Appliquées de Grenoble, Grenoble, France, the M.S. degree in applied mathematics from the Université Joseph Fourier, Grenoble, France, both in 2001 and the Ph.D. degree in applied mathematics from the Institut National Polytechnique de Grenoble, France, in September 2004.

From October 2004 to December 2005, he was a postdoctoral researcher at the Department of Electrical and Systems Engineering of the University of Pennsylvania, Philadelphia and from January to August 2006, he was a postdoctoral researcher at the Verimag laboratory, Grenoble, France. Since September 2006, he has been an Assistant Professor at the Université Joseph Fourier, Grenoble, France.

His research interests include algorithmic analysis and approximation theory of discrete, continuous and hybrid systems as well as hierarchical and multiscale approaches to control.

George J. Pappas received the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, in 1998. He is currently a Professor at the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia. He holds secondary appointments in the Departments of Computer and Information Sciences, and Mechanical Engineering and Applied Mechanics, and is a member of the General Robotics and Active Sensory Perception (GRASP) Laboratory. He currently serves as the Deputy Dean in the School of Engineering and Applied Science. He is the Co-Editor of Hybrid Systems: Computation and Control (Springer-Verlag, 2004). His research interests include the areas of hybrid and embedded systems, hierarchical control systems, distributed control systems, nonlinear control systems, and geometric control theory, with applications to robotics, unmanned aerial vehicles, and biomolecular networks.

Prof. Pappas is the recipient of numerous awards including the National Science Foundation (NSF) Career Award in 2002, the NSF Presidential Early Career Award for Scientists and Engineers (PECASE) in 2002, and the Eliahu Jury Award for Excellence in Systems Research from the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley in 1999. His and his students' papers were finalists for the Best Student Paper Award at the IEEE Conference on Decision and Control in 1998, 2001, 2004, and 2006, the American Control Conference in 2001 and 2004, the IEEE Conference on Robotics and Automation in 2007, and the 2007 IEEE/RSJ Conference on Intelligent Robots and Systems.