Bisimulation of dynamical systems*

Arjan van der Schaft†
Department of Applied Mathematics, University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
a.j.vanderschaft@math.utwente.nl

Abstract

A general notion of bisimulation is studied for linear input-state-output systems, using analogies with the notion of bisimulation for concurrent processes. A characterization of bisimulation and an algorithm for computing the maximal bisimulation relation is derived using geometric control theory. Bisimulation is shown to be a concept which unifies the concepts of state space equivalence and state space reduction, and which allows to study equivalence of systems with non-minimal state space dimension. The notion of bisimulation is especially powerful for 'non-deterministic' dynamical systems, and leads in this case to a notion of equivalence which is finer than equality of external behavior. Finally a notion of structural bisimulation is developed for hybrid systems with continuous input and output variables.

Keywords Bisimulation, continuous dynamical systems, controlled invariance, maximal bisimulation relation, non-deterministic systems, abstraction, hybrid systems.

1 Introduction

A crucial notion in the theory of concurrent processes and model-checking is the concept of *bisimulation* which expresses when a (sub-) process can be considered to be externally equivalent to another (hopefully simpler) process. On the other hand, classical notions in systems and control theory are *state space equivalence*, and *reduction* of an input-state-output system to an equivalent system with minimal state space dimension. These notions have been instrumental in e.g. linking input-output models to state space models, and in studying the properties of interconnected systems.

Developments in both areas have been rather independent, one of the reasons being that the mathematical formalisms for describing both types of systems (discrete processes on the one hand, and continuous dynamical systems on the other hand) are rather different. However, with the rise of interest in *hybrid systems*, there is a clear need to bring these theories together.

The aim of this paper is to make a further step in the reapproachment between the theory of concurrent processes on the one hand and mathematical systems theory on the other hand by defining and studying a notion of *bisimulation for continuous dynamical systems*, and to relate it to system-theoretic notions of state space equivalence and state space reduction. Furthermore, we will make some initial steps towards a general notion of bisimulation for hybrid

*Submitted to the 7th Int. Workshop on *Hybrid Systems: Computation and Control* (HSCC 2004)
†Research partially supported by the EU-project HYBRIDGE, IST-2001-32460

systems with communication variables by *merging* the notion of bisimulation of continuous dynamical systems with the established notion of bisimulation for concurrent processes.

Extensions of the notion of bisimulation to continuous dynamical systems have been explored before in a series of innovative papers by Pappas and co-authors [7, 8, 1, 11, 12, 13, 4, 19], and this work is very much inspired by these papers. A main difference is that in [11, 12, 13, 19] the focus is on characterizing bisimulation of a dynamical system by a "projected" dynamical system with lower state space dimension ("an abstraction"; see Section 4), while in the present paper we deal with a general notion of bisimulation between two continuous dynamical systems and make precise the relations with system-theoretic notions of state space equivalence and state space reduction. (We note that the definition of bisimulation relations for dynamical systems has been given before in a general context in [4].)

We study continuous dynamical systems of the form

$$\begin{array}{rcll} \dot{x} &=& Ax + Bu + Gd, & x \in \mathbb{R}^n, u \in \mathbb{R}^m, d \in \mathbb{R}^s \\ y &=& Cx, & y \in \mathbb{R}^p \end{array} \tag{1}$$

where x are as usual the state variables, u are the input variables, y are the output variables, while d are additional input variables, which can be thought of as *disturbances.* Thus we restrict to *linear input-state-output systems* although most of the theory can be generalized to the nonlinear case, cf. [17], see also [19]. The basic problem we address is when two systems of the form (1) can be considered to be externally *equivalent*, in the sense that for all time instants t_0 the solution trajectories for $t \geq t_0$ of one system are mimicked by the other in such a way that the input and output trajectories $u(t)$ and $y(t)$ of both systems are the same for $t \geq t_0$, *without* imposing any relation between the values of their disturbance variables d. The intuitive idea is that we only want to distinguish between two systems if the distinction can be detected by an external system interacting with these systems. This is a fundamental notion in design, enabling us to take a 'divide and conquer' strategy, and in analysis, allowing us to switch between externally equivalent representations of the same system and to reduce sub-systems to externally equivalent but simpler sub-systems.

In case the disturbance variables d are absent this problem comes down to the usual system-theoretic notion of *state space equivalence*, if *additionally* the systems have *minimal state space dimension.* At the same time the problem bears much similarity with the notion of *bisimulation*, as introduced by Milner and Park [9, 14] in the study of *concurrent processes*, especially if the disturbance d is present. Thirdly, it is closely linked to the notion of *controlled invariance*, as introduced in linear systems theory by Wonham & Morse [21] and Basile & Marro [2]. The last two connections were already explored by Pappas and co-authors [12, 11, 13, 19] in the more restricted context when a system is bisimilar to an "abstraction" of itself, and we shall make explicit how our results generalize the results of [12, 11, 13, 19, 4] in Section 4. Note that in [11, 12, 19, 19] the input term Bu plays the same role as the disturbance term Gd in our setting, and thus bisimulation only involves equality of the outputs $y = Cx$.

The reason why the above notion of equivalence is similar to the notion of bisimulation as used in computer science can be briefly explained as follows. The notion of bisimulation has been introduced in order to handle external equivalence of general (interactive) concurrent processes. The concept is especially powerful for concurrent processes which are *non-deterministic* in the sense that branching in the (discrete) state may occur while the traces ("words") generated by the transition system are the same. In fact, the existence of a bisimulation relation between two *deterministic* processes is equivalent to equality of their external behaviors (the set of "traces", or the "language" generated by the process), and in this case bisimulation provides an efficient way to check equality of external behavior. For non-deterministic processes, however, bisimulation provides a finer equivalence than equality of external behavior, and, for example, also captures the *deadlock behavior* of concurrent processes.

A similar picture appears to arise for bisimulation of continuous dynamical systems (1). First, a type of "non-determinism" is present in systems (1) if we consider u and y as the external variables of the system (analogously to the labels of the discrete transitions of a process), while d denotes a generator for non-determinism in the evolution of the state x. If d is absent then (1) reduces to an ordinary "deterministic" system. For *deterministic* systems (that is, d absent) bisimulation can be shown (cf. [17]) to be equivalent to equality of external behavior, while generalizing the notion of state space equivalence to systems with non-minimal state space dimension. On the other hand, for *non-deterministic* systems bisimulation will be a stronger (finer) type of equivalence than equality of external behavior.

The structure of the paper is as follows. In Section 2 a (linear-)algebraic characterization of bisimulation is given, based on geometric control theory. The maximal bisimulation relation is computed in Section 3, and reduction of dynamical systems is treated using the notion of a bisimulation relation between the system and itself. Simulation and abstraction is treated in Section 4, making explicit the relation with previously obtained results in [12]. In Section 5 the definition of a hybrid automaton with continuous input and output variables is given, based on the definition in [18]. Furthermore, a notion of structural bisimulation for such hybrid systems is provided, *combining* the notion of bisimulation of continuous dynamical systems with the usual notion of bisimulation for concurrent processes. Finally, Section 6 contains the conclusions and questions for further research.

2 Bisimilar linear dynamical systems

Consider two dynamical systems of the form (1):

$$\Sigma_i : \begin{array}{rcll} \dot{x}_i & = & A_i x_i + B_i u_i + G_i d_i, & x_i \in \mathcal{X}_i, u_i \in \mathcal{U}, d_i \in \mathcal{D}_i \\ y_i & = & C_i x_i, & y_i \in \mathcal{Y} \quad i = 1,2 \end{array} \tag{2}$$

with $\mathcal{X}_i, \mathcal{D}_i, \mathcal{U}, \mathcal{Y}$ finite-dimensional linear spaces (over $\mathbb{R}$).

Before defining bisimulation we need to specify the solution trajectories of the systems (the *"semantics"*). That is, we have to specify the function classes of admissible input functions $u : [0, \infty) \to \mathcal{U}$ and admissible disturbance functions $d : [0, \infty) \to \mathcal{D}$, together with compatible function classes of state and output solutions $x : [0, \infty) \to \mathcal{X}$ and $y : [0, \infty) \to \mathcal{Y}$. For compactness of notation we will usually denote these time-functions respectively by $u(\cdot), d(\cdot), x(\cdot)$ and $y(\cdot)$. The exact class from which the functions are chosen is not very important. For example, we can take all functions to be C^∞. In some cases it may be natural/advantageous to require the property that if $d_1(\cdot)$ and $d_2(\cdot)$ are admissible disturbance functions then for every $\tau \geq 0$ also the function $d_3(\cdot)$ defined by $d_3(\cdot) = d_1(\cdot)(0 \leq t < \tau)$ and $d_3(\cdot) = d_2(\cdot)(t \geq \tau)$ is admissible. For this purpose, we could take the admissible disturbance and input functions to be *piecewise-continuous* with $x(\cdot)$ being the corresponding piecewise-differentiable solutions of the differential equations, and also the output functions to be piecewise-differentiable.

Definition 2.1. A (linear) bisimulation relation between Σ_1 and Σ_2 is a linear subspace

$$\mathcal{R} \subset \mathcal{X}_1 \times \mathcal{X}_2$$

with the following property. Take any $(x_{10}, x_{20}) \in \mathcal{R}$ and any joint input function $u_1(\cdot) = u_2(\cdot)$. Then for every disturbance function $d_1(\cdot)$ there should exist a disturbance function $d_2(\cdot)$ such that the resulting state solution trajectories $x_1(\cdot)$, with $x_1(0) = x_{10}$, and $x_2(\cdot)$, with $x_2(0) = x_{20}$, satisfy

$$(i) \quad (x_1(t), x_2(t)) \in \mathcal{R}, \quad \text{for all } t \geq 0 \tag{3}$$

$$(ii) \quad C_1 x_1(t) = C_2 x_2(t), \quad \text{for all } t \geq 0 \tag{4}$$

(or more precisely, for all $t \geq 0$ for which the trajectories are defined). Conversely, for every disturbance function $d_2(\cdot)$ there should exist a disturbance function $d_1(\cdot)$ such that again the resulting state trajectories $x_1(\cdot)$ and $x_2(\cdot)$ satisfy (3) and (4).

Hence for every pair $(x_{10}, x_{20}) \in \mathcal{R}$ all possible trajectories $x_1(\cdot)$ with $x_1(0) = x_{10}$ can be "simulated" by a trajectory $x_2(\cdot)$ with $x_2(\cdot) = x_{20}$ in the sense of giving the same input-output data for all future times while $(x_1(t), x_2(t)) \in \mathcal{R}$ for all $t \geq 0$, and conversely.

Remark 2.2. A similar definition (in the nonlinear case, and for the case that u_i is absent) has been given before in [4].

Remark 2.3. Note that the existence of a bisimulation relation between systems defines an *equivalence relation* between systems. Clearly $\mathcal{R}_{id} := \{(x_1, x_1) \mid x_1 \in \mathcal{X}_1\}$ is a bisimulation relation between Σ_1 given by (2) and itself. Furthermore, the existence of a bisimulation relation between Σ_1 and Σ_2 is clearly symmetric. Finally, if $\mathcal{R}_{12} \subset \mathcal{X}_1 \times \mathcal{X}_2$ is a bisimulation relation between Σ_1 and Σ_2, and $\mathcal{R}_{23} \subset \mathcal{X}_2 \times \mathcal{X}_3$ between Σ_2 and Σ_3 then $\mathcal{R}_{13} := \{(x_1, x_3) \mid \exists x_2 \in \mathcal{X}_2 \text{ s.t. } (x_1, x_2) \in \mathcal{R}_{12}, (x_2, x_3) \in \mathcal{R}_{23}\}$ is a bisimulation relation between Σ_1 and Σ_3.

We shall only deal with *linear* bisimulation relations, that is, $\mathcal{R}$ is throughout assumed to be a linear subspace of $\mathcal{X}_1 \times \mathcal{X}_2$. Hence we will drop the adjective "linear", and simply call $\mathcal{R}$ a *bisimulation relation*. (See [17] for nonlinear bisimulation of nonlinear systems.)

Definition 2.4. Two systems Σ_1 and Σ_2 as in (2) are *bisimilar* if there exists a bisimulation relation $\mathcal{R} \subset \mathcal{X}_1 \times \mathcal{X}_2$ with the property that

$$\pi_1(\mathcal{R}) = \mathcal{X}_1, \;\; \pi_2(\mathcal{R}) = \mathcal{X}_2 \tag{5}$$

where $\pi_i : \mathcal{X}_1 \times \mathcal{X}_2 \to \mathcal{X}_i, \;\; i = 1, 2$, denote the canonical projections.

Remark 2.5. Actually Definition 2.4 constitutes a slight departure from the definition of bisimulation relation as usually given for discrete processes [9, 14], by imposing the extra requirement (5). The reason is that in computer science discrete processes are usually defined with respect to a *fixed* initial state (or, a subset of initial conditions). In our setting we consider the behavior of the systems Σ_i for *arbitrary* initial states. Hence for every initial condition x_{10} of Σ_1 there should exist an initial condition x_{20} of Σ_2 with $(x_{10}, x_{20}) \in \mathcal{R}$ and vice versa; thus implying (5). The generalization to *subsets* of initial conditions $\mathcal{X}_{i0} \subset \mathcal{X}_i$ obviously can be done by relaxing (5) to $\pi_i(\mathcal{R}) = \mathcal{X}_{i0}, i = 1, 2$.

Remark 2.6. For $G_1 = G_2 = 0$ the above notion of bisimilarity is close to the usual notion of *state space equivalence* of two input-state-output systems

$$\Sigma_i : \begin{array}{rcl} \dot{x}_i & = & A_i x_i + B_i u_i, \quad x_i \in \mathcal{X}_i, u_i \in \mathcal{U}, y_i \in \mathcal{Y} \\ y_i & = & C_i x_i \qquad\qquad i = 1, 2 \end{array} \tag{6}$$

Indeed, in this case one usually starts with a linear equivalence *mapping*

$$S : \mathcal{X}_1 \to \mathcal{X}_2 \tag{7}$$

which is assumed to be invertible (implying that dim $\mathcal{X}_1$ = dim $\mathcal{X}_2$) with the property that

$$Sx_1(t) = x_2(t), \text{ for all } t \geq 0 \tag{8}$$

$$Cx_1(t) = Cx_2(t), \text{ for all } t \geq 0 \tag{9}$$

for all state trajectories $x_1(\cdot)$ and $x_2(\cdot)$ resulting from initial conditions x_{10} and x_{20} related by $Sx_{10} = x_{20}$ and all input-functions $u_1(\cdot) = u_2(\cdot)$. Defining the linear subspace

$$\mathcal{R} = \{(x_1, x_2) \in \mathcal{X}_1 \times \mathcal{X}_2 \mid x_2 = Sx_1\} \tag{10}$$

(i.e., the graph of the mapping S) it is easily seen that $\mathcal{R}$ is a bisimulation relation which satisfies $\pi_1(\mathcal{R}) = \mathcal{X}_1$ trivially and $\pi_2(\mathcal{R}) = \mathcal{X}_2$ because of invertibility of S.

Clearly, by allowing $\mathcal{R}$ to be a *relation* instead of the graph of a mapping, the notion of bisimilarity even in the case $G_1 = G_2 = 0$ is more general than state space equivalence. In particular, we may allow $\mathcal{X}_1$ and $\mathcal{X}_2$ to be of *different dimension*. Furthermore, by doing so we incorporate in the notion of bisimilarity the notion of *reduction* of an input-state-output system to a lower-dimensional input-state-output system, and especially the reduction to a *minimal* input-state-output system. This is worked out in [17].

Using well-known ideas from state space equivalence of linear dynamical systems and especially from the theory of controlled invariance, see e.g. [21, 2], it is easy to derive an algebraic characterization of the notion of a bisimulation relation.

Proposition 2.7. *A subspace $R \subset \mathcal{X}_1 \times \mathcal{X}_2$ is a bisimulation relation between Σ_1 and Σ_2 if and only if for all $(x_1, x_2) \in \mathcal{R}$ and all $u \in \mathcal{U}$ the following properties hold:*

(i) For all $d_1 \in \mathcal{D}_1$ there should exist a $d_2 \in \mathcal{D}_2$ such that

$$(A_1x_1 + B_1u + G_1d_1, A_2x_2 + B_2u + G_2d_2) \in \mathcal{R}, \tag{11}$$

and conversely for every $d_2 \in \mathcal{D}_2$ there should exist a $d_1 \in \mathcal{D}_1$ such that (11) holds.

(ii)

$$C_1x_1 = C_2x_2 \tag{12}$$

Proof. Consider (3). Then by differentiating $x_1(t)$ and $x_2(t)$ with respect to t and evaluating at any t we obtain (11), with $x_1 = x_1(t), x_2 = x_2(t), u = u_1(t) = u_2(t), d_1 = d_1(t), d_2 = d_2(t)$. Conversely, if (11) holds then $(\dot{x}_1(t), \dot{x}_2(t)) \in \mathcal{R}$ for all $t \geq 0$, thus implying (3). Equivalence of (4) and (12) is obvious. ■

Theorem 2.8. *A subspace $\mathcal{R} \subset \mathcal{X}_1 \times \mathcal{X}_2$ is a bisimulation relation between Σ_1 and Σ_2 if and only if*

$$\begin{aligned}
&(a) \quad \mathcal{R} + \text{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} = \mathcal{R} + \text{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} =: \mathcal{R}_e \\
&(b) \quad \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathcal{R} \subset \mathcal{R}_e \\
&(c) \quad \text{im} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \subset \mathcal{R}_e \\
&(d) \quad \mathcal{R} \subset \ker \begin{bmatrix} C_1 \vdots - C_2 \end{bmatrix}
\end{aligned} \tag{13}$$

Proof. Clearly (12) is equivalent with (13d). Let $\mathcal{R}$ satisfy property (i) of Proposition 2.7. Take $u = 0$. Then for all $(x_1, x_2) \in \mathcal{R}$, and for all d_1 there exists a d_2 such that

$$(A_1 x_1 + G_1 d_1, A_2 x_2 + G_2 d_2) \in \mathcal{R} \tag{14}$$

This is a version of the so-called Modified Disturbance Decoupling Problem (with d_1 being the "disturbance", which is assumed to be available for feedforward control action, and d_2 being the "control", [21]). It follows from the solution of this problem (see [21]) that (14) is equivalent to

$$\begin{aligned}
&\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathcal{R} \subset \mathcal{R} + \text{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \\
&\text{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \subset \mathcal{R} + \text{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix}
\end{aligned} \tag{15}$$

Analogously, property (i) of Proposition 2.7 implies that for all $(x_1, x_2) \in \mathcal{R}$, and all d_2 there exists d_1 such that (14) holds, implying similarly

$$\begin{aligned}
&\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathcal{R} \subset \mathcal{R} + \text{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \\
&\text{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \subset \mathcal{R} + \text{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix}
\end{aligned} \tag{16}$$

The second lines of (15) and (16) are readily seen to be equivalent to (13a). Consequently, the first lines of (15) and (16) are equivalent with (13b).

Finally, consider property (i) for $x_1 = x_2 = 0$ and $d_1 = 0$. Then (11) implies

$$\text{im} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \subset \mathcal{R} + \text{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} = \mathcal{R}_e$$

(the same result follows by taking $d_2 = 0$ instead of $d_1 = 0$), and thus (13c). By linearity it follows that conversely (13a,b,c) imply property (i). ∎

Remark 2.9. Note that a subspace $\mathcal{R} \subset \mathcal{X}_1 \times \mathcal{X}_2$ satisfies properties (13a,b) if and only if the mapping F (from subspaces $\mathcal{S} \subset \mathcal{X}_1 \times \mathcal{X}_2$ to subspaces $F(\mathcal{S}) \subset \mathcal{X}_1 \times \mathcal{X}_2$) defined by

$$\mathcal{S} \overset{F}{\mapsto} \{z \in \mathcal{X}_1 \times \mathcal{X}_2 \mid \begin{array}{l} \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} z + \mathrm{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \subset \mathcal{S} + \mathrm{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix}, \\ \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} z + \mathrm{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \subset \mathcal{S} + \mathrm{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \} \end{array}$$

satisfies $\mathcal{R} \subset F(\mathcal{R})$. This will be instrumental to compute the maximal bisimulation relation, see Algorithm 3.3; in fact, the maximal bisimulation relation turns out to be a *fixed-point* of this mapping. (This is well-known in the theory of bisimulation for concurrent processes [9].)

Bisimilarity is easily seen to imply *equality of external behavior*. Consider two systems $\Sigma_i, i = 1, 2$, as in (2), with external behavior $\mathcal{B}_i$ defined as

$$\mathcal{B}_i := \{(u_i(\cdot), y_i(\cdot) \mid \exists x_i(\cdot), d_i(\cdot) \text{ such that (2) is satisfied } \} \tag{17}$$

Proposition 2.10. *Let $\Sigma_i, i = 1, 2$, be bisimilar. Then their external behaviors $\mathcal{B}_i$ are equal.*

However, in the case of non-deterministic systems, that is, d_i is present, systems may have the same external behavior, while *not* being bisimilar. This is illustrated by the following example.

Example 2.11. Consider the two systems

$$\Sigma_1 : \begin{array}{rcl} \dot{x}^1 & = & x^2 \\ \dot{x}^2 & = & d_1 \\ y_1 & = & x^1 \end{array} \tag{18}$$

and

$$\Sigma_2 : \begin{array}{rcl} \dot{z} & = & d_2 \\ y_2 & = & z \end{array} \tag{19}$$

It can be readily seen that there does not exist any bisimulation relation between Σ_1 and Σ_2 (consider condition (13a)). On the other hand, if we restrict e.g. to C^∞ external behaviors then obviously $\mathcal{B}_1 = \mathcal{B}_2$. (Note the different logical quantifiers in the definition of bisimilarity and in equality of external behavior. For bisimilarity there should exist for every x^1, x^2 a z such that for every d_1 there exists a d_2 with equal external trajectories and conversely, while for equality of external behavior there should exist for every x^1, x^2, d_1 a pair z, d_2 with equal external trajectories, and conversely.)

An *interpretation* of the fact that Σ_1 and Σ_2 are not bisimilar can be given as follows. Suppose we "test" the system Σ_1 at some time instant $t = t_0$ in the sense of observing one of its possible external trajectories $y_1(t), t \geq t_0$. At $t = t_0$ the system Σ_1 is in a given, but unknown, initial state $(x^1(t_0), x^2(t_0))$. Hence, all possible runs $y_1(t), t \geq t_0$, starting from this fixed initial state will have a *fixed* time-derivative $\dot{y}_1(t_0) = x^2(t_0)$ at $t = t_0$. On the other hand, for Σ_2 the possible runs $y_2(t), t \geq t_0$, can have arbitrary time-derivative at $t = t_0$. Hence, Σ_1 and Σ_2 can be considered to be externally *different*.

For deterministic systems (d_i void) it is shown in [17] that equality of external behavior *does* imply bisimilarity, and how the bisimulation relation can be easily derived.

3 The maximal bisimulation relation and reduction of dynamical systems

In this section we first show how to compute the *maximal* bisimulation relation $R \subset \mathcal{X}_1 \times \mathcal{X}_2$ for two linear dynamical systems Σ_1 and Σ_2. The way to do this is very similar to the computation of the maximal controlled invariant subspace contained in a given subspace, which is the central algorithm in linear geometric control theory [21]. Furthermore, structurally the algorithm is the same as the existing algorithms to compute the maximal bisimulation relation for two discrete processes, see e.g. [6].

First we remark that the *maximal* bisimulation relation exists if there exists at least one bisimulation relation (contrary to e.g. the *minimal* bisimulation relation). The argument is similar to the argument showing the existence of a maximal controlled invariant subspace, and is based on the following simple observations.

Proposition 3.1. *Let $\mathcal{R}_a \subset \mathcal{X}_1 \times \mathcal{X}_2$ and $\mathcal{R}_b \subset \mathcal{X}_1 \times \mathcal{X}_2$ be bisimulation relations. Then also $\mathcal{R}_a + \mathcal{R}_b \subset \mathcal{X}_1 \times \mathcal{X}_2$ is a bisimulation relation.*

Proof. Since $\mathcal{R}_a, \mathcal{R}_b$ are bisimulation relations they satisfy properties (13). It follows that also $\mathcal{R}_a + \mathcal{R}_b$ satisfies (13), and thus is a bisimulation relation. ■

Proposition 3.2. *Given Σ_1 and Σ_2 and suppose there exists a bisimulation relation between Σ_1 and Σ_2. Then the* maximal *bisimulation relation exists.*

Proof. Suppose there exists a bisimulation relation. Let $\mathcal{R}^{max}$ be a bisimulation relation of *maximal dimension*. Take any other bisimulation relation $\mathcal{R}$. Then $\mathcal{R} \subset \mathcal{R}^{max}$, since otherwise dim $(\mathcal{R} + \mathcal{R}^{max}) >$ dim $\mathcal{R}^{max}$ while also $\mathcal{R} + \mathcal{R}^{max}$ is a bisimulation relation; a contradiction with the maximality of dimension of $\mathcal{R}^{max}$. ■

The maximal bisimulation relation $\mathcal{R}^{max}$ can be computed in the following way, similarly to the algorithm to compute the maximal controlled invariant subspace [21]. For notational convenience define

$$A^{\times} := \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, \quad G_1^{\times} := \begin{bmatrix} G_1 \\ 0 \end{bmatrix}, \quad G_2^{\times} := \begin{bmatrix} 0 \\ G_2 \end{bmatrix}, \quad C^{\times} := \begin{bmatrix} C_1 \vdots -C_2 \end{bmatrix} \tag{20}$$

Algorithm 3.3. *Given two dynamical systems Σ_1 and Σ_2. Define the following sequence $\mathcal{R}^j, j = 0, 1, 2, \cdots$ of linear subspaces of $\mathcal{X}_1 \times \mathcal{X}_2$*

$$\begin{aligned} \mathcal{R}^0 &= \mathcal{X}_1 \times \mathcal{X}_2 \\ \mathcal{R}^1 &= \left\{ z \in \mathcal{R}^0 \mid z \in \ker C^{\times} \right\} \\ \mathcal{R}^2 &= \left\{ z \in \mathcal{R}^1 \mid A^{\times} z + \operatorname{im} G_1^{\times} \subset \mathcal{R}^1 + \operatorname{im} G_2^{\times}, A^{\times} z + \operatorname{im} G_2^{\times} \subset \mathcal{R}^1 + \operatorname{im} G_1^{\times} \right\} \\ &\vdots \\ \mathcal{R}^{j+1} &= \left\{ z \in \mathcal{R}^j \mid A^{\times} z + \operatorname{im} G_1^{\times} \subset \mathcal{R}^j + \operatorname{im} G_2^{\times}, A^{\times} z + \operatorname{im} G_2^{\times} \subset \mathcal{R}^j + \operatorname{im} G_1^{\times} \right\} \end{aligned} \tag{21}$$

Assumption 3.4. *Assume that the subspaces $\mathcal{R}^j$ in (21) are* non-empty.

Theorem 3.5. *Let Assumption 3.4 be satisfied. The sequence of subspaces $\mathcal{R}^0, \mathcal{R}^1, \cdots, \mathcal{R}^j, \cdots$ satisfies the following properties.*

1. $\mathcal{R}^0 \supset \mathcal{R}^1 \supset \mathcal{R}^2 \cdots \supset \mathcal{R}^j \supset \mathcal{R}^{j+1} \supset \cdots$
2. *There exists a finite k such that $\mathcal{R}^k = \mathcal{R}^{k+1} =: \mathcal{R}^*$ and then $\mathcal{R}^j = \mathcal{R}^*$ for all $j \geq k$.*
3. *$\mathcal{R}^*$ is the maximal subspace of $\mathcal{X}_1 \times \mathcal{X}_2$ satisfying properties (13a,b,d) of Proposition 2.8.*

The proof is very similar to the proof of the corresponding properties of the algorithm for computing the maximal controlled invariant subspace [21], and is given in [17].

If $\mathcal{R}^*$ as obtained from Algorithm 3.3 satisfies property (13c), then it follows that $\mathcal{R}^*$ equals the *maximal bisimulation relation* $\mathcal{R}^{max}$ between Σ_1 and Σ_2, while if $\mathcal{R}^*$ does *not* satisfy property (13c) then there does *not* exist any bisimulation relation between Σ_1 and Σ_2. With regard to bisimilarity (Definition 2.4), we have the following immediate consequence.

Corollary 3.6. *Σ_1 and Σ_2 are* bisimilar *if and only if Assumption 3.4 is satisfied and $\mathcal{R}^*$ satisfies Property (13c) and Equation (5).*

In the rest of this section we study the question how to *reduce* a linear dynamical system to a system with *lower state space dimension*, which is *bisimilar* to the original system, and in particular how to reduce the system to a bisimilar system with *minimal* state space dimension.

This can be achieved by considering bisimulation relations between the system and *a copy of itself*. Furthermore, the reduction to a bisimilar system with minimal state space dimension can be performed by using the same algorithm as given in the previous section for computing the maximal bisimulation relation. Actually, this idea is well-known in the context of concurrent processes, see e.g. [6]. So let us consider a linear dynamical system as in (1):

$$\Sigma : \begin{array}{rcll} \dot{x} & = & Ax + Bu + Gd, & x \in \mathcal{X}, u \in \mathcal{U}, d \in \mathcal{D} \\ y & = & Cx, & y \in \mathcal{Y} \end{array} \tag{22}$$

with $\mathcal{X}, \mathcal{U}, \mathcal{Y}$ and $\mathcal{D}$ finite-dimensional linear spaces. Now consider a bisimulation relation between Σ and itself, that is, in view of Theorem 2.8, subspaces $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ satisfying

$$\begin{array}{ll} (a) & \mathcal{R} + \mathrm{im} \begin{bmatrix} G \\ 0 \end{bmatrix} = \mathcal{R} + \mathrm{im} \begin{bmatrix} 0 \\ G \end{bmatrix} =: \mathcal{R}_e \\ (b) & \begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix} \mathcal{R} \subset \mathcal{R}_e \\ (c) & \mathrm{im} \begin{bmatrix} B \\ B \end{bmatrix} \subset \mathcal{R}_e \\ (d) & \mathcal{R} \subset \ker \begin{bmatrix} C \vdots -C \end{bmatrix} \end{array} \tag{23}$$

First of all we note the following obvious fact:

Proposition 3.7. *The identity relation $\mathcal{R}_{id} = \{(x, x) \mid x \in \mathcal{X}\}$ is a bisimulation between Σ and itself.*

Every $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ defines a *relation* on $\mathcal{X}$ by saying that $x_a, x_b \in \mathcal{X}$ are related by $\mathcal{R}$ if and only if $(x_a, x_b) \in \mathcal{R}$. For reduction we should restrict attention to $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ such that the corresponding relation on $\mathcal{X}$ is an *equivalence* relation, i.e., $\mathcal{R}$ is reflexive ($(x,x) \in \mathcal{R}$ for all $x \in \mathcal{X}$), symmetric ($(x_a, x_b) \in \mathcal{R} \iff (x_b, x_a) \in \mathcal{R}$), and transitive ($(x_a, x_b) \in \mathcal{R}, (x_b, x_c) \in \mathcal{R} \Rightarrow (x_a, x_c) \in \mathcal{R}$). This can be done without loss of generality. Indeed, by Proposition 3.7 we may always add to any bisimulation relation $\mathcal{R}$ the identity bisimulation relation $\mathcal{R}_{id}$, thus enforcing reflexivity. Furthermore, let $\mathcal{R}$ satisfy (23), then also the inverse relation $\mathcal{R}^{-1} := \{(x_a, x_b) \mid (x_b, x_a) \in \mathcal{R}\}$ satisfies (23), implying that the symmetric closure $\mathcal{R} + \mathcal{R}^{-1}$ satisfies (23). Finally, for *linear* relations reflexivity and symmetry already *implies* transivity: if $(x_a, x_b), (x_b, x_c) \in \mathcal{R}$, then $(x_a - x_c, 0) = (x_a, x_b) - (x_c, x_b) \in \mathcal{R}$, and thus $(x_a, x_c) = (x_a - x_c, 0) + (x_c, x_c) \in \mathcal{R}$.

Any equivalence relation $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ can be uniquely associated with a linear subspace $\bar{\mathcal{R}} \subset \mathcal{X}$ as follows:

$$\bar{\mathcal{R}} := \{x_a - x_b \mid (x_a, x_b) \in \mathcal{R}\} \tag{24}$$

Indeed, $\bar{\mathcal{R}}$ defined by (24) is a *linear space* if and only if $\mathcal{R}$ is reflexive and symmetric (and therefore an equivalence relation). In terms of $\bar{\mathcal{R}}$ conditions (23) reduce as follows.

Theorem 3.8. *Let $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ be an equivalence relation, and define $\bar{\mathcal{R}} \subset \mathcal{X}$ as in (23). Conditions (23a,b,c,d) for $\mathcal{R}$ are equivalent to*

$$\begin{array}{rcl} A\bar{\mathcal{R}} & \subset & \bar{\mathcal{R}} + \operatorname{im} G \\ \bar{\mathcal{R}} & \subset & \ker C \end{array} \tag{25}$$

Proof. It is readily seen that (23b,d) are equivalent to (25). Satisfaction of (23a,c) follows from reflexivity of $\mathcal{R}$. ■

A subspace $\bar{\mathcal{R}}$ satisfying the first line of (25) is called a *controlled invariant subspace* (with respect to A and $\operatorname{im} G$), cf.[2, 21]. Thus there is a one-to-one correspondence between *bisimulation equivalence relations* $\mathcal{R}$ and *controlled invariant subspaces* $\bar{\mathcal{R}}$ which are contained in $\ker C$.

As a consequence of Proposition 3.7 the *maximal* bisimulation $\mathcal{R}^{max} = \mathcal{R}^*$ between Σ and itself exists, and contains $\mathcal{R}_{id}$. Hence $\mathcal{R}^*$ is reflexive, while by symmetry of the data it follows that the symmetric closure of $\mathcal{R}^*$ (adjoining (x_b, x_a) if $(x_a, x_b) \in \mathcal{R}^*$) also satisfies (23a,b,d), and hence $\mathcal{R}^*$ is symmetric. Thus the maximal bisimulation relation $\mathcal{R}^*$ is an equivalence relation. The corresponding subspace $\bar{\mathcal{R}}^* \subset \mathcal{X}$ is precisely the *maximal* controlled invariant subspace contained in $\ker C$, and can be computed in this way, cf. [21, 2].

It is now clear how to reduce Σ to a lower-dimensional system that is bisimilar to Σ. Let R be a bisimulation equivalence relation. Define the reduced state space

$$\mathcal{X}_{\mathcal{R}} := \mathcal{X}/_{\bar{\mathcal{R}}} \tag{26}$$

with canonical projection $\Pi_{\mathcal{R}} : \mathcal{X} \to \mathcal{X}/_{\bar{\mathcal{R}}}$. By the first line of (24) there exists a "feedback" map K such that

$$(A + GK)\bar{\mathcal{R}} \subset \bar{\mathcal{R}} \tag{27}$$

and thus $A + GK$ projects to a linear map

$$A_{\mathcal{R}} : \mathcal{X}_{\mathcal{R}} \to \mathcal{X}_{\mathcal{R}} \tag{28}$$

satisfying $A_\mathcal{R}\Pi_\mathcal{R} = \Pi_\mathcal{R}(A + GK)$. Furthermore, define

$$G_\mathcal{R} := \Pi_\mathcal{R} G, \quad B_\mathcal{R} := \Pi_\mathcal{R} B, \tag{29}$$

and by the second line of (24) we may define

$$C_R : \mathcal{X}_\mathcal{R} \to \mathcal{Y} \tag{30}$$

such that $C_\mathcal{R}\Pi_\mathcal{R} = C$. Together this defines a reduced system

$$\Sigma_\mathcal{R} : \begin{array}{rcl} \dot{x}_\mathcal{R} & = & A_\mathcal{R} x_\mathcal{R} + B_\mathcal{R} u + G_\mathcal{R} d \\ y & = & C_\mathcal{R} x_\mathcal{R} \end{array} \tag{31}$$

Proposition 3.9. *Let $\mathcal{R}$ be a bisimulation equivalence relation between Σ and itself, and construct $\Sigma_\mathcal{R}$ as above. Then $\Sigma_\mathcal{R}$ is bisimilar to Σ. Furthermore, let $\mathcal{R}^*$ denote the maximal bisimulation relation between Σ and itself. Then $\Sigma_{\mathcal{R}^*}$ is the smallest system that is bisimilar by reduction to Σ.*

Proof. Define the following relation $\mathcal{S}_\mathcal{R} \subset \mathcal{X} \times \mathcal{X}_\mathcal{R}$

$$(x, x_\mathcal{R}) \in \mathcal{S}_\mathcal{R} \Longleftrightarrow x_\mathcal{R} = \Pi_\mathcal{R}(x) \tag{32}$$

This is readily seen to be a bisimulation relation which obviously satisfies (5). ■

4 Simulation and abstraction

A one-sided version of the notions of bisimulation relation and bisimilarity as provided in Definitions 2.1 and 2.4 can be stated as follows:

Definition 4.1. Consider Σ_1 and Σ_2 as given in (2). A *simulation relation* of Σ_1 by Σ_2 is a linear subspace

$$\mathcal{S} \subset \mathcal{X}_1 \times \mathcal{X}_2$$

with the following property. Take any $(x_{10}, x_{20}) \in \mathcal{S}$ and any joint input function $u_1(\cdot) = u_2(\cdot)$. Then for every disturbance function $d_1(\cdot)$ there should exist a disturbance function $d_2(\cdot)$ such that the resulting state solution trajectories $x_1(\cdot)$, with $x_1(0) = x_{10}$, and $x_2(\cdot)$, with $x_2(0) = x_{20}$, satisfy

$$(i) \quad (x_1(t), x_2(t)) \in \mathcal{S}, \quad \text{for all } t \geq 0 \tag{33}$$

$$(ii) \quad C_1 x_1(t) = C_2 x_2(t), \quad \text{for all } t \geq 0 \tag{34}$$

Furthermore, Σ_1 is *simulated* by Σ_2 if the simulation relation $\mathcal{S}$ satisfies $\pi_1(\mathcal{S}) = \mathcal{X}_1$ with π_1 the canonical projection of $\mathcal{X}_1 \times \mathcal{X}_2$ to $\mathcal{X}_1$.

For several purposes the existence of a simulation relation instead of a bisimulation relation is already sufficient. One may for example think of a system description capturing the required specifications on a system to be designed: if the actual closed-loop system is simulated by the specified system then at least one knows that there is no unwanted behavior in the actual closed-loop system. Or, one may "approximate' a given system by a lower-dimensional system in the sense that the given system is simulated by the approximating system. (This is the idea of *abstraction* as will be discussed later on.) Then checking certain properties for the approximating system will also guarantee these properties for the given system.

By a "one-sided" version of Theorem 2.8, see especially Equation (15), we obtain

Proposition 4.2. *$\mathcal{S} \subset \mathcal{X}_1 \times \mathcal{X}_2$ is a simulation relation of Σ_1 by Σ_2 if and only if*

$$\begin{array}{ll} (a) & \mathcal{S} + \operatorname{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \subset \mathcal{S} + \operatorname{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \\ (b) & \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathcal{S} \subset \mathcal{S} + \operatorname{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \\ (c) & \operatorname{im} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \subset \mathcal{S} + \operatorname{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \\ (d) & \mathcal{S} \subset \ker \begin{bmatrix} C_1 \vdots - C_2 \end{bmatrix} \end{array} \tag{35}$$

For discrete processes it may occur that there is a simulation relation of Σ_1 by Σ_2 *and* a simulation relation of Σ_2 by Σ_1 while there does *not* exist a bisimulation relation between Σ_1 and Σ_2, cf. [9, 10]. In the present context, however, we have the following

Proposition 4.3. *Let $\mathcal{S} \subset \mathcal{X}_1 \times \mathcal{X}_2$ be a simulation relation of Σ_1 by Σ_2 and let $\mathcal{T} \subset \mathcal{X}_2 \times \mathcal{X}_1$ be a simulation relation of Σ_2 by Σ_1. Then $\mathcal{R} := \mathcal{S} + \mathcal{T}^{-1}$ is a bisimulation relation between Σ_1 and Σ_2.*

Proof. Let $\mathcal{S}$ satisfy (35) and let $\mathcal{T}$ satisfy (35) with index 1 throughout replaced by 2. It follows that

$$\mathcal{T}^{-1} + \mathcal{S} + \operatorname{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \subset \mathcal{S} + \mathcal{T}^{-1} + \operatorname{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} \subset \mathcal{S} + \mathcal{T}^{-1} + \operatorname{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix}$$

and thus (13a) results for $\mathcal{R} = \mathcal{S} + \mathcal{T}^{-1}$. Similarly, $\mathcal{R}$ satisfies (13b,c,d). ■

Now let us recall the definition of an *abstraction* as introduced in a general context in [1, 5, 11, 13], and more specifically in [12, 19]. Consider a dynamical system (22), together with a surjective linear map $H : \mathcal{X} \to \mathcal{Z}$, $\mathcal{Z}$ being another linear space, satisfying throughout

$$\ker H \subset \ker C \tag{36}$$

Clearly this implies that there exist a a unique linear map $\bar{C} : \mathcal{Z} \to \mathcal{Y}$ such that

$$C = \bar{C} H \tag{37}$$

Then define the following dynamical system on $\mathcal{Z}$

$$\bar{\Sigma} : \begin{array}{rcll} \dot{z} & = & \bar{A} z + \bar{B} u + \bar{G} d, & z \in \mathcal{Z}, u \in \mathcal{U}, d \in \mathcal{D} \\ y & = & \bar{C} z, & y \in \mathcal{Y} \end{array} \tag{38}$$

with

$$\begin{array}{rcl} \bar{A} & := & HAH^+ \\ \bar{B} & := & HB \\ \bar{G} & := & [HG \vdots HAv_1 \vdots \cdots \vdots HAv_k] \end{array} \tag{39}$$

where H^+ denotes the Moore-Penrose pseudo-inverse of H, and $v_1, \cdots, v_k$ span $\ker H$.

We think of $\bar{\Sigma}$ as an "*abstraction*" of Σ in the sense that we factor out the part of the state variables $x \in \mathcal{X}$ corresponding to $\ker H$. This idea was put forward in a series of innovative papers by Pappas and co-authors (see especially [12]) in the slightly more restricted case where $B = 0$. (N.B., actually in [12] the "generator for non-determinism" d is denoted by u.)

It has been shown in [12] that Σ is simulated by $\bar{\Sigma}$, as also follows from Proposition 4.2 by taking the simulation relation $\mathcal{R} := \{(x,z) \mid z = Hx\}$. A main result of [12] is that $\mathcal{R} = \{(x,z) \mid z = Hx\}$ is a *bisimulation* relation between Σ and $\bar{\Sigma}$ if and only if $\ker H$ is *controlled invariant* with respect to A and $\operatorname{im} G$. This also follows from Theorem 3.8. Generalization of this result to the nonlinear case was obtained in [19], see also [17].

5 Structural bisimulation of hybrid systems with continuous input-output behavior

Aim of this section is to give a definition of bisimulation for hybrid systems with discrete *and* continuous external variables. The discrete external variables are the actions corresponding to the discrete transitions, while the continuous external variables are the continuous inputs and outputs as before. The bisimulation relation should thus respect the *total* external behavior of the hybrid system, that is, with respect to the actions, *as well as* with respect to the continuous external variables. The inclusion of continuous external variables makes the setting different from previous notions of bisimulation of hybrid systems, which only involve the *discrete* external behavior, see e.g. [5, 1, 7, 8, 20].

We start from the definition of a hybrid automaton with continuous external variables as given in [18], where we restrict ourselves to the case that the continuous dynamics in every location is a linear input-state-output system as in (22) and that the reset rule for the continous state is linear.

Definition 5.1 (Hybrid automaton with linear input-state-output dynamics). A hybrid automaton with linear input-state-output dynamics is described by a seven-tuple $\Sigma^{hyb} := (\mathcal{L}, \mathcal{X}, \mathcal{A}, \mathcal{U}, \mathcal{Y}, E, F)$, where the symbols have the following meanings.

- $\mathcal{L}$ is a finite set, called the set of *discrete states* or *locations*.
- $\mathcal{X}$ is a finite-dimensional linear space called the *continuous state space*.
- $\mathcal{A}$ is a finite set of symbols called the set of *discrete communication variables*, or *actions*.
- $\mathcal{U}$ and $\mathcal{Y}$ are finite-dimensional linear spaces, denoting respectively the continuous *input* and *output* space.
- $E \subset \mathcal{L} \times \mathcal{L} \times \mathcal{A} \times \mathcal{X} \times \mathcal{X}$ specifies the discrete transitions, both in the discrete and continuous states. It is assumed that for every $l^-, l^+ \in \mathcal{L}$ and $a \in \mathcal{A}$ the subset $\{(x^-, x^+) \mid (l^-, l^+, a, x^-, x^+) \in E\}$ is a *linear* subspace of $\mathcal{X} \times \mathcal{X}$ or empty.
- F is a mapping which assigns to every $l \in \mathcal{L}$ a linear non-deterministic input-state-output system

$$\begin{aligned} \dot{x} &= A^l x + B^l u + G^l d, \quad x \in \mathcal{X}, u \in \mathcal{U}, d \in \mathcal{D} \\ y &= C^l x, \qquad\qquad\qquad\quad y \in \mathcal{Y} \end{aligned} \tag{40}$$

  where $d \in \mathcal{D}$ is as before an unknown disturbance generator.

A *trajectory* (or run) of the hybrid system Σ^{hyb} on the time-interval $[0,T]$ consists of the following ingredients. First such a trajectory involves a discrete set $\mathcal{E} \subset [0,T]$ denoting the *event times* $t \in [0,T]$ associated with the trajectory. Secondly, there is a function $l : [0,T] \to \mathcal{L}$ which is constant on every subinterval between subsequent event times $t_a, t_b \in \mathcal{E}$, and which specifies the location of the hybrid system for $t \in (t_a, t_b)$. Thirdly, the trajectory involves admissible time-functions

$$x : [0,T] \to \mathcal{X}, \quad u : [0,T] \to \mathcal{U}, \quad y : [0,T] \to \mathcal{Y} \tag{41}$$

satisfying for all $t \notin \mathcal{E}$ the dynamics (40), with l the location between subsequent event times $t_a, t_b \in \mathcal{E}$. Finally, the trajectory includes a discrete function

$$a : \mathcal{E} \to \mathcal{A} \tag{42}$$

such that for all $t \in \mathcal{E}$

$$(l(t^-), l(t^+), a(t), x(t^-), x(t^+)) \in E \tag{43}$$

Here, of course, $x(t^-)$ and $x(t^+)$ denote the limit values of the variables x when approaching t from the left, respectively from the right, and the same for $l(t^-)$ and $l(t^+)$. (Hence we throughout assume that the class of admissible functions x is chosen in such a way that these left and right limits are defined.)

Thus the mapping F (the *flow* conditions) specifies the continuous dynamics of the hybrid system depending on the location the system is in, and this continuous dynamics remains the same between subsequent event times. On the other hand, E (the *event* conditions) stands for the event behavior at the event times, entailing the discrete state variables $l \in \mathcal{L}$ and the discrete communication variables $a \in \mathcal{A}$, together with a (linear) reset of the continuous state variables x. In [18] it is discussed how the flow conditions F incorporate the notion of *location invariant*, while the event conditions E include the notion of *guard*.

Remark 5.2. In [18] a more general definition of hybrid automaton with continuous communication variables has been given in the same format by replacing the two linear spaces $\mathcal{U}$ and $\mathcal{Y}$ by a *single* linear space $\mathcal{W}$, denoting the space of *continuous communication variables*, and by letting $E \subset \mathcal{L} \times \mathcal{L} \times \mathcal{A} \times \mathcal{X} \times \mathcal{X}$ to be a general subset, and finally by letting F to be a mapping which assigns to every location $l \in \mathcal{L}$ a (possibly nonlinear and constrained dynamics)

$$F^l(\dot{x}, x, w) = 0$$

in the continuous variables $x \in \mathcal{X}$ and $w \in \mathcal{W}$.

Remark 5.3. Note that the discrete part of the dynamics is allowed to be non-deterministic in the sense that for a given a and $l(t^-), x(t^-)$ there may be more than one $l(t^+), x(t^+)$ such that $(l(t^-), l(t^+), a, x(t^-), x(t^+)) \in E$. In this sense there is an asymmetry in the model since for the continuous dynamics the non-determinism is represented by an *additional* disturbance generator d. This asymmetry may be resolved by replacing the explicit differential description (22) by the *differential inclusion*

$$\begin{aligned} &\dot{x} - Ax - Bu \in \operatorname{im} G \\ &y = Cx \end{aligned} \tag{44}$$

or to rewrite the first line of (44) as the differential-algebraic equation $G^\perp(\dot{x} - Ax - Bu) = 0$, with $G^\perp$ an annihilating matrix of G.

Remark 5.4. Of course, much more can be said about the possible semantics of the hybrid automaton defined above. In particular, additional requirements can be imposed on the set $\mathcal{E} \subset [0, T]$ of event times, while on the other hand the notion of a trajectory can be further generalized by allowing for *multiple events* at the same event time. For a discussion of these issues we refer to [18].

By *merging* the definition of a linear bisimulation relation for linear input-state-output systems (2) with the common notion of bisimulation for concurrent processes we obtain the following notion of a hybrid bisimulation relation.

Definition 5.5 (Hybrid bisimulation relation). Consider two hybrid automata with linear input-state-output dynamics $\Sigma_i^{hyb} = (\mathcal{L}_i, \mathcal{X}_i, \mathcal{A}_i, \mathcal{U}_i, \mathcal{Y}_i, E_i, F_i), i = 1, 2$, as above. A linear hybrid bisimulation between Σ_1^{hyb} and Σ_2^{hyb} is a subset

$$\mathcal{R} \subset (\mathcal{L}_1 \times \mathcal{X}_1) \times (\mathcal{L}_2 \times \mathcal{X}_2)$$

such that for all l_1, l_2 the sets $\{(x_1, x_2) \mid (l_1, x_1, l_2, x_2) \in \mathcal{R}\}$ are linear subspaces of $\mathcal{X}_1 \times \mathcal{X}_2$ or empty, and have the following property. Take any $(l_{10}, x_{10}, l_{20}, x_{20}) \in \mathcal{R}$. Furthermore take any joint input function $u_1(\cdot) = u_2(\cdot)$ and any joint sequence of actions $a^1a^2\ldots$ on a set $\mathcal{E}$ of event times. Then for every disturbance function $d_1(\cdot)$ there should exist a disturbance function $d_2(\cdot)$ such that the resulting state trajectories $(l_1(\cdot), x_1(\cdot))$, with $(l_1(0), x_1(0)) = (l_{10}, x_{10})$, and $(l_2(\cdot), x_2(\cdot))$, with $(l_2(0), x_2(0)) = (l_{20}, x_{20})$, have the same set $\mathcal{E}$ of event times and satisfy

$$(i) \quad ((l_1(t), x_1(t)), (l_2(t), x_2(t))) \in \mathcal{R}, \quad \text{for all } t \notin \mathcal{E} \tag{45}$$

$$(ii) \quad C_1^{l_1(t)} x_1(t) = C_2^{l_2(t)} x_2(t), \quad \text{for all } t \notin \mathcal{E} \tag{46}$$

and the same holds with d_1 replaced by d_2 and conversely.

Hence for every $(l_{10}, x_{10}, l_{20}, x_{20}) \in \mathcal{R}$ every possible hybrid state trajectory $(l_1(\cdot), x_1(\cdot))$ of Σ_1^{hyb} emanating from the hybrid initial state (l_{10}, x_{10}) can be simulated by a hybrid state trajectory $(l_2(\cdot), x_2(\cdot))$ of Σ_2^{hyb} emanating from (l_{20}, x_{20}) in the sense of giving the same actions a^i at the event times and the same continuous input-output data for all time instants between the event times, and conversely.

Note that the above definition of a hybrid bisimulation relation is very much based on splitting the hybrid dynamics into its continuous part (described by the flow conditions F) and its discrete part (described by the event conditions E). It can therefore be called a *structural* bisimulation relation, in much the same style as previously developed structural bisimulation relations for timed automata, see e.g. [3].

Remark 5.6. Note that in between subsequent event times the notion of hybrid bisimulation is the same as bisimulation for linear input-state-output dynamics. 'Dually', if we abstract away from the continuous behavior between event times, then hybrid bisimulation reduces to ordinary bisimulation for the resulting concurrent process with discrete states $l \in \mathcal{L}$ and actions $a \in \mathcal{A}$. (Of course, the resulting discrete dynamics generally depends crucially on the 'hidden' continuous dynamics. Also, the event times are not just an ordered sequence of time instants, but they are embedded in the physical time-axis of the underlying continuous dynamics.)

6 Conclusions and outlook

We have studied a notion of bisimulation for continuous dynamical systems, motivated by the theory of bisimulation for concurrent processes and by previously obtained results by Pappas and co-authors. The notion of bisimulation appears to be a notion which *unifies* the concepts of state space equivalence and state space reduction, and which allows to study equivalence of systems with non-minimal state space dimension, cf. [17].

Compared with classical systems theory a new twist to the problem is given by the idea of considering *non-deterministic* continuous dynamical systems. For concurrent discrete processes the advantages of allowing non-determinism are clear [9, 6]. For continuous dynamical systems such arguments have not been explicitated except in the context of abstraction. In particular, for verification of hybrid systems it is natural to look for notions of abstraction which allow to extend methods for verification of discrete processes to the hybrid case, see [5, 1, 7, 8, 20]. This naturally leads to considering ways of abstracting continuous dynamical systems. Apart from abstraction we believe that there are other good reasons to study some type of "non-determinism" in continuous dynamical systems. Indeed, it would be interesting to investigate if uncertainty and robustness issues can be fruitfully cast in this framework. Current research is aimed at exploring such ideas in the extension of the results on achievable closed-loop behavior obtained in [16] to the situation where the closed-loop system is specified up to bisimulation.

We have provided a notion of *structural hybrid bisimulation* for hybrid systems. Main difference with existing notions is that we consider hybrid systems which interact with their environment not only via their discrete actions but also via their continuous (input-output) behavior. This raises many questions; two of them being the following. First, it needs to be investigated if and how the theory of maximal bisimulation relations, reduction and abstraction for continuous dynamical systems, as developed in this paper and in [17], can be combined with the well-established theory of concurrent processes into a similar theory of hybrid systems with discrete and continuous external behavior. This is currently under study. Secondly, it is important to relate more closely the notion of hybrid bisimulation with previously proposed notions for hybrid systems without (or with 'abstracted') continuous external behavior, see e.g. [5, 1, 7, 8, 20, 12, 4].

Obvious generalization of the results in this paper is the extension to the nonlinear case, continuing on [11, 19, 17]. Another interesting generalization would be to suppress the a priori distinction of the external variables into inputs u and outputs y. This can be achieved by considering a behavioral version of (1), e.g. given as

$$K\dot{x} + Lx + Mw + Nd = 0, \tag{47}$$

where now w denotes the *whole* vector of external variables (not split into u and y), cf. [15]. Note that (47) may also include algebraic constraints on the state variables, and thus is eminently suited to model systems as resulting from interconnection of systems, or multi-modal systems arising from varying (event-driven) state constraints. For modelling hybrid systems this is a natural setting; see also Remark 5.3.

Acknowledgements

Initial discussions with George Pappas and Paulo Tabuada (University of Pennsylvania) and ongoing conversations with Jan Willem Polderman, Rom Langerak, Agung Julius and Stefan Strubbe (all University of Twente) are gratefully acknowledged.

References

[1] R. Alur, T.A. Henzinger, G. Lafferriere, G.J. Pappas, Discrete abstractions of hybrid systems, *Proceedings of the IEEE, vol. 88*, pp. 971–984, 2000.

[2] G. Basile, G. Marro, *Controlled and conditioned invariants in linear system theory*, Prentice Hall, Englewood Cliffs, 1992.

[3] P. D'Argenio, *Algebras and Automata for Timed and Stochastic Systems*, Ph.D. Thesis, University of Twente, 1999.

[4] E. Haghverdi, P. Tabuada, G.J. Pappas, Unifying bisimulation relations for discrete and continuous systems, *Category Theory and Computer Science, Electronic Notes in Theoretical Computer Science*, August 2002.

[5] T.A. Henzinger, "Hybrid automata with finite bisimulations", in ICIALP 95: Automata, Languages, and programming, eds. Z. Fulop, F. Gecseg, Lecture Notes in Computer Science 944, pp. 324–335, Springer, New York, 1995

[6] H. Hermanns, *Interactive Markov Chains*, Lecture Notes in Computer Science 2428, Springer, 2002.

[7] G. Lafferriere, G.J. Pappas, S. Sastry, "Hybrid systems with finite bisimulations", in *Hybrid Systems V*, eds. P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, S. Sastry, Lecture Notes in Computer Science, Springer, 1998.

[8] G. Lafferriere, G.J. Pappas, S. Sastry, "O-minimal hybrid systems", *Math. Contr. Signals, Syst.*, vol. 13, pp. 1-21, 2000.

[9] R. Milner, *Communication and Concurrency*, Prentice Hall International Series in Computer Science, 1989.

[10] R. Milner, *Communication and Mobile Systems: the π- Calculus*, Cambridge University Press, Cambridge, UK, 1999.

[11] G.J. Pappas, G. Lafferriere, S. Sastry, Hierarchically consistent control systems, *IEEE Transactions on Automatic Control, 45(6)*, pp. 1144–1160, 2000.

[12] G.J. Pappas, Bisimilar linear systems. Technical Report MS-CIS-01-19, University of Pennsylvania, Philadelphia, PA, June 2001, to appear in *Automatica*.

[13] G.J. Pappas, S. Simic, Consistent abstractions of affine control systems, *IEEE Transactions on Automatic Control, 47*, pp. 745–756, 2002.

[14] D. Park, Concurrency and automata on infinite sequences. In P. Deussen, Editor, *Fifth GI Conference on Theoretical Computer Science*, vol. 104 of *Lecture Notes in Computer Science*, Springer, 1981.

[15] J.W. Polderman, J.C. Willems, *Introduction to Mathematical Systems Theory*, Springer-Verlag, New York, 1998.

[16] A.J. van der Schaft, Achievable behavior of general systems. *Systems & Control Letters, vol. 49*, pp. 141–149, 2003.

[17] A.J. van der Schaft, "Equivalence of dynamical systems by bisimulation", to be submitted.

[18] A.J. van der Schaft, J.M. Schumacher, *An Introduction to Hybrid Dynamical Systems*, Springer Lecture Notes in Control and Information Sciences, Vol.251, Springer-Verlag, London, 2000. Second revised edition to appear in *Communications and Control Engineering Series*, Springer, London, 2004.

[19] P. Tabuada, G.J. Pappas, "Bisimilar control affine systems", *Systems and Control Letters*, to appear.

[20] P. Tabuada, G.J. Pappas, P. Lima, "Composing abstractions of hybrid systems", in *Hybrid Systems: Computation and Control*, eds. C. Tomlin, M.R. Greenstreet, Lecture Notes in Computer Science, pp. 436–450, Springer, 2002.

[21] W.M. Wonham, *Linear multivariable control: a geometric approach*, Springer, New York, Third Edition, 1985.