Bisimulation of Dynamical Systems*

Arjan van der Schaft**

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

Abstract. A general notion of bisimulation is studied for dynamical systems. An algebraic characterization of bisimulation together with an algorithm for computing the maximal bisimulation relation is derived using geometric control theory. Bisimulation of dynamical systems is shown to be a concept which unifies the system-theoretic 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. Furthermore, by merging bisimulation of dynamical systems with bisimulation of concurrent processes a notion of structural bisimulation is developed for hybrid systems with continuous input and output variables.

1 Introduction

A crucial notion in the theory of concurrent processes and model-checking is the concept of *bisimulation*. This notion expresses when a (sub-) process can be considered to be externally equivalent to another process. At the same time, 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 and mathematical systems theory by defining and characterizing a notion of *bisimulation for continuous dynamical systems*, and to relate it to system-theoretic notions of state space equivalence

* Appeared in Hybrid Systems: Computation and Control, Eds. R. Alur, G.J. Pappas, Lecture Notes in Computer Science 2293, pp. 555–569, Springer, 2004.
** Research partially supported by the EU-project HYBRIDGE, IST-2001-32460.

and state space reduction. Furthermore, by *merging* the notion of bisimulation of continuous dynamical systems with the established notion of bisimulation for concurrent processes we give a definition of structural bisimulation for hybrid systems with continuous communication variables.

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 [6, 7, 1, 10–12, 3, 17], and the present paper is very much inspired by this work. A main difference is that while in [10–12, 17] the focus is on characterizing bisimulation of a dynamical system by a "projected" dynamical system with lower state space dimension ("an abstraction"), the present paper (see also [15]) deals with a general notion of bisimulation between two continuous dynamical systems and gives algebraic conditions when two systems are bisimilar. (Note that the abstract definition of bisimulation relations for dynamical systems has been given before in a general context in [3].) Furthermore, the present paper as well as [15] makes precise the relations with system-theoretic notions of state space equivalence and state space reduction.

The continuous dynamical systems that we study are of the form

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

where x are 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. [15]; see also [17] in the case of abstraction. The basic problem of bisimulation 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.

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.* Furthermore, it will become clear that the notion of bisimulation of continuous dynamical systems and its algebraic characterization is closely linked to the notion of *controlled invariance*, as introduced in linear systems in [19, 2]. This last connection was already explored by Pappas and co-authors [11, 10, 12, 17] in the more restricted context when a system is bisimilar to an "abstraction" of itself, and [15] makes explicit how these previously obtained results fit within the framework of the present paper. (Note that in [10, 11, 17] the input term Bu plays the same role as the disturbance term Gd in our setting.)

The notion of bisimulation for concurrent processes as introduced in [8, 13]) is especially powerful for concurrent processes which are *non-deterministic* in the sense that branching in the (discrete) state may occur while the traces 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 disturbance generator in the evolution of the state x. If d is absent then (1) reduces to an ordinary "deterministic" system, and bisimulation can be shown (cf. [15]) 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.

By *combining* the characterization of bisimulation of continuous dynamical systems with the usual notion of bisimulation for concurrent processes we will propose a notion of structural bisimulation for hybrid systems with continuous communication variables, which lends itself to algebraic characterizations. Indeed, in [14] this has been worked out in the special case of switching linear systems without invariants and guards, including an algorithm for computing the maximal bisimulation relation. The general proposed notion of structural hybrid bisimulation makes use of the definition of hybrid automata with continuous communication variables as recently provided in [16].

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. In Section 4 a notion of structural bisimulation for hybrid system automata with continuous input and output variables is provided. Finally, Section 5 contains the conclusions and questions for further research.

2 Bisimilar linear dynamical systems

Consider two dynamical systems of the form (1):

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

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 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^∞ although 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.

Definition 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 1. A similar definition (for the case that u_i is absent) has been given before in a general context in [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 [15] for a treatment of bisimulation of nonlinear systems.)

Definition 2. *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. Definition 2 constitutes a slight departure from the definition of bisimulation relation as usually given for discrete processes [8, 13], 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 3. 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

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

Indeed, in this case one usually starts with a linear equivalence *mapping* $S : \mathcal{X}_1 \to \mathcal{X}_2$,
which is assumed to be invertible (implying that dim $\mathcal{X}_1 =$ dim $\mathcal{X}_2$) with the property that

$$Sx_1(t) = x_2(t), \text{ for all } t \geq 0 \tag{7}$$

$$Cx_1(t) = Cx_2(t), \text{ for all } t \geq 0 \tag{8}$$

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

$$\mathcal{R} = \{(x_1, x_2) \in \mathcal{X}_1 \times \mathcal{X}_2 \mid x_2 = Sx_1\} \tag{9}$$

(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 [15].

Using well-known ideas from state space equivalence of linear dynamical systems and especially from the theory of controlled invariance, see e.g. [19, 2], it is easy to derive an algebraic characterization of the notion of a bisimulation relation.

Proposition 1. *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

$$(A_1 x_1 + B_1 u + G_1 d_1, A_2 x_2 + B_2 u + G_2 d_2) \in \mathcal{R}, \tag{10}$$

and conversely for every $d_2 \in \mathcal{D}_2$ there should exist a $d_1 \in \mathcal{D}_1$ such that (10) holds.

(ii)

$$C_1 x_1 = C_2 x_2 \tag{11}$$

Proof. Consider (3). Then by differentiating $x_1(t)$ and $x_2(t)$ with respect to t and evaluating at any t we obtain (10), 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 (10) 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 (11) is obvious.

A main theorem proved in [15] is the following.

Theorem 1. *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*

$$\begin{aligned}
&(a)\ \mathcal{R} + \operatorname{im} \begin{bmatrix} G_1 \\ 0 \end{bmatrix} = \mathcal{R} + \operatorname{im} \begin{bmatrix} 0 \\ G_2 \end{bmatrix} =: \mathcal{R}_e \\
&(b)\ \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathcal{R} \subset \mathcal{R}_e \\
&(c)\ \operatorname{im} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \subset \mathcal{R}_e \\
&(d)\ \mathcal{R} \subset \ker \begin{bmatrix} C_1 \vdots -C_2 \end{bmatrix}
\end{aligned} \tag{12}$$

Remark 4. Note that a subspace $\mathcal{R} \subset \mathcal{X}_1 \times \mathcal{X}_2$ satisfies properties (12a,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

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

satisfies $\mathcal{R} \subset F(\mathcal{R})$. This will be instrumental to compute the maximal bisimulation relation, see Algorithm 2; 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 [8].)

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

$$\mathcal{B}_i := \{(u_i(\cdot), y_i(\cdot) \mid \exists x_i(\cdot), d_i(\cdot) \text{ such that (2) is satisfied } \} \tag{13}$$

Proposition 2. *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 1. Consider the two systems

$$\Sigma_1 : \begin{aligned} \dot{x}^1 &= x^2 \\ \dot{x}^2 &= d_1 \\ y_1 &= x^1 \end{aligned} \tag{14}$$

and

$$\Sigma_2 : \begin{aligned} \dot{z} &= d_2 \\ y_2 &= z \end{aligned} \tag{15}$$

It can be readily seen that there does not exist any bisimulation relation between Σ_1 and Σ_2 (consider condition (12a)). 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 [15] that equality of external behavior *does* imply bisimilarity, and how the bisimulation relation can be easily derived.

3 Maximal bisimulation relation and reduction

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 [19]. Furthermore, structurally the algorithm is the same as the existing algorithms to compute the maximal bisimulation relation for two discrete processes, see e.g. [5]. For details we refer to [15].

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. *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 (12). It follows that also $\mathcal{R}_a + \mathcal{R}_b$ satisfies (12), and thus is a bisimulation relation.

Proposition 4. *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 [19]. For notational convenience define

$$A^{\times} := \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, \quad G_1^{\times} := \begin{bmatrix} G_1 \\ 0 \end{bmatrix}, \quad G_2^{\times} := \begin{bmatrix} 0 \\ G_2 \end{bmatrix}, \quad C^{\times} := \begin{bmatrix} C_1 \vdots - C_2 \end{bmatrix} \tag{16}$$

Algorithm 2. *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 $\mathcal{X}_1 \times \mathcal{X}_2$*

$$\begin{aligned}
\mathcal{R}^0 &= \mathcal{X}_1 \times \mathcal{X}_2 \\
\mathcal{R}^1 &= \left\{ z \in \mathcal{R}^0 \mid z \in \ker C^{\times} \right\} \\
\mathcal{R}^2 &= \left\{ z \in \mathcal{R}^1 \mid A^{\times} z + \operatorname{im} G_1^{\times} \subset \mathcal{R}^1 + \operatorname{im} G_2^{\times}, A^{\times} z + \operatorname{im} G_2^{\times} \subset \mathcal{R}^1 + \operatorname{im} G_1^{\times} \right\} \\
&\vdots \\
\mathcal{R}^{j+1} &= \left\{ z \in \mathcal{R}^j \mid A^{\times} z + \operatorname{im} G_1^{\times} \subset \mathcal{R}^j + \operatorname{im} G_2^{\times}, A^{\times} z + \operatorname{im} G_2^{\times} \subset \mathcal{R}^j + \operatorname{im} G_1^{\times} \right\}
\end{aligned} \tag{17}$$

Assumption 3 *Assume that the subspaces $\mathcal{R}^j$ in (17) are* non-empty*.*

Theorem 4. *Let Assumption 3 be satisfied. The sequence of subspaces $\mathcal{R}^j$ 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 (12a,b,d) of Proposition 1.*

The proof is very similar to the proof of the corresponding properties of the algorithm for computing the maximal controlled invariant subspace [19], and is given in [15].

If $\mathcal{R}^*$ as obtained from Algorithm 2 satisfies property (12c), 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 (12c) then there does *not* exist any bisimulation relation between Σ_1 and Σ_2. With regard to bisimilarity (Definition 2), we have the following immediate consequence.

Corollary 1. *Σ_1 and Σ_2 are* bisimilar *if and only if Assumption 3 is satisfied and $\mathcal{R}^*$ satisfies Property (12c) 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 is 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. [5]. Consider a linear dynamical system as in (1)

$$\Sigma : \begin{array}{ll} \dot{x} = Ax + Bu + Gd, & x \in \mathcal{X}, u \in \mathcal{U}, d \in \mathcal{D} \\ y = Cx, & y \in \mathcal{Y} \end{array} \tag{18}$$

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 1, subspaces $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ satisfying

$$\begin{aligned} &(a)\ \mathcal{R} + \operatorname{im} \begin{bmatrix} G \\ 0 \end{bmatrix} = \mathcal{R} + \operatorname{im} \begin{bmatrix} 0 \\ G \end{bmatrix} =: \mathcal{R}_e \\ &(b)\ \begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix} \mathcal{R} \subset \mathcal{R}_e \\ &(c)\ \operatorname{im} \begin{bmatrix} B \\ B \end{bmatrix} \subset \mathcal{R}_e \\ &(d)\ \mathcal{R} \subset \ker \begin{bmatrix} C \vdots -C \end{bmatrix} \end{aligned} \tag{19}$$

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 we may always add to any bisimulation relation $\mathcal{R}$ the identity bisimulation relation $\mathcal{R}_{id} := \{(x, x) \mid x \in \mathcal{X}\}$, thus enforcing reflexivity. Furthermore, let $\mathcal{R}$ satisfy (19), then also the inverse relation $\mathcal{R}^{-1} := \{(x_a, x_b) \mid (x_b, x_a) \in \mathcal{R}\}$ satisfies (19), implying that the symmetric closure $\mathcal{R} + \mathcal{R}^{-1}$ satisfies (19). 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:

$$\bar{\mathcal{R}} := \{x_a - x_b \mid (x_a, x_b) \in \mathcal{R}\} \tag{20}$$

Indeed, $\bar{\mathcal{R}}$ defined by (20) 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 (19) reduce as follows.

Theorem 5. *Let $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ be an equivalence relation, and define $\bar{\mathcal{R}} \subset \mathcal{X}$ as in (19). Conditions (19a,b,c,d) for $\mathcal{R}$ are equivalent to*

$$\begin{aligned} A\bar{\mathcal{R}} &\subset \bar{\mathcal{R}} + \operatorname{im} G \\ \bar{\mathcal{R}} &\subset \ker C \end{aligned} \tag{21}$$

Proof. It is readily seen that (19b,d) are equivalent to (21). Satisfaction of (19a,c) follows from reflexivity of $\mathcal{R}$.

A subspace $\bar{\mathcal{R}}$ satisfying the first line of (21) is called in geometric control theory a *controlled invariant subspace* , cf.[2, 19]. 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 $\ker C$.

The *maximal* bisimulation $\mathcal{R}^{max} = \mathcal{R}^*$ between Σ and itself always exists, since it 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 (19a,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. [19, 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

$$\mathcal{X}_{\mathcal{R}} := \mathcal{X}/_{\bar{\mathcal{R}}} \tag{22}$$

with canonical projection $\Pi_{\mathcal{R}} : \mathcal{X} \to \mathcal{X}/_{\bar{\mathcal{R}}}$. By the first line of (21) there exists a "feedback" map K such that

$$(A + GK)\bar{\mathcal{R}} \subset \bar{\mathcal{R}} \tag{23}$$

and thus $A + GK$ projects to a linear map $A_{\mathcal{R}} : \mathcal{X}_{\mathcal{R}} \to \mathcal{X}_{\mathcal{R}}$ satisfying $A_{\mathcal{R}}\Pi_{\mathcal{R}} = \Pi_{\mathcal{R}}(A + GK)$. Furthermore, define $G_{\mathcal{R}} := \Pi_{\mathcal{R}}G, \quad B_{\mathcal{R}} := \Pi_{\mathcal{R}}B$, and by the second line of (21) we may define $C_R : \mathcal{X}_{\mathcal{R}} \to \mathcal{Y}$ such that $C_{\mathcal{R}}\Pi_{\mathcal{R}} = C$. Together this defines a reduced system

$$\Sigma_{\mathcal{R}} : \quad \begin{aligned} \dot{x}_{\mathcal{R}} &= A_{\mathcal{R}}x_{\mathcal{R}} + B_{\mathcal{R}}u + G_{\mathcal{R}}d \\ y &= C_{\mathcal{R}}x_{\mathcal{R}} \end{aligned} \tag{24}$$

Proposition 5. *(See [15] for the proof.) 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.*

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

Aim of this section is to characterize 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. [4, 1, 6, 7, 18].

We start from the definition of a hybrid automaton with continuous external variables as given in [16].

Definition 3 (Hybrid automaton). *A hybrid automaton is described by a six-tuple $\Sigma^{hyb} := (\mathcal{L}, \mathcal{X}, \mathcal{A}, \mathcal{W}, E, F)$, 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 manifold called the continuous state space.*
- *$\mathcal{A}$ is a finite set of symbols called the set of discrete communication variables, or actions.*
- *$\mathcal{W}$ is a finite-dimensional linear space called the space of continuous communication variables. In the sequel the vector $w \in \mathcal{W}$ will be often partitioned into an input vector u and an output vector y.*
- *E is a subset of $\mathcal{L} \times \mathcal{L} \times \mathcal{A} \times \mathcal{X} \times \mathcal{X}$; a typical element of this set is denoted by (l^-, l^+, a, x^-, x^+).*
- *F is a subset $\mathcal{L} \times T\mathcal{X} \times \mathcal{W}$, where $T\mathcal{X}$ denotes the tangent bundle of $\mathcal{X}$; a typical element of this set is denoted by $(l, x, \dot{x}, w)$.*

A hybrid 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 $x : [0, T] \to \mathcal{X}, \quad w : [0, T] \to \mathcal{W}$, satisfying for all $t \notin \mathcal{E}$ the dynamics

$$(l, x(t), \dot{x}(t), w(t)) \in F \tag{25}$$

with l the location between subsequent event times $t_a, t_b \in \mathcal{E}$. Finally, the trajectory includes a discrete function $a : \mathcal{E} \to \mathcal{A}$ such that for all $t \in \mathcal{E}$

$$(l(t^-), l(t^+), a(t), x(t^-), x(t^+)) \in E \tag{26}$$

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 a hybrid run is specified by a five-tuple

$$r = (\mathcal{E}, l, x, a, w) \tag{27}$$

Note that the subset 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 reset of the continuous state variables x. In [16] 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. 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 [16].

In terms of the hybrid runs a natural definition of hybrid bisimulation is given as follows:

Definition 4 (Hybrid bisimulation relation). *Consider two hybrid automata* $\Sigma_i^{hyb} = (\mathcal{L}_i, \mathcal{X}_i, \mathcal{A}_i, \mathcal{W}_i, E_i, F_i), i = 1, 2,$ *as above. A hybrid bisimulation between* Σ_1^{hyb} *and* Σ_2^{hyb} *is a subset*

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

with the following property. Take any $(l_{10}, x_{10}, l_{20}, x_{20}) \in \mathcal{R}$. *Then for every hybrid run* $r_1 = (\mathcal{E}_1, l_1, x_1, a_1, w_1)$ *of* Σ_1^{hyb} *with* $(l_1(0), x_1(0)) = (l_{10}, x_{10})$ *there should exist a hybrid run* $r_2 = (\mathcal{E}_2, l_2, x_2, a_2, w_2)$ *of* Σ_1^{hyb} *with* $(l_2(0), x_2(0)) = (l_{20}, x_{20})$ *such that for all times t for which the hybrid run* r_1 *is defined*

- $\mathcal{E}_1 = \mathcal{E}_2 =: \mathcal{E}$
- $w_1(t) = w_2(t)$ *for all* $t \geq 0$ *with* $t \notin \mathcal{E}$
- $a_1(t) = a_2(t)$ *for all* $t \geq 0$ *with* $t \in \mathcal{E}$

– $(l_1(t), x_1(t), l_2(t), x_2(t)) \in \mathcal{R}$ *for all* $t \geq 0$ *with* $t \notin \mathcal{E}$,

and conversely for every hybrid run r_2 *of* Σ_2^{hyb} *there should exist a hybrid run* r_1 *of* Σ_1^{hyb} *with the same properties.*

A more checkable version of hybrid bisimulation is obtained by merging the previous type of algebraic characterization of bisimulation relations for dynamical systems with the common notion of bisimulation for concurrent processes. Hereto we throughout assume that the continuous state space parts of the bisimulation relation $\mathcal{R}$, namely all sets

$$\mathcal{R}_{l_1 l_2} := \{(x_1, x_2) \mid (l_1, x_1, l_2, x_2) \in \mathcal{R}\} \subset \mathcal{X}_1 \times \mathcal{X}_2 \tag{28}$$

are submanifolds.

Definition 5 (Structural hybrid bisimulation relation). *Consider two hybrid automata* $\Sigma_i^{hyb} = (\mathcal{L}_i, \mathcal{X}_i, \mathcal{A}_i, \mathcal{W}_i, E_i, F_i), i = 1, 2,$ *as above. A structural hybrid bisimulation relation 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 all sets $\mathcal{R}_{l_1 l_2}$ *are submanifolds and have the following property. Take any* $(l_1^-, x_1^-, l_2^-, x_2^-) \in \mathcal{R}$*. Then for every* l_1^+, x_1^+, a *for which*

$$(l_1^-, l_1^+, a, x_1^-, x_1^+) \in E_1,$$

there should exist l_2^+, x_2^+ *such that*

$$(l_2^-, l_2^+, a, x_2^-, x_2^+) \in E_2$$

while $(l_1^+, x_1^+, l_2^+, x_2^+) \in \mathcal{R}$*, and conversely.*
Furthermore, take any $(l_1, x_1, l_2, x_2) \in \mathcal{R}$*. Then for every* $\dot{x}_1, w$ *for which*

$$(l_1, x_1, \dot{x}_1, w) \in F_1$$

there should exist $\dot{x}_2$ *such that*

$$(l_2, x_2, \dot{x}_2, w) \in F_2$$

while $(\dot{x}_1, \dot{x}_2) \in T_{(x_1,x_2)}\mathcal{R}_{l_1 l_2}$*, and conversely.*

It is easily seen that any structural hybrid bisimulation relation is a hybrid bisimulation relation in the sense of Definition 4. The basic observation is that the infinitesimal invariance condition $(\dot{x}_1(t), \dot{x}_2(t)) \in T_{(x_1,x_2)}\mathcal{R}_{l_1 l_2}$ implies that the trajectory $(l_1, l_2, x_1(t), x_2(t))$ remains in $\mathcal{R}$.

Definition 5 provides a checkable condition for bisimulation once we have derived algebraic conditions for $\mathcal{R}$ being a structural hybrid bisimulation relation. In particular, let $\mathcal{X}_1, \mathcal{X}_2$ be *linear* spaces with *linear* subspaces $\mathcal{R}_{l_1 l_2} \subset \mathcal{X}_1 \times \mathcal{X}_2$, while the flow conditions F assign to every $l \in \mathcal{L}$ a linear non-deterministic input-state-output system

$$\begin{aligned} \dot{x} &= A^l x + B^l u + G^l d, \; x \in \mathcal{X}, u \in \mathcal{U}, d \in \mathcal{D} \\ y &= C^l x, \qquad\qquad\qquad y \in \mathcal{Y} \end{aligned} \tag{29}$$

with $w = (u, y) \in \mathcal{W} := \mathcal{U} \times \mathcal{Y}$ and d as before a disturbance generator. Then we may use Theorem 1 to characterize the continuous part of the bisimulation.

For the important special case of *switching linear systems*, where the discrete dynamics is independent of the continuous dynamics (no invariants nor guards, reset map is the identity map) and all discrete transitions have the same action label, this has been worked out in [14]. This paper also shows how to compute in this case the *maximal bisimulation relation*, based on Algorithm 2 and the underlying discrete dynamics.

5 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 on abstraction 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. [15].

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 [8, 5]. Apart from abstraction we believe that there are other good reasons to allow 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.

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 the environment not only via their discrete actions but also via their continuous (input-output) behavior. Next step is to give an algorithm for computing the maximal structural hybrid bisimulation relation, extending the results obtained in [14] for switching linear systems without invariants and guards. Secondly, it is important to relate the proposed notion of structural hybrid bisimulation with previously proposed notions of bisimulation for hybrid systems without (or with 'abstracted') continuous external behavior, see e.g. [4, 1, 6, 7, 18, 11, 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. 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.
4. 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
5. H. Hermanns, *Interactive Markov Chains*, Lecture Notes in Computer Science 2428, Springer, 2002.
6. 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.
7. G. Lafferriere, G.J. Pappas, S. Sastry, "O-minimal hybrid systems", *Math. Contr. Signals, Syst.*, vol. 13, pp. 1-21, 2000.
8. R. Milner, *Communication and Concurrency*, Prentice Hall International Series in Computer Science, 1989.
9. R. Milner, *Communication and Mobile Systems: the* π*- Calculus*, Cambridge University Press, Cambridge, UK, 1999.
10. G.J. Pappas, G. Lafferriere, S. Sastry, Hierarchically consistent control systems, *IEEE Transactions on Automatic Control, 45(6)*, pp. 1144–1160, 2000.
11. G.J. Pappas, Bisimilar linear systems, *Automatica, 39*, pp. 2035–2047, 2003.
12. G.J. Pappas, S. Simic, Consistent abstractions of affine control systems, *IEEE Transactions on Automatic Control, 47*, pp. 745–756, 2002.
13. 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.
14. G. Pola, A.J. van der Schaft, M.D. Di Benedetto, Bisimulation theory for switching linear systems, in preparation.
15. A.J. van der Schaft, "Equivalence of dynamical systems by bisimulation", Technical Report Department of Applied Mathematics, University of Twente, October 2003, submitted for publication.
16. 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.
17. P. Tabuada, G.J. Pappas, "Bisimilar control affine systems", *Systems and Control Letters*, to appear.
18. 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.
19. W.M. Wonham, *Linear multivariable control: a geometric approach*, Springer, New York, Third Edition, 1985.