The Achievable Dynamics via Control by Interconnection

Vinjamoor, Harsh; Schaft, Arjan J. van der

Published in:
IEEE Transactions on Automatic Control

DOI:
10.1109/TAC.2010.2072650

Publication date:
2011

The Achievable Dynamics via Control by Interconnection

Harsh Vinjamoor and Arjan J. van der Schaft

***Abstract*—We consider the problem of finding a controller such that, when interconnected to the plant, we obtain a system that is equivalent to a desired system. Here, "equivalence" is formalized as "bisimilarity." We give necessary and sufficient conditions for the existence of such a controller. The systems we consider are linear input-state-output systems. A comparison is made to previously obtained results about achievable/implementable behaviors in the behavioral approach to systems theory. Among the advantages of using the notion of bisimilarity is the fact that it directly applies to state-space systems, while the computations involved are operations on constant matrices.**

***Index Terms*—Achievability, bisimulations, canonical controller, interconnection, linear systems.**

I. Motivation

A BASIC question in systems and control theory is the following: Given a plant system, by constructing another dynamical system (called a controller) and interconnecting this to the plant, what are the possibilities of modifying its input-output behavior? Before we begin addressing this question, we first explain and motivate the setting we shall work with.

We consider plant systems with two types of inputs f and u and two types of outputs z and y; see Fig. 1. The first type of input f together with the first type of output z describes the interaction of the system with its environment and can be used for performance specifications (for example, as in $\mathcal{H}_\infty$ control). We call the pair (f, z) the *manifest variables* of the system. The second type of input u and the second type of output y are variables that are to be connected to the controller system. Hence, we call the pair (u, y) the *control variables* of the system as depicted in Fig. 1.

Now suppose we are given a plant P together with another system called the desired system (or specification) S, having the same set of manifest variables (f, z). The aim is to design a controller C, if it exists, so that when we attach the controller C to the plant P, this controlled system behaves exactly like the desired system S.

A. More General Interconnections

The allowed controller interconnections that we consider are more general than the ones usually seen in controller design techniques and depicted in Fig. 1. We first motivate our more general controller interconnection and then state precisely what type of interconnections we allow in the next section. Usually, control theory deals with *feedback controllers*, i.e., controllers that accept the output y of the plant as their input and produce an output that acts as an input u to the plant. Thus, a controller is looked at as a signal processing unit. These controllers have many advantages. For instance, in the case of linear state-space systems without feed-through terms, a feedback interconnection is guaranteed to be well-posed, in the sense that, *after* attaching the controller, all plant states are allowed as initial conditions.

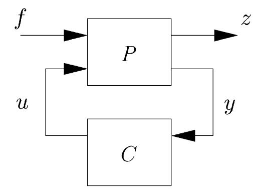


Fig. 1. Standard control configuration.

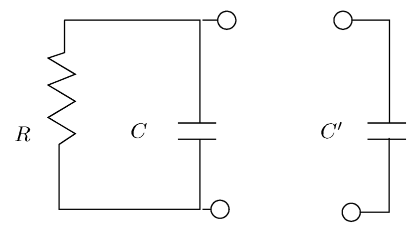


Fig. 2. Nonfeedback interconnection.

However, there are desired systems that can be achieved by interconnecting a controller to the plant, but *not* by the standard feedback type of interconnection. These considerations are not new and have already been addressed; see, for instance, the example of the "door closing mechanism" in [4] and [14]. Consider also the example of an RC-circuit, Fig. 2, in which we can attach another capacitance C' in a parallel connection to the first capacitance C. This interconnection is not a standard feedback interconnection. In fact, the input to the port of the RC-circuit is the current, and its output is the voltage across C, while for the C'-circuit, the input is again the current and the output the voltage across C'. Hence, the interconnection amounts to equating two inputs and the two outputs. Note that the resulting capacitance is $C + C' > C$ and cannot be obtained by connecting a positive (physical) capacitance in series with C. Moreover, one has to ensure that the voltages of the two capacitors are equal before interconnecting the C'-circuit to the RC-circuit. In other words, the initial conditions of the plant and the controller have to be adjusted before closing the switch. In general, this

Manuscript received July 23, 2009; revised March 05, 2010; June 29, 2010; July 19, 2010; and July 29, 2010; accepted August 17, 2010. Date of publication September 02, 2010; date of current version May 11, 2011. Recommended by Associate Editor A. Astolfi.

The authors are with the Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Groningen 9700 AK, The Netherlands (e-mail: h.g.vinjamoor@rug.nl; a.j.van.der.schaft@rug.nl).

Digital Object Identifier 10.1109/TAC.2010.2072650

type of interconnection often occurs in physical systems' interconnection.

Finally, we show a mathematical example that illustrates the possible need for interconnections other than the standard feedback interconnection. Suppose we have a linear plant

$$\begin{aligned} \dot{x}_P &= \begin{bmatrix} 1 & 0 \\ a & 1 \end{bmatrix} x_P + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u_P, \quad x_P \in \mathbb{R}^2 \\ z_P &= [1 \quad 0] x_P \\ y_P &= [1 \quad 0] x_P \end{aligned} \tag{1}$$

where $a \in \mathbb{R}$. Suppose the desired system is just the zero system; hence, the aim is to design a controller so that the output z_P of the plant is identically zero. The first step is to compute the largest controlled invariant subspace contained in the kernel of the output map corresponding to the output z_P (see [1] or [16]). In this case, it is the subspace spanned by $\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$. To ensure that z_P is identically zero, the initial conditions of the plant have to be restricted to this subspace. This will be achieved by setting $y_P = 0$, and the input u_P is uniquely determined as $u_P = 0$. Thus, $u_P = 0$, $y_P = 0$ is a controller that achieves the desired behavior and is *not* in the standard feedback configuration.

B. Equivalence of Dynamical Systems

In the problem statement, we had left one more issue vague. When do we say that the controlled system behaves exactly like the desired system? Thus, we need a notion of equivalence between systems. For input–output systems, the classical notion of equivalence is the equality of transfer matrices, while for state-space systems, the classical notion of equivalence is the existence of an invertible state-space transformation (similarity transformation). In the behavioral approach (see Remark 16 later), two systems are equivalent if their behaviors are equal. An intuitive idea that combines all these approaches is the following: We shall say that a state-space system A is "equal" to another state-space system B if for every initial state of A, there exists an initial state of B, and for every initial state of B, there exists an initial state of A such that for every input applied simultaneously to A and B, the outputs of A and B are identical. This has been correctly formalized with the language of bisimulation relations. We give the precise definitions in Section II.

The notion of bisimulations originates from computer science. It was introduced by [5] and [7] in the context of concurrent processes. For deterministic automata, equivalence in the sense of bisimulations is the same as language equality. For nondeterministic automata, however, bisimilarity is a stronger notion than language equality. Recently, this notion has been fruitfully extended to continuous input-state-output systems in [6] and [12]. Also it has been found that this notion is stronger than behavior equality; see [12, Example 2.15]. Moreover, via the concept of *reduction to a minimal state-space system*, the notion of bisimulation combines the ideas of input–output behavior equality and state-space equivalence. An important reason for using bisimulation equivalence is that one can avail of various ideas and algorithms from the geometric theory of linear systems. Analogously, using the geometric theory for nonlinear systems, the definition of bisimulation is directly extendable to nonlinear systems (see [12]). It will turn out that the main theorem of this paper can be formulated in terms of the one-sided notion of bisimulation called simulation.

Thus, the problem that we address is the following: Given the plant dynamics and a system with desired dynamics, find necessary and sufficient conditions for the existence of a controller (another dynamical system) such that when the controller is interconnected to the plant, the resulting interconnected system is bisimilar to the system with desired dynamics.

Related questions have been addressed for abstract state systems in [8]. The recent paper [9] also addresses this question in a very general category theory framework, thus encompassing a much larger class of systems. However, the problem is not identical to the one we address. In [9], the variables of the plant are not partitioned into manifest and control variables. In fact, the manifest variables are also the control variables. Thus, it is a special case of our setting. Consequently, as expected, when we restrict our main result (Section III) to this special case, we do indeed recover the condition for existence of a controller as stated in [9]. We elaborate more on the relation between the results in [9] and our results in the conclusion in Section IV.

This paper is organized as follows. In Section II, we state the necessary definitions of bisimulations, state exactly what interconnecting two systems means, and recall and derive some preliminary results. In Section III, we first describe the equations defining our plant and desired dynamics, and then formulate the problem statement in terms of these systems. Thereafter, we prove the main result of this paper. We wrap up with some concluding comments in Section IV.

II. Definitions

In this section, we give precise definitions of various notions needed and also derive some results that are useful for proving the main result. We first define a bisimulation relation as introduced in [6] and [12]. Consider two dynamical systems described by the following equations:

$$\Sigma_i : \quad \begin{aligned} \dot{x}_i &= A_i x_i + B_i^u u_i + B_i^f f_i \\ y_i &= C_i^y x_i \\ z_i &= C_i^z x_i \end{aligned} \tag{2}$$

where $x_i \in X_i, u_i \in U, f_i \in F, z_i \in Z, y_i \in Y$ and $i \in \{1, 2\}$. Here, x_i is the state of the system Σ_i; u_i, f_i are inputs, while y_i and z_i are outputs that take values in the finite dimensional real vector spaces U, F, Y, and Z, respectively. The state space X_i is also a finite dimensional real vector space. In the sequel, we will denote by $\mathcal{U}$, $\mathcal{Y}$, $\mathcal{Z}$, and $\mathcal{F}$ function spaces of real valued functions that take values in the vector spaces U, Y, Z, and F, respectively. For simplicity of notation, we will denote the elements of $\mathcal{U}$, $\mathcal{Y}$, $\mathcal{Z}$, and $\mathcal{F}$ also by u_i, y_i, z_i, and f_i, respectively.

Definition 1: [12, Definition 2.1]: A *bisimulation relation* between two linear systems Σ_1 and Σ_2 with respect to the variables f_i and z_i is a linear subspace $\mathrm{R} \subset X_1 \times X_2$ with the following property. Take any $(x_{10}, x_{20}) \in \mathrm{R}$ and any common input function $f_1 = f_2$. Then, for any input function u_1, there exists an input function u_2 such that the resulting state trajectories $x_1(t)$ with $x_1(0) = x_{10}$ and $x_2(t)$ with $x_2(0) = x_{20}$ satisfy

$$\begin{aligned} (x_1(t), x_2(t)) &\in \mathrm{R} \text{ for all } t \geq 0 \\ z_1(t) &= z_2(t) \text{ for all } t \geq 0 \end{aligned} \tag{3}$$

and conversely, for any input function u_2, there should exist a function u_1 such that the state trajectories $x_1(t)$ and $x_2(t)$ and outputs z_1 and z_2 satisfy (3).

Two systems Σ_1 and Σ_2 are said to be *bisimilar*, denoted $\Sigma_1 \approx \Sigma_2$, if there exists a bisimulation relation $\mathrm{R} \subset X_1 \times X_2$ such that $\pi_1(\mathrm{R}) = X_1$ and $\pi_2(\mathrm{R}) = X_2$, where $\pi_i : X_1 \times X_2 \to X_i$, $i = 1, 2$, denote the canonical projections. Such a bisimulation relation is called a *full* bisimulation relation.

For linear time-invariant systems, one can assume without loss of generality that bisimulation relations are linear subspaces; see [12, Remark 2.2]. Note that in the definition of a bisimulation relation with respect to the variables f_i and z_i, the output y_i does not play any role. However, it is used when we interconnect the plant to a controller (u_i and y_i are the variables available to the controller).

A bisimulation relation can be explicitly characterized by conditions involving the matrices describing the two systems ([12]; see also [6]).

Proposition 2: [12, Theorem 2.10]: Let Σ_1 and Σ_2 be two systems of the form given in (2). A subspace $\mathrm{R} \subseteq X_1 \times X_2$ is a bisimulation relation with respect to f_i and z_i if and only if the following hold true:

$$\begin{aligned} \mathrm{R} + \mathrm{im} \begin{bmatrix} B_1^u \\ 0 \end{bmatrix} &= \mathrm{R} + \mathrm{im} \begin{bmatrix} 0 \\ B_2^u \end{bmatrix} =: \mathrm{R_e} \\ \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathrm{R} &\subseteq \mathrm{R}_e \\ \mathrm{im} \begin{bmatrix} B_1^f \\ B_2^f \end{bmatrix} &\subseteq \mathrm{R}_e \\ \mathrm{R} &\subseteq \ker \begin{bmatrix} C_1^z & -C_2^z \end{bmatrix}. \end{aligned} \tag{4}$$

There is also a one-sided notion of bisimulation called simulation. The main theorem that we prove is in terms of simulation relations that are defined as follows.

Definition 3: A *simulation relation* of Σ_1 by Σ_2 with respect to f_i and z_i is a linear subspace $L \subset X_1 \times X_2$ with the following property. Take any $(x_{10}, x_{20}) \in L$ and any joint input function $f_1 = f_2$. Then, for any input function u_1, there exists an input function u_2 such that the resulting state trajectories $x_1(t)$ with $x_1(0) = x_{10}$ and $x_2(t)$ with $x_2(0) = x_{20}$ satisfy

$$\begin{aligned} (x_1(t), x_2(t)) &\in L \text{ for all } t \geq 0 \\ z_1(t) &= z_2(t) \text{ for all } t \geq 0. \end{aligned} \tag{5}$$

System Σ_1 is said to be *simulated* by system Σ_2 (or equivalently, Σ_2 simulates Σ_1), denoted $\Sigma_1 \preceq \Sigma_2$, if there exists a simulation relation L of Σ_1 by Σ_2 such that $\pi_1(L) = X_1$. Such a simulation relation is called a *full* simulation relation of Σ_1 by Σ_2.

A subspace $L \subseteq X_1 \times X_2$ is a simulation relation of Σ_1 by Σ_2 with respect to f_i and z_i if and only if the following are true ([12, Proposition 5.2]):

$$\begin{aligned} L + \mathrm{im} \begin{bmatrix} B_1^u \\ 0 \end{bmatrix} &\subseteq L + \mathrm{im} \begin{bmatrix} 0 \\ B_2^u \end{bmatrix} =: \mathrm{L_e} \\ \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} L &\subseteq L_e \\ \mathrm{im} \begin{bmatrix} B_1^f \\ B_2^f \end{bmatrix} &\subseteq L_e \\ L &\subseteq \ker \begin{bmatrix} C_1^z & -C_2^z \end{bmatrix}. \end{aligned} \tag{6}$$

The following lemma shows that the relation $\preceq$ is transitive.

Lemma 4: Let Σ_1, Σ_2, and Σ_3 be three systems of the form of (2). If $\Sigma_1 \preceq \Sigma_2$ and $\Sigma_2 \preceq \Sigma_3$, then $\Sigma_1 \preceq \Sigma_3$.

Proof: The simulation relation of Σ_1 by Σ_3 can be constructed as follows. Let $R_{12} \subseteq X_1 \times X_2$ and $R_{23} \subseteq X_2 \times X_3$ be full simulation relations of Σ_1 by Σ_2 and of Σ_2 by Σ_3, respectively. Then, a full simulation relation of Σ_1 by Σ_3 is given by $\{(x_1, x_3) \in X_1 \times X_3 | \exists x_2 \in X_2 \text{ such that } (x_1, x_2) \in R_{12} \text{ and } (x_2, x_3) \in R_{23}\}$. ●●

The definition of simulation relations seems to suggest that two systems Σ_1 and Σ_2 are bisimilar if and only if $\Sigma_1 \preceq \Sigma_2$ and $\Sigma_2 \preceq \Sigma_1$. As it turns out, for linear time-invariant systems, this is indeed true. Note that this is not true in general; see [5] and also [12, Section 5].

Proposition 5: [12, Proposition 5.3]: Let $L \subset X_1 \times X_2$ be a full simulation relation of Σ_1 by Σ_2 and $T \subset X_2 \times X_1$ be a full simulation relation of Σ_2 by Σ_1. Then, $\Sigma_1 \approx \Sigma_2$, where the full bisimulation relation is given by $L + T^{-1}$, with $T^{-1} := \{(x_a, x_b) | (x_b, x_a) \in T\}$.

Whenever there exists a (bi)simulation, also the maximal (bi)simulation relation exists. In [12], the following algorithm is given for computing the maximal simulation of Σ_1 by Σ_2, where Σ_1 and Σ_2 are of the form given in (2). The algorithm is very similar to the algorithm used to find the maximal controlled invariant subspace contained in a given subspace of the state space (see [1] or [16]).

Define $A := \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}$, $G_1 := \begin{bmatrix} B_1^u \\ 0 \end{bmatrix}$, $G_2 := \begin{bmatrix} 0 \\ B_2^u \end{bmatrix}$, and $C := [C_1^z \;\; -C_2^z]$. Consider the following descending sequence of subspaces L^j:

$$\begin{aligned} L^0 &= X_1 \times X_2 \\ L^1 &= \{z \in L^0 | z \in \ker C\} \\ L^2 &= \{z \in L^1 | Az + \mathrm{im}\, G_1 \subset L^1 + \mathrm{im}\, G_2\} \\ &\;\;\vdots \\ L^{j+1} &= \{z \in L^j | Az + \mathrm{im}\, G_1 \subset L^j + \mathrm{im}\, G_2\}. \end{aligned} \tag{7}$$

The algorithm terminates whenever L^j is empty or $L^j = L^{j+1}$ for some $j \geqslant 0$. In the first case, there does not exist a simulation relation of Σ_1 by Σ_2, while in the second case $L^j = L^{j+1} =: L$

is the maximal simulation relation if and only if $\operatorname{im}\begin{bmatrix} B_1^f \\ B_2^f \end{bmatrix} \subseteq L + \operatorname{im}\begin{bmatrix} 0 \\ B_2^u \end{bmatrix}$. Furthermore, the number of steps j is bounded by the dimension of $X_1 \times X_2$; see [12, Equation 44]. The algorithm for computing a bisimulation relation is analogous (see [12]). In fact, when a bisimulation relation exists, it can be computed using the algorithm mentioned in (7); see [12, Proposition 5.4]. The above algorithm for computing the simulation relations can thus be used to verify the necessary and sufficient condition derived in the next section.

We now introduce some notation that we will use extensively in this paper. Recall that throughout the paper we refer to the variables (f_i, z_i) as the *manifest* variables and the variables (u_i, y_i) as the *control* variables. Note that simulation and bisimulation relations are always defined with respect to the manifest variables (f_i, z_i).

The interconnection of two systems can be with respect to either the manifest variables or the control variables; we shall indicate this by subscripts m and c, respectively. In order to allow for more general interconnections than the standard feedback one, we will use a permutation matrix Π in the following definition.

Definition 6: Let Σ_1 and Σ_2 be two systems of the form of (2). Their interconnection through the manifest variables via the interconnection matrix Π, denoted by $\Sigma_1 \|_m^\Pi \Sigma_2$, is defined by the following two sets of equations:

1)

$$\begin{aligned} \dot{x}_i &= A_i x_i + B_i^u u_i + B_i^f f_i \\ y_i &= C_i^y x_i \qquad i = 1, 2 \\ z_i &= C_i^z x_i \end{aligned}$$

2)

$$\begin{bmatrix} f_1(t) \\ z_1(t) \end{bmatrix} = \Pi \begin{bmatrix} f_2(t) \\ z_2(t) \end{bmatrix} \qquad \forall t \geq 0$$

where Π is a permutation matrix.

In general, the system $\Sigma_1 \|_m^\Pi \Sigma_2$ is a differential-algebraic system with constraints on the state variables (x_1, x_2). The state space of such a system, denoted by X_{Σ_1, Σ_2}, is defined by

$$\Big\{ (x_1, x_2) \in X_{\Sigma_1} \times X_{\Sigma_2} | x_1(0) = x_1, \\ x_2(0) = x_2 \text{ and } \exists \text{ functions } f_1, f_2, u_1, u_2 \\ \text{such that } \begin{bmatrix} f_1(t) \\ z_1(t) \end{bmatrix} = \Pi \begin{bmatrix} f_2(t) \\ z_2(t) \end{bmatrix} \forall t \geq 0 \Big\}.$$

It is clear that this is a controlled invariant subspace.

Similarly, interconnection through the control variables and a suitable permutation matrix Π is denoted by $\Sigma_1 \|_c^\Pi \Sigma_2$, where the first set of equations are as in Definition 6, while the second set of equations are now $\begin{bmatrix} u_1 \\ y_1 \end{bmatrix} = \Pi \begin{bmatrix} u_2 \\ y_2 \end{bmatrix}$. We shall denote the state space of such an interconnected system also simply by X_{Σ_1, Σ_2} whenever Π and the interconnection variables (m or c) are clear from the context. Furthermore, whenever the permutation matrix $\Pi = I$ (the identity matrix), we shall simply drop it from our notation, i.e., we shall write $\Sigma_1 \|_m \Sigma_2$ instead of $\Sigma_1 \|_m^I \Sigma_2$. Note that the standard feedback interconnection is given by $\begin{bmatrix} u_1 \\ y_1 \end{bmatrix} = \Pi \begin{bmatrix} u_2 \\ y_2 \end{bmatrix}$ with $\Pi = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$, with I an identity matrix of appropriate dimensions.

A particular subsystem of the original system Σ_i, of the form in (2), which will be important in the sequel, is obtained by setting the control variables (u_i, y_i) of the system to zero. The resulting system denoted by Σ_i^c is defined by the differential-algebraic system

$$\Sigma_i^c: \quad \begin{aligned} \dot{x}_i &= A_i x_i + B_i^f f_i \\ 0 &= y_i = C_i^y x_i \\ z_i &= C_i^z x_i. \end{aligned} \tag{8}$$

The (constrained) state space of this system, denoted by $X_{\Sigma_i^c}$, is the largest (A_i, B_i^f)-invariant subspace (see [16] for details) contained in $\ker(C_i^y)$. In the original system Σ_i, f_i is free to be any input function. Now, however, since $y_i = 0$, every input function f_i for Σ_i^c can be written as $f_i = F x_i + L w$, where F is such that $(A_i + B_i^f F) X_{\Sigma_i^c} \subseteq X_{\Sigma_i^c}$, L is such that $\operatorname{im} B_i^f L = \operatorname{im} B_i^f \cap X_{\Sigma_i^c}$ and w is any function that takes values in $\mathbb{R}^d$, where d is the dimension of the subspace $\operatorname{im} B_i^f \cap X_{\Sigma_i^c}$. This set of input functions f_i in the system Σ_i^c will be denoted by $\mathcal{I}(f_i, \Sigma_i^c)$.

We shall use the notation $(x_1(0), u_1, y_1, f_1, z_1) \in \Sigma_1$ to indicate that starting with an initial condition $x_1(0)$, if we apply the input functions (u_1, f_1) to the system Σ_1, then (y_1, z_1) will be the resulting output functions. Similarly, we shall denote a trajectory in an interconnected system $\Sigma_1 \|_m^\Pi \Sigma_2$ by the notation $(x_1(0), u_1, y_1, f_1, z_1) \times (x_2(0), u_2, y_2, f_2, z_2) \in \Sigma_1 \|_m^\Pi \Sigma_2$ to indicate that $(x_i(0), u_i, y_i, f_i, z_i) \in \Sigma_i$ for $i = 1, 2$ while the functions (f_i, z_i) satisfy $\begin{bmatrix} f_1 \\ z_1 \end{bmatrix} = \Pi \begin{bmatrix} f_2 \\ z_2 \end{bmatrix}$.

III. Existence of a Controller

We now formulate the problem statement precisely. Let P denote the plant system given by the following equations:

$$P: \quad \begin{aligned} \dot{x}_P &= A_P x_p + B_P^u u_P + B_P^f f_P, \qquad x_P \in X_P \\ z_P &= C_P^z x_P \\ y_P &= C_P^y x_P. \end{aligned} \tag{9}$$

Let S denote the desired system with equations as

$$S: \quad \begin{aligned} \dot{x}_S &= A_S x_S + B_S^f f_S, \qquad x_S \in X_S \\ z_S &= C_S^z x_S. \end{aligned} \tag{10}$$

Consider a controller system[1] defined by the equations

$$C: \quad \begin{aligned} \dot{x}_C &= A_C x_C + B_C^u u_C, \qquad x_C \in X_C \\ y_C &= C_C^y x_C. \end{aligned} \tag{11}$$

Thus, the definitions of simulation, bisimulation, interconnection, etc., can all be applied to these systems.

[1]Note that all the above systems are special cases of the system Σ_i given in (2).

Problem Statement: Given P and S, find necessary and sufficient conditions for the existence of a controller C such that $P\|_c^{\Pi}C$ is bisimilar to S for some permutation matrix Π. When such a controller exists, we shall say that S is *achievable* from P, or just *achievable* when the plant is clear from the context.

We now state the main result of this paper. It provides necessary and sufficient conditions for the achievability of S from P.

Theorem 7: S is achievable if and only if $P^c \preceq S \preceq P$.

The necessity of the conditions is intuitively expected. The condition $S \preceq P$ is necessary, for otherwise one would have trajectories in S that cannot be generated by P. The necessity of $P^c \preceq S$ is a little more subtle. P^c is the behavior that is present whenever $u_P = 0$ and $y_P = 0$; see (8). Owing to linearity, whichever controller we attach, the trajectories of P^c will continue to exist in the controlled system, and hence must be contained in S.

For proving the sufficiency of the condition $P^c \preceq S \preceq P$, we explicitly construct a controller that achieves S. The controller we shall use is the canonical controller, denoted by $C_{\mathtt{can}}$ (see Fig. 3) introduced in [11] in a behavioral setting. $C_{\mathtt{can}}$ is defined by $C_{\mathtt{can}} := S\|_m P$ (recall that this notation implies that $\Pi = I$). The equations governing the dynamics of $C_{\mathtt{can}}$ are as follows:

$$\begin{aligned}\begin{bmatrix}\dot{x}_S\\ \dot{x}_P\end{bmatrix} &= \begin{bmatrix}A_S & 0\\ 0 & A_P\end{bmatrix}\begin{bmatrix}x_S\\ x_P\end{bmatrix} + \begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix} f + \begin{bmatrix}0\\ B_P^u\end{bmatrix} u\\ 0 &= [C_S^z \quad -C_P^z]\begin{bmatrix}x_S\\ x_P\end{bmatrix} = z_S - z_P\\ y &= [0 \quad C_P^y]\begin{bmatrix}x_S\\ x_P\end{bmatrix}.\end{aligned} \tag{12}$$

Let $A_{\text{can}} = \begin{bmatrix}A_S & 0\\ 0 & A_P\end{bmatrix}$ and $B_{\text{can}} := \begin{bmatrix}B_S^f & 0\\ B_P^f & B_P^u\end{bmatrix}$ where $f = f_P = f_S$ and $u = u_P$. By construction, the (constrained) state space X_{can} for $C_{\mathtt{can}}$ is the largest $(A_{\text{can}}, B_{\text{can}})$-invariant subspace contained in $\ker[C_S^z \;\; -C_P^z]$, i.e., X_{can} is the largest subspace of $X_S \times X_P$ such that

$$\begin{aligned}A_{\text{can}}X_{\text{can}} &\subseteq X_{\text{can}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix} + \operatorname{im}\begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix}\\ X_{\text{can}} &\subseteq \ker\left[C_S^z | - C_P^z\right].\end{aligned} \tag{13}$$

Before proving Theorem 7, we need a couple of preliminary results that are of interest by themselves. The maximal simulation relation of S by P is the largest subspace $\mathrm{R_{SP}} \subseteq X_S \times X_P$ such that

$$\begin{aligned}A_{\text{can}}\mathrm{R_{SP}} &\subseteq \mathrm{R_{SP}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix}\\ \mathrm{R_{SP}} &\subseteq \ker\left[C_S^z \quad -C_P^z\right]\\ \operatorname{im}\begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix} &\subseteq \mathrm{R_{SP}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix}.\end{aligned} \tag{14}$$

Observe that in (12) if we set $f = 0$, then the resulting set of equations are exactly those that are used to compute $\mathrm{R_{SP}}$. This suggests a close relationship between X_{can} and $\mathrm{R_{SP}}$. In fact, the following lemma is true.

Lemma 8: Given that $\mathrm{R_{SP}}$ exists, $\mathrm{R_{SP}} = X_{\text{can}}$.

Proof: ($\mathrm{R_{SP}} \subseteq X_{\text{can}}$): We know that $A_{can}\mathrm{R_{SP}} \subseteq \mathrm{R_{SP}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix}$. Hence, $A_{\text{can}}\mathrm{R_{SP}} \subseteq \mathrm{R_{SP}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix} + \operatorname{im}\begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix}$.

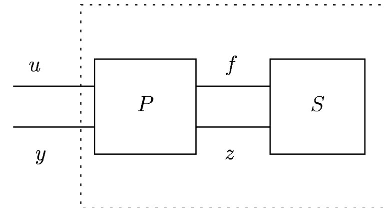


Fig. 3. Canonical controller.

Also, we know that $\mathrm{R_{SP}} \subseteq \ker[C_S^z | \; -C_P^z]$. Hence, $\mathrm{R_{SP}} \subseteq X_{\text{can}}$.

($X_{\text{can}} \subseteq \mathrm{R_{SP}}$): Since $\mathrm{R_{SP}}$ exists we know that $\operatorname{im}\begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix} \subseteq \mathrm{R_{SP}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix}$. As just proved, $\mathrm{R_{SP}} \subseteq X_{\text{can}}$. Hence, $\operatorname{im}\begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix} \subseteq X_{\text{can}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix}$. Thus, since $A_{\text{can}}X_{\text{can}} \subseteq X_{\text{can}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix} + \operatorname{im}\begin{bmatrix}B_S^f\\ B_P^f\end{bmatrix}$, we have $A_{\text{can}}X_{\text{can}} \subseteq X_{\text{can}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix} + X_{\text{can}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix} = X_{\text{can}} + \operatorname{im}\begin{bmatrix}0\\ B_P^u\end{bmatrix}$. Also, by definition, $X_{\text{can}} \subseteq \ker[C_S^z | -C_P^z]$. Thus, $X_{\text{can}} \subseteq \mathrm{R_{SP}}$. ●●

We need one more result about the plant system before we can prove Theorem 7. Suppose $(x_P(0), u, y, f, z) \in P$. Now, assuming u and y are fixed, we characterize the set of inputs f' and the set of states $x'_P(0)$ such that $(x'_P(0), u, y, f', z'_P) \in P$ for some z'_P.

Lemma 9: The two trajectories $(x_P(0), u, y, f, z)$ and $(x'_P(0), u, y, f', z')$ are *both* trajectories in P if and only if $x'_P(0) - x_P(0) \in X_{P^c}$ and $f' - f \in \mathcal{I}(f_P, P^c)$ [recall that $\mathcal{I}(f_P, P^c)$ is the set of allowed inputs in P^c; see (8) and the following text].

Proof: The output due to initial state $x_P(0)$ and input functions u and f is

$$\begin{aligned}y_P(t) = {} & C_P^y e^{(A_P t)} x_P(0) + C_P^y \int_0^t e^{(A_P(t-\tau))} B_f^P f_P(\tau) d\tau\\ & + C_P^y \int_0^t e^{(A_P(t-\tau))} B_u^P u_P(\tau) d\tau.\end{aligned} \tag{15}$$

The output due to state $x'_P(0)$ and input f'_P is obtained by replacing $x_P(0)$ by $x'_P(0)$, and likewise f_P by f'_P in the above equation (because the input u_P is the same).

($\Rightarrow$): Subtracting the output due to $x_P(0)$ from the output due to $x'_P(0)$ we get

$$\begin{aligned}0 = {} & C_P^y e^{(A_P t)} \left(x_P(0) - x'_P(0)\right)\\ & + C_P^y \int_0^t e^{(A_P(t-\tau))} B_f^P \left(f_P(\tau) - f'_P(\tau)\right) d\tau.\end{aligned} \tag{16}$$

Thus, $(x_P(0) - x'_P(0)) \in X_{P^c}$ and $f_P - f'_P \in \mathcal{I}(f_P, P^c)$.

($\Leftarrow$): Given $(x_P(0) - x'_P(0)) \in X_{P^c}$ and $f_P - f'_P \in \mathcal{I}(f_P, P^c)$, we know that (16) holds true. Hence, $C_P^y e^{(A_P t)} x_P(0) + C_P^y \int_0^t e^{(A_P(t-\tau))} B_f^P f_P(\tau) d\tau =$

$C_P^y e^{(A_P t)} x_P'(0) + C_P^y \int_0^t e^{(A_P(t-\tau))} B_f^P f_P'(\tau) d\tau$. Adding $C_P^y \int_0^t e^{(A_P(t-\tau))} B_u^P u_P(\tau) d\tau$ to both sides yields the desired result. ●●

We now prove Theorem 7.

Proof of Theorem 7: ($\Leftarrow$:) Suppose $P\|_c^\Pi C \approx S$ for some Π. By Lemma 4, it is sufficient to show that $P\|_c^\Pi C$ is simulated by P and itself simulates P^c. Let $(x_P(0), u_P, y_P, f_P, z_P) \times (x_C(0), u_C, y_C) \in P\|_c^\Pi C$ and $(x_P(0), u_P, y_P', f_P, z_P') \in P$. Then, by the uniqueness of the solutions of a linear differential equation, we have that $z_P = z_P'$ and $y_P = y_P'$. Thus

$$\{(a,b,c) \in X_P \times X_C \times X_P | a = c \text{ and } (a,b) \in X_{P,C}\}$$

is a simulation relation of $P\|_c^\Pi C$ by P. Now, consider any $(x_P(0), 0, 0, f, z_P) \in P^c$. Then, take $(x_P(0), 0, 0, f, z_p') \times (0,0,0) \in P\|_c^\Pi C$; this can be done since $(x_P(0), 0)$ always is in $X_{P,C}$. Then, by the uniqueness of solutions of differential equations, $z_P = z_P'$. Thus

$$\{(a,b,c) \in X_{P^c} \times X_P \times X_C | a = b \text{ and } c = 0\}$$

is a simulation relation of P^c by $P\|_c^\Pi C$.

($\Rightarrow$:) Here we have to construct a controller C, choose a permutation matrix Π, and then show that $P\|_c^\Pi C \approx S$. For this, we construct the canonical controller $C_{\mathtt{can}}$ [see (12)]. We will show that under the conditions $P^c \preceq S \preceq P$, we have $C_{\mathtt{can}}\|_c P \approx S$. Proving this is equivalent (see Proposition 5) to proving the following two statements.

1) $S \preceq C_{\mathtt{can}}\|_c P$.
2) $C_{\mathtt{can}}\|_c P \preceq S$.

1) Let $(x_S(0), f, z_S) \in S$. Since $S \preceq P$, there exist $x_P(0)$, u, y, and z_P such that $(x_S(0), f, z_S) \times (x_P(0), u, y, f, z_P) \times (x_P(0), u, y, f, z_P) \in C_{\mathtt{can}}\|_c P$ with $(x_S(0), x_P(0)) \in \mathrm{R_{SP}}$. Then, by the uniqueness of the solutions of a linear differential equation, $z_P = z_S$. Thus, the simulation relation $S \preceq C_{\mathtt{can}}\|_c P$ is given by

$$\{(a,b,c,d) \in X_S \times X_S \times X_P \times X_P | \\ a = b, c = d \text{ and } (b,c) \in \mathrm{R_{SP}}\}.$$

2) We shall denote the maximal simulation relation of P^c by S by the symbol $\mathrm{R_{P^cS}}$. Let $(x_S(0), f, z_S = z_P) \times (x_P(0), u, y, f, z_P) \times (x_P'(0), y, y, f', z_P') \in C_{\mathtt{can}}\|_c P$. Note that by Lemma 8 we have that $(x_S(0), x_P(0)) \in \mathrm{R_{SP}}$, and by Lemma 9 $(x_P'(0) - x_P(0)) \in X_{P^c}$ and $f' - f \in \mathcal{I}(f_P, P^c)$. Since $P^c \preceq S$, there exists a state $x_{P^cS}(0) \in X_S$ such that $(x_P'(0) - x_P(0), 0, 0, f' - f, z_P' - z_P) \in P^c$ and $(x_{NS}(0), f' - f, z_P' - z_P) \in S$ where $(x_P'(0) - x_P(0), x_{NS}) \in \mathrm{R_{P^cS}}$. To this trajectory in S, we can add the trajectory $(x_S(0), f, z_S = z_P)$ to get $(x_S(0) + x_{NS}(0), f', z_P') \in S$. Thus, we have found a state in S such that for the same input $f' = f_P = f_S$, the outputs of S and $C_{\mathtt{can}}\|_c P$ are equal to z_P'. Hence, the simulation relation $C_{\mathtt{can}}\|_c P \preceq S$ is given by

$$\{(a,b,c,d) \in X_S \times X_P \times X_P \times X_S | (a,b) \in X_{\mathrm{can}}, \\ c - b \in X_{P^c}, ((c-b),(d-a)) \in \mathrm{R_{P^cS}}\}.$$

This proves the result. ●●

Remark 10: From the previous proof, one can see that Theorem 7 can also be rephrased as follows:

$$C_{\mathtt{can}} \underset{c}{\|} P \approx S \Leftrightarrow S \text{ is achievable.}$$

In other words, if there exists a controller for some interconnection matrix Π, then the canonical controller, with the identity matrix as the interconnection matrix, also works. As a result, one just has to check whether the canonical controller achieves S or not to decide about the existence of a controller that achieves S. This greatly narrows down the search for controllers—hence the word "canonical." Furthermore, given P and S, one can immediately construct the canonical controller. Observe that for the example stated in Section I-A [see (1)] we can just as well use the canonical controller instead of the controller $y_P = 0$, $u_P = 0$, which is a controller designed by *ad hoc* means.

Remark 11: Another interesting consequence of Theorem 7 (and one more reason for the word *canonical*) is that any controller that achieves S is *simulated by the canonical controller* in the following sense. Suppose C is a controller such that $P\|_c^\Pi C \approx S$ for some interconnection matrix Π. By Theorem 7, we also have $S \approx C_{\mathtt{can}}\|_c P$. Now let $(x_P(0), f, z, u_P, y_P) \times (x_C(0), u_C, y_C) \in P\|_c^\Pi C$. Since $P\|_c^\Pi C \approx S$, there exists some $(x_S(0), f_S, z_S) \in S$ such that $f_S = f$ and $z_S = z$. Hence, $(x_S(0), f, z) \times (x_P(0), f, z, u, y) \times (x_P(0), f, z, u, y) \in C_{\mathtt{can}}\|_c P$. Thus, $C\|_c^\Pi P \preceq_c C_{\mathtt{can}}\|_c P$ with simulation relation $\{(a,b,c,d,e) \in X_C \times X_P \times X_S \times X_P \times X_P | d = e = b \text{ and } (c,b) \in \mathrm{R_{SP}} \text{ and } (b,a) \in X_{P,C}\}$, where $\preceq_c$ is a simulation with respect to the control variables. This observation implies that controllers other than the canonical controller may still be obtained as subsystems of the canonical controller. A result very much in this spirit has been obtained in the behavioral approach in [3].

Remark 12: A question that may arise about the interpretation of Theorem 7 is the following: How can a system S that has no inputs other than f_S be bisimilar to $C_{\mathtt{can}}\|_c P$? (Recall that $C_{\mathtt{can}}$ itself might have some freedom in the input u_P and has also an input f_P'.) To see what is happening, observe the following: In the notation used in the proof of Theorem 7, $C_{\mathtt{can}}\|_c P$ has input f' (see part 2 of the ($\Rightarrow$) implication of the proof of Theorem 7) and the input functions are u and f. As noted in the proof, $f' - f \in \mathcal{I}(f_P, P^c)$. Thus, for such input functions, the proof shows that $C_{\mathtt{can}}\|_c P \preceq S$. However, there could also be nonuniqueness in the input function u, i.e., $(x_S(0), f, z) \times (x_P(0), u, y, f, z) \times (x_P'(0), u, y, f', z')$ and $(x_S(0), f, z) \times (x_P(0), u', y', f, z) \times (x_P'(0), u', y', f', z'')$ may *both* be trajectories in $C_{\mathtt{can}}\|_c P$. For the two systems to be bisimilar, we must have that $z' = z''$. For linear time-invariant systems, this is always true and can be seen as follows: Subtracting the two trajectories yields a trajectory $(0,0,0) \times (0, u - u', y - y', 0, 0) \times (0, u - u', y - y', 0, z' - z'')$, where $(0, u - u', y - y', 0, 0)$ and $(0, u - u', y - y', 0, z' - z'')$ are both trajectories in P with the same states and inputs. By the uniqueness of solutions of differential equations, we have that $z' = z''$. This shows that the nonuniqueness of the input function u does not affect the bisimilarity of the two systems as it does not influence the input–output behavior of the system

$C_{\mathtt{can}}\|_c P$. This explains why Theorem 7 works despite the presence of u in $C_{\mathtt{can}}\|_c P$. In fact, this is a crucial issue when extending these results to the nonlinear case; see [13].

Remark 13: It is interesting to see what happens if $(P^c \preceq S \preceq P)$ does not hold true but we still interconnect $C_{\mathtt{can}}$ to the plant system. Suppose $P^c \preceq S$ but $S \not\preceq P$. In this case, $C_{\mathtt{can}}\|_c^\Pi P \preceq S$ but not $S \preceq C_{\mathtt{can}}\|_c^\Pi P$, i.e., $C_{\mathtt{can}}\|_c^\Pi P$ is "smaller" than required. If, on the other hand, $P^c \not\preceq S$ but $S \preceq P$, then $S \preceq C_{\mathtt{can}}\|_c^\Pi P$ holds true, but not $C_{\mathtt{can}}\|_c^\Pi P \preceq S$, i.e., $C_{\mathtt{can}}\|_c^\Pi P$ "contains" the desired system, but is still "larger" than it. See [11] for a related discussion in the behavioral framework.

Remark 14: Note that, in general, the algorithm for computing the maximal simulation relation [see (7) and the text after it] can terminate if for some j we have that S^j is empty. However, for the systems P^c, S, and P, used in the above condition, this cannot happen because for the systems being simulated, i.e., P^c and S, each has only one type of input. Thus, the algorithm in (7) reduces to that of computing the maximal controlled invariant subspace contained in a certain subspace and hence always terminates to yield some subspace (possibly the zero subspace).

Finally, we derive two immediate extensions of Theorem 7. In the first extension, we replace S by a general system D of the form

$$D: \quad \begin{aligned} \dot{x}_D &= A_D x_D + B_D^{f_1} f_1 + B_D^{f_2} f_2 \\ z_1 &= C_D^1 x_D \\ z_2 &= C_2^z x_D \end{aligned} \tag{17}$$

which is a system with two input variables f_1 and f_2 and two output variables z_1 and z_2. Now, given a system Σ_1 of the form of (2) with variables (f_1, z_1, u, y), the objective is to construct another system Σ_2, also of the form of (2) with variables (f_2, z_2, u, y), such that $\Sigma_1\|_c^\Pi \Sigma_2$ is bisimilar to D with respect to the variables (f_1, z_1, f_2, z_2). We need the following notation to state the theorem precisely: $\preceq_{f_1,z_1}$ indicates simulation with respect to the manifest variables (f_1, z_1), $\approx_{f_1,z_1,f_2,z_2}$ indicates bisimulation with respect to all the input–output variables of system D, and D^{f_2,z_2} is the system obtained by setting $f_2 = 0$, $z_2 = 0$ in the equations of D.

Theorem 15: Let Σ_1 be as in (2) and D be system of the form of (17). Then, the following statements are equivalent:

1) $\Sigma_1^c \preceq_{f_1,z_1} D^{f_2,z_2}$ and $D \preceq_{f_1,z_1} \Sigma_1$.
2) There exists a permutation matrix Π and a system Σ_2 of the form of (2) such that $\Sigma_1\|_c^\Pi \Sigma_2 \approx_{f_1,z_1,f_2,z_2} D$.

An application of the above theorem is, for example, the situation where Σ_1 is the plant and Σ_2 is an inner-loop controller system, while D is the controlled system that can be further controlled by interconnecting it with an outer-loop controller that acts on the variables (f_2, z_2); see Fig. 4.

This result too is proved by constructing the canonical controller, which in this case is the system $\Sigma_1\|_{(f_1,z_1)}^\Pi D$. Note that when the signals (f_2, z_2) are absent in D, then $D = D^{f_2,z_2}$, and we recover Theorem 7 from Theorem 15.

Another natural extension of Theorem 7 concerns systems with a direct feed-through term. Note that the proof of Theorem 7 does not make use of the fact that there is no

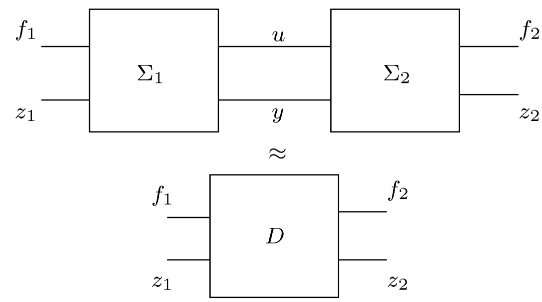


Fig. 4. Generalized configuration with additional (f_2, z_2).

feed-through term in the equations, and thus the theorem holds true for systems with direct feed-through also. However, for the theorem to be useful, it needs to be shown that the simulation conditions $P^c \preceq S$ and $S \preceq P$ can be checked by some algorithm.

We now explain how $S \preceq P$ can be checked. Let P and S be described by the following equations:

$$P: \quad \begin{aligned} \dot{x}_P &= A_P x_p + B_P^u u_P + B_P^f f_P \\ \begin{bmatrix} z_P \\ y_P \end{bmatrix} &= \begin{bmatrix} C_P^z \\ C_P^y \end{bmatrix} x_P + \begin{bmatrix} D_P^z f_P + E_P^z u_P \\ D_P^y f_P + E_P^y u_P \end{bmatrix} \end{aligned}$$

$$S: \quad \begin{aligned} \dot{x}_S &= A_S x_S + B_S^f f_S \\ z_S &= C_S^z x_S + D_S^z f_S. \end{aligned}$$

To compute $\mathrm{R_{SP}}$, we consider the following equations:

$$\begin{aligned} \begin{bmatrix} \dot{x}_S \\ \dot{x}_P \end{bmatrix} &= \begin{bmatrix} A_S & 0 \\ 0 & A_P \end{bmatrix} \begin{bmatrix} x_S \\ x_P \end{bmatrix} + \begin{bmatrix} B_S^f \\ B_P^f \end{bmatrix} f + \begin{bmatrix} 0 \\ B_P^u \end{bmatrix} u \\ 0 &= [C_S^z \quad -C_P^z] \begin{bmatrix} x_S \\ x_P \end{bmatrix} + (D_S^z - D_P^z) f - E_P^z u \\ y &= [0 \quad C_P^y] \begin{bmatrix} x_S \\ x_P \end{bmatrix} \end{aligned}$$

where $f = f_P = f_S$. Similarly, z_P and z_S have been set to be equal. The plant input is $u_P = u$, and the output is $y_P = y$. To compute $\mathrm{R_{SP}}$, we use the following construction (see [16, Ex. 4.6, p. 99]): Define $\dot{\phi} = z_S - z_P$ where ϕ is a vector of new state variables. Also define a new output $Y := \phi$. Consider the following equations:

$$\begin{aligned} \begin{bmatrix} \dot{x}_S \\ \dot{x}_P \\ \dot{\phi} \end{bmatrix} &= \begin{bmatrix} A_S & 0 & 0 \\ 0 & A_P & 0 \\ C_S^z & -C_P^z & 0 \end{bmatrix} \begin{bmatrix} x_S \\ x_P \\ \phi \end{bmatrix} \\ &\quad + \begin{bmatrix} B_S^f \\ B_P^f \\ D_S^z - D_P^z \end{bmatrix} f + \begin{bmatrix} 0 \\ B_P^u \\ -E_P^z \end{bmatrix} u \\ Y &= [0 \quad 0 \quad I] \begin{bmatrix} x_S \\ x_P \\ \phi \end{bmatrix}. \end{aligned}$$

Set $f = 0$. Let $\mathcal{V}$ be the largest controlled invariant subspace contained in $\ker[0\ 0\ I]$. $\mathcal{V}$ is the maximal simulation relation of S by P if $\mathrm{im}\begin{bmatrix} B_S^f \\ B_P^f \\ D_S^z - D_P^z \end{bmatrix} \subseteq \mathcal{V}$; otherwise a simulation relation of S by P does not exist. The existence of the simulation relation $P^c \preceq S$ can be checked along the same lines; it requires a little more work since P^c has to be computed first.

Remark 16: We comment briefly on the relation of the conditions $P^c \preceq S \preceq P$ to the necessary and sufficient conditions obtained in the control by interconnection of behavioral systems. In the behavioral setting, the problem is formulated as follows. Given a behavior $\mathcal{P}$ and a desired behavior $\mathcal{S}$, we say that $\mathcal{S}$ is *implementable* if there exists another behavior $\mathcal{C}$ such that $\mathcal{P} \cap_c \mathcal{C} = \mathcal{S}$, where the c indicates that the interconnection is through the control variables. It has been shown that $\mathcal{N} \subseteq \mathcal{S} \subseteq \mathcal{P}_m$ is equivalent to implementability, where $\mathcal{P}_m$ is the plant behavior projected onto the manifest variables and $\mathcal{N}$ (analogous to P^c) is the behavior obtained by setting the control variables of the plant to zero; see [15]. Observe that the condition is very similar to the one we have derived. However, the two conditions are not equivalent since behavioral equality does not in general imply bisimilarity (see [12, Example 2.15]).

IV. Conclusion

We have derived necessary and sufficient conditions for the existence of a controller C such that $P||_c^\Pi C \approx S$ for some permutation matrix Π, namely $P^c \preceq S \preceq P$. Moreover, the conditions $P^c \preceq S \preceq P$ can be verified by computing the maximal simulation relation of P^c by S and, respectively, of S by P, and checking that they are full.

The notion of (bi)simulation is readily extendable to nonlinear systems. Some steps in this direction have been taken; see for example [2], [10], and [12]. The problem of achievability for general systems has been addressed in [9]. As mentioned earlier, in [9] conditions similar to the ones we have derived have been presented. However, in [9] there is no partitioning of the external variables into manifest and control variables; indeed, all manifest variables are also available for control. Hence, the system P^c is the system obtained by setting *all* the variables to zero. Consequently, the only condition that remains is $S \preceq P$. This is exactly the condition obtained in [9] when restricted to linear time-invariant systems.

From the proof of Theorem 7, we see that the canonical controller plays a central role. However, contrary to the linear case, it turns out that, for nonlinear systems, the canonical controller does not always achieve the desired dynamics. There are achievable desired dynamics that require one to *restrict* the canonical controller to an invariant subset of its state space; this has also been observed in [8, Theorem 4]. The extent to which we can extract achievability-related information from the canonical controller for general systems is a topic of current research. Some preliminary results have been obtained in [13].

Another important question that arises is the following: When does there exist a standard feedback controller (so $\Pi = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$) that achieves S? Furthermore, can we parametrize these controllers using the canonical controller (see Remark 11)? These issues are currently under investigation.

References

[1] G. Basile and G. Marro, *Controlled and Conditioned Invariants in Linear System Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[2] K. A. Grasse, "Simulation and bisimulation on nonlinear control systems with admissible classes of inputs and disturbances," *SIAM J. Control Optim.*, vol. 46, no. 2, pp. 562–584, 2007.

[3] A. A. Julius, J. W. Polderman, and A. J. van der Schaft, "Parametrization of the regular equivalences of the canonical controller," *IEEE Trans. Autom. Control*, vol. 53, no. 4, pp. 1032–1036, May 2008.

[4] M. Kuijper, "Why do stabilizing controllers stabilize?," *Automatica*, vol. 31, no. 4, pp. 621–625, 1995.

[5] R. Milner, *Communication and Concurrency*, ser. Computer Science. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[6] G. J. Pappas, "Bisimilar Linear Systems," *Automatica*, vol. 39, no. 12, pp. 2035–2047, 2003.

[7] D. Park, *Concurrency and Automata on Infinite Sequences*. New York: Springer, 1981, vol. 104, Lecture Notes in Computer Science.

[8] G. Pola, A. J. van der Schaft, and M. D. Di Benedetto, "Achievable bisimilar behavior of abstract state systems," in *Proc. 44th IEEE Conf. Decision Control & Eur. Control Conf.*, Seville, Spain, Dec. 2005, pp. 1535–1540.

[9] P. Tabuada, "Controller synthesis for bisimulation equivalence," *Syst. Control Lett.*, vol. 57, pp. 443–452, 2008.

[10] P. Tabuada and G. J. Pappas, "Bismilar control affine systems," *Syst. Control Lett.*, vol. 52, pp. 49–58, 2004.

[11] A. J. van der Schaft, "Achievable behavior of general systems," *Syst. Control Lett.*, vol. 49, pp. 141–149, 2003.

[12] A. J. van der Schaft, "Equivalence of dynamical systems by bisimulation," *IEEE Trans. Autom. Control*, vol. 49, no. 12, pp. 2160–2172, Dec. 2004.

[13] H. Vinjamoor and A. J. van der Schaft, "Achievable dynamics for a class of non-linear systems," presented at the NOLCOS 2010, Bologna, Italy, 2010.

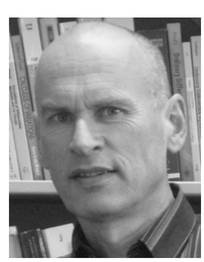
[14] J. C. Willems, "On interconnections, control and feedback," *IEEE Trans. Autom. Control*, vol. 42, no. 3, pp. 326–339, Mar. 1997.

[15] J. C. Willems and H. L. Trentelman, "Synthesis of dissipative systems using quadratic differential forms: Part I," *IEEE Trans. Autom. Control*, vol. 47, no. 1, pp. 53–69, Jan. 2002.

[16] W. M. Wonham, *Linear Multivariable Control: A Geometric Approach*, 3rd ed. New York: Springer, 1985.

Harsh Vinjamoor studied instrumentation engineering at the Government College of Engineering, Pune, India, received the Master's degree in systems and control from the Indian Institute of Technology, Bombay, India, and is currently pursuing the Ph.D. degree at the University of Groningen, Groningen, The Netherlands, with a focus on control-by-interconnection of state-space systems.

During his Master's degree education, he studied questions related to the interconnection of systems in the behavioral framework.



Arjan J. van der Schaft received the Ph.D. degree in mathematics from the University of Groningen, Groningen, The Netherlands, in 1983.

In 1982, he joined the Department of Applied Mathematics, University of Twente, Enschede, The Netherlands, where he was appointed as a Full Professor in mathematical systems and control theory in 2000. In 2005, he returned to the University of Groningen as a Full Professor in mathematics.

Prof. van der Schaft has served as Associate Editor for *Systems & Control Letters*, the *Journal of Nonlinear Science*, *SIAM Journal on Control*, and the IEEE Transactions on Automatic Control. Currently, he is Associate Editor for *Systems and Control Letters* and the *Journal of Geometric Mechanics*, and Editor-at-Large for the *European Journal of Control*. He was an Invited Lecturer at the International Congress of Mathematicians, Madrid, Spain, in 2006, and a co-recipient of the SICE Takeda Best Paper Prize 2008.