What can the canonical controller in principle tell us?

Harsh Vinjamoor*, Arjan van der Schaft

Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, P.O. Box 407, 9700AK Groningen, Netherlands

Abstract

Given a plant and a desired specification our goal is to construct a controller system which, when interconnected with the plant, yields a system that behaves like the desired specification. We can always construct the canonical controller introduced in van der Schaft (2003) [10]. For linear systems there exists a controller which when interconnected to the plant yields the desired behaviour if and only if the canonical controller is itself one such controller, see Vinjamoor and van der Schaft (2011) [4]. In this paper we extend this result to nonlinear systems. It turns out that one has to look at the canonical controller together with its subsystems. We obtain necessary and sufficient conditions for the existence of a controller for a class of nonlinear systems. We end with examples which show that in certain cases looking at subsystems of the canonical controller also does not suffice.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Throughout this paper we will have to deal with three state space systems, namely, the plant $P$, the desired system $S$ and the controller system $C$. The goal is to find necessary and sufficient conditions under which there exists a controller $C$ such that when interconnected to $P$, the resulting systems behaves exactly like $S$: we will say that $C$ achieves $S$. We assume that only the variables $(u_0, y_0)$ of the plant are available for interconnection with the controller, see Fig. 1. We shall call the variable $z_0$ manifest (denoted by $m$) as it is the variable whose behaviour we are interested in. For the variables $(u_0, y_0)$ we shall use the term control (denoted by $c$) variables since they are available for control.

The class of controller interconnections that we consider is more general than the ones usually seen in controller design techniques. Classical control theory deals with feedback controllers (see Fig. 1), i.e., controllers which accept the output of the plant as their input and produce an output which acts as an input to the plant. Thus a controller is looked at as a signal processing unit. These controllers have certain advantages. For instance, in the case of linear time-invariant state space systems without feed-through terms, a feedback interconnection is guaranteed to be well-posed, in the sense that after interconnecting the controller it is not necessary to restrict the set of initial conditions of the plant to a proper subspace of its state space.

However, there are desired systems $S$ which can be achieved, but not by this class of interconnections precisely because the state space of the plant does not get restricted after interconnecting a controller. These considerations are not new and have already been addressed; see for instance the example of the ‘door closing mechanism’ in [1–3]. In general these other types of interconnections occur frequently in physical system interconnections. In this paper we shall allow for ‘non-feedback’ interconnections in which outputs of two systems get equated, thus resulting in state constraints on the interconnected system. For a detailed discussion with examples see [4, Section I.A].

This paper is a generalization to nonlinear systems of the results found in [4]. Note that a similar problem was addressed in [5]. The difference is that we partition the variables into manifest and control variables while in [5] all the variables are available for control purposes. In the next section we state precisely the class of systems we consider, followed by definitions of system equivalence that we use in this paper. The main result of the paper (Theorem 4) is then stated and proved. This is followed by a discussion of the main theorem and illustrative examples. We then present some examples of desired specifications which can be achieved but for which the canonical controller yields no information about their achievability. We conclude with some remarks and future directions in Section 4. Preliminary results of this paper were presented in [6].

2. Definitions and the main result

Consider the following plant

$$\dot{x}_P = f_P(x_P) + g_P(x_P)u_P$$

$$y_P = h_P(x_P)$$

$$z_P = c_P(x_P).$$

(1)
Let the desired system be the following
\[ X_S = f_S(x_S) \]
\[ Z_S = c_S(z_S) \] (2)

Under standard technical assumptions we have that given the initial condition \( x\theta(0) \) and an input function \( u_p \), the state trajectory \( x\theta \) is uniquely determined. Consequently the outputs \( y_p \) and \( z_p \) are also determined. Similarly, given \( x\theta(0) \) the state trajectory \( x\theta \) is uniquely determined and hence so is the output \( z_S \). The main result of this paper applies to classes of systems with exactly these properties and is hence not limited to smooth differential systems (1) and (2). We now state precisely the class of systems we consider.

Let \( X_P, U, Y \) and \( Z \) be sets and \( X_p, U, Y \) and \( Z \) be functions from \( R \) to \( X_P, U, Y \) and \( Z \) respectively. Let \( P \subseteq X_P \times U \times Y \times Z \). Assume that the set \( P \) has the following differential-equation-like property: for all \( x\theta(0) \in X_P \) and \( u_p \in U \), there exist unique \( x_p \in X_p, y_p \in Y \) and \( z_p \in Z \) such that \( (x_p, u_p, y_p, z_p) \in P \). We shall call this set \( P \) the plant and the set \( X_p \) its state space. The variables \( y_p, z_p \) are called the outputs of \( P \) while \( u_p \) is called an input. Let \( X_C \) be the set of functions from \( R \) to a set \( X_c \) and suppose \( S \subseteq X_C \times Z \) has the following property: for all \( x\theta(0) \in X_C \) there exists a unique \( (x_c, z_c) \in S \) where \( x_c \) will be called the state space of \( S \). We shall refer to this set \( S \) as the desired system. The variables \( z_c \) are called outputs of \( S \). Consider \( X_C \times U \times Y \) where again \( X_C \) is the set of functions from \( R \) to a set \( X_c \). We shall call a set \( C \subseteq X_C \times U \times Y \) a controller with state space \( X_c \). Further we assume that all systems are time-invariant. For nonlinear systems the domains of definition of the functions involved might have to be restricted to subintervals of \( R \) containing zero. Although the main theorem holds true with such domain restrictions, this leads to cumbersome notation and is also not central to the main result of the paper, hence we ignore such phenomena.

Let the real numbers be denoted by \( R \). Given a finite collection of sets of functions of time \( A_1, \ldots, A_m \) and integers \( 1 \leq i_1 < \cdots < i_k \leq m \) we define a function \( \kappa_{i_1, i_2, \ldots, i_k} : A_1 \times \cdots \times A_m \longrightarrow A_{i_1} \times A_{i_2} \times \cdots \times A_{i_k} \) which projects onto the indicated factors and evaluates the corresponding functions at \( t = 0 \).

We now discuss the notion of equivalence that we use in this paper. Given a controller \( C \), when do we say that a \( P \)-interconnected-to-\( C \) behaves like \( S \)? One intuitive idea is that for every initial condition in \( X_S \) there should exist an initial condition in the state space of \( P \)-interconnected-to-\( C \) such that the outputs \( z_p \) and \( z_S \) of \( P \) and \( S \) are identical. The definition of bisimulation as introduced in [7] (inspired by [8]) and followed up in [9]) makes this idea precise. The following is a generalized definition.

**Definition 1.** Consider \( \Sigma_1 \subseteq X_{\Sigma_1} \times V_1 \times Z \) where \( Z \) are the outputs, \( X_{\Sigma_1} \) is the set of functions from \( R \) to a set \( X_c \) and \( V_1 \) are sets of functions from \( R \) to the set \( V_1 \); \( i = 1, 2 \). We shall say that \( R \subseteq X_{\Sigma_1} \times X_{\Sigma_2} \) is a bisimulation relation between \( \Sigma_1 \) and \( \Sigma_2 \) if \( R \) has the following property: take any \( (x\theta(0), x\zeta(0)) \in R \) and then for all \( v_1 \) such that \( (x\theta_1 (0), v_1, z_1) \in \Sigma_1 \) there exists \( v_2 \) such that \( (x\theta_2 (v_2, v_1, z_1) \in \Sigma_2 \) and \( (x\zeta(t), y\zeta(t)) \in R \) for all \( t \geq 0 \), and conversely, for all \( v_2 \) such that \( (x\theta_2 (v_2, v_1, z_1) \in \Sigma_1 \) there exists \( v_1 \) such that \( (x\theta_1 (0), v_1, z_2) \in \Sigma_2 \) and \( (x\zeta(t), y\zeta(t)) \in R \) for all \( t \geq 0 \).

A bisimulation relation is said to be full if \( \kappa_1 (R) = X_{\Sigma_1} \) and \( \kappa_2 (R) = X_{\Sigma_2} \). Two systems \( \Sigma_1 \) and \( \Sigma_2 \) are called bisimilar, denoted \( \Sigma_1 \approx \Sigma_2 \), if there exists a full bisimulation relation \( R \) between \( \Sigma_1 \) and \( \Sigma_2 \).

A one-sided version of bisimulation is the following.

**Definition 2.** Consider \( \Sigma_1 \subseteq X_{\Sigma_1} \times V_1 \times Z \) where \( X_{\Sigma_1} \) is the set of functions from \( R \) to a set \( X_c \) and \( V_1 \) are sets of functions from \( R \) to the sets \( V_1 ; i = 1, 2 \). We shall say that \( R \subseteq X_{\Sigma_1} \times X_{\Sigma_2} \) is a simulation relation of \( \Sigma_1 \) by \( \Sigma_2 \) if \( R \) has the following property: take any \( (x\theta_1 (0), x\zeta(0)) \in R \). Then for all \( v_1 \) such that \( (x\theta_1 (0), v_1, z_1) \in \Sigma_1 \) there exists \( v_2 \) such that \( (x\theta_2 (v_2, v_1, z_1) \in \Sigma_2 \) and \( (x\zeta(t), y\zeta(t)) \in R \) for all \( t \geq 0 \). A simulation relation is said to be full if \( \kappa_1 (R) = X_{\Sigma_1} \). We shall say that \( \Sigma_1 \) is simulated by \( \Sigma_2 \), denoted \( \Sigma_1 \preceq \Sigma_2 \), if there exists a full simulation relation \( R \) of \( \Sigma_1 \) by \( \Sigma_2 \).

In the above definitions we allow the set \( V_i \) to be a singleton but not empty. For instance, for the simulation relation of \( S := \Sigma_1 \) by \( \Sigma_2 \), \( V_1 \) is a singleton while \( V_2 = U \). Whenever a set \( V_i \) is a singleton we suppress it in the notation of the system. Precise details of what \( V_i \) depends on the pair of systems between which simulations are being considered. In the text after Theorem 4 these details have been stated. For smooth differential systems the above definitions coincide with those in [7].

We now define four systems which will be needed to state and prove the main result. Let \( \Pi \) be a permutation matrix and \( C \subseteq X_C \times U \times Y \), a controller system.

\[ C \subset P \quad \text{if} \quad C \ni \left\{ (x_C, u_C, y_C, x_P, y_P, z_P) \right\} \in C \times P \quad \text{such that} \quad \Pi \left( \begin{array}{c} u_C \\ y_C \end{array} \right) = \left( \begin{array}{c} u_P \\ y_P \end{array} \right) \]

The subscript \( c \) in \( C \subset P \) indicates that the interconnection constraints are via the control variables \( (u_C, y_C, u_P, y_P, C) \). For \( \Pi = \left( \begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array} \right) \) we recover our usual feedback interconnection, see Fig. 2. For \( \Pi = I \) we obtain an interconnection that imposes constraints induced by the equation \( y_C = y_P \). This usually results in a state space smaller than \( X_P \times X_c \). The state space of \( C \subset P \), denoted by \( \Sigma_{C \subset P} \), is the projection of \( C \subset P \) onto the state spaces of \( C \) and \( P \).

\[ \Sigma_{C \subset P} \] respectively, i.e., \( \Sigma_{C \subset P} = \kappa_{1,4}(C \subset P) \). Recall that \( \kappa_{1,4} \) is the map which projects onto the first and the fourth component and then evaluates the resulting two factors at \( t = 0 \) (see Figs. 3 and 4).

We extend the definition of the canonical controller as introduced for linear systems in [10] to the generalized systems considered in this paper as follows.

\[ S \subset P : \right\{ (x_S, z_S, x_P, y_P, z_P) \in S \times P \mid z_S = z_P \} \]

We will continue to call this system the canonical controller. As earlier, in the notation \( S \subset P \), the subscript \( m \) indicates that the
The statespace of this system is the largest controlled invariant manifold of the plant. For the special case when P and S are as in Eqs. (1) and (2) respectively, the explicit equations defining the canonical controller \( S \parallel P \) are

\[
\begin{align*}
\dot{x}_S &= f_S(x_S) \\
\dot{z}_P &= f_P(x_P) + g_P(x_P) u_P \\
l &= z_P - z_S = c_P(x_P) - c_S(x_S) = 0 \\
y_P &= h_P(x_P).
\end{align*}
\]

The statespace of this system is the largest controlled invariant output nulling submanifold with \( I \) as the output (see [11,12] for a treatment of smooth systems). Also, computing the statespace of the above interconnection is in this case equivalent to finding the largest simulation relation as explained for smooth systems in [7, Section 7]. Thus the statespaces of interconnected systems are in general analogues of controlled invariant subspaces for linear systems and controlled invariant submanifolds for smooth systems.

Given a full simulation relation \( R \) of \( S \) by \( P \) we define a restriction \( C_R \) of the canonical controller by

\[
C_R := \{ (x_S, z_S, x_P, u_P, y_P, z_P) \in S \times P | z_S = z_P, (x_S(t), x_P(t)) \in R, \forall t \geq 0 \}.
\]

Note that since \( S \) has no input variables, \( R \) is analogous to a controlled invariant submanifold (see [11,12]) and hence we can restrict \( S \parallel P \) to it. In the special case of Eqs. (1) and (2), restricting to a full simulation relation \( R \) is thus restricting the canonical controller to a controlled invariant submanifold, possibly smaller than the largest controlled invariant output nulling submanifold. This is the system in the dotted box in Fig. 5.

When we interconnect \( C_R \) to the plant we obtain a system that is given as follows, see also Fig. 5.

\[
C_R \parallel P := \{ (x_S, z_S, x_P, u_P, y_P, z_P, x_P', u_P', y_P') \in S \times P \times \mathbb{Z} | (x_S(t), x_P(t)) \in R, \forall t \geq 0, \\
\quad z_S = z_P, u_P = u_P', y_P = y_P', z_P' \}.
\]

The state space of this system is the projection of \( C_R \parallel P \) onto the state spaces of the \( C_R \) and \( P \), i.e., \( \kappa_{1,3,7}(C_R \parallel P) \) and is denoted by \( \mathbb{X}_{C_R \parallel P} \). Note that \( \kappa_{1,3,7} \) is the projection onto the first, third and the seventh component followed by evaluation of the resulting three factors at \( t = 0 \).

**Problem statement:** Given \( P \) and \( S \), find necessary and sufficient conditions for the existence of a controller \( C \) and an interconnection matrix \( R \) such that \( P \parallel C \approx S \). If such a \( C \) exists then \( S \) is said to be achievable and \( C \) is said to achieve \( S \).

For linear time-invariant systems, there exists a controller if and only if the canonical controller is such a controller (see [4]). In Section 3.2 we illustrate that for nonlinear systems one has to look at subsystems of the canonical controller. In this case the controller is obtained as a restriction of the canonical controller, i.e., \( C_R \) for some full simulation relation \( R \). That one has to look at subsystems of the canonical controller has also been observed in [13]. However, it turns out that

\[
(C_R | R \text{ a full simulation relation of } S \text{ by } P)
\]

does not always contain a controller even if \( S \) is achievable, see Section 3.3. Using the following assumption on \( P \) and \( S \) we obtain a class of systems for which \( S \) is achievable if and only if

\[
(C_R | R \text{ a full simulation relation of } S \text{ by } P)
\]

does contain an achieving controller.

**Assumption 3.** For any full simulation relation \( R \) of \( S \) by \( P \), given

\[
(x_S, z, x_P, u, y, z, x_P', u, y, z') \in C_R \parallel P
\]

and

\[
(x_S, z, x_P, u', y', z) \in C_R.
\]

there exists \( z'' \in \mathbb{Z} \) such that

\[
(x_S, z, x_P, u', y', z, x_P', u', y', z'') \in C_R \parallel P
\]

and for all such \( z'', z' = z'' \).

In the next section we list some important classes of systems \( P, S \) where **Assumption 3** holds true.
Main result

**Theorem 4.** Given $S$ and $P$ satisfying Assumption 3 the following are equivalent:

1. There exists a controller $C$ and an interconnection matrix $\Pi$ such that $P \overset{\Pi}{\lceil} C \approx S$.
2. There exists a full simulation relation $R$ of $S$ by $P$ such that $C_R \overset{\Pi}{\lceil} P \approx S$.

In the above theorem, to check the condition $P \overset{\Pi}{\lceil} C \approx S$ we apply Definition 1 with $\Sigma_1 := P \overset{\Pi}{\lceil} C$, $\Sigma_2 := S$, $\mathcal{X}_1 = \mathcal{X} \overset{\Pi}{\lceil} C$, $\mathcal{X}_2 = \mathcal{X}_S$, $\mathcal{V}_1 = \mathcal{U} \times \mathcal{Y}$ and $\mathcal{V}_2$ is a singleton. Similarly, to check the condition $C_R \overset{\Pi}{\lceil} P \approx S$ we apply Definition 2 with $\Sigma_1 := C_R \overset{\Pi}{\lceil} P$, $\Sigma_2 := S$, $\mathcal{X}_1 := \mathcal{X}$, $\mathcal{X}_2 := \mathcal{X}_S$, $\mathcal{V}_1 := \mathcal{Z} \times \mathcal{U} \times \mathcal{Y}$ and $\mathcal{V}_2$ is again a singleton. When considering simulation relations $R$ of $S$ by $P$ with $\Sigma_1 := S$, $\Sigma_2 = P$ we have that $\mathcal{V}_1$ is a singleton and $\mathcal{V}_2 = \mathcal{U} \times \mathcal{Y}$. As mentioned earlier, whenever $\mathcal{V}_1$ is a singleton we suppress it in the notation of the system trajectories. We now prove Theorem 4.

**Proof of Theorem 4.** (1 $\Rightarrow$ 2): Let $B \subseteq X_S \times X_P \times \mathcal{X}_C$ be a full bisimulation relation between $S$ and $P \overset{\Pi}{\lceil} C$. Consequently, for every $x \in X_S$ there exists a state $y \in \mathcal{X}_P$ and some state in $X_C$ such that $z_S = y_P$. Thus $R := \kappa_{1,2}(B)$ is a full simulation relation of $S$ by $P$.

Suppose $(x_S, z, x_P, u, y, z', x_P', u, y') \in C_R \overset{\Pi}{\lceil} P$. Since $(x_S(0), x_P(0)) \in R$, there exists $x_C(0) \in \mathcal{X}_C$ and $(u', y') \in \mathcal{U} \times \mathcal{Y}$ such that

$$(x_S, z, x_P, u', y', z, x_C, u', y') \in S \overset{\Pi}{\lceil} m \overset{\Pi}{\lceil} C.$$

Thus we have that

1. $(x_S, z, x_P, u, y, z', x_P', u, y') \in C_R \overset{\Pi}{\lceil} P$
2. $(x_S, z, x_P', u', y') \in C_R \overset{\Pi}{\lceil} P$.

By Assumption 3 there exists $z'' \in Z$ such that

$(x_S, z, x_P', u', y', z, x_C, u', y', z'') \in C_R \overset{\Pi}{\lceil} P$

and for all such $z''$, $z' = z''$.

Thus $(x_S', u', y', z') \in P$.

Hence $(x_C, u', y', x_P', u', y', z') \in C \overset{\Pi}{\lceil} P$. Thus we have shown that $C_R \overset{\Pi}{\lceil} P \approx S \overset{\Pi}{\lceil} C$. As $S \approx C \overset{\Pi}{\lceil} P$ we are done.

(2 $\Rightarrow$ 1): Let $B \subseteq X_S \times X_P \times X_P \times X_C$ be a full simulation relation of $C_R \overset{\Pi}{\lceil} P$ by $S$. Since $R$ is a full simulation relation, for all $x_S(0) \in X_S$, there exists $x_P(0) \in X_P$ such that for some $(u, y, z) \in \mathcal{U} \times \mathcal{Y} \times Z$, $(x_S, z, x_P, u, y, z) \in C_R$. Also, $(x_S, z, x_P', u, y, z, x_P', u, y, z, x_S', z) \in C_R \overset{\Pi}{\lceil} P$. Hence $\kappa_2(B) = X_S$ and $B$ is a bisimulation relation. Thus choosing $\Pi = I$ and $C_R$ as the controller we have that $C_R \overset{\Pi}{\lceil} P \approx S$ with $B$ as the bisimulation relation.

Note that in Theorem 4 the second condition is equivalent to $C_R \overset{\Pi}{\lceil} P \approx S$. This follows from the latter half of the above proof.

3. Discussion

In this section we discuss some aspects of Theorem 4 along with a few illustrative examples.

3.1. Systems satisfying Assumption 3

**Proposition 5.** Suppose $S$ and $P$ are linear subspaces of $X_S \times X_P \times X_P \times \mathcal{U} \times \mathcal{Y} \times \mathcal{Z}$ respectively. Then $S$ and $P$ satisfy Assumption 3 for all full simulation relations of $S$ by $P$.

**Proof.** Let $(x_S, z, x_P, u, y, z', x_P', u, y') \in C_R \overset{\Pi}{\lceil} P$ and $(x_S, z, x_P', u', y', z') \in C_R \overset{\Pi}{\lceil} P$, where $R$ is a full simulation relation of $S$ by $P$. Since we have linear subspaces, subtracting trajectories we obtain $(0, 0, 0, u' - u, y' - y, 0) \in C_R$. Thus $(0, u' - u, y' - y, 0) \in P$. Hence we obtain that $(x_S', u', y', z') \in P$. Thus there exists $z'' := z'$ such that $(x_S, z, x_P, u', y', z, x_P', u', y', z'') \in C_R \overset{\Pi}{\lceil} P$. We now show that $z''$ must be equal to $z'$. Suppose there exists some $z''$, not necessarily equal to $z'$ such that $(x_S, z, x_P, u', y', z, x_P', u', y', z'') \in C_R \overset{\Pi}{\lceil} P$. Since these are linear systems we can subtract the two trajectories to get $(0, 0, 0, u' - u, y' - y, 0, 0, u' - u, y' - y, z' - z'' \in C_R \overset{\Pi}{\lceil} P$. Consider the two trajectories $(0, u' - u, y' - y, 0, 0, u' - u, y' - y, z' - z'') \in C_R \overset{\Pi}{\lceil} P$. Since the initial conditions are the same, i.e., zero and the inputs are the same, i.e., $u' - u$, by the uniqueness of solutions, we must have that the outputs are the same, i.e., $z' = z''$. $\square$

**Remark 6.** For the linear time-invariant case, $C_R \overset{\Pi}{\lceil} P \approx S$ is equivalent to $S \approx P$ and $N \approx S$ where $N$ is the system obtained by setting the variables $u_P$ and $y_P$ to zero in the plant. Hence we recover the result obtained in [4, Theorem 7]: there exists a controller which achieves $S$ if and only if $N \approx S$. $\square$

Let $(x_S, z, x_P, u, y, z, x_P', y, z') \in C_R \overset{\Pi}{\lceil} P$ and $(x_S, z, x_P, u, y, z, x_P, u, y, z') \in C_R \overset{\Pi}{\lceil} P$. Subtracting the two we get $(0, 0, 0, 0, 0, 0, 0, 0, z - z') \in C_R \overset{\Pi}{\lceil} P$ where $(x_P - x_P', 0, 0, z - z') \in N$.

Since $C_R \overset{\Pi}{\lceil} P \approx S$ we have that $N \approx S$ and since $R$ is a full simulation relation $S \approx P$. Thus $N \approx S \approx P$.

Another class of problems which satisfy Assumption 3 is described as follows. Consider $S \overset{\Pi}{\lceil} P$. If it satisfies the property that given $(x_S(0), x_P(0)) \in X_i\overset{\Pi}{\lceil} m \overset{\Pi}{\lceil} \mathcal{U}$ the input $u$ is uniquely determined, then Assumption 3 is automatically satisfied. For example, consider a plant as described in Eq. (1). Suppose the desired system $S$ is just the zero system, i.e., after interconnecting a controller the output $z_P$, is required to be identically zero. Further assume that the plant has a well-defined relative degree $r \leq n_P$ with respect to the input $u_P$ and output $z_P$; we assume that $r$ is the same at every point of the state space and that $L_{z_P}^{-1} L_{u_P}^{r+1} C_P(x_P) \neq 0$ for all $x_P \in \{ \{x\mid L_{z_P}^{r+1} C_P(x) = 0; 0 \leq k < r\} \}$.

Then we know that starting from an initial condition on $V$ we can keep the output zero by choosing $u_P := -\frac{L_{z_P}^{-1} C_P(x_P(t))}{L_{u_P}^{r+1} C_P(x_P(t))}$, see [12, Section 4.3, page 169] or [11]. Thus, given the state $x_P$, the input $u_P$ is uniquely defined. Consequently the problem of zeroing the output (or equivalently that of keeping the output constant) for
a SISO control affine nonlinear system satisfies Assumption 3. We summarise this in the following lemma.

**Proposition 7.** Consider a plant of the form (1). Suppose $S$ is the zero system. Assume the plant has a well-defined relative degree with respect to the input $u_p$ and output $z_p$. Then Assumption 3 is satisfied for all simulation relations of $S$ by $P$.

### 3.2. Illustration of the main result

We now state an example where the canonical controller does not achieve the desired system, but when restricted to an invariant subset it does indeed achieve the desired system. Consider a plant given by the equations

$$
\begin{align*}
x_p &= x_p(1 + u_p) \\
y_p &= x_p^2 \\
z_p &= (x_p^2 - 4)(x_p - 1).
\end{align*}
$$

Suppose the desired system is the zero system $S = \{0, 0\} \in X_0 \times Z$, i.e., we require $z_p$ to be identically zero in the controlled system. It is clear that one must have an initial condition in the set $[2, -2]$. By choosing $u_p = -1$ these points become equilibria. Further note that this is the only choice of $u_p$ which ensures that $z_p$ is identically zero. Thus Assumption 3 is satisfied in this case; see the previous subsection. Consider $R := \{(0, x) \mid x \in [2, -2] \}$ which is a full simulation relation (where $0$ is assumed to be the state space of $S$). Consider the system $C_R \parallel P$. On computing the system trajectories\(^1\) one finds that $(x_5 = 0, z_5 = 0, x_{p1} = 1, u_{p1} = -1, y_{p1} = 1, z_{p1} = 1, x_{p2} = 0, u_{p2} = -1, y_{p2} = 1, z_{p2} = 0) \in C_R \parallel P$. Thus $z_{p2} \neq 0$. Hence $C_R \parallel P$ is not bisimilar to $S$. However consider $R' := \{(0, x) \mid x \in [2, -2] \}$. Then on carrying out the computation one finds that $C_R \parallel P \approx S$.

We summarise this section with a method that could in principle be used to construct controllers: given a plant $P$ and a system $S$ which satisfy Assumption 3, first construct $C_R \parallel P$ and check if it is a controller which achieves $S$. If yes, then we have a controller. If not, compute the full simulation relations (not necessarily maximal) of $S$ by $P$. Restrict the system $C_R \parallel P$ to each of these simulation relations and check if one of them yields a controller. If none of the simulation relations yield a controller, then no controller exists.

### 3.3. Why do we need Assumption 3?

In this section we present two examples which show that if Assumption 3 is not satisfied then there exist systems $S$ which can be achieved but not by any controller system from the set $\{C_R \mid R \text{ a full simulation relation of } S \text{ by } P\}$.

**Example 8.** Consider the following plant.

$$
P := \{(x_p, u_1, u_2, y, z), (x_p, u_1, u_2', y, z'), (x_p, u_1', u_2, y, z), (x_p, u_1', u_2, y, z), (x_p', u_1, u_2', y, z'), (x_p', u_1', u_2', y, z'), (x_p', u_1', u_2', y, z'), (x_p', u_1', u_2', y, z')\}.
$$

Let $S := \{(x_5, z), (x_5', z')\}$.

Note that this example has no time dependence at all. Both the systems are just sets which can be interconnected through the control variables.

Let $C := \{(x_C, u_1, u_2, y)\}$.

Then,

$$
P \parallel C = \{(x_p, u_1, u_2, y, z, x_C, u_1, u_2, y), (x_p', u_1, u_2, y, z', x_C, u_1, u_2, y)\}.
$$

Clearly $P \parallel C \approx S$ with the bisimulation relation given by

$$
\{(x_p, x_C), (x_5, x_C), (x_5', x_C')\}.
$$

Thus $S$ is achievable.

We will now show that no system in the set

$$
\{C_R \mid R \text{ a full simulation relation of } S \text{ by } P\}
$$

works as a controller. First consider the canonical controller without any state space restriction.

$$
S \parallel m P = \{(x_5, z, x_p, u_1, u_2, y, z), (x_5, z, x_p, u_1', u_2', y, z), (x_5', z', x_p, u_1, u_2, y, z'), (x_5', z', x_p, u_1', u_2', y, z')\}
$$

are trajectories in $S \parallel m P \parallel P$. Consequently Assumption 3 is not satisfied. Since $z' \neq z$ we have that $S \parallel m P$ is not a controller. Its state space is

$$
\{(x_5, x_p), (x_5', x_p), (x_5', x_p')\}.
$$

It is easy to see that restricting to either of these states does not yield a controller precisely because $z' \neq z$. Thus there is no full simulation relation $R$ such that $C_R$ is an achieving controller. The above example suggests that Assumption 3 is also almost necessary for the existence of a controller as a subsystem of $S \parallel m P$.

We now present another example which illustrates the difference between behavioural equality (see [10]) and bisimilarity.

**Example 9.** Consider the same plant as above with the third entry replaced by $(x_p, u_1', u_2, y, z')$. Let $S$ be as in the previous example.

Once again the same controller $C$ achieves $S$. As earlier, $S \parallel m P$ is not a controller because

$$
(x_5, z, x_p, u_1', u_2', y, z, x_p', u_1', u_2', y, z) \in S \parallel m P \parallel P \text{ and } z' \neq z.
$$

The state space of $S \parallel m P$ is as earlier. Restricting the state space to $(x_5, x_p)$ or $(x_5', x_p)$ does not yield controllers for the same reason, namely, $z' \neq z$. Now consider $S \parallel m P$ with the state space restricted to $(x_5', x_p')$. As usual denote the interconnection of this restricted
system with $P$ by $C_R \parallel_c P$. Then

$$C_R \parallel_c P = \{ (x'_s, z', x'_p, u_1, u_2, y, z') \mid (x'_s, z', x'_p, u_1, u_2, y, z') \}.$$

Thus the output of this system is either $z$ or $z'$. Hence the set of output trajectories is equal to the set of output trajectories of $S$. However, $C_R \parallel_c P$ is still not bisimilar (see Definition 1) to $S$.

4. Conclusions

Given a plant and a desired system satisfying Assumption 3, we have obtained necessary and sufficient conditions for the achievability of a given desired system. From the proof of Theorem 4 we see that under Assumption 3, if there at all exists a controller such that $P \parallel_C \approx S$, then there exists a full simulation $R$ of $S$ by $P$ (not necessarily the maximal simulation relation) such that $S \parallel_m P$ with state space restricted to $R$ is a controller, i.e., $C_R \parallel_c P \approx S$. Thus, provided Assumption 3 is satisfied, the system $S \parallel_m P$ contains all the information needed to draw conclusions about the existence of a controller, and $S \parallel_m P$ is in this sense still canonical. However, as seen in Section 3.3, there exist systems for which we need different methods to solve the problem of achieving $S$.

Our results are theoretical and show when we can resort to the canonical controller. Constructive methods to find all simulation relations of $S$ by $P$ depend on the class of systems being studied. For most nonlinear systems this is still an open problem. For polynomial systems some steps in this direction have been taken in [14].

Note that for $S$ we consider systems without inputs: if the desired system $S$ has inputs then simulation relations and controlled invariant subsets are not the same objects. Extending the above results to desired systems with inputs is currently being investigated.

References