On achievable bisimulations for linear time-invariant systems.

Harsh Vinjamoor* and Arjan van der Schaft*

Abstract—We consider here the problem of finding a controller for a system such that when interconnected to the plant, we get a system which is bisimilar to the desired system. We give necessary and sufficient conditions for the existence of such a controller. The systems we consider are ordinary linear time invariant dynamical systems described by state space equations. We briefly compare our results with similar results in the behavioral approach to systems' theory. The advantage of using the notion of bisimilarity is that it applies to state space systems and the computations involved are operations on real matrices. Keywords: Linear systems, bisimulations, achievability, canonical controller.

I. INTRODUCTION

A common question in systems and control theory is the following: given a plant, can one suitably alter it so that we have a modified system that suits our needs. We achieve this objective by constructing another system called a 'controller' and interconnecting/'attaching' this to the plant so that the new interconnected system has the desired dynamics. Now, to decide whether the interconnected system indeed does have the desired dynamics, we need some notion of equivalence between systems. For state space systems, one would call two systems equivalent if they are related by an invertible state space transform (also called the similarity transform). In the 'behavioral' approach, two systems are equivalent if the behaviors are equal. A notion that is used in computer science is that of bisimulation. This notion has been used to study the equivalence of automata. The idea of a bisimulation has been extended to include continuous time dynamical systems (see [Pap03], [vdS04]). It has been found that this notion actually is stronger than the idea of behavior equality. Moreover, it inherently uses the concept of a state and combines the ideas of behavior equivalence and state space equivalence. In this paper we use this notion of equivalence between systems. For deterministic systems (i.e. without disturbances) bisimulation reduces to the idea of behavioral equivalence. For two state space systems which are controllable and observable, equivalence in the sense of bisimulations and state space equivalence are the same. Furthermore, any state space system is bisimilar to its minimal realization. Thus equivalence in the sense of bisimulations is more general. Moreover, it is a good blend (as we will see later) of state space equivalence and input-output behavior equality. Also, the definition of bisimulation is easily extendable to nonlinear systems (see [vdS04]).

In the rest of the paper, we establish necessary and sufficient conditions which allow us to decide whether there

exists a controller which when interconnected to the plant, yields a system which is bisimilar to the desired system. Similar issues have been addressed for more general abstract state systems in [PvdSB05]. The paper is organized as follows: In section II we introduce bisimulations and relevant results needed in the paper. Section III states the problem and provides necessary and sufficient conditions for existence of a solution to the problem. Finally we conclude with some future directions.

II. DEFINITIONS

We now state precisely the notion of bisimulation for continuous time linear time invariant systems as introduced in [vdS04]. Consider two dynamical systems described by the following equations.

$$\begin{aligned} \dot{x}_i &= A_i x_i + B_i^u u_i + B_i^J f_i, x_i \in \mathcal{X}_i, u_i \in \mathcal{U} \\ y_i &= C_i^y x_i, y_i \in \mathcal{Y} \\ z_i &= C_i^z x_i, z_i \in \mathcal{Z} \end{aligned}$$
(1)

where i = 1, 2, and each system is denoted by Σ_i . x_i is the state of the system and takes values in \mathcal{X}_i , u_i is an input vector in \mathcal{U} , f_i is also an input vector and, y_i and z_i are output vectors in \mathcal{Y} and \mathcal{Z} respectively. All the variables take values in finite dimensional linear spaces.

Definition 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

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

with the following property. Take any $(x_{10}, x_{20}) \in \mathcal{R}$ and any joint input function $f_1 = f_2$. Then, for any input function u_1 there should exist 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

$$(x_1(t), x_2(t)) \in \mathcal{R} \text{ for all } t \ge 0$$
(2)

$$z_1(t) = z_2(t) \text{ for all } t \ge 0 \tag{3}$$

and conversely, for any input function u_2 there should exist a function u_1 such that the state trajectories $x_1(t)$ and $x_2(t)$ satisfy (2) and (3).

Note that in the definition of a bisimulation relation with respect to the variables f and z, the output y_i does not play any role. However, it is used when we interconnect the plant to a controller (u and y are the variables available to the controller). Besides, the set of pair of states x and inputs f, resulting in the same output y in the plant, play an important role in proving the main result.

^{*} The authors are with the University of Groningen. Email: h.g.vinjamoor@rug.nl, a.j.van.der.schaft@math.rug.nl

A bisimulation relation can be explicitly characterized by conditions involving the matrices describing the two systems (see [vdS04]).

Proposition 2: Let Σ_1 and Σ_2 be two systems of the form given in equation (1). A subspace $\mathcal{R} \subseteq \mathcal{X}_1 \times \mathcal{X}_2$ is a bisimulation relation if and only if the following are true:

$$\mathcal{R} + im \begin{bmatrix} B_1^u \\ 0 \end{bmatrix} = \mathcal{R} + im \begin{bmatrix} 0 \\ B_2^u \end{bmatrix} =: \mathcal{R}_e$$

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \mathcal{R} \subseteq \mathcal{R}_e$$

$$im \begin{bmatrix} B_1^f \\ B_2^f \end{bmatrix} \subseteq \mathcal{R}_e$$
(4)

$$\mathcal{R} \subseteq ker \begin{bmatrix} C_1^z & -C_2^z \end{bmatrix}$$

Definition 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 $\mathcal{R} \subset \mathcal{X}_1 \times \mathcal{X}_2$ such that $\pi_1(\mathcal{R}) = \mathcal{X}_1$ and $\pi_2(\mathcal{R}) = \mathcal{X}_2$, where $\pi_i : \mathcal{X}_1 \times \mathcal{X}_2 \to \mathcal{X}_i$, i = 1, 2, denote the canonical projections. Such a bisimulation relation is called a "full" bisimulation relation.

A weaker notion called 'simulation' is also defined.

Definition 4: A simulation relation of Σ_1 by Σ_2 is a linear subspace

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

with the following property. Take any $(x_{10}, x_{20}) \in \mathcal{R}$ and any joint input function $f_1 = f_2$. Then, for any input function u_1 there should exist 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

$$(t), x_2(t)) \in \mathcal{R} \text{ for all } t \ge 0$$
 (5)

$$z_1(t) = z_2(t) \text{ for all } t \ge 0 \tag{6}$$

Analogous to the conditions for bisimulations, we have: A subspace $S \subseteq \mathcal{X}_1 \times \mathcal{X}_2$ is a simulation relation of Σ_1 by Σ_2 if and only if the following are true

 $(x_1$

$$S + im \begin{bmatrix} B_1^u \\ 0 \end{bmatrix} \subseteq S + im \begin{bmatrix} 0 \\ B_2^u \end{bmatrix} =: S_e$$

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} S \subseteq S_e$$

$$im \begin{bmatrix} B_1^f \\ B_2^f \end{bmatrix} \subseteq S_e$$

$$S \subseteq ker \begin{bmatrix} C_1^z & -C_2^z \end{bmatrix}$$
(7)

Definition 5: System Σ_1 is said to be simulated by system Σ_2 , denoted by $\Sigma_1 \preccurlyeq \Sigma_2$, if there exists a simulation relation S of Σ_1 by Σ_2 such that $\Pi_1(S) = \mathcal{X}_1$. Such a simulation relation is called a "full" simulation relation of Σ_1 by Σ_2 .

The following lemma shows that \preccurlyeq is transitive.

Lemma 6: Let Σ_1 , Σ_2 and Σ_3 be three systems of the form of equation (1). If $\Sigma_1 \preccurlyeq \Sigma_2$ and $\Sigma_2 \preccurlyeq \Sigma_3$, then $\Sigma_1 \preccurlyeq \Sigma_3$.

We state one more proposition from [vdS04] which will be useful for proving the main result.

Proposition 7: Let $S \subset \mathcal{X}_1 \times \mathcal{X}_2$ be a full simulation relation of Σ_1 by Σ_2 and $\mathcal{T} \subset \mathcal{X}_2 \times \mathcal{X}_1$ be a full simulation relation of Σ_2 by Σ_1 . Then $\Sigma_1 \approx \Sigma_2$ where the full bisimulation relation is give by $S + T^{-1}$, with $T^{-1} = \{(x_a, x_b) | (x_b, x_a) \in T\}.$

The maximal (bi-)simulation relations exist and can be computed. We state a constructive algorithm (see [vdS04]) for computing the maximal simulation of Σ_1 by Σ_2 where Σ_1 and Σ_2 are of the form given in equation (1). The algorithm is very similar to the algorithm used to find the maximal controlled invariant subspace contained in given subspace (see [Won85]) of the state space; this algorithm also terminates in a finite number of steps.

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

$$S^{0} = \mathcal{X}_{1} \times \mathcal{X}_{2}$$

$$S^{1} = \{z \in S^{0} | z \in kerC\}$$

$$S^{2} = \{z \in S^{1} | Az + imG_{1} \subset S^{1} + imG_{2}\}$$

$$S^{j+1} = \{z \in S^j | Az + imG_1 \subset S^j + imG_2\}$$

The algorithm terminates when $S^j = S^{j+1}$. It can be proved that the termination occurs in a finite number of steps (bound by the dimension of S_0) to yield the maximal simulation relation (see [vdS04]). The algorithm for computing a bisimulation relation is similar. The above algorithm for simulation relations can be used to verify the necessary and sufficient condition derived in the next section.

III. MAIN RESULT

We now formulate the problem statement precisely. Let P denote the plant and C the controller system. Let S denote the desired system. P, C and S are all represented as state space systems with states x_P, x_C and x_S respectively. P has inputs f_P and u_P and outputs z_P and y_P . The controller shares the variables y and u with the plant; that is, only the variables u and y can be influenced by the controller. S has input f_S and output z_S . The plant is represented as

$$\dot{x}_P = A_P x_p + B_P^u u_P + B_P^f f_P$$

$$\begin{bmatrix} z_P \\ y_P \end{bmatrix} = \begin{bmatrix} C_P^z \\ C_P^y \end{bmatrix} x_P$$
(8)

 X_P denotes the state space of P.

We define the system N as follows: Setting u_P and y_P to zero in the plant we get the following system,

$$\dot{x}_P = A_P x_P + B_P^J f_P$$

$$\begin{bmatrix} z_P \\ 0 \end{bmatrix} = \begin{bmatrix} C_P^z \\ C_P^y \end{bmatrix} x_p$$
(9)

The state space for this system equals the largest (A_P, B_P^J) invariant subspace (see [Won85]) contained in $kerC_P^y$; we denote it by X_N . The explicit equations for the dynamics of N are given as follows:

$$\dot{x}_P = (A_P + B_P^f F)x_P + B_P^f Lw \tag{10}$$

where F is such that $(A_P + B_P^f F)X_N \subset X_N$, $x_P(0) \in X_N$ and L is such that $im(L) = imB_P^f \cap X_N$. Now use a basis adapted to X_N so that only the first few components of the state vector are non-zero in X_N . This yields an explicit expression for N.

Remark 8: Any input in N is given as $Fx_P + Lw$ where w can take any arbitrary value in the function space considered. We denote this set of inputs by \mathcal{F} . Observe that \mathcal{F} is in general not the whole function space.

S is represented as

$$\dot{x}_S = A_S x_S + B_S^J f_S z_S = C_S^z x_S$$
(11)

Let X_S be the state space for S.

 $P \parallel C$ is the system obtained when the variables u and y are shared by the two systems, i.e., u and y satisfy the equations of the plant P as well as of the controller C. Let (u_P, y_P) be the variables that are available for control in the plant and (u_C, y_C) the variables in the controller which we interconnect with the plant variables (u_P, y_P) . Then $\begin{bmatrix} u_P \\ y_P \end{bmatrix} = \prod \begin{bmatrix} u_C \\ y_C \end{bmatrix}$ where Π is a square *permutation* matrix (we choose controllers with the same number of control variables as those of the plant). Note that as a result of the interconnection the state space of the interconnected system may be smaller than the product space $X_P \times X_C$. The equations for $P \parallel C$ are given by

$$\dot{x}_P = A_P x_P + B_P^u u_P + B_P^f f_P$$
$$\begin{bmatrix} z_P \\ y_P \end{bmatrix} = \begin{bmatrix} C_P^z \\ C_P^y \end{bmatrix} x_P$$
$$\dot{x}_C = A_C x_C + B_C^u u_C + B_C g$$
$$\begin{bmatrix} h_C \\ y_C \end{bmatrix} = \begin{bmatrix} C_C^z \\ C_C^y \\ C_C^y \end{bmatrix} x_C$$

subject to the constraint

$$\begin{bmatrix} u_C \\ y_C \end{bmatrix} = \Pi \begin{bmatrix} u_P \\ y_P \end{bmatrix}$$

where Π is a permutation matrix as mentioned earlier and g and h_C are additional variables in the controller that are not available for interconnection with the plant. We are now ready to state the problem.

Problem statement: Given P and S, find necessary and sufficient conditions for the existence of a controller C such that $P \parallel C$ is bisimilar to S.

Theorem 9: $(N \preccurlyeq S \preccurlyeq P) \Leftrightarrow (\exists C \text{ such that } P \parallel C \approx S)$. Before proving this theorem let us first make some comments. For proving the necessity, we must allow for any kind of interconnection, since we do not a priori know how the plant and controller are interconnected, i.e., we do not know the permutation matrix Π . For proving sufficiency however, we have to construct our own controller and hence can choose the kind of interconnection.

The existence of $N \preccurlyeq S$ plays a key role in proving that the controller we construct is actually one that achieves the desired specification S up to bisimulation. This is analogous to a similar result that has been proved in the behavioral approach, where the relation \preccurlyeq is replaced by a set inclusion \subseteq (see [WT02]). However, here we take explicit account of the state of the system as against the purely behavioral approach. Let R_{SP} denote the maximal simulation relation of S by P and similarly let R_{NS} be the maximal simulation relation of N by S.

It is worth while noting that the condition $N \preccurlyeq S \preccurlyeq P$



Fig. 1. The canonical controller

can actually be computationally checked without too much difficulty using the algorithm stated in the previous section. The condition $S \preccurlyeq P$ can be checked by directly applying the algorithm since we have explicit expressions of S and P. Checking $N \preccurlyeq S$ requires some more work as we have to compute the explicit equations for N as in (10). Having done so, we can then apply the algorithm to compute the maximal simulation relation of N by S. as follows: The inputs allowed in N are given as $Fx_P + Lw$. Setting the input to this and writing the equations for computing $N \preccurlyeq S$ we have (see equation (10))

$$\begin{bmatrix} \dot{x}_P \\ \dot{x}_S \end{bmatrix} = \begin{bmatrix} A_P + B_P^f F & 0 \\ B_S^f F & A_S \end{bmatrix} \begin{bmatrix} x_P \\ x_S \end{bmatrix} + \begin{bmatrix} B_P^f L \\ B_S^f F L \end{bmatrix} w$$
(12)
$$\begin{bmatrix} C_P^z & -C_S^z \\ C_P^y & 0 \end{bmatrix} \begin{bmatrix} x_P \\ x_S \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} =: C_{NS} \begin{bmatrix} x_P \\ x_S \end{bmatrix}$$

Let A_{NS} and B_{NS} denote the A and B matrices for the above equation. R_{NS} is then the largest A_{NS} -invariant subspace contained in $kerC_{NS}$ such that $imB_{NS} \subset R_{NS}$. Thus $N \preccurlyeq S$ can be checked.

In order to prove theorem III we define one more interconnection viz. the canonical controller (see figure 1). The canonical controller was introduced in [vdS03] in a behavioral setting. Here we use the same idea, but for state space systems. Denote $C_{can} = P \parallel S$ where the interconnection is with respect to the variables f and z. The equations governing the dynamics of C_{can} are given by,

$$\begin{bmatrix} \dot{x}_S \\ \dot{x}_P \end{bmatrix} = \begin{bmatrix} A_S & 0 \\ 0 & A_P \end{bmatrix} \begin{bmatrix} x_S \\ x_P \end{bmatrix} + \begin{bmatrix} B_S^f \\ B_P^f \end{bmatrix} f + \begin{bmatrix} 0 \\ B_P^u \end{bmatrix} u$$
$$\begin{bmatrix} C_S^z & -C_P^z \end{bmatrix} \begin{bmatrix} x_S \\ x_P \end{bmatrix} = 0$$
$$y = \begin{bmatrix} 0 & C_P^y \end{bmatrix} \begin{bmatrix} x_S \\ x_P \end{bmatrix}$$
(13)

Let the A-matrix of the above system be denote by A_{can} and the matrix $\begin{bmatrix} B_S^f & 0\\ B_P^f & B_P^u \end{bmatrix}$ as B_{can} . The state space X_{can} for C_{can} is defined as the largest (A_{can}, B_{can}) invariant subspace contained in $ker [C_S^z - C_P^z]$.

Lemma 10: $R_{SP} \subseteq X_{can}$

Proof : As defined, X_{can} is the largest subspace R such that

$$A_{can}R \subseteq R + imB_{can}$$
$$R \subseteq ker \begin{bmatrix} C_P^z & -C_S^z \end{bmatrix}$$

On the other hand, the maximal simulation relation of S by P is given by the largest subspace \bar{R} such that

$$\begin{aligned} A_{can}\bar{R} &\subseteq \bar{R} + im \begin{bmatrix} 0\\ B_P^u \end{bmatrix}\\ \bar{R} &\subseteq ker \begin{bmatrix} C_P^z & -C_S^z \end{bmatrix}\\ im \begin{bmatrix} B_S^f \\ B_P^f \end{bmatrix} &\subseteq \bar{R} \end{aligned}$$

Since $im \begin{bmatrix} B_S^I \\ B_P^I \end{bmatrix} \subseteq \overline{R}, \overline{R}$ also satisfies the conditions defining R. Thus $R_{SP} \subseteq X_{can}$.

We now prove theorem 9.

Proof : (\Leftarrow :) Since $P \parallel C \approx S$, by lemma 6 it is sufficient to show that $P \parallel C$ is simulated by P and itself simulates N. Let (x_P, x_C) be an initial state in $P \parallel C$. Let f be some input and u_{PC} the signal u that evolves with time in $P \parallel C$. Let z_{PC} be the corresponding output that evolves with time. Now consider a stand alone plant (i.e. without controller attached) with initial state x_P with input f and input $u = u_{PC}$. The output z_P is uniquely determined by the initial state and the inputs. Since these are the same for the plant in the interconnection and the stand alone plant that we have considered, $z_P = z_{PC}$. The above argument is true for any state in $P \parallel C$. The simulation relation of $P \parallel C$ by P is thus given by

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

Now consider N. Let x_N be any state in N and $f \in \mathcal{F}$; see remark 8. By the definition of N, the output y_N will be zero; let z_N be the output corresponding to state x_N . Now, choose the state $(x_N, 0)$ in $P \parallel C$ with input f. (Given the initial state and input, the state trajectory is uniquely defined. Since $u_P = 0$, $y_P = 0$ and $x_C = 0$ satisfy the system equations, this is the only possible set of signals.) Due to the state $(x_N, 0)$, the input and output to the controller will be zero and its state trajectory will be identically zero. Thus the output z_{PC} in $P \parallel C$ is determined uniquely by the state x_N and input f_P of the plant. Since $u_P = y_P = 0$, and $x_N \in X_N$, $z_{PC} = z_N$. This argument holds true for any state in N. The simulation relation of N by $P \parallel C$ is given by

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

We now prove the other direction of the claim:

 $(\Rightarrow:)$ We now have to construct a controller C and show

that $P \parallel C$ is bisimilar to S. For this we construct the 'canonical controller'.

1. We define the system $C_{can} = P \parallel S$ with state space $X_{can} \subset X_S \times X_P$, input $f_{can}(f_{can} = f_S = f_P)$, input u_{can} , output y_{can} and $z_S = z_P$. (Intuitively, S and P share the z variable and we allow only those combinations of the states in X_P and X_S which ensure that $z_S = z_P$.) Note that since $S \preccurlyeq P$, for all $x_S \in X_S$, $\exists x_P \in X_P$ such that for $f_S = f_P$, $\exists u$ such that $z_S = z_P$. Let R_{SP} be the simulation relation. The existence of R_{SP} together with lemma 10 ensures that there exist states and inputs for which the interconnection is not ill-posed.

2. Now consider the interconnection of C_{can} and P obtained by setting $u_P = u_{can}$ and $y_P = y_{can}$. We denote the output z of P by z_{pcan} . Note that we are equating the outputs of two systems ($y_P = y_{can}$). As a result the state space of $C_{can} \parallel P$ may not be the whole product space $X_{can} \times X_P$. The states $\pi_P X_{can}$ and x_P should be such that $y_P = y_{can}$. Moreover f_P and f_{can} are not entirely 'free' inputs and are related. To characterize such states and the relation between the inputs, we state the following lemma.

Lemma 11: Let $x_P, x'_P \in X_P$ with input f_P and f'_P respectively. Then, for the same input $u_P, y_P = y'_P$ if and only if $x_P - x'_P \in X_N$ and $f_P - f'_P \in \mathcal{F}$.

Proof : The output due to initial state x_p , and inputs u and f is

$$y_{P}(t) = C_{P}^{y} e^{(A_{P}t)} x_{P} + C_{P}^{y} \int_{0}^{t} e^{(A_{P}(t-\tau))} B_{f}^{P} f_{P}(\tau) d\tau + C_{P}^{y} \int_{0}^{t} e^{(A_{P}(t-\tau))} B_{u}^{P} u_{P}(\tau) d\tau$$
(14)

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

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

$$0 = C_P^y e^{(A_P t)} (x_P - x'_P) + C_P^y \int_0^t e^{(A_P (t-\tau))} B_f^P (f_P(\tau) - f'_P(\tau)) d\tau$$
(15)

Thus $(x_P - x'_P) \in X_N$ and $f_P - f'_P \in \mathcal{F}$. (\Leftarrow): Given $(x_P - x'_P) \in X_N$ and $f_P - f'_P \in \mathcal{F}$, subtracting the expressions for outputs y_P and y'_P , we get zero. (because input u_P is the same).

Thus the states allowed in $C_{can} \parallel P$ are a subset of $X_{can} \times X_p$ such that $\pi_P X_{can} - \pi_2(X_{can} \times X_P) \in X_N$ where $\pi_2(R_{SP} \times X_P)$ is the projection on the second component of the state space of $C_{can} \parallel P$. Also, $f_P - f_{can} \in \mathcal{F}$.

We will now prove the other direction of the claim by showing that

•
$$S \preccurlyeq C_{can} \parallel P$$

• $C_{can} \parallel P \preccurlyeq S$

Proving these two statements is equivalent (see proposition 7) to proving that S and $C_{can} \parallel P$ are bisimilar.

 $S \preccurlyeq C_{can} \parallel P$: Let $x_S \in X_S$ and f_S be the corresponding

input. Choose the state of $C_{can} \parallel P$ as $((x_S, x_P), x_P)$ where $(x_S, x_P) \in R_{SP}$ and select $f_P = f_{can} = f_S$. Note that this is allowed because the zero state with corresponding input $f_P = 0$ is valid pair of (x_N, f_P) in N. The output z_S and z_{pcan} will be the same. The simulation relation is given by

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

 $C_{can} \parallel P \preccurlyeq S$: Let $((x_S, x_P), x'_P)$ be any state in $C_{can} \parallel P$. Then we can write $x'_P = x_P + x_N$ for some state $x_N \in X_N$ (by the result proved above). Choose state $x_S + x'_S$ in Swhere $(x_N, x'_S) \in R_{NS}$ where R_{ns} is the simulation relation between N and S. Thus the state of $C_{can} \parallel P \times X_s$ that we have chosen is $(((x_S, x_P), x_P + x_N), x_S + x'_S)$ which can be written as $(((x_S, x_P), x_P + x_N), x_S + x'_S)$ which can be written as $(((x_S, x_P), x_P), x_S) + (((0, 0), x_N), x'_S))$. Further, we can write $f_P = f_{can} + f_N$ where $f_N \in \mathcal{F}$. The output $z_S = z_{pcan}$ for both states. Since the system is linear, the output due the sum of the states is also the same. The simulation relation is given by

$$\{(a, b, c, d) \in X_S \times X_P \times X_P \times X_S | \\ (a, b) \in X_{can}, c - b \in X_N \text{ and } ((c - b), (d - a)) \in R_{NS} \}$$

This proves the result. \Box

IV. EXAMPLE

We present here a mathematical example to illustrate the main theorem.

Consider a plant given by

$$\dot{x}_P^1 = x_P^1 + x_P^2 + u_P$$
$$\dot{x}_P^2 = x_P^1 + x_P^2 + f$$
$$y_P = x_P^2$$
$$z_P = x_P^1$$

Let S be given by

$$\begin{split} \dot{x}_{S}^{1} &= x_{S}^{1} + x_{S}^{2} + bx_{S}^{3} \\ \dot{x}_{S}^{2} &= x_{S}^{1} + x_{S}^{2} + f \\ \dot{x}_{S}^{3} &= ax_{S}^{2} \\ z_{S} &= x_{S}^{1} \end{split}$$

where a and b are non-zero real numbers. The state space of N is found to be the span of $\begin{bmatrix} 1 & 0 \end{bmatrix}'$. Note that $X_N \cap imB_P^f = 0$. Therefore, f is uniquely determined; in fact $f = -x_P^1$. Thus, the equations for N are

$$\dot{x}_P^1 = x_P^1$$
$$z_P = x_P^1$$

where $x_P \in X_N$. Consider any state $x_P^1(0)$ in N. Choose state $x_S^2(0) = 0$ and $x_S^3(0) = 0$ in S with $f_S = -x_S^1$ where $x_S^1(0) = x_P^1(0)$. Then the equations for S reduce to

$$\dot{x}_S^1 = x_S^1$$
$$z_S = x_S^1$$

Thus $N \preccurlyeq S$.

Now, consider S with any initial state with input f_S . Set $f_P = f_S$. Choose $x_P^1(0) = x_S^1(0)$, $x_P^2(0) = x_S^2(0)$ and $u_P = bx_S^3(t)$. Then it is clear that $z_P = z_S$. Thus $S \preccurlyeq P$.

Hence, by theorem 9, there exists a controller which when interconnected with the plant yields a system which is bisimilar to S. We can see this as follows: Choose a controller given by

$$\dot{x}_C = au_C$$

$$y_C = bx_C$$

subject to (16)

$$y_P = u_C$$

$$y_C = u_P$$

Using this as the controller it is easily seen that we get a system bisimilar to S. Interestingly enough, we can actually arrive at the same controller using the canonical controller. The equations for the canonical controller are as follows:

$$\begin{split} \dot{x}_{P}^{1} &= x_{P}^{1} + x_{P}^{2} + u_{P} \\ \dot{x}_{P}^{2} &= x_{P}^{1} + x_{P}^{2} + f \\ \dot{x}_{S}^{1} &= x_{S}^{1} + x_{S}^{2} + bx_{S}^{3} \\ \dot{x}_{S}^{2} &= x_{S}^{1} + x_{S}^{2} + f \\ \dot{x}_{S}^{3} &= ax_{S}^{2} \\ y_{P} &= x_{P}^{2} \end{split}$$

subject to the constraints

$$x_P^1 = z_P = z_S = x_S^1$$
$$f_P = f_S$$

Now, let us impose an additional constraint $x_P^2 = x_S^2$. The first and third equation (together with the other constraints) yield $u_P = bx_S^3$. Further, the second and the fourth equation become the same equation. Eliminating redundant equations and ignoring equations that are present in the plant equations we get,

$$\dot{x}_S^3 = ay_P$$
$$u_P = bx_S^3$$

Observe that these are precisely the equations of the controller in equation (16).

V. CONCLUSION

We have derived necessary and sufficient conditions for the existence of a controller C such that $P \parallel C \approx S$ namely $N \preccurlyeq S \preccurlyeq P$. The condition $S \preccurlyeq P$ is expected. The critical condition is $N \preccurlyeq S$. This condition enables us to prove sufficiency. Moreover, the conditions derived can be verified computationally.

Although elegant, the canonical controller may not be very useful in practice. One reason is that it is not likely to be regular, i.e., we may not be able to connect it to the plant in such a way that inputs of the plant are connected to the outputs of the controller and vice versa. Also, the canonical controller is generally of a high state space dimension and is in a sense redundant (i.e. contains a copy of the plant and the desired system). Despite these apparent drawbacks, a result in the behavioral approach (see [JPvdS08]) indicates that regular implementability can be characterized using the canonical controller. It will hence be fruitful to look for a similar condition in terms of simulation relations.

REFERENCES

- [JPvdS08] A. Agung Julius, Jan Willem Polderman, and Arjan van der Schaft. Parametrization of the regualar equivalences of the canonical controller. *IEEE Transaction on Automatic Control*, 53(4):1032–1036, May 2008.
- [Pap03] G.J. Pappas. Bisimilar linear systems. Automatica, 39(12):2035–2047, 2003.
- [PvdSB05] G. Pola, A.J. van der Schaft, and M.D. Di Benedetto. Achievable bisimilar behavior of abstract state systems. pages 1535– 1540. in Proc. 44th IEEE Conference on Decision and Control and the European Control Conference, Seville, Spain, December 2005.
- [vdS03] Arjan van der Schaft. Achievable behavior of general systems. *Systems & Control Letters*, 49:141–149, 2003.
- [vdS04] Arjan van der Schaft. Equivalence of dynamical systems by bisimulation. IEEE Transaction on Automatic Control, December 2004.
- [Won85] W.M. Wonham. *Linear multivariable control: a geometric approach*. Springer, New York, Third Edition, 1985.
- [WT02] J.C. Willems and H.L. Trentelman. Synthesis of dissipative systems using quadratic differential forms: part i. *IEEE Trans*actions on Automatic Control, 47:53–69, 2002.