Proceedings of the 41st IEEE
Conference on Decision and Control
Las Vegas, Nevada USA, December 2002

TuA01-2

Achievable behavior by composition

A.J. van der Schaft[1], A.A. Julius
Faculty of Mathematical Sciences, University of Twente
P.O. Box 217, 7500 AE Enschede, the Netherlands
Fax 31-53-4340733
{a.j.vanderschaft,a.a.julius}@math.utwente.nl

Abstract

A fundamental question in systems and control theory concerns the characterization of the set of achievable closed-loop systems for a given plant system and a controller system to be designed. This problem, for example, shows up in assessing the 'limits of performance' of a controlled system. Similar problems have been studied by researchers in automata theory and discrete event systems replacing the notion of closed-loop system by the composition of a given system and its controller. In this paper this problem is addressed in a general behavioral context. Necessary and often sufficient conditions for a behavior to be achievable are given, and for any achievable behavior a canonical controller is defined. These results generalize previously obtained results obtained for finite-dimensional linear systems. Next these general results are applied to classes of automata and hybrid systems.

1 Introduction

One can compactly express a plenitude of problems by considering the solution of equations of the form

$$\mathcal{P} \,||\, \mathcal{C} \cong \mathcal{S} \tag{1}$$

with $\mathcal{P}$ and $\mathcal{S}$ given, say continuous, discrete, or hybrid systems, in the unknown system $\mathcal{C}$. This equation lies at the heart of modularity; for analysis and design. Of course, in order to make sense of this equation one has to make precise the definition of *system composition* ($||$) and *system equivalence* ($\cong$).

Versions of this problem have been investigated (from different points of view) by researchers in many areas, including automata theory, several process algebra formalisms and control theory. For example, within systems and control theory the fundamental question of the 'limits of performance' of a controlled system, and the parameterization of H_∞ controllers for a given plant system, can be regarded as instances of this general problem, recognizing $\mathcal{P}$ as the plant system, $\mathcal{S}$ as the desired (closed-loop) system behavior, and $\mathcal{C}$ as the controller to be constructed.

[1] Research funded by the EU Program HYBRIDGE (IST-2001-32460)

The general solvability of equation (1) has been addressed and solved within a general behavioral framework in [10] (see also [7, 4]) for *finite-dimensional linear differential* systems. In this version of the general problem $\mathcal{P}, \mathcal{S}$ and $\mathcal{C}$ are all linear system behaviors (that is, the trajectories or 'traces' generated by the finite-dimensional linear system). Furthermore, in this behavioral setting the system composition $||$ denotes *intersection* of behaviors, while the system equivalence $\cong$ is just *equality* of behaviors.

Very recently, the results of [10] have been extended and generalized to *general behavioral systems* in [6]; obtaining sufficient and often necessary conditions for solvability of (1), and, in case of solvability, the construction of a controller solving (1). In the current paper we will summarize and extend the results of [6] in Section 2, and then apply in Section 3 these results to the case when $\mathcal{P}, \mathcal{S}$ and $\mathcal{C}$ are (subclasses of) automata and hybrid systems.

2 General behavioral results

Consider a system $\mathcal{P}$ (the 'plant') with two types of external variables, namely the variables z which can be interconnected to another system $\mathcal{C}$ (the 'controller') sharing the same variables z, and remaining variables w which represent the interaction (or communication) of the system with (the rest of) its environment; see Figure 1.

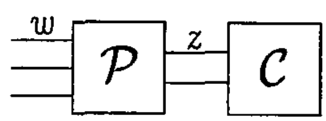

Figure 1: Plant controller configuration

We consider $\mathcal{P}$ and $\mathcal{C}$ to be systems in a general *behavioral* sense, that is, as a collection of allowable system trajectories. Formally, let W be a general set where the variables w take value, and let Z be the set where the variables z take value. Furthermore, let T be a general set denoting the time-axis. (Note that although we primarily think of T as $\mathcal{R}$ or $\mathcal{Z}$ we do not impose any conditions on the set T.) The plant system $\mathcal{P}$ is given as a collection of time-functions (w, z)

0-7803-7516-5/02/$17.00 ©2002 IEEE

with

$$w : T \to W, \qquad z : T \to Z, \tag{2}$$

that is, $\mathcal{P} \subset (W \times Z)^T$. Note that we do not require the spaces W and Z to be disjoint; indeed, some of the components of w and z may coincide.

Similarly, the controller system $\mathcal{C}$ is given as a collection of time-functions

$$z : T \to Z \tag{3}$$

that is, $\mathcal{C} \subset Z^T$. The *composition* of $\mathcal{P}$ and $\mathcal{C}$ via the shared variables z, denoted $\mathcal{P} \|_z \mathcal{C}$, is given by

$$\mathcal{P} \|_z \mathcal{C} = \{w : T \to W \mid \exists z : T \to Z \text{ such that } (w,z) \in \mathcal{P}, z \in \mathcal{C}\} \tag{4}$$

(Note that the shared variables z become *hidden* variables in the composition.) A basic question in systems and control theory is to characterize the set of composed behaviors $\mathcal{P} \|_z \mathcal{C}$ that are achievable by selecting the controller system $\mathcal{C}$ in an appropriate way. This can be regarded as a fundamental issue in characterizing the 'limits of performance' of a given plant system $\mathcal{P}$ by considering all possible controller systems $\mathcal{C}$.

The following theorem has been recently derived in [6], generalizing a result obtained for linear finite-dimensional systems in [10]. Denote by $\pi_w(\mathcal{P}) \subset W^T$ the plant behavior projected on W^T, that is

$$\pi_w(\mathcal{P}) = \{w : T \to W \mid \exists z : T \to Z \text{ such that } (w,z) \in \mathcal{P}\} \tag{5}$$

Theorem 1 *Let $\mathcal{P} \subset (W \times Z)^T$ be a given plant system, and let $\mathcal{C} \subset Z^T$ be a controller system to be designed. Let $\mathcal{S} \subset W^T$ be a desired behavior. Then there exists $\mathcal{C}$ such that $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ if*

(i) $\mathcal{S} \subset \pi_w(\mathcal{P})$

(ii) The following implication holds: for any $(w,z), (\tilde{w},z) \in \mathcal{P}$ whenever $\tilde{w} \in \mathcal{S}$ then also $w \in \mathcal{S}$.

Proof Define the controller system $\mathcal{C}_{can}$ (called the *canonical controller*) in the following implicit way; see Figure 2.

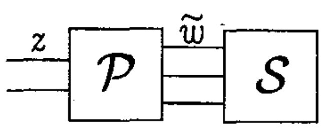

Figure 2: Canonical controller $\mathcal{C}_{can}$

$$\mathcal{C}_{can} := \{z : T \to Z \mid \exists \tilde{w} : T \to W \text{ such that } (\tilde{w},z) \in \mathcal{P} \text{ and } \tilde{w} \in \mathcal{S}\} \tag{6}$$

We prove that $\mathcal{P} \|_z \mathcal{C}_{can} = \mathcal{S}$; see Figure 3.
$\supset$: Let $w \in \mathcal{S}$. Because of $(i) \exists z : T \to Z$ such that $(w,z) \in \mathcal{P}$. Hence also $z \in \mathcal{C}_{can}$ (take $\tilde{w} = w$), and thus $w \in P \|_z \mathcal{C}$.
$\subset$: Let $w \in \mathcal{P} \|_z \mathcal{C}_{can}$. Thus $\exists z : T \to Z, \tilde{w} : T \to W$ such that $(w,z) \in \mathcal{P}, (\tilde{w},z) \in \mathcal{P}$ and $\tilde{w} \in \mathcal{S}$. By (ii) this implies that $w \in \mathcal{S}$. □

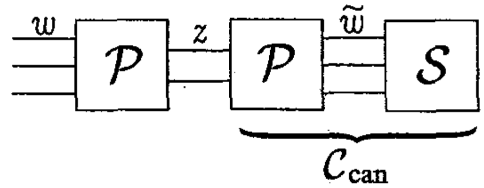

Figure 3: Composed behavior

Remark 2 *Note that although the results of Theorem 1 are formulated in terms of the behaviors $\mathcal{P} \subset (W \times Z)^T, \mathcal{C} \subset Z^T, \mathcal{S} \subset W^T$, we did not really use the time-function structure of these sets. Indeed, all statements remain equally valid if we replace $(W \times Z)^T = W^T \times Z^T, Z^T, W^T$ by general sets $\mathcal{W} \times \mathcal{Z}, \mathcal{Z}, \mathcal{W}$, and consider $\mathcal{P} \subset \mathcal{W} \times \mathcal{Z}, \mathcal{C} \subset \mathcal{Z}, \mathcal{S} \subset \mathcal{W}$.*

Remark 3 *In some sense the action of the canonical controller $\mathcal{C}_{can}$ can be seen as 'inverting' the plant $\mathcal{P}$ and substituting the desired behavior $\mathcal{S}$. Note however that we have not split the variables z and w into input and output components. Furthermore, $\mathcal{C}_{can}$ is defined in an implicit way (using the auxiliary variables $\tilde{w}$), and elimination of the variables $\tilde{w}$ from $\mathcal{C}_{can}$ will result in a controller of quite a different form.*

Remark 4 *It immediately follows from the proof of Theorem 1 that if $\mathcal{S}$ only satisfies condition (i) then still $\mathcal{S} \subset \mathcal{P} \|_z \mathcal{C}_{can}$, while if $\mathcal{S}$ only satisfies condition (ii) then $\mathcal{P} \|_z \mathcal{C}_{can} \subset \mathcal{S}$. The first case guarantees a kind of liveness property (the composed system contains a desired behavior $\mathcal{S}$), while in the second case the composed system $\mathcal{P} \|_z \mathcal{C}_{can}$ satisfies at least the 'specifications' given by $\mathcal{S}$ (see also [8]).*

Remark 5 *In [6] it has been shown how Theorem 1 generalizes the result obtained for finite-dimensional linear systems in [10].*

As discussed in [6] the conditions of Theorem 1 are often close to be *necessary* as well. Indeed, let $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ for some controller $\mathcal{C}$. Then it immediately follows that for every $w \in \mathcal{S} = \mathcal{P} \|_z \mathcal{C}$ there exists $z \in \mathcal{C}$ such that $(w,z) \in \mathcal{P}$, and hence $w \in \pi_w(\mathcal{P})$. Thus condition (i) is a *necessary condition* as well.
Necessity of condition (ii) is more subtle. Let $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$. Then for every $\tilde{w} \in \mathcal{S} = \mathcal{P} \|_z \mathcal{C}$ there exists $z' \in \mathcal{C}$ such that

$(\tilde{w},z') \in \mathcal{P}$. Let now $(w,z') \in \mathcal{P}$. Then also $w \in \mathcal{P} \|_z \mathcal{C} = \mathcal{S}$. Hence condition (*ii*) is necessary for a *non-empty subset* of $z' \in Z^T$ such that $(w,z'),(\tilde{w},z') \in \mathcal{P}$.

Complete necessity of condition (*ii*) is ensured if the plant $\mathcal{P}$ satisfies the following additional 'homogeneity' property H^W (see [6]):

$\mathcal{P}$ satisfies **Property** H^W if: Let $(\tilde{w},z),(w,z) \in \mathcal{P}$. Then if $(\tilde{w},z') \in \mathcal{P}$ also $(w,z') \in \mathcal{P}$.

Remark 6 *A simple example where condition (ii) is not necessary is given as follows. We consider the set-theoretic setting of Remark 2, with* $\mathcal{W} = \{w_1, w_2\}$ *(two elements), and* $\mathcal{Z} = \{z_1, z_2\}$ *(again two elements). Let the plant system be given as* $\mathcal{P} = \{(w_1,z_1),(w_2,z_1),(w_2,z_2)\}$ *and the desired behavior as* $\mathcal{S} = \{w_2\}$*. Then clearly condition (ii) is not satisfied (since* $(w_1,z_1),(w_2,z_1) \in \mathcal{P}, w_2 \in \mathcal{S}$ *does not imply* $w_1 \in \mathcal{S}$*). However, the controller* $\mathcal{C} := \{z_2\}$ *is such that* $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$*. Note that* $\mathcal{P}$ *does not satisfy Property* H^W.

Furthermore, it is clear that the canonical controller $\mathcal{C}_{can}$ *does not do the required job, since it is given as* $\mathcal{C}_{can} = \{z_1, z_2\}$.[1]

In general, if there exists a controller system $\mathcal{C}$ such that $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ then there can be many *different* controller systems $\mathcal{C}'$ also yielding $\mathcal{P} \|_z \mathcal{C}' = \mathcal{S}$. Among all these controllers the canonical controller $\mathcal{C}_{can}$ has the property of being the *least restrictive* controller (see [6]):

Proposition 7 *Consider the controller system* $\mathcal{C}_{can}$ *such that* $\mathcal{P} \| \mathcal{C}_{can} = \mathcal{S}$*. Let* $\mathcal{C}$ *be another controller such that* $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$*. Then for every* $z \in \mathcal{C}$ *with* $(w,z) \in \mathcal{P}$*, also* $z \in \mathcal{C}_{can}$.

The canonical controllers $\mathcal{C}_{can} := \mathcal{P} \|_w \mathcal{S}$, with $\mathcal{S}$ any system, are 'universal' in the following sense. Let $\mathcal{C}$ be any controller, and denote $\mathcal{S} := \mathcal{P} \|_z \mathcal{C}$. Then define $\mathcal{C}_{can} := \mathcal{P} \|_w \mathcal{S}$. If $\mathcal{P}$ satisfies the 'dual' homogeneity Property H^Z:

$\mathcal{P}$ satisfies **Property** H^Z if: Let $(w,\tilde{z}),(w,z) \in \mathcal{P}$. Then if $(w',\tilde{z}) \in \mathcal{P}$ also $(w',z) \in \mathcal{P}$,

then it follows that

$$\mathcal{P} \|_z \mathcal{C}_{can} = \mathcal{S}$$

Indeed, let $w \in \mathcal{S}$. Then $\exists z \in \mathcal{C}_{can}$ with $(w,z) \in \mathcal{P}$. Therefore $w \in \mathcal{P} \|_z \mathcal{C}_{can}$ (see Figure 4). Conversely, let $w \in \mathcal{P} \|_z \mathcal{C}_{can}$. Then there exist $z, \tilde{w}$ and $\tilde{z}$ such that $(w,z) \in \mathcal{P}$, $(\tilde{w},z) \in \mathcal{P}$, $(\tilde{w},\tilde{z}) \in \mathcal{P}$, $\tilde{z} \in \mathcal{C}$; see Figure 5. Since $\mathcal{P}$ satisfies Property H^Z, it follows that also $(w,\tilde{z}) \in \mathcal{P}$, and hence $w \in \mathcal{P} \|_z \mathcal{C} = \mathcal{S}$.

[1] We thank Jan C. Willems for a useful discussion on this issue.

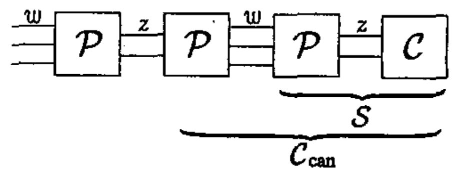

Figure 4: $\mathcal{P} \|_z \mathcal{C}_{can} \supset \mathcal{S}$

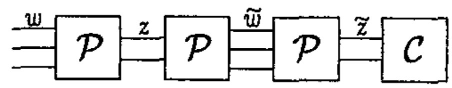

Figure 5: $\mathcal{P} \|_z \mathcal{C}_{can} \subset \mathcal{S}$

For more properties of the canonical controller $\mathcal{C}_{can}$ we refer to [6].

¿From an implementation point of view a basic problem that remains in the construction of the canonical controllers concerns the presence of the auxiliary variables $\tilde{w}$. Indeed, we would like to have an algorithmic procedure for eliminating these latent variables, and so to obtain an equivalent *explicit* controller. For the linear time-invariant case this can be easily done (see [6]), and extensions of this procedure to *nonlinear* systems are sketched in [6].

We conclude this section by giving the following extension of Theorem 1 where the controller system $\mathcal{C}$ is allowed to have additional external variables $v : T \to V$.

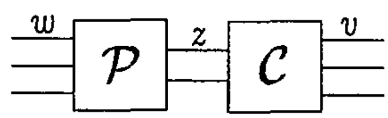

Figure 6: Plant-controller interconnection described in Theorem 8

Theorem 8 *Let* $\mathcal{P} \subset (W \times Z)^T$ *be a given plant system, and let* $\mathcal{C} \subset (Z \times V)^T$ *be a controller system to be designed, with additional external variables* $v \in V$*, see Figure 6. Let* $\mathcal{S} \subset (W \times V)^T$ *be a desired behavior. Then there exists* $\mathcal{C}$ *such that* $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ *if*

(i) $\pi_w \mathcal{S} \subset \pi_w(\mathcal{P})$

(ii) The following implication holds: for any $(w,z),(\tilde{w},z) \in \mathcal{P}$ *whenever* $(\tilde{w},v) \in \mathcal{S}$ *then also* $(w,v) \in \mathcal{S}$.

Proof Define the canonical controller system $\mathcal{C}_{can}$ as

$$\mathcal{C}_{can} := \{z : T \to Z, v : T \to V \mid \exists \tilde{w} : T \to W \text{ such that } (\tilde{w}, z) \in \mathcal{P} \text{ and } (\tilde{w}, v) \in \mathcal{S}\} \quad (7)$$

In the same way as in the proof of Theorem 1 it is shown that $\mathcal{P} \|_z \mathcal{C}_{can} = \mathcal{S}$. □

3 Discrete-event and hybrid systems

In this section we provide a preliminary discussion of applications of the general results described in Section 2 to automata and hybrid systems.

Most straightforward application of the results of Section 2 are to discrete-event systems or automata represented in a purely "behavioral" form, that is as *languages*. Indeed, let us define an event set $E := W \times Z$, and consider plant systems $\mathcal{P}$ to be given as a language over E, that is

$$\mathcal{P} \subset E^* \quad (8)$$

with E^* denoting as usual the set of all finite strings of elements of E. Similarly, we consider the desired behavior $\mathcal{S}$ to be a language over W, that is $\mathcal{S} \subset W^*$, and the to be constructed controller $\mathcal{C}$ to be a language over Z, that is, $\mathcal{C} \subset Z^*$. Then we immediately obtain the following version of Theorem 1:

Proposition 9 *Let $\mathcal{P} \subset (W \times Z)^*$ be a given plant system, and let $\mathcal{C} \subset Z^*$ be a controller system to be designed. Let $\mathcal{S} \subset W^*$ be a desired behavior. Then there exists $\mathcal{C}$ such that $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ if*

(i) $\mathcal{S} \subset \pi_w(\mathcal{P})$

(ii) The following implication holds: for any $(w,z), (\tilde{w},z) \in \mathcal{P}$ whenever $\tilde{w} \in \mathcal{S}$ then also $w \in \mathcal{S}$.

Remark 10 *The only technical difference with Theorem 1 is that the time axis T is not the same for all strings (although we could fix this by letting $T = \mathbb{N}$ and by adding an extra "sink" state). In view of Remark 2 this however does not pose a problem.*

Remark 11 *The conditions of Proposition 9 can be compared to the solution of the supervisory control problem in the case of partial event observation, as derived in the framework developed by Wonham and Ramadge; see e.g. the exposition of the Controllability and Observability Theorem in [1]. In order to do so one may associate with the shared variables z the events that are both controllable and observable. Then condition (i) can be interpreted as a controllability condition on the required language $\mathcal{S}$, and condition (ii) as an observability condition.*

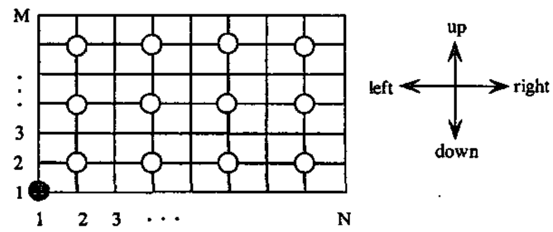

Figure 7: The lattice in Example 12. Remarks : black circle = car, white circle = road sign.

Example 12 *Consider the following situation. A dot is moving on a lattice of N by M points, as illustrated in Figure 7. We can think of the nodes of the lattice as the states of an automaton, with the initial state at, say, the lower left corner. Each state can be naturally named as a pair (i,j), with $1 \le i \le N, 1 \le j \le M$. Let the set W consist of $2 \times (N-1) \times (M-1)$ transitions associated with movement on the $(N-1) \times (M-1)$ vertices in the lattice, and let $Z = \{up, down, left, right\}$. It is obvious that every transition in this automaton can be naturally described as a pair $(w,z) \in W \times Z$. Imagine the lattice as, for example a street map of a city and the dot as a car. The vertices are then streets. The events in Z are the transitions at the disposal of the car driver.*

Referring to Proposition 9, we associate $\pi_w(\mathcal{P})$ with the language generated by the automaton. Using the car navigation interpretation, we can say that $\pi_w(\mathcal{P})$ represents all continuous trajectory originating from the initial state. We also observe that each $w \in \pi_w(\mathcal{P})$ is paired with exactly one $z \in Z^*$ such that $(w,z) \in \mathcal{P}$. Hence, the second condition given in the proposition is always satisfied. The proposition then implies that if a specification $\mathcal{S}$ consists of elements of the language generated by the automaton (or continuous car trajectories), we can find a controller (or a driver) $\mathcal{C}$ that realizes it. This implication is trivial.

The most important limitation of Proposition 9 is that the events in W and Z are assumed to be *synchronized*; indeed they appear in *pairs* $(w,z) \in E$.

Consider on the other hand an automaton A, whose set of events $E = W \cup Z$. One somewhat artificial way of treating this case is to *force* W and Z to synchronize in the following way. Introduce the extended sets of events

$$E_z := Z \cup \{\tau\},$$
$$E_w := W \cup \{\tau\},$$

and the mappings $F_z : E \to E_z$ and $F_w : E \to E_w$

$$F_z(a) = \begin{cases} a, & a \in Z \\ \tau, & otherwise \end{cases},$$

$$F_w(a) = \begin{cases} a, & a \in W \\ \tau, & otherwise \end{cases}.$$

The symbol τ denotes the silent event.

Let $\mathcal{A}$ be the language generated by the automaton A. Denote each element of $\mathcal{A}$ as $(a_1, a_2, a_3, \cdots) \in E^*$. We now define the behavior of the synchronized plant system $\mathcal{P}$ as the collection of all traces $((w_1, z_1), (w_2, z_2), \cdots) \in (E_w \times E_z)^*$ such that $(w_i, z_i) = (F_w(a_i), F_z(a_i))$, $i \in \mathbb{N}$ for some $(a_1, a_2, \cdots) \in \mathcal{A}$.

Notice that we need to introduce the silent action τ in order to have the forced synchronization between W and Z.

The case where W and Z are not synchronized is somehow analogous to the case where not all variables are observable from the variables used for interconnection in linear behaviors. For example, in the case of full synchronization between W and Z, it is possible to achieve the null behavior (the one that contains no traces at all) as a specification by interconnecting the plant with a null behavior as a controller. In the non-synchronized case, this is not necessarily true. For linear behaviors, the null behavior is achieved by using the null controller (the one that contains only the zero trajectory), if and only if all variables are observable from the interconnection variables.

Example 13 (continued) *Refer to our car navigation example. If we assume that instead of driving the car, we control its trajectory by using several traffic lights/road signs on the lattice, a different approach should be taken. Suppose that on every state (i, j) where i and j are both even, there is a road sign we can control. We refer to these states as the* even-even *states. Hence, the command $\{up, down, left, right\}$ can only be given if the car is in an even-even state, and Z is the set of transitions originating from the even-even states. It is clear that we do not have full synchronization between W and Z anymore. It is fairly easy to see that it is not possible to reject the trajectory going straight from* $(1,1)$ *to* $(N,1)$*, even if the most restrictive controller (the null behavior) is used.*

Indeed, we can easily observe that any $z \in E_z^*$ can be paired with more than one $w \in E_w^*$. We can associate this situation with non-observability of the whole behavior from Z.

The notion of observability for linear behaviors has been treated, for example in [3, 9]. However, the extension of this notion to behaviors related to discrete automata is not trivial. Recall that in linear behaviors, partitioning of information flow (i.e. the information that can be extracted from or fed to a behavior) is done based on partitioning of variables. This is not the case in discrete automata, where partitioning of events is done instead. We argue that flow of information is intimately related to the concept of controllability and observability, as controlling a behavior can be associated with feeding information to it and observing with extracting information from it. We suggest developing more general notions of controllability and observability in the behavioral framework as a potentially fruitful course for further research.

Another venue to the extension of Proposition 9 to the non-synchronized case is to replace the equality sign in $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ by an equivalence relation such as *(weak) bisimulation* ([2]). This will necessitate to extend the behavioral approach taken in this paper to *non-deterministic* automata, which are not completely specified by their generated languages. We leave this as a topic for future research.

The extension of Proposition 9 to *hybrid behaviors*, as defined in [5], is again straightforward. In this context the time axis T is taken to be $\mathcal{R}$ (for the continuous-time behavior), punctuated by a discrete set $\mathfrak{E}$ of times at which the events take place. (For simplicity of exposition we assume that there are no multiple events; see otherwise [5].) Let us now define signal spaces W_c and Z_c for the *continuous* variables w_c and z_c, and signal spaces W_d and Z_d for the *discrete* variables w_d and z_d. The behavior of a plant system $\mathcal{P}$ is then defined by a quadruple (w_c, w_d, z_c, z_d) with

$$w_c : \mathcal{R} \to W_c z_c : \mathcal{R} \to Z_c w_d : \mathfrak{E} \to W_d z_d : \mathfrak{E} \to Z_d.$$

(Again this implies that the events in W_d and Z_d are synchronized.)

Similarly, the desired behavior $\mathcal{S}$ is defined by pairs (w_c, w_d) and the controller system $\mathcal{C}$ by pairs (z_c, z_d). The analogon of Proposition 9 reads (in self-explanatory notation) as

Proposition 14 *Let $\mathcal{P}$ be a given plant system, and let $\mathcal{C}$ be a controller system to be designed. Let $\mathcal{S}$ be a desired behavior. Then there exists $\mathcal{C}$ such that $\mathcal{P} \|_z \mathcal{C} = \mathcal{S}$ if*

(i) $\mathcal{S} \subset \pi_{(w_c, w_d)}(\mathcal{P})$

(ii) The following implication holds: for any $(w_c, w_d, z_c, z_d), (\tilde{w}_c, \tilde{w}_d, z_c, z_d) \in \mathcal{P}$ whenever $(\tilde{w}_c, \tilde{w}_d) \in \mathcal{S}$ then also $(w_c, w_d) \in \mathcal{S}$.

4 Open problems

This paper exhibits some results in the application and extension of the general behavioral results, as discussed in Section 2, to discrete-event and hybrid systems. However, there are still some issues left untreated, which are themselves challenging.

From the implementation point of view, we identify two issues to be treated further. First, we recognize that not every controller behavior, which is created as a collection of trajectories, is suited for implementation. Some structures need to be imposed, for example linearity and time-invariance if we are interested in realization by differential systems. Second, even if we obtain an implementable controller, there is no guarantee that the interconnection can be done properly. Other authors have addressed this issue, for example Willems in [9] introduced the notion of regularity of interconnections of linear behaviors.

From the general behavior theoretical point of view, there are also things to be done. Many of the concepts and theory in the arsenal of tools of behavior theory were developed for linear behaviors. We may need to generalize and extend the existing tools to be able to handle general behaviors. General behavioral approach undoubtedly will overlap with the existing bodies of theory concerning the systems in consideration. To find and expose the relation between these more classical theories and the behavioral approach, as it has been done in linear systems, is an appealing research problem. In particular, we consider translating the conditions of Theorem 1 to a process-algebraic setting as one topic that matches this idea.

References

[1] C.G. Cassandras, S. Lafortune, *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, Boston, 1999.

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

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

[4] J.W. Polderman. Sequential Continuous Time Adaptive Control: A Behavioral Approach, pp. 2484-2487 in Proc. 39th IEEE Conf. Decision and Control, Sydney, Australia, 2000.

[5] A.J. van der Schaft, J.M. Schumacher, *An Introduction to Hybrid Dynamical Systems*, Springer Lecture Notes in Control and Information Sciences, Vol.251, Springer-Verlag, London, 2000.

[6] A.J. van der Schaft. Achievable behavior of general systems. Manuscript February 14, 2002. Submitted for publication.

[7] H.L. Trentelman. A truly behavioral approach to the H_∞- control problem, pp. 177-190 in *The Mathematics of Systems and Control: From Intelligent Control to Behavioral Systems*, (editors J.W. Polderman and H.L. Trentelman), Foundations Systems and Control Groningen, 1999.

[8] S. Weiland, A.A. Stoorvogel, B.D. de Jager, A behavioral approach to the H_∞ optimal control problem, *Systems and Control Letters, 32*, pp. 323–334, 1997.

[9] J.C. Willems, On Interconnections, Control and Feedback, *IEEE Transactions on Automatic Control*, Vol. 42(3), pp 326-339, 1997.

[10] J.C. Willems, H.L. Trentelman. Synthesis of dissipative systems using quadratic differential forms, Part I. *IEEE Transactions on Automatic Control*, Vol. 47, (No. 1), pp. 53-69, 2002.