A Partial Order Approach to Discrete Dynamic Feedback in a Class of Hybrid Systems

Domitilla Del Vecchio

University of Michigan, Department of Electrical Engineering and Computer Science,
1301 Beal Avenue, Ann Arbor, MI 48109
ddv@umich.edu
http://www.eecs.umich.edu/~ddv

Abstract. We consider the dynamic feedback problem in a class of hybrid systems modeled as (infinite) state deterministic transition systems, in which the continuous variables are available for measurement. The contribution of the present paper is twofold. First, a novel framework for performing dynamic feedback is proposed which relies on partial orders on the sets of inputs and of discrete states. Within this framework, a state estimator updates a lower and an upper bound of the set of current states. A controller then uses such upper and lower bounds to compute the upper and lower bounds of the set of inputs that maintain the current state in a desired set. Second, we show that under dynamic controllability assumptions, the conditions that allow to apply the developed algorithms can always be verified. Therefore, the partial order approach to dynamic feedback is general. A multi-robot system is presented to show the computational advantages in a system in which the size of the state set can be so large as to render enumeration and exhaustive techniques inapplicable.

1 Introduction

Controller design problems under language specification have been extensively studied for discrete systems in the computer science literature (see [10] for an overview). A control perspective in the context of discrete event systems was given by [7]. The approach has been extended to specific classes of hybrid systems such as timed automata [1] and rectangular automata [11]. These works are mainly concerned with state feedback. An output map is considered in the literature of viability theory for hybrid systems (see for example [2] and [5]), in which static output feedback is usually performed. In this paper, we consider the dynamic control problem for systems with continuous and discrete variables in the case in which the continuous variables are measured. This simplified scenario has practical interest in multi-robot systems in which the continuous variables represent the position and the velocity of a robot, while the discrete variables regulate the internal communication and coordination protocol. This work thus relates also to the computer science literature addressing control under partial observation of automata and of discrete event systems. In [7], the control problem of discrete event systems under language specifications is considered. The proposed control algorithms with full observation have polynomial complexity in the number of states. In the case of partial observations, the control problem becomes NP complete at worse. In a practical system, the number of states can be exponential in the number of constituent processes,

A. Bemporad, A. Bicchi, and G. Buttazzo (Eds.): HSCC 2007, LNCS 4416, pp. 159–173, 2007.

and therefore these control methodologies are prohibitive. Caines and Wang [6], consider the problem of steering the state of a partially observed automata to a final desired state. A dynamic programming methodology is proposed, which leads to a complexity of the control computation that is polynomial in the size of the state set, of the input set, and of the output set. Modular synthesis and special structures on the process are suggested (by [8], for example) in order to reduce computation.

In this work, we exploit a partial order structure on the set of inputs and of states to construct a feedback system that updates the lower and upper bound of the set of possible current system states and gives as output the lower and upper bounds of the set of inputs that satisfy the system specifications. This can be achieved under suitable order preserving assumptions of the system dynamics with respect to the state and to the input. We then show that if the system is controllable by dynamic output feedback one can always find partial orders on which the assumptions needed for the construction of the proposed controller are verified. We finally show how these assumptions can be relaxed. A multi-robot example is proposed, which shows how to apply the proposed methodology in an attack-defense scenario. This paper is organized as follows. In section 2, we introduce the system model. In section 3, we introduce the control problem on a partial order. In section 4, we give a solution to the problem and in Section 5 we show that the proposed construction is possible if the system is controllable. In section 6, a relaxed version of the main theorem is proposed and a multi-robot example is illustrated. An appendix contains notions on partial order theory and the proofs.

2 Deterministic Transition Systems

Definition 1. A *deterministic transition system* is a tuple $\Sigma = (Q, \mathcal{I}, \mathcal{Y}, F, g)$ in which Q is a set of states, $\mathcal{Y}$ is a set of outputs, $\mathcal{I}$ is a set of inputs, $F : Q \times \mathcal{I} \longrightarrow Q$ is a transition function, and $g : Q \longrightarrow \mathcal{Y}$ is an output function.

An *execution* of Σ is any sequence $\sigma = \{s(k)\}_{k\in\mathbb{N}}$ such that $s(0) \in Q$ and $s(k+1) = F(s(k), u(k))$ for $u(k) \in \mathcal{I}$ for all $k \in \mathbb{N}$. The set of all executions of Σ is denoted $\mathcal{E}(\Sigma)$. The output sequence $g(\sigma)$ is also denoted $\{y(k)\}_{k\in\mathbb{N}}$ with $y(k) = g(s(k))$. Given a system execution σ, $s(k) = \sigma(k)(s)$ denotes the value of the state at step k along such an execution. Let $S \subseteq Q$ be a subset of the state set. We would like to design a control algorithm that based on the output sequence $\{y(k)\}_{k\in\mathbb{N}}$ of Σ determines control inputs that guarantee that $\sigma(k)(s) \in S$ for all k. The initial set, denoted $X_0 \subseteq Q$ is the set in which the initial condition of the system Σ is constrained to lie, that is, $s(0) \in X_0$. The next definition proposes a concept of dynamic output feedback analogous to the one proposed by [9].

Definition 2. The system Σ is said to be *controllable by dynamic output feedback* with respect to set S and initial set $X_0 \subseteq S$ if there exist a feedback system $\Sigma_f = (\mathcal{P}(Q), \mathcal{Y}, \mathcal{P}(\mathcal{I}), H_2, H_1)$ such that for all executions $\sigma \in \mathcal{E}(\Sigma)$ with output sequence $\{y(k)\}_{k\in\mathbb{N}}$ if $X(k+1) = H_2(X(k), y(k))$, $u(k) \in H_1(X(k), y(k))$, with $X(0) = X_0$, then (i) $\sigma(k)(s) \in X(k)$ and (ii) $X(k) \subseteq S$ for all k.

In this definition, $X(k)$ is the set of all possible states compatible with the system dynamics and with the output sequence, while H_2 is the update function of a state estimator.

The function H_1 for all set of possible states X, determines the set of inputs that map such a set inside S. Let $y \in \mathcal{Y}$, we denote the set of all possible states compatible with such an output by $O_y(\Sigma) = \{s \in Q \mid g(s) = y\}$. We refer to this set as an *output set*.

Proposition 1. *Let $X_0 \subseteq S$. System Σ is controllable by dynamic output feedback with respect to set S and initial set $X_0 \subseteq S$ if and only if $\{u \in \mathcal{I} \mid F(O_y(\Sigma) \cap S, u) \subseteq S\} \neq \emptyset$.*

The theorems that will be proven rely on the condition that a system is controllable by dynamic output feedback with respect to a set S. This proposition allows to replace such controllability condition by $\{u \in \mathcal{I} \mid F(O_y(\Sigma) \cap S, u) \subseteq S\} \neq \emptyset$ for all $y \in \mathcal{Y}$. In this paper, we do not focus on the problem of checking whether the condition of Proposition 1 is verified in a given system, but we focus on how to construct a dynamic feedback controller when such a condition is verified. For completeness, a system Σ is said to be *controllable by static output feedback* if for all $y \in \mathcal{Y}$ the set $\{u \in \mathcal{I} \mid F(O_y(\Sigma), u) \subseteq S\}$ is not empty. A system that is controllable by dynamic output feedback is not necessarily controllable by static output feedback. In fact, in the static output feedback no memory is needed in the controller. This memory is instead used in the dynamic output feedback case, in which a state estimator on-line restricts at each step the set of all possible current system states. We next specialize the structure of system Σ to explicitly model the evolution of continuous and discrete variables.

3 Problem Setup

Given a deterministic transition system $\Sigma = (Q, \mathcal{I}, \mathcal{Y}, F, g)$, we specialize it to the case $Q = \mathcal{A} \times \mathcal{Z}$, in which $\mathcal{A}$ is a discrete set of variables denoted $\alpha \in \mathcal{A}$, $\mathcal{Z}$ is a set of continuous variables denoted $z \in \mathcal{Z}$, and $\mathcal{I}$ is a discrete set of inputs denoted $u \in \mathcal{I}$. The transition function is the pair $F = (f, h)$, in which $f : \mathcal{A} \times \mathcal{Z} \times \mathcal{I} \longrightarrow \mathcal{A}$ and $h : \mathcal{A} \times \mathcal{Z} \longrightarrow \mathcal{Z}$. The set of outputs is defined as $\mathcal{Y} = \mathcal{Z} \times \mathcal{Z}$ and the output function is $g : \mathcal{A} \times \mathcal{Z} \longrightarrow \mathcal{Y}$. For the remainder of this paper, we denote by $\Sigma = (\mathcal{A} \times \mathcal{Z}, \mathcal{I}, \mathcal{Y}, (f, h), g)$ the system represented by the following difference equations

$$\alpha(k+1) = f(\alpha(k), z(k), u(k)), \qquad z(k+1) = h(\alpha(k), z(k)) \tag{1}$$
$$(y_1(k), y_2(k)) = (z(k), h(\alpha(k), z(k))).$$

Any execution of the system Σ is of the form $\sigma = \{\alpha(k), z(k)\}_{k \in \mathbb{N}}$ and the output sequence is given by $\{y(k)\}_{k \in \mathbb{N}} = \{y_1(k), y_2(k)\}_{k \in \mathbb{N}}$. Given any execution σ of the system, we will denote the values of z and α at step k along such an execution by $\sigma(k)(z)$ and $\sigma(k)(\alpha)$, respectively. Given the measured variables z, we consider the problem of determining the input u such that the discrete state α is kept inside a set $S \subseteq \mathcal{A}$. If $\mathcal{A}$ and $\mathcal{I}$ are finite and discrete, in order to compute the set of inputs that map a set $X \subseteq \mathcal{A}$ inside S, we can compute $f(\alpha, z, u)$ for all $u \in \mathcal{I}$ and all $\alpha \in X$ and check whether it is contained in S. Assuming the size of X and the size of S of the order of the size of $\mathcal{A}$, this requires a number of computations of order $|\mathcal{I}||\mathcal{A}|^2$. If $\mathcal{A}$ is given by the product of a number of sets (as it is in the multi-agent systems that we consider) this approach is not practical as the number of computations is exponential in the number of agents. We thus propose an alternative procedure, whose idea can be explained in the following simple example.

Assume $\alpha \in \mathbb{N}$, $X = [2, 11]$, $S = [1, 10]$, $u \in \mathbb{Z}$, and that $f(\alpha, z, u) = f(\alpha, u) = \alpha + u$. For computing the set of inputs in $\mathbb{Z}$ such that $f(X, u) \subset S$, it is enough to compute the set of $u \in \mathbb{Z}$ such that $f(2, u) \geq 1$ and the set of $u \in \mathbb{Z}$ such that $f(11, u) \leq 10$, and then intersect these two sets. These two sets are intervals in $\mathbb{Z}$: $[-1, \infty)$ and $(-\infty, -1]$, respectively. The intersection of these two sets gives the answer $u = \{-1\}$. This simplification is due to the fact that the spaces $\mathcal{A}$ and $\mathcal{I}$ are equipped with an order (total in this case), while the function f preserves such orders. This argument will be formalized in a general framework in this paper by using partial order theory. We next state the problem of determining a feedback system Σ_f that updates the lower and upper bounds of the set of possible current states and gives as output the lower and upper bounds of the set of allowed inputs.

Problem 1. (Dynamic Output Feedback on a Lattice) Given system $\Sigma = (\mathcal{A} \times \mathcal{Z}, \mathcal{I}, \mathcal{Y}, (f, h), g)$ with initial set $X_0 \subseteq S$, find a deterministic transition system $\Sigma_f = (\chi \times \chi, \mathcal{Y}, \tilde{\mathcal{I}} \times \tilde{\mathcal{I}}, (H_{21}, H_{22}), (H_{11}, H_{12}))$ with $H_{21} : \chi \times \chi \times \mathcal{Y} \to \chi$, $H_{22} : \chi \times \chi \times \mathcal{Y} \to \chi$, $H_{11} : \chi \times \chi \times \mathcal{Y} \to \tilde{\mathcal{I}}$, $H_{12} : \chi \times \chi \times \mathcal{Y} \to \tilde{\mathcal{I}}$, $(\chi, \leq)$ and $(\tilde{\mathcal{I}}, \leq)$ lattices, with $\mathcal{A} \subseteq \chi$ and $\mathcal{I} \subseteq \tilde{\mathcal{I}}$, such that if $u(k) \in [H_{11}(L(k), U(k), y(k)), H_{12}(L(k), U(k), y(k))] \cap \mathcal{I}$, $L(k+1) = H_{21}(L(k), U(k), y(k))$, $U(k+1) = H_{22}(L(k), U(k), y(k))$, $L(0), U(0) \in \chi$, and $\{y(k)\}_{k \geq 0} = g(\sigma)$, we have (i) $\sigma(k)(\alpha) \in [L(k), U(k)] \cap \mathcal{A}$ and (ii) $[L(k), U(k)] \cap \mathcal{A} \subseteq S$.

The variables $L(k)$ and $U(k)$ are the lower and the upper bounds in a partial order $(\chi, \leq)$ of the set of possible current states. The functions H_{11} and H_{12} determine the lower and upper bounds of the set of inputs that map the set $[L(k), U(k)] \cap \mathcal{A}$ inside S. In the next section, we determine the form of the functions $H_{11}, H_{12}, H_{21}, H_{22}$ that solve this problem.

4 Problem Solution

To solve Problem 1, we need to re-define the original system on the partial orders.

Definition 3. Consider the system $\Sigma = (\mathcal{A} \times \mathcal{Z}, \mathcal{I}, \mathcal{Y}, (f, h), g)$. An extension of Σ on partial orders $(\chi, \leq)$ and $(\tilde{\mathcal{I}}, \leq)$ with $\mathcal{A} \subseteq \chi$ and $\mathcal{I} \subseteq \tilde{\mathcal{I}}$ is given by a new system $\tilde{\Sigma} = (\chi \times \mathcal{Z}, \tilde{\mathcal{I}}, \mathcal{Y}, (\tilde{f}, \tilde{h}), \tilde{g})$, in which

(i) $(\tilde{\mathcal{I}}, \leq) = \bigcup_x (\tilde{\mathcal{I}}(x), \leq)$, where for all $x \in \chi$, $(\tilde{\mathcal{I}}(x), \leq)$ is a sublattice of $(\tilde{\mathcal{I}}, \leq)$ with $\mathcal{I} \subseteq \tilde{\mathcal{I}}(x)$ and with the sublattices $(\tilde{\mathcal{I}}(x), \leq)$ for all $x \in \chi$ compatible partial orders;
(ii) $\tilde{f}|_{\mathcal{A} \times \mathcal{Z} \times \mathcal{I}} = f$, $\tilde{h}|_{\mathcal{A} \times \mathcal{Z}} = h$, and $\tilde{g}|_{\mathcal{A} \times \mathcal{Z}} = g$.

Item (i) requires to have input set extensions allowed at different states in χ, which all contain the inputs in $\mathcal{I}$. Item (ii) requires that the extended system is equal to the original system when restricted to the original sets $\mathcal{A}$ and $\mathcal{I}$. In the sequel, we will denote by $\tilde{\Sigma}|_{\mathcal{I}}$ the system $\tilde{\Sigma}$ in which the input set is restricted to $\mathcal{I}$.

Definition 4. The pair $(\tilde{\Sigma}, (\chi, \leq))$ is said to be *output interval compatible* if

(i) for all $y \in \mathcal{Y}$, $O_y(\tilde{\Sigma})$ is an interval lattice, that is, $O_y(\tilde{\Sigma}) = [\wedge O_y(\tilde{\Sigma}), \vee O_y(\tilde{\Sigma})]$;
(ii) $\tilde{f} : ([\wedge O_y(\tilde{\Sigma}), \vee O_y(\tilde{\Sigma})], z, u) \to [\tilde{f}(\wedge O_y(\tilde{\Sigma}), z, u), \tilde{f}(\vee O_y(\tilde{\Sigma}), z, u)]$ is an order isomorphism for all $(z, u) \in \mathcal{Z} \times \mathcal{I}$.

If the pair $(\tilde{\Sigma}, (\chi, \leq))$ is output interval compatible, we can use the result of [4], in which a state estimator on a partial order that updates a lower bound L and an upper bound U of the set of all possible current states is given by the update laws

$$L(k+1) = \tilde{f}(L(k) \vee \bigwedge O_{y(k)}(\tilde{\Sigma}), z(k), u(k)) \tag{2}$$

$$U(k+1) = \tilde{f}(U(k) \wedge \bigvee O_{y(k)}(\tilde{\Sigma}), z(k), u(k)). \tag{3}$$

These update laws are such that $\sigma(k)(\alpha) \in [L(k), U(k)] \cap \mathcal{A}$. As a consequence, the functions H_{21} and H_{22} that solve item (i) of Problem 1 are given by equations (2) and (3), respectively. One contribution of this work is to determine also the functions H_{11} and H_{12} of Problem 1, which determine the dynamic feedback law. In order to proceed, we give the following definition.

Definition 5. The pair $(\tilde{\Sigma}, (\tilde{\mathcal{I}}, \leq))$ is *input interval compatible* if for all $x \in \chi$ and $z \in \mathcal{Z}$

(i) $\tilde{f} : (x, z, \tilde{\mathcal{I}}(x)) \to [\tilde{f}(x, z, \wedge\tilde{\mathcal{I}}(x)), \tilde{f}(x, z, \vee\tilde{\mathcal{I}}(x))]$ is order preserving and onto;
(ii) $\tilde{f}$: $(x, z, \tilde{\mathcal{I}}(x)) \to [\tilde{f}(x, z, \wedge\tilde{\mathcal{I}}(x)), \tilde{f}(x, z, \vee\tilde{\mathcal{I}}(x)]$ is either $\vee$-preserving or $\tilde{f}(x, z, \vee\tilde{\mathcal{I}}(x)) = \vee\tilde{S}$, and it is either $\wedge$-preserving or $\tilde{f}(x, z, \wedge\tilde{\mathcal{I}}(x)) = \wedge\tilde{S}$.

This definition establishes that $\tilde{f}$ preserves the order in the second argument. The $\vee$ ($\wedge$) preserving properties guarantee that the set of inputs that is mapped to the same point through $\tilde{f}$ is a lattice. The following theorem gives the expressions of the functions H_{11} and H_{12}. A pictorial interpretation of H_{11} and H_{12} is given in Figure 1. Denote $\tilde{f}^{-1}_{x,z}(w) := \{u \in \tilde{\mathcal{I}}(x) \mid f(x, z, u) = w\}$.

Theorem 1. *Let system $\Sigma = (\mathcal{A} \times \mathcal{Z}, \mathcal{I}, \mathcal{Y}, (f, h), g)$ be controllable by dynamic output feedback with respect to $S \subseteq \mathcal{A}$ and initial set $X_0 \subseteq S$. Let $(\chi, \leq)$ and $(\tilde{\mathcal{I}}, \leq)$ be such that $\mathcal{A} \subseteq \chi$ and $\mathcal{I} \subseteq \tilde{\mathcal{I}}$. Let $\tilde{S} \subseteq \chi$ be an interval lattice such that $\tilde{S} \cap \mathcal{A} = S$. Assume that the extension $\tilde{\Sigma} = (\chi \times \mathcal{Z}, \tilde{\mathcal{I}}, \mathcal{Y}, (\tilde{f}, \tilde{h}), \tilde{g})$ is such that*

(i) $\tilde{\Sigma}|_{\mathcal{I}}$ is controllable by dynamic output feedback with respect to $\tilde{S}$;
(ii) the pair $(\tilde{\Sigma}, (\chi, \leq))$ is output interval compatible;
(iii) the pair $(\tilde{\Sigma}, (\tilde{\mathcal{I}}, \leq))$ is input interval compatible.

Then, a solution to Problem 1, Σ_f, is given by functions H_{21} and H_{22} given by expressions (2) and (3), respectively, with $L(0) = \wedge\tilde{S}$, $U(0) = \vee\tilde{S}$, and

$$\begin{aligned} H_{11}(L(k), U(k), y(k)) &= \bigwedge \tilde{f}^{-1}_{L'(k),z(k)}\left(\tilde{f}(L'(k), z(k), \bigwedge\tilde{\mathcal{I}}(L'(k))) \vee \bigwedge\tilde{S}\right) \\ &\quad \vee \bigwedge \tilde{f}^{-1}_{U'(k),z(k)}\left(\tilde{f}(U'(k), z(k), \bigwedge\tilde{\mathcal{I}}(U'(k))) \vee \bigwedge\tilde{S}\right) \\ H_{12}(L(k), U(k), y(k)) &= \bigvee \tilde{f}^{-1}_{L'(k),z(k)}\left(\tilde{f}(L'(k), z(k), \bigvee\tilde{\mathcal{I}}(L'(k))) \wedge \bigvee\tilde{S}\right) \\ &\quad \wedge \bigvee \tilde{f}^{-1}_{U'(k),z(k)}\left(\tilde{f}(U'(k), z(k), \bigvee\tilde{\mathcal{I}}(U'(k))) \wedge \bigvee\tilde{S}\right), \end{aligned} \tag{4}$$

in which $L'(k) = L(k) \vee \wedge O_{y(k)}(\tilde{\Sigma})$, $U'(k) = U(k) \wedge \vee O_{y(k)}(\tilde{\Sigma})$.

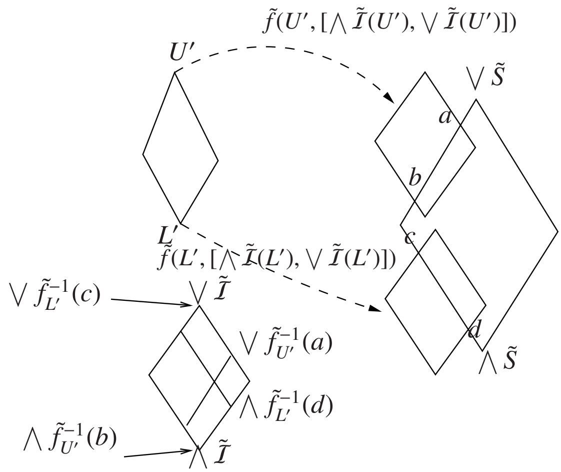

Fig. 1. Abstraction of Hasse diagrams to rhombi. In the picture, $a = \tilde{f}(U', \bigvee\tilde{\mathcal{I}}(U')) \wedge \bigvee\tilde{S}$, $b = \tilde{f}(U', \bigwedge\tilde{\mathcal{I}}(U')) \vee \bigwedge\tilde{S}$, $c = \tilde{f}(L', \bigvee\tilde{\mathcal{I}}(L')) \wedge \bigvee\tilde{S}$, $d = \tilde{f}(L', \bigwedge\tilde{\mathcal{I}}(L')) \vee \bigwedge\tilde{S}$, $H_{11} = \bigvee\tilde{f}_{U'}^{-1}(a)$, and $H_{12} = \bigwedge\tilde{f}_{L'}^{-1}(d)$. The dependencies on z and on k have been omitted.

5 Generality of the Partial Order Approach

We next show that if the system Σ is controllable by dynamic output feedback, the assumptions of Theorem 1 can be verified by suitable choices of $(\chi, \leq)$, $(\tilde{\mathcal{I}}, \leq)$, and $\tilde{\Sigma}$.

Theorem 2. *If system Σ is controllable by dynamic output feedback with respect to $S \subseteq \mathcal{A}$ and initial set $X_0 \subseteq S$, then there are partial orders $(\chi, \leq)$ and $(\tilde{\mathcal{I}}, \leq)$, an interval lattice $\tilde{S} \subseteq \chi$ with $\tilde{S} \cap \mathcal{A} = S$, and an extension $\tilde{\Sigma}$, such that $\tilde{\Sigma}|_{\mathcal{I}}$ is controllable by dynamic output feedback with respect to $\tilde{S}$, $(\tilde{\Sigma}, (\chi, \leq))$ is output interval compatible, and $(\tilde{\Sigma}, (\tilde{\mathcal{I}}, \leq))$ is input interval compatible.*

The assumption that Σ is controllable by dynamic output feedback with respect to S is needed to show that $\tilde{\Sigma}|_{\mathcal{I}}$ is also controllable by dynamic output feedback with respect to the interval lattice $\tilde{S}$. Such assumption is not needed to show output and input interval compatibility of $(\tilde{\Sigma}, (\chi, \leq))$ and of $(\tilde{\Sigma}, (\tilde{\mathcal{I}}, \leq))$, respectively. In case Σ is not controllable by dynamic output feedback with respect to S, $\tilde{\Sigma}|_{\mathcal{I}}$ will also not be controllable by dynamic output feedback with respect to any interval lattice $\tilde{S}$ such that $\tilde{S} \cap \mathcal{A} = S$ and for any choice of partial orders. This implies that $[H_{11}, H_{12}] \cap \mathcal{I}$ might be empty.

Example 1. The proof of Theorem 2 is constructive (see Appendix). We illustrate in this example how to construct the extended input partial order on a finite state/finite input system. Let $\mathcal{A} = \{\alpha_1, \alpha_2, \alpha_3\}$, $\mathcal{I} = \{u_1, u_2, u_3\}$, and let the update function F be given in the table of Figure 2. According to the proof of Theorem 2, we have $(\chi, \leq) = (\mathcal{P}(\mathcal{A}), \subseteq)$ and for all $x \in \chi$, we have $\tilde{\mathcal{I}}(x) = \mathcal{P}(\mathcal{I}) \cup I_x$, in which I_x contains "silent inputs" introduced to satisfy the onto property of item (i) of Definition 5. We start with $x = \alpha_1 \vee \alpha_2 \vee \alpha_3$. By computing $\tilde{f}(x, \tilde{u})$ (with $\tilde{f}(x, \tilde{u}) = f(x, \tilde{u})$ where $x \subseteq \mathcal{A}$ and $\tilde{u} \subseteq \mathcal{I}$ in the righthand side) for all $\tilde{u} \in \mathcal{P}(\mathcal{I})$, we note that $\tilde{f}(x, \tilde{u}) = \alpha_1 \vee \alpha_2 \vee \alpha_3$ for

$\tilde{u} \in \{u_1 \curlyvee u_2 \curlyvee u_3, u_1 \curlyvee u_2, u_2, u_2 \curlyvee u_3\}$, $\tilde{f}(x, \tilde{u}) = \alpha_1 \curlyvee \alpha_2$ for $\tilde{u} \in \{u_1, u_2 \curlyvee u_3\}$, $\tilde{f}(x, u_3) = \alpha_1$. As a consequence, the elements in χ that are less than $\tilde{f}(x, u_1 \curlyvee u_2 \curlyvee u_3)$ (where $u_1 \curlyvee u_2 \curlyvee u_3 = \bigvee \tilde{\mathcal{I}}(x)$) for which there is not an input in $\mathcal{P}(\mathcal{I})$ that map x to them are given by $\{\alpha_1 \curlyvee \alpha_2, \alpha_2, \alpha_2 \curlyvee \alpha_3, \alpha_3\}$. Thus, the set of silent inputs is $I_x = \{\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4\}$ such that $\tilde{f}(x, \epsilon_1) = \alpha_1 \curlyvee \alpha_2$, $\tilde{f}(x, \epsilon_2) = \alpha_2$, $\tilde{f}(x, \epsilon_3) = \alpha_2 \curlyvee \alpha_3$, $\tilde{f}(x, \epsilon_4) = \alpha_3$. We then establish the order among the elements in $I_x \cup \mathcal{P}(\mathcal{I})$ by following the procedure outlined in item 3 of the proof of Theorem 2 to guarantee the $\curlyvee$-preserving property of item (ii) of Definition 5. Since $\tilde{f}(x, \epsilon_1) \prec \alpha_1 \curlyvee \alpha_2 \curlyvee \alpha_3$ and $\sup_{\tilde{u} \in \mathcal{P}(\mathcal{I})}\{\tilde{u} \mid \tilde{f}(x, \tilde{u}) = \alpha_1 \curlyvee \alpha_2 \curlyvee \alpha_3\} = u_1 \curlyvee u_2 \curlyvee u_3$, we set $\epsilon_1 \prec u_1 \curlyvee u_2 \curlyvee u_3$. Also, $\tilde{f}(x, \epsilon_1) \succ \alpha_1$ and $\sup_{\tilde{u} \in \mathcal{P}(\mathcal{I})}\{\tilde{u} \mid \tilde{f}(x, \tilde{u}) = \alpha_1\} = u_3$. Then, we set $\epsilon_1 \succ u_3$. Finally, $\epsilon_1 \succ \epsilon_2$ because $\tilde{f}(x, \epsilon_1) \succ \tilde{f}(x, \epsilon_2)$. Proceeding in a similar way for all of the other silent inputs, we obtain the additional relations: $\epsilon_2 \prec \epsilon_3$, $\epsilon_4 \prec \epsilon_3$, $\epsilon_4 \prec u_1 \curlyvee u_3$, $\epsilon_3 \prec u_1 \curlyvee u_2 \curlyvee u_3$. The resulting extended input partial order $\tilde{\mathcal{I}}(x)$ is shown in the left plot of Figure 2. For $x = \alpha_2 \curlyvee \alpha_3$, the resulting $\tilde{\mathcal{I}}(x)$ is shown in the right plot of Figure 2.The reader can verify that when $x = \alpha_i$ for some i, $I_x = \emptyset$.

	α_1	α_2	α_3
u_1	α_1	α_3	α_1
u_2	α_2	α_1	α_3
u_3	α_1	α_1	α_1

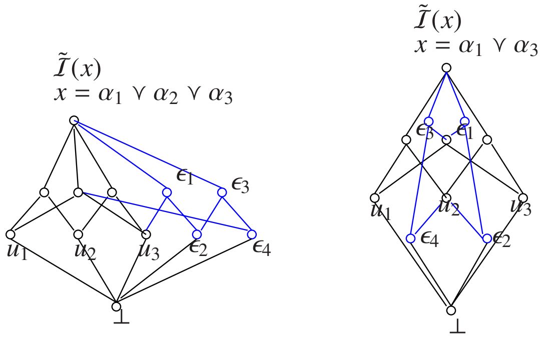

Fig. 2. Example 1. The table represents the update function $F(\alpha, u)$. The pictures at the center and at the right represent the extended input sets for $x = \alpha_1 \curlyvee \alpha_2 \curlyvee \alpha_3$ and $x = \alpha_1 \curlyvee \alpha_3$ with the associated partial orders, respectively. The blue elements are the silent inputs I_x.

Computational considerations. For a finite state-finite input system, the sizes of $\tilde{\mathcal{I}}$ and of χ are related to the computational load of the proposed algorithms as these partial order structures need to be computed and stored in memory. The size of these partial orders does not affect computation in those systems in which the partial orders have algebraic properties as we will see in Example 2 of the next section. The amounts of computation c needed for computing such partial orders can be estimated to be $c \leq K \sum_{i=1}^{|\mathcal{A}|^2} |X_i||\mathcal{I}||S|$, in which X_i are the sets on which the estimator evolves, $|\mathcal{A}|^2$ is the number of such sets, and $K > 0$. This amount of computation is comparable to the one obtained by using enumeration and exhaustive techniques.

In this section, we have shown that the partial order approach to dynamic feedback is general and that the worst case computation is proportional to the one of exhaustive searches. The partial orders constructed in the proof of Theorem 2 and in Example 1 are not unique and have mainly a theoretical relevance as they are impractical for implementation in systems with a large number of states and inputs. Thus, we propose in the next section a relaxed version of Theorem 1.

6 Relaxations and Application to a Multi-robot Example

Consider the case in which partial orders $(\chi, \leq)$ and $(\tilde{\mathcal{I}}, \leq)$ have been chosen and the assumptions of Theorem 1 do not all hold. Some possible relaxations of the basic assumptions of Theorem 1 are as follows:

(R1) the set $\tilde{S} \subseteq \chi$ such that $\tilde{S} \cap \mathcal{A} = S$ is given by $\tilde{S} = \bigcup_{i=1}^{M} \tilde{S}^i$, in which $\tilde{S}^i$ are intervals and $\tilde{S} \cap O_y(\tilde{\Sigma})$ is an interval;

(R2) $\tilde{f} : \chi \times \mathcal{Z} \times \mathcal{I} \rightarrow \chi$ is a piece-wise order isomorphism, that is, for all interval $[L, U] \subseteq \chi$, we have that there are disjoint intervals $[L^j, U^j]$ with $\bigcup_j [L^j, U^j] = [L, U]$ such that $\tilde{f}([L^j, U^j], z, u) \rightarrow [\tilde{f}(L^j, z, u), \tilde{f}(U^j, z, u)]$ is an order isomorphism for all j and any $u \in \mathcal{I}$;

(R3) for all interval $[L, U] \subseteq \tilde{S} \cap O_y(\tilde{\Sigma})$ there are a function $\tilde{f}' : \chi \times \tilde{\mathcal{I}} \rightarrow \chi$ with $\tilde{f}' : (x, [\bigwedge\tilde{\mathcal{I}}, \bigwedge\tilde{\mathcal{I}}]) \rightarrow [\tilde{f}'(x, \bigwedge\tilde{\mathcal{I}}), \tilde{f}'(x, \bigvee\tilde{\mathcal{I}})]$ an order isomorphism for all $x \in \chi$ and an order preserving map in the first argument, constants $L^* \leq U^* \in \chi$, and constants $L^S \leq U^S \in \chi$ such that $\{u \in \mathcal{I} \mid \tilde{f}([L, U], z, u) \subseteq \tilde{S}\} \supseteq \mathcal{I} \cap \{\tilde{u} \in \tilde{\mathcal{I}} \mid \tilde{f}'([L^*, U^*], \tilde{u}) \subseteq [L^S, U^S]\}$, with the righthand set not empty.

It is always possible to determine a set $\tilde{S}$ that is a union of intervals and any function can always be broken into order isomorphisms. The expressions of the functions H_{12} and H_{11} as given in formulas (4) stay the same, but one should substitute $\tilde{f}'$ in place of $\tilde{f}$, L^* and U^* in place of L' and U', and L^S and U^S in place of $\bigwedge\tilde{S}$ and $\bigvee\tilde{S}$. Due to the piecewise isomorphic nature of the function $\tilde{f}$, the update laws (2-3) become: $L(k+1) = \bigwedge_{\bar{L}^j \leq \bar{U}^j} \bar{L}^j$, $\bar{L}^j = \tilde{f}(L^j(k), z(k), u(k)) \vee \bigwedge O_{y(k+1)}(\tilde{\Sigma})$ and $U(k+1) = \bigvee_{\bar{L}^j \leq \bar{U}^j} \bar{U}^j$, $\bar{U}^j = \tilde{f}(U^j(k), z(k), u(k)) \vee \bigvee O_{y(k+1)}(\tilde{\Sigma})$, in which L^j, U^j establish the intervals where $\tilde{f}$ is an order isomorphism in the first argument, and $L(0) = \bigwedge O_{y(0)}(\tilde{\Sigma})$, $U(0) = \bigvee O_{y(0)}(\tilde{\Sigma})$.

Example 2. We consider a version of the "capture the flag" game for robots called "RoboFlag Drill" already considered in [4], in which now the attackers can use their estimates of the assignments of the opponents to decide the next action to take. Briefly, some number of robots with positions $(z_i, 0) \in \mathbb{R}^2$, which we refer to as blue robots, must defend their zone $\{(x, y) \in \mathbb{R}^2 \mid y \leq 0\}$ from an equal number of incoming robots, which we refer to as red robots. The positions of the red robots are $(x_i, y_i) \in \mathbb{R}^2$. The red robots move toward the blue defensive zone. The blue robots are assigned each to a red robot and they coordinate to intercept the red robots. In this work, we allow the red robots to swap their horizontal location with a nearby red robot as appropriate. Let N represent the number of robots in each team. The RoboFlag Drill system can be specified by the rules $y_i(k+1) = y_i(k) - \delta$ if $y_i(k) \geq \delta$,

$$z_i(k+1) = z_i(k) + \delta \text{ if } z_i(k) < x_{\alpha_i(k)}, \quad z_i(k+1) = z_i(k) - \delta \text{ if } z_i(k) > x_{\alpha_i(k)} \tag{5}$$

$$(\alpha_i(k+1), \alpha_{i+1}(k+1)) = (\alpha_{i+1}(k), \alpha_i(k)) \text{ if } x_{\alpha_i(k)} \geq z_{i+1}(k) \wedge x_{\alpha_{i+1}(k)} \leq z_{i+1}(k) \tag{6}$$

$$(x_i(k+1), x_{i+1}(k+1)) = (x_{i+1}(k), x_i(k)) \text{ if } swap_{i,i+1}(k). \tag{7}$$

The variable α_i is the red robot that blue robot i is required to intercept. Equation (6) establishes that two blue robots trade their assignments only when the current assignments cause them to go toward each other. Rule (7) allows two adjacent red robots to swap their horizontal position. If the red robots never swap horizontal position, the assignments of the blue robot reaches an equilibrium value in which no more conflicts among the assignments of the blue robots are present (the attackers have all been intercepted). In this work, we want to solve the following problem: *Given measurements $z(k)$ determine control inputs $swap_{i,i+1}(k)$ such that there are always at least two pairs of blue robots with conflicting assignments*. To formalize this problem, we translate the rules (5-6-7) to the form $\Sigma = (\mathcal{A} \times \mathcal{Z}, \mathcal{I}, \mathcal{Y}, (f, h), g)$. Thus, let $\{1, ..., N\}$ be the locations at which the red robots can reside, that is, the location denotes the order along the x direction at which the red robots are displaced. With abuse of notation, let x_i denote the x coordinate of location i and α_i the location to which blue robot i is assigned. We assume $x_i \leq z_i \leq x_{i+1}$ for all i and for all time. We set $\mathcal{A} = \text{perm}(N)$, $\mathcal{Z} = \mathbb{R}^N$, $\mathcal{I} = \{u \in \{-1, 0, 1\}^N \mid u_i = 1 \Leftrightarrow u_{i+1} = -1, u_N \neq 1, u_1 \neq -1\}$ ($u_i = 1$ iff $swap_{i,i+1}$ is true), $\mathcal{Y} = \mathbb{R}^N \times \mathbb{R}^N$. The functions are defined as follows: $f(\alpha, z, u) = G(F(\alpha, z), u)$, in which $F(\alpha, z)$ is represented by relations (6) and $G(\beta, u) = \beta'$, with $u_j = 1 \Rightarrow$ [(if $\beta_i = j \Rightarrow \beta'_i = j + 1$) and (if $\beta_i = j + 1 \Rightarrow \beta'_i = j$)]. The function $h(\alpha, z)$ is represented by relations (5). Let the entropy of the blue robots be defined by $E = \frac{1}{2} \sum_{i=1}^{N} |\alpha_i - i|$. In the absence of input to the system (i.e. $u(k) = 0$ for all k), E converges to zero. We define the set S as $S = \{\alpha \mid E \geq 2\}$, which can be computed and is given by $S = \{\alpha \mid \exists\, i, j,$ with $j > i + 1$ such that $\alpha_i \neq i$ and $\alpha_j \neq j\}$. If $\alpha \in S$, there are at least two pairs of blue robots with conflicting assignments.

To apply the dynamic control algorithm using the relaxations (R1-R2-R3), we need to determine the partial orders $(\chi, \leq)$ and $(\tilde{\mathcal{I}}, \leq)$, the set $\tilde{S}$ satisfying item (R1), the extended function $\tilde{f}$, and finally the function $\tilde{f}'$ with the intervals $[L^*, U^*]$ and $[L^S, U^S]$ as given in item (R3). Set $(\chi, \leq) = (\mathbb{N}^N, \leq)$ with order established component-wise. Given any set $X \subseteq \mathbb{N}^N$, we denote $[X]_j$ the projection along j of such set. Then, a set $\tilde{S} \subseteq \chi$ is given by $\tilde{S} = \bigcup_i \tilde{S}^i$, in which $\tilde{S}^i$ for each i are intervals of four types: (a) there are $l < j$ such that $[S^i]_l = [l + 1, N]$ and $[S^i]_j = [j + 1, N]$; (b) there are $l < j$ such that $[S^i]_l = [1, l - 1]$ and $[S^i]_j = [j + 1, N]$; (c) there are $l < j$ with $j > l + 1$ such that $[S^i]_l = [l + 1, N]$ and $[S^i]_j = [1, j - 1]$; (d) there are $l < j$ such that $[S^i]_l = [1, l - 1]$ and $[S^i]_j = [1, j - 1]$. This can be checked by recalling that $\alpha \in \bigcup_i \tilde{S}_i$ if $\alpha \in \tilde{S}^i$ for some i. Define the extension $\tilde{F} : \chi \times \mathcal{Z} \to \chi$ as F with now $\alpha \in \mathbb{N}^N$. Clearly, $\tilde{F}|_{\mathcal{A} \times \mathcal{Z}} = F$. Also, we define $\tilde{h} : \chi \times \mathcal{Z} \to \mathcal{Z}$ as h with $\alpha \in \mathbb{N}^N$, for which $\tilde{h}|_{\mathcal{A} \times \mathcal{Z}} = h$. One can check that the output set is an interval and that the function $\tilde{F}$ is an order isomorphism on the output set. The function $\tilde{G} : \chi \times \mathcal{I} \to \chi$ is defined as G in which the first argument belongs to $\mathbb{N}^N$. Then $\tilde{f} = \tilde{G} \circ \tilde{F}$, in which one can check that $\tilde{f}_{\mathcal{A} \times \mathcal{Z} \times \mathcal{I}} = f$. We thus have defined the extended system $\tilde{\Sigma} = (\chi \times \mathcal{Z}, \mathcal{Y}, \mathcal{I}, (\tilde{f}, \tilde{h}), \tilde{g})$. For the input set, we consider $\tilde{\mathcal{I}} = \{-1, 0, 1\}^N$ with order established componentwise. It is easy to show that the extended system $\tilde{\Sigma}|_{\mathcal{I}} = (\chi \times \mathcal{Z}, \mathcal{I}, \mathcal{Y}, (\tilde{f}, \tilde{h}), \tilde{g})$ satisfies the dynamic controllability condition with respect to $\tilde{S}$ if $N > 4$. We are left to determine the function $\tilde{f}'$ with the intervals $[L^*, U^*]$ and $[L^S, U^S]$ as given in item (R3). For all $x = (x_1, ..., x_N) \in \chi$ and $\tilde{u} = (\tilde{u}_1, ..., \tilde{u}_N) \in \tilde{\mathcal{I}}$, we set $\tilde{f}'(x, \tilde{u}) = (\tilde{f}'_1(x_1, \tilde{u}_1), ..., \tilde{f}'_N(x_N, \tilde{u}_N))$, in which $\tilde{f}'(x_i, \tilde{u}_i) := x_i + \tilde{u}_i$. The following algorithm, computes the sets $[L^*, U^*]$ and $[L^S, U^S]$ component-wise for all intervals $[L, U] \subseteq \tilde{S} \cap O_y(\tilde{\Sigma})$. Let $P' = \tilde{F}([L, U], z)$

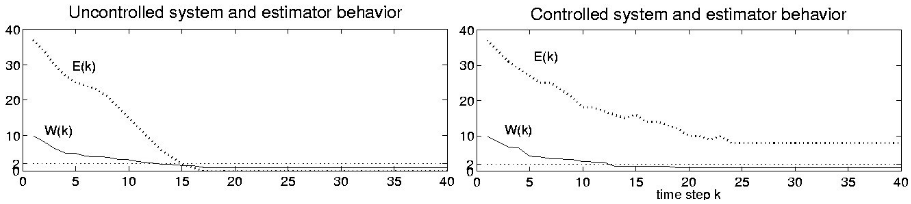

Fig. 3. Convergence plots of the estimator and of the entropy with $N = 15$ and $\alpha(0) = (4, 8, 9, 2, 13, 15, 6, 5, 12, 10, 1, 14, 3, 7, 11)$. For the controlled system $E > 2$ always. From the uncontrolled system plot, one realizes that a strategy that estimates the state first and then computes the controller once the estimator has converged does not work.

Algorithm

Initialize $flag_i = 0$, $L_i^* = U_i^* = i$, $L_i^S = 1$, and $U_i^S = N$ for all i
For $i = 1 : N$
  If $\min(P_i') = i$ and $flag_{i-1} = 0 \Longrightarrow L_i^S = 1,\ U_i^S = i - 1$ and $flag_i = 1$
End
For $i = 2 : N$
  If $\max(P_{i-1}') = i - 1$ and $flag_{i-1} = 0 \Longrightarrow L_{i-1}^S = i,\ U_{i-1}^S = N$ and $flag_i = 1$
End
For $i = 1 : N$
  If $\min(P_i') \geq i + 1$ and $flag_{i+1} = 0 \Longrightarrow L_i^* = U_i^* = i + 1,\ L_i^S = i + 1,\ U_i^S = N$
  If $\max(P_i') \leq i - 1$ and $flag_i = 0 \Longrightarrow L_i^* = U_i^* = i - 1,\ L_i^S = 1\ U_i^S = i - 1$
End.

The idea behind this algorithm is as follows. Say that $[P']_i = [i, N]$ and that we want to remove i from it by swapping red robot i with red robot $i-1$. This can be done by asking that $i + \tilde{u}_i \in [1, i - 1]$, which gives $\tilde{u}_i \leq -1$. Finally, note that the function $\tilde{f}$ is a composition of a function $\tilde{F}$, which is an order isomorphism, and a function $\tilde{G}$, which is a piecewise order isomorphism. To see this, let $\tilde{u}_i = -1$ and $P_i' = [i, N]$ for example, then we can re-write $[i, N] = [i, i] \cup [i + 1, N]$ so that $\tilde{G}_i : ([i, i], -1) \rightarrow [\tilde{G}_i(i, -1), \tilde{G}_i(i, 1)] = [i - 1, i - 1]$ and $\tilde{G}_i : ([i + 1, N], -1) \rightarrow [\tilde{G}_i(i + 1, -1), \tilde{G}_i(N, -1)] = [i + 1, N]$ are order isomorphisms. Figure 3 shows the behavior of the estimator error (given by $W(k) = 1/N \sum_{i=1}^{N} |m_i(k)|$, in which $m_i(k)$ is the coordinate set $[L_i(k), U_i(k)]$ minus all the singletons that occur at other coordinates) and of the entropy $E(k) = 1/2 \sum_{i=1}^{N} |\alpha_i(k) - i|$. In this example, the computation requirement for the implementation of the dynamic controller is proportional to N (number of variables to control and estimate). If we had not used any structure, we would have had a number of computations at least of the order of $(N!)^2$ as the size of the output set and of the set S are both of the order of $N!$. Note also that this simplification is not due to the fact that the dynamics decouples as it is heavily coupled between the robots.

7 Conclusions

We have proposed a partial order approach to dynamic feedback for the discrete variables of a hybrid system, which relies on partial order theory to compute only suitable lower and upper bounds to determine the dynamic controller. We have shown that such an approach is general as it can be applied to any system that is controllable by dynamic output feedback. The worst case computation load of the proposed approach does not exceed the one of exhaustive searches under partial observations. The main computational advantage is obtained when one can choose suitable partial orders in which the computation of joins and meets is efficiently performed. A multi-robot example showed this point. The next step is to consider the dynamic feedback problem also for the continuous variables and establish system structures that allow efficient choices of partial orders. As we mentioned, this work was not concerned with analysis problems: these are left to our future work, in which we would like to determine efficient computations of escape tubes and controlled invariance kernels by the computation of suitable lower and upper bounds, only. Finally, we plan to consider in our future work uncertainty in the system dynamics by modeling it as nondeterminism. On the application side, we will extend these results to the design of safety controllers under partial observation in the context of intelligent transportation systems.

References

1. E. Asarin, O. Maler, and A. Pnueli. Symbolic controller design for discrete and timed systems. *Hybrid Systems: Computation and Control,* Lecture Notes in Computer Science, volume 999, P. Antsakilis, W. Kohn, A. Nerode, and S. Sastry, Eds. Springer Verlag, pages 1–20, 1995.
2. J. Aubin, J. Lygeros, M. Quincampoix, S. Sastry, and N. Seube. Impulse differential inclusions: A viability approach to hybrid systems. *IEEE Transactions on Automatic Control*, 47(1):2–20, 2002.
3. B. A. Davey and H. A. Priesteley. *Introduction to Lattices and Order*. Cambridge University Press, 2002.
4. D. DelVecchio, R. M. Murray, and E. Klavins. Discrete state estimators for systems on a lattice. *Automatica*, 42(2):271–285, 2006.
5. A. Deshpande and P. Varaiya. Viable control of hybrid systems. In *Hybrid Systems II,* Lecture Notes in Computer Science, volume 999, P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, Eds. Springer Verlag, pages 128–147, 1995.
6. P. E.Caines and S. Wang. Classical and logic based regulator design and its complexity for partially observed automata. In *Conf. on Decision and Control*, pages 132–137, 1989.
7. P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–98, 1989.
8. K. Rohloff and S. Lafortune. On the synthesis of safe control policies in decentralized control of discrete event systems. *IEEE Transactions on Automatic Control*, 48(6):1064–1068, 2003.
9. E. D. Sontag. *Mathematical Control Theory*. Springer, 1998.
10. W. Thomas. On the synthesis of strategies in infinite games. In *Proceedings of the STACS 95,* E. W. Mayr and C. Puech, Eds. Springer Verlag, pages 1–13, 1995.
11. H. Wong-Toi. The synthesis of controllers for linear hybrid automata. In *Conf. on Decision and Control*, pages 4607–4613, 1997.

Appendix I. Partial Order Theory

In this section, we introduce the main notation and definitions about partial orders that will be used in this work. For a complete overview, the reader is referred to [3]. A partial order is a set χ with a partial order relation "$\leq$", and we denote it by the pair $(\chi, \leq)$. for all $x, w \in \chi$, the $\sup\{x, w\}$ is the smallest element that is larger than both x and w. In a similar way, the $\inf\{x, w\}$ is the largest element that is smaller than both x and w. We define the *join* "$\vee$" and the *meet* "$\wedge$" of two elements x and w in χ as $x \vee w = \sup\{x, w\}$ and $x \wedge w = \inf\{x, w\}$. If $S \subseteq \chi$, $\bigvee S = \sup\ S$ and $\bigwedge S = \inf\ S$. If $x \wedge w \in \chi$ and $x \vee w \in \chi$ for all $x, w \in \chi$, then $(\chi, \leq)$ is a *lattice*. Let $(\chi, \leq)$ be a lattice and let $S \subseteq \chi$ be a non-empty subset of χ. Then, $(S, \leq)$ is a *sublattice* of χ if $a, b \in S$ implies that $a \vee b \in S$ and $a \wedge b \in S$. Any interval sublattice of $(\chi, \leq)$ is given by $[L, U] = \{w \in \chi \mid L \leq w \leq U\}$ for $L, U \in \chi$. That is, this special sublattice can be represented by only two elements. for all set S, we denote by $\mathcal{P}(S)$ the set of all subsets of S. On $\mathcal{P}(S)$, it is possible to establish a partial order relation determined by the inclusion relation. Therefore, $(\mathcal{P}(S), \leq)$ with "$\leq$" established by the inclusion relation is a lattice. Let $(\chi, \leq)$ be a partial order and let $x, w \in \chi$. We will use the notation $x \prec w$ to say that $x \leq w$ and there is not an element that is larger than x and smaller than w; we will use the notation $x \succ w$ to say that $x \geq w$ and there is not an element that is larger than x and smaller than w. for all $w, x \in \chi$, we denote $w \parallel x$ if they are not related by the order relation. Let $(P, \leq_P)$ and $(Q, \leq_Q)$ be two partial orders and let $X = P \cap Q$. They are said to be *compatible partial orders* if for all pair $x_1, x_2 \in X$ we have that $x_1 \leq_P x_2$ if and only if $x_1 \leq_Q x_2$. Let $(P, \leq_P)$ and $(Q, \leq_Q)$ be two compatible partial orders. Then, the union of the two partial orders, denoted $(P, \leq_P) \cup (Q, \leq_Q)$ is the new partial order $(R, \leq)$, in which $R = P \cup Q$ and for all $x_1, x_2 \in R$ we have that $x_1 \leq x_2$ if and only if $x_1 \leq_Q x_2$ or $x_1 \leq_P x_2$. In the sequel, when we will have two compatible partial orders, we will omit the subscript of "$\leq$" as there will be no ambiguity on the partial order relation between any two elements. We now consider maps on partial orders. Let $(P, \leq)$ and $(Q, \leq)$ be partially ordered sets. A map $f : P \to Q$ is (i) an *order preserving map* if $x \leq w \implies f(x) \leq f(w)$; (ii) an *order embedding* if $x \leq w \iff f(x) \leq f(w)$; (iii) an *order isomorphism* if it is order embedding and it maps P *onto* Q. The map $f : P \to Q$ is said to be $\vee$-preserving if for all $x, w \in P$, we have that $f(x \vee w) = f(x) \vee f(w)$. It is said to be $\wedge$-preserving if for all $x, w \in P$, we have that $f(x \wedge w) = f(x) \wedge f(w)$. One can show that if f is order preserving, then for all $x, y \in P$, we have that $f(x) \vee f(w) \leq f(x \vee w)$ and $f(x) \wedge f(w) \geq f(x \wedge w)$.

Appendix II. Proof of Theorems and Propositions

Proof. (Proof of Proposition 1) ($\Leftarrow$)Choose functions H_1 and H_2 as $H_2(X(k), y(k)) = F(X(k) \cap O_{y(k)}(\Sigma), u(k))$, $u(k) \in H_1(X(k), y(k))$ with $H_1(X(k), y(k)) = \{u \in \mathcal{I} \mid F(X(k) \cap O_{y(k)}(\Sigma), u) \subset S\}$ and $X(0) = X_0$. We show that the set $H_1(X(k), y(k))$ is not empty for all k and that properties (i) and (ii) of Definition 2 are satisfied. We proceed by induction argument on the step k. (Base case) By assumption, $X(0) \subseteq S$ and $s(0) \in X(0)$. As a consequence, $\{u \in \mathcal{I} \mid F(X(0) \cap O_{y(0)}(\Sigma), u) \subseteq S\}$ is not empty. (Induction step) Assume $X(k) \subseteq S$ and $s(k) \in X(k)$, then $H_1(X(k), y(k)) = \{u \in \mathcal{I} \mid F(X(k) \cap O_{y(k)}(\Sigma), u) \subseteq S\} \neq \emptyset$ because $\{u \in \mathcal{I} \mid F(X(k) \cap O_{y(k)}(\Sigma), u) \subseteq S\} \supseteq \{u \in \mathcal{I} \mid F(S \cap O_{y(k)}(\Sigma), u) \subseteq S\}$ and the latter set is nonempty by assumption. Thus, if $u(k) \in H_1(X(k), y(k))$ we have by construction that $X(k+1) \subseteq S$. Also, since $s(k) \in X(k)$ and $s(k) \in O_{y(k)}(\Sigma)$, we have that $s(k+1) \in X(k+1)$.

($\Rightarrow$) Assume that $\{u \in \mathcal{I} \mid F(O_y(\Sigma) \cap S, u) \subseteq S\} = \emptyset$ for some y. Let $s(0) \in S$ be such that $y(0) = g(s(0))$ and $\{u \in \mathcal{I} \mid F(O_{y(0)}(\Sigma) \cap S, u) \subseteq S\} = \emptyset$. Thus $\{u \in \mathcal{I} \mid F(X_0 \cap O_{y(0)}(\Sigma), u) \subseteq S\} = \emptyset$. Assume that the system is controllable by dynamic output feedback with respect to $X_0 \subseteq S$. Then, there are functions $H_1 : \mathcal{P}(Q) \times \mathcal{Y} \longrightarrow \mathcal{P}(\mathcal{I})$ and $H_2 : \mathcal{P}(Q) \times \mathcal{Y} \times \mathcal{I} \longrightarrow \mathcal{P}(Q)$ such that $X(1) = H_2(X_0, y(0)) \subseteq S$, $s(1) \in X(1)$ and $u(0) \in H_1(X_0, y(0))$. For guaranteeing $s(1) \in X(1)$ with $s(1) = F(s(0), u(0))$ and $s(0) \in X_0 \cap O_{y(0)}(\Sigma)$, we need that $F(X_0 \cap O_{y(0)}(\Sigma), u(0)) \subseteq X(1)$. However, $F(X_0 \cap O_{y(0)}(\Sigma), u(0)) \nsubseteq S$ and $X(1) \subseteq S$. This leads to a contradiction.

Proof. (Proof of Theorem 1) The dependencies on z are neglected. Equations (2-3) imply that $\alpha(k) \in [L(k), U(k)] \cap \mathcal{A}$. Thus, property (i) of Problem 1 is true. We next show that

(a) $\{u \in \mathcal{I} \mid \tilde{f}([L'(k), U'(k)], u) \subseteq [\wedge\tilde{S}, \vee\tilde{S}]\} = \mathcal{I} \cap [H_{11}(L(k), U(k), y(k)), H_{12}(L(k), U(k), y(k))]$;

(b) $[H_{11}(L(k), U(k), y(k)), H_{12}(L(k), U(k), y(k))] \cap \mathcal{I}$ is not empty.

Proof of (a). Since $[L'(k), U'(k)] \subseteq O_{y(k)}(\tilde{\Sigma})$, the function $\tilde{f}$ preserves the ordering in the first argument. As a consequence, we have that $\{u \in \mathcal{I} \mid \tilde{f}([L'(k), U'(k)], u) \subseteq [\wedge\tilde{S}, \vee\tilde{S}]\} = \{u \in \mathcal{I} \mid \wedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \vee\tilde{S}\} \cap \{u \in \mathcal{I} \mid \wedge\tilde{S} \leq \tilde{f}(U'(k), u) \leq \vee\tilde{S}\}$. Also, we have that $\{u \in \mathcal{I} \mid \wedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \vee\tilde{S}\} = \mathcal{I} \cap \{u \in \tilde{\mathcal{I}}(L'(k)) \mid \wedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \vee\tilde{S}\}$ and that $\{u \in \mathcal{I} \mid \wedge\tilde{S} \leq \tilde{f}(U'(k), u) \leq \vee\tilde{S}\} = \mathcal{I} \cap \{u \in \tilde{\mathcal{I}}(U'(k)) \mid \wedge\tilde{S} \leq \tilde{f}(U'(k), u) \leq \vee\tilde{S}\}$. As a consequence, we have that

$$\begin{aligned} &\{u \in \mathcal{I} \mid \tilde{f}([L'(k), U'(k)], u) \subseteq [\bigwedge\tilde{S}, \bigvee\tilde{S}]\} = \\ &\mathcal{I} \cap \{u \in \tilde{\mathcal{I}}(L'(k)) \mid \bigwedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \bigvee\tilde{S}\} \cap \\ &\{u \in \tilde{\mathcal{I}}(U'(k)) \mid \bigwedge\tilde{S} \leq \tilde{f}(U'(k), u) \leq \bigvee\tilde{S}\}. \end{aligned} \tag{8}$$

One can readily verify that $\{u \in \tilde{\mathcal{I}}(L'(k)) \mid \wedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \vee\tilde{S}\} = \tilde{f}^{-1}_{L'(k)}((\tilde{f}(L'(k), \tilde{\mathcal{I}}(L'(k))) \cap [\wedge\tilde{S}, \vee\tilde{S}])$, which derives from the definition of $\tilde{f}^{-1}_{L'(k)}$. By the onto property in item (i) of Definition 5, we also have that $\tilde{f}(L'(k), \tilde{\mathcal{I}}(L'(k))) = [\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))), \tilde{f}(L'(k), \vee\tilde{\mathcal{I}}(L'(k)))]$. As a consequence, we obtain that $\{u \in \tilde{\mathcal{I}}(L'(k)) \mid \wedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \vee\tilde{S}\} = \tilde{f}^{-1}_{L'(k)}([\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S}, \tilde{f}(L'(k), \vee\tilde{\mathcal{I}}(L'(k))) \wedge \vee\tilde{S}])$. We are thus left to show that $\tilde{f}^{-1}_{L'(k)}([\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S}, \tilde{f}(L'(k), \vee\tilde{\mathcal{I}}(L'(k))) \wedge \vee\tilde{S}]) = [\wedge\tilde{f}^{-1}_{L'(k)}(\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S}), \vee\tilde{f}^{-1}_{L'(k)}(\tilde{f}(L'(k), \vee\tilde{\mathcal{I}}(L'(k))) \wedge \vee\tilde{S})]$. To show this, we show that any element of the first set belongs to the second and *viceversa*. Any element of the second set is also an element of the first set due to the order preserving property of $\tilde{f}$ in the second argument as established in item (i) of Definition 5. Assume now that u is in the first set, then $\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \tilde{f}(L'(k), \vee\tilde{\mathcal{I}}(L'(k))) \wedge \vee\tilde{S}$. We next show that $w \leq u$ in which $w = \wedge\tilde{f}^{-1}_{L'(k)}\left(\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S}\right)$. If $\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) = \wedge\tilde{S}$, we have that $\wedge\tilde{f}^{-1}_{L'(k)}\left(\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S}\right) = \wedge\tilde{\mathcal{I}}(L'(k))$ and therefore we have that $\wedge\tilde{f}^{-1}_{L'(k)}(\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \vee \wedge\tilde{S}) \leq u$. If instead $\tilde{f}(L'(k), \wedge\tilde{\mathcal{I}}(L'(k))) \neq \wedge\tilde{S}$, by item (ii) of Definition 5, we have that $\tilde{f}$ is $\wedge$-preserving in the second argument. Since $w \leq \tilde{f}(L'(k), u)$, it must be that either $\wedge\tilde{f}^{-1}(w) \leq u$ or $\wedge\tilde{f}^{-1}(w) \parallel u$ by the order preserving property of $\tilde{f}$ in the second argument. Let us show that $\wedge\tilde{f}^{-1}(w) \parallel u$ is not possible. By the $\wedge$-preserving property, we have that $\tilde{f}(\wedge\tilde{f}^{-1}_{L'(k)}(w) \wedge u) = w \wedge$

$\tilde{f}(L'(k), u)$. Since $w \leq \tilde{f}(L'(k), u)$, we have that $w \wedge \tilde{f}(L'(k), u) = w$, which in turn implies that $\tilde{f}(L'(k), \wedge\tilde{f}^{-1}(w) \wedge u) = w$. By the definition of $\wedge\tilde{f}^{-1}_{L'(k)}(w)$, it follows that we must have $\wedge\tilde{f}^{-1}_{L'(k)}(w) \leq u$. One can proceed in a similar way to show that $u \leq \vee\tilde{f}^{-1}_{L'(k)}\left(\tilde{f}(L'(k), \vee\tilde{\mathcal{I}}(L'(k))) \wedge \vee\tilde{S}\right)$. As a consequence, we have concluded that

$$\begin{aligned} \{u \in \tilde{\mathcal{I}}(L'(k)) \mid \bigwedge\tilde{S} \leq \tilde{f}(L'(k), u) \leq \bigvee\tilde{S}\} = \\ [\bigwedge\tilde{f}^{-1}_{L'(k)}\left(\tilde{f}(L'(k), \bigwedge\tilde{\mathcal{I}}(L'(k))) \vee \bigwedge\tilde{S}\right), \bigvee\tilde{f}^{-1}_{L'(k)}\left(\tilde{f}(L'(k), \bigvee\tilde{\mathcal{I}}(L'(k))) \wedge \bigvee\tilde{S}\right)]. \end{aligned} \tag{9}$$

Similar reasonings can be used to show that equation (9) holds for $U'(k)$. Equations (8), (9) and (9) with $U'(k)$ in place of $L'(k)$ prove (a). Given (a), to show (b) one can show that $\{u \in \mathcal{I} \mid \tilde{f}([L'(k), U'(k)], u) \subseteq [\wedge\tilde{S}, \vee\tilde{S}]\}$ is not empty. This is true if $[L'(k), U'(k)] \subseteq O_y(\tilde{\Sigma}) \cap \tilde{S}$ as by assumption $\tilde{\Sigma}|_{\mathcal{I}}$ is controllable by dynamic output feedback with respect to $\tilde{S}$. We can show that $[L'(k), U'(k)] \subseteq O_{y(k)}(\tilde{\Sigma}) \cap \tilde{S}$ by induction on the step k. In fact, $[L'(0), U'(0)] \subseteq O_y(\tilde{\Sigma}) \cap \tilde{S}$ as $L(0) = \wedge\tilde{S}$ and $U(0) = \vee\tilde{S}$. Assume that $[L'(k), U'(k)] \subseteq O_{y(k)}(\tilde{\Sigma}) \cap \tilde{S}$, let us show that also $[L'(k+1), U'(k+1)] \subseteq O_{y(k+1)}(\tilde{\Sigma}) \cap \tilde{S}$. Since $[L'(k), U'(k)] \subseteq O_{y(k)}(\tilde{\Sigma}) \cap \tilde{S}$, we have that $\mathcal{I} \cap [H_{11}(L(k), U(k), y(k)), H_{12}(L(k), U(k), y(k))]$ is not empty. We thus can take $u(k) \in \mathcal{I} \cap [H_{11}(L(k), U(k), y(k)), H_{12}(L(k), U(k), y(k))]$ and apply it to the system. By construction of $H_{11}(L(k), U(k), y(k))$ and $H_{12}(L(k), U(k), y(k))$, we have that $[L(k+1), U(k+1)] \subseteq \tilde{S}$. Thus, $[L'(k+1), U'(k+1)] \subseteq \tilde{S} \cap O_{y(k+1)}(\tilde{\Sigma})$. Therefore, (b) is shown.

Proof. (Proof of Theorem 2) We determine a system extension $\tilde{\Sigma}$ and we show that the properties of Definition 5 are satisfied.

1. Define $\chi = \mathcal{P}(\mathcal{A})$ and $(\chi, \leq) = (\mathcal{P}(\mathcal{A}), \subseteq)$. The bottom element is $\perp_\chi = \emptyset$. Let $x \in \chi$ be given by $x = \alpha_1 \vee ... \vee \alpha_n$ with $\alpha_i \in \mathcal{A}$, for all $u \in \mathcal{I}$ we define the function $\tilde{f} : \chi \times \mathcal{I} \to \chi$ as $\tilde{f}(x, u) = f(\alpha_1, u) \vee ... \vee f(\alpha_n, u)$ for $u \in \mathcal{I}$. Output interval compatibility of the pair $(\tilde{\Sigma}, (\chi, \leq))$ follows immediately.

2. for all $x \in \chi$, the extended input set $\tilde{\mathcal{I}}(x)$ is defined as $\mathcal{P}(\mathcal{I}) \bigcup I_x$, in which the order among the elements in $\mathcal{P}(\mathcal{I})$ is established according to inclusion relation, and the sets I_x for all x are called the sets of *silent inputs* and are defined as follows. for all $\tilde{u} \in \mathcal{P}(\mathcal{I})$, we have $\tilde{u} = u_1 \vee ... \vee u_p$ for some $u_i \in \mathcal{I}$. Then, we define $\tilde{f}(x, \tilde{u}) = \tilde{f}(x, u_1) \vee ... \vee \tilde{f}(x, u_p)$. Let us initialize $I_x = \emptyset$ and let $\mathcal{I} = \{u_1, ..., u_m\}$. for all $w \in \chi$ such that $w \leq \tilde{f}(x, u_1 \vee ... \vee u_m)$, if there is not a $\tilde{u} \in \mathcal{P}(\mathcal{I})$ such that $\tilde{f}(x, \tilde{u}) = w$, define a silent input ϵ such that $\tilde{f}(x, \epsilon) = w$. Thus, we add such silent input to I_x, that is, $I_x = I_x \cup \epsilon$.

3. We next establish the order among the silent inputs and the inputs in $\mathcal{P}(\mathcal{I})$. for all $\epsilon \in I_x$, let $w = \tilde{f}(x, \epsilon)$. By construction, $\tilde{f}(x, \epsilon) \leq \tilde{f}(x, u_1 \vee ... \vee u_m)$. Let $\{w_1, ..., w_k\}$ be the set of elements with $w_i \leq \tilde{f}(x, u_1 \vee ... \vee u_m)$ such that either $w_i \prec \tilde{f}(x, \epsilon)$ or $w_i \succ \tilde{f}(x, \epsilon_i)$. Let $\tilde{u}_i \in \tilde{\mathcal{I}}(x)$ be such that $\tilde{f}(x, \tilde{u}_i) = w_i$. If $\tilde{u}_i \in I_x$ then set $\tilde{u}_i \succ \epsilon$ if and only if $w_i \succ \tilde{f}(x, \epsilon_i)$ and $\tilde{u}_i \prec \epsilon$ if and only if $w_i \prec \tilde{f}(x, \epsilon_i)$. If $\tilde{u}_i \in \mathcal{P}(\mathcal{I})$, there may be several such inputs so that $\tilde{f}(x, \tilde{u}_i) = w_i$. Let $\tilde{u}_i$ be the greatest of such inputs, that is, $\tilde{u}_i = \sup_{(\mathcal{P}(\mathcal{I}), \leq)}\{\tilde{u} \mid \tilde{f}(x, \tilde{u}) = w_i\}$. (By the way $\tilde{f}$ has been defined on elements of $\mathcal{P}(\mathcal{I})$ it follows that $\tilde{f}(x, \tilde{u}_i) = w_i$.) Then, we set $\tilde{u}_i \succ \epsilon$ if and only if $w_i \succ \tilde{f}(x, \epsilon_i)$ and $\tilde{u}_i \prec \epsilon$ if and only if $w_i \prec \tilde{f}(x, \epsilon_i)$. Let $\perp_{\mathcal{I}} = \wedge\tilde{\mathcal{I}}(x)$ such that every element that does not have a lower element is strictly greater than it. We also define $\tilde{f}(x, \perp_{\mathcal{I}}) = \perp_\chi$. Note that by construction, the top element of $\tilde{\mathcal{I}}(x)$ is given by $u_1 \vee ... \vee u_m = \vee\tilde{\mathcal{I}}(x)$. By construction, $(\tilde{\mathcal{I}}(x), \leq)$ are compatible partial orders.

From item 2., it follows that $\tilde{f} : (x, \tilde{\mathcal{I}}(x)) \to [\tilde{f}(x, \bigwedge\tilde{\mathcal{I}}(x)), \tilde{f}(x, \bigvee\tilde{\mathcal{I}}(x))]$ is onto. To show that it is also order preserving, we show that for all $\tilde{u}_1 \leq \tilde{u}_2$ in $\tilde{\mathcal{I}}(x)$ also $\tilde{f}(x, \tilde{u}_1) \leq \tilde{f}(x, \tilde{u}_2)$. Let $\tilde{u}_{1,1} \prec \tilde{u}_{1,2} \prec \ldots \prec \tilde{u}_{1,k}$ be the chain between $\tilde{u}_{1,1} = \tilde{u}_1$ and $\tilde{u}_{1,k} = \tilde{u}_2$. Consider any consecutive pair $\tilde{u}_{1,i} \prec \tilde{u}_{1,i+1}$. Then if $\tilde{u}_{1,i}, \tilde{u}_{1,i+1} \in \mathcal{P}(\mathcal{I})$, by the definition of $\tilde{f}$ on elements in $\mathcal{P}(\mathcal{I})$ given in item 1., we have $\tilde{f}(x, \tilde{u}_{1,i}) \leq \tilde{f}(x, \tilde{u}_{1,i+1})$. If either one of $\tilde{u}_{1,i}, \tilde{u}_{1,i+1}$ is in I_x (that is, it is a silent input), by the definition of the order in item 3., we have that $\tilde{u}_{1,i} \prec \tilde{u}_{1,i+1}$ if and only if $\tilde{f}(x, \tilde{u}_{1,i}) \prec \tilde{f}(x, \tilde{u}_{1,i+1})$. Since, this holds for all consecutive pair $(\tilde{u}_{1,i}, \tilde{u}_{1,i+1})$, we thus have that $\tilde{f}(x, \tilde{u}_1) \leq \tilde{f}(x, \tilde{u}_2)$.

To show property (ii) of Definition 5, note that $\tilde{S} = [\bot_\chi, \bigvee\tilde{S}]$ for $\bigvee\tilde{S} = \mathcal{P}(S) \in \chi$. As a consequence, we have that $\tilde{f}(x, \bigwedge\mathcal{I}(x)) = \bigwedge S$. Thus, we are left to show that for all $x \in \chi$, $\tilde{f}(x, \cdot)$ is $\vee$-preserving in the second argument when the second argument is ranging in $\tilde{\mathcal{I}}(x)$. Let $\tilde{u}_1, \tilde{u}_2 \in \tilde{\mathcal{I}}(x)$, we need to show that $\tilde{f}(x, \tilde{u}_1 \vee \tilde{u}_2) = \tilde{f}(x, \tilde{u}_1) \vee \tilde{f}(x, \tilde{u}_2)$ for all $x \in \chi$. By the order preserving property of $\tilde{f}$ in the second argument, we already know that $\tilde{f}(x, \tilde{u}_1) \vee \tilde{f}(x, \tilde{u}_2) \leq \tilde{f}(x, \tilde{u}_1 \vee \tilde{u}_2)$. Let us denote $\tilde{f}(x, \tilde{u}_1) \vee \tilde{f}(x, \tilde{u}_2) = a$ and let us in fact show that $a = \tilde{f}(x, \tilde{u}_1 \vee \tilde{u}_2)$. Let $\tilde{u}$ be such that $\tilde{f}(x, \tilde{u}) = a$. If $\tilde{u} \in \mathcal{P}(\mathcal{I})$, then let it be the largest such $\tilde{u}$. Consider the two chains $w_1 \prec w_2 \prec \ldots \prec w_{k_1}$ and $v_1 \prec v_2 \prec \ldots \prec v_{k_2}$, in which $w_1 = \tilde{f}(x, \tilde{u}_1)$, $v_{k_2} = w_{k_1} = a$, and $\tilde{f}(x, \tilde{u}_2) = v_1$. for all two consecutive elements on such chains $w_i \prec w_{i+1}$, there are $\tilde{u}_{1,i}, \tilde{u}_{1,i+1} \in \tilde{\mathcal{I}}(x)$ such that $\tilde{f}(x, \tilde{u}_{1,i}) = w_i$ and $\tilde{f}(x, \tilde{u}_{1,i+1}) = w_{i+1}$. If $\tilde{u}_{1,i}, \tilde{u}_{1,i+1}$ are both in $\mathcal{P}(\mathcal{I})$ then $w_i \leq w_{i+1}$. Also, if $\tilde{u}_{1,i} \in \mathcal{P}(\mathcal{I})$, we assume it is the largest input such that $\tilde{f}(x, \tilde{u}_{1,i}) = w_i$. If one or both of the inputs $\tilde{u}_{1,i}, \tilde{u}_{1,i+1}$ is a silent input, by item 3., we have that $\tilde{u}_{1,i} \prec \tilde{u}_{1,i+1}$. Since this is true for all $i \in \{1, \ldots, k-1\}$, we finally obtain that $\tilde{u}_1 \leq \tilde{u}$. Repeating this process for the chain $v_1 \prec v_2 \prec \ldots \prec v_{k_2}$, one also obtains that $\tilde{u}_2 \leq \tilde{u}$. Since $\tilde{u}_1$ and $\tilde{u}_2$ cannot have two different joins, it must be that either $\tilde{u} \leq \tilde{u}_1 \vee \tilde{u}_2$ or $\tilde{u}_1 \vee \tilde{u}_2 \leq \tilde{u}$. By the order preserving property of $\tilde{f}$, we have that $\tilde{u}_1 \vee \tilde{u}_2 \leq \tilde{u}$ implies $\tilde{f}(x, \tilde{u}_1 \vee \tilde{u}_2) \leq \tilde{f}(x, \tilde{u}) = a$. But, we assumed that $a < \tilde{f}(x, \tilde{u}_1 \vee \tilde{u}_2)$, as a consequence it must be that $\tilde{u} \leq \tilde{u}_1 \vee \tilde{u}_2$. However, by definition $\tilde{u}_1 \vee \tilde{u}_2$ is the smallest element that is larger than both $\tilde{u}_1$ and $\tilde{u}_2$. This in turn implies $\tilde{u} = \tilde{u}_1 \vee \tilde{u}_2$ and therefore $a = \tilde{f}(x, \tilde{u}_1 \vee \tilde{u}_2)$. Finally, we set $(\tilde{\mathcal{I}}, \leq)$ as the union of the lattices $(\tilde{\mathcal{I}}(x), \leq)$ constructed above. This union is well defined as all of the partial orders $(\tilde{\mathcal{I}}(x), \leq)$ are compatible by construction. Thus, one can add a bottom and a top element for $\tilde{\mathcal{I}}$ to make $(\tilde{\mathcal{I}}, \leq)$ a lattice. To conclude the proof, we need to show that $\{u \in \mathcal{I} \mid \tilde{f}([\bot, x], u) \subseteq [\bot, \bigvee\tilde{S}]\}$ is not empty for $[\bot, x] \subseteq \tilde{S} \cap O_y(\tilde{\Sigma})$. Note that $\{u \in \mathcal{I} \mid \tilde{f}([\bot, x], u) \subseteq [\bot, \bigvee\tilde{S}]\} = \{u \in \mathcal{I} \mid \tilde{f}(x, u) \leq \bigvee\tilde{S}\}$. The latter set is also equal to $\{u \in \mathcal{I} \mid f(x, x) \subseteq S\}$. This is not empty as $x \subseteq S \cap O_y(\Sigma)$ and Σ is controllable by dynamic output feedback with respect to S. Thus, $\tilde{\Sigma}|_{\mathcal{I}}$ is controllable by dynamic output feedback with respect to $\tilde{S}$.