2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

A Small Model Theorem for Bisimilarity Control under Partial Observation

Changyan Zhou and Ratnesh Kumar
Department of Electrical & Computer Engineering
Iowa State University, Ames, IA 50014

Abstract— **In our prior work on control under complete observation of a nondeterministic system to satisfy bisimilarity with a nondeterministic specification [21], [22], we established a "small model theorem" showing that a control-compatible (Σ_u-compatible for short) supervisor exists if and only if it exists over a certain finite state space, namely the power set of Cartesian product of system and specification state spaces. In this paper we show that the small model theorem remains valid even when there is partial observation of events so that a supervisor must be both control and observation compatible ((Σ_u, M)-compatible for short). The result proves the decidability of bisimilarity enforcing control under partial observation for general nondeterministic systems and nondeterministic specification.**

Keywords: Discrete event systems, supervisory control, nondeterministic systems, bisimulation equivalence, controllability, partial observation

I. Introduction

Extensive research on supervisory control has been done for deterministic discrete event systems (DESs) [17], [13] and also for nondeterministic discrete event systems [11], [12], [4], [6], using language as a means of specification. Also, the supervisors can be deterministic as well as nondeterministic. Inan [5] advocated the use of nondeterministic supervisors for control under partial observation for language specification. The notion of nondeterministic control was formalized in [10] and used for control of possibly nondeterministic plant under partial observation for language specification, and the notion of achievability (a property weaker than controllability and observability combined) was introduced as a necessary and sufficient condition for existence. Nondeterministic supervisors were also used in [7] where nondeterministic specification was specified in the temporal logic of CTL*.

In general, plant, specification, and supervisor all can be nondeterministic. Nondeterministic plant and specification are useful when designing a system at a higher level of abstraction so that lower level details of system and its specification are omitted to obtain higher level models that are nondeterministic. Nondeterministic specifications are also meaningful when the system to be controlled has a nondeterministic model due to lack of information (caused for example by partial observation or unmodeled dynamics). For nondeterministic systems numerous notions of behavioral equivalence that are finer than the language equivalence have been proposed ([20] provides a classification of these equivalences). In this paper we study the control of (possibly nondeterministic) plants subject to nondeterministic specifications using nondeterministic supervisors under partial observation. The notion of behavioral equivalence used between specification and controlled plant is bisimulation equivalence.

Choice of bisimulation equivalence is supported by the fact that bisimulation equivalence specification is equivalent to a specification in the temporal logic of μ-calculus that subsumes the complete branching-time logic CTL* [3]. Control for achieving CTL* specification was studied by Jiang and Kumar in [7], under the assumption that plant model is deterministic and all plant events are observable. In this paper we allow both plant and specification models to be nondeterministic, and plant is partially observed.

Bisimulation relation has been used as a technique for supervisory control of deterministic systems subject to language specifications in [18], [9], [2], [15], [8]. In [18], [8] the controllability and observability is characterized as a bisimulation type relation. [16] studied the problem of synthesizing a supervisor so that the controlled system is bisimilar to a deterministic specification. In that setting, the event set of the system and specification need not be same, and all events are treated controllable. [14] studied control for bisimulation equivalence for a partial specification (defined over an "external event set"). The plant is taken to be deterministic and all events are treated controllable. Further it is required that all events treated indistinguishable from the partial specifications point of view be either all enabled or all disabled at a state. Such a requirement does not make sense in supervisory control context. [19] studied the controller synthesis problem for deterministic plants subject to a possibly nondeterministic partial specification such that the controlled system is bisimulation equivalent to the specification. This is the same problem as that studied in [14] except the aforementioned control requirement is removed. [1] studied the synthesis of controllers for deterministic plants subject to μ-calculus based specifications under partial observation, where the observation mask is restricted to be projection type. A mu-calculus specification is equivalent to a bisimulation equivalence specification, but expressed very differently. The control problem is solved by reduction to a discrete-event game problem, and explicit

The research was supported in part by the National Science Foundation under the grants NSF-ECS-9709796, NSF-ECS-0099851, NSF-ECS-0218207, NSF-ECS-0244732, NSF-EPNES-0323379, and NSF-0424048, and a DoD-EPSCoR grant through the Office of Naval Research under the grant N000140110621.

conditions for the existence of a supervisor are not provided.

This paper extends our past work [21], [22] where we studied bisimilarity enforcing control under complete observation. A small model theorem was obtained showing that a supervisor exists if and only if it exists over a finite state space, proving the decidability of the control problem (both existence and synthesis). A supervisor must be control-compatible (also called Σ_u-compatible), meaning it should never disable any uncontrollable events.

When there is partial observation, supervisor must also be observation compatible (also called M-compatible) besides being Σ_u-compatible, meaning state-updates on indistinguishable events be identical. In this paper, we extend the small model theorem by showing that a control and observation compatible supervisor for enforcing bisimulation equivalence between the specification and the controlled system exists if and only if it exists over a certain finite state space, namely the power set of Cartesian product of the plant and the specification state spaces. This proves the decidability of bisimilarity enforcing control under partial observation for general nondeterministic systems and nondeterministic specifications. The results are illustrated through a simple example.

Rest of the paper is organized as follows. Section 2 gives notation and preliminaries. Section 3 studies supervisory control under partial observation for achieving bisimilarity for nondeterministic systems. The paper concludes with Section 4.

II. Notation and Preliminaries

In this paper nondeterministic state machines (NSMs) are used to model discrete event systems. A NSM G is a five tuple: $G := (X, \Sigma, \alpha, X_0, X_m)$, where X is its set of states, Σ is its set of events, $\alpha : X \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^X$ is its transition function, $X_0 \subseteq X$ is its set of initial states, and $X_m \subseteq X$ is its set of marked states. For an event set Σ, we use $\overline{\Sigma}$ to denote $\Sigma \cup \{\epsilon\}$. A triple $(x, \sigma, x') \in X \times \overline{\Sigma} \times X$ is called a *transition* if $x' \in \alpha(x, \sigma)$; if $\sigma = \epsilon$, the transition is called an ϵ-transition. Σ^* denotes the set of all finite-length sequences of events from Σ, called *traces* including the trace of zero length, denoted ϵ. The ϵ-closure of $x \in X$, denoted as $\epsilon^*(x)$, is the set of states reached by the execution of zero or more ϵ-transitions from state x. By using ϵ-closure map, we can extend the definition of transition function from events to traces, $\alpha^* : X \times \Sigma^* \rightarrow 2^X$, which is defined inductively as:

$$\forall x \in X, \alpha^*(x, \epsilon) := \epsilon^*(x);$$
$$\forall s \in \Sigma^*, \sigma \in \Sigma : \alpha^*(x, s\sigma) := \epsilon^*(\alpha(\alpha^*(x, s), \sigma)),$$

where for $\hat{X} \subseteq X$ and $\hat{\Sigma} \subseteq \Sigma$, $\alpha(\hat{X}, \hat{\Sigma}) := \cup_{x \in \hat{X}, \sigma \in \hat{\Sigma}} \alpha(x, \sigma)$, and $\epsilon^*(\hat{X}) := \cup_{x \in \hat{X}} \epsilon^*(x)$. The language generated (resp., marked) by G, is denoted as $L(G)$ (resp., $L_m(G)$). $L(G)$ is the sequence of events generated starting from the initial state, i.e., $L(G) = \{s \in \Sigma^* \mid \alpha^*(X_0, s) \neq \emptyset\}$, and $L_m(G)$ is the set of generated sequences that end in a marked state, i.e., $L_m(G) = \{s \in L(G) \mid \alpha^*(X_0, s) \cap X_m \neq \emptyset\}$. For $x \in X$, we define $\Sigma(x) := \{\sigma \in \overline{\Sigma} \mid \alpha(x, \sigma) \neq \emptyset\}$ to denote the set of all labels on which transitions are defined at state x.

One way to model control interaction between plant and supervisor is via the synchronous composition of their state machine (or automaton) representations. The *synchronous composition* of two automata G_1 and G_2, where $G_i = (X_i, \Sigma, \alpha_i, X_{0i}, X_{mi})$, is the automaton $G_1 \| G_2 = (X_1 \times X_2, \Sigma, \alpha_{\|}, X_{01} \times X_{02}, X_{m1} \times X_{m2})$, where for $x_1 \in X_1$, $x_2 \in X_2$, $\sigma \in \overline{\Sigma}$:

$$\alpha_{\|}((x_1, x_2), \sigma) := \begin{cases} \alpha_1(x_1, \sigma) \times \alpha_2(x_2, \sigma) & \text{if } \sigma \neq \epsilon \\ (\alpha_1(x_1, \epsilon) \times \{x_2\}) \cup (\{x_1\} \times \alpha_2(x_2, \epsilon)) & \text{if } \sigma = \epsilon \end{cases}$$

We also define the *union* of G_1 and G_2 as the automaton $G_1 \cup G_2 = (X_1 \cup X_2, \Sigma, \alpha_{\cup}, X_{01} \cup X_{02}, X_{m1} \cup X_{m2})$, where for $x \in X_1 \cup X_2$, $\sigma \in \overline{\Sigma}$:

$$\alpha_{\cup}(x, \sigma) = \begin{cases} \alpha_1(x, \sigma) & \text{if } x \in X_1 \\ \alpha_2(x, \sigma) & \text{if } x \in X_2 \end{cases}$$

The events executed by a plant are partially observed by a supervisor owing to the type of event-sensors used. Such a partial observation is represented using an observation mask function $M : \overline{\Sigma} \rightarrow \overline{\Delta}$ (Δ is the set of observed symbols), satisfying $M(\epsilon) = \epsilon$. M is said to be projection type if $\Delta \subseteq \Sigma$. $\sigma \in \Sigma$ is said to be an unobservable event if $M(\sigma) = \epsilon$, and otherwise it is said to be an observable event. Two events $\sigma_1, \sigma_2 \in \overline{\Sigma}$ are said to be indistinguishable if $M(\sigma_1) = M(\sigma_2)$. The observation mask M is extended to be defined over traces in Σ^* as follows: $M(\epsilon) := \epsilon$; $\forall s \in \Sigma^*, \sigma \in \Sigma : M(s\sigma) := M(s)M(\sigma)$.

Bisimulation equivalence is a type of behavioral equivalence that is used to describe equivalence between nondeterministic systems. A bisimulation relation over two state machines is a symmetric simulation relation. We next introduce the notion of a simulation relation.

Definition 1: Given automata $G_1 = (X_1, \Sigma, \alpha_1, X_{01}, X_{m1})$ and $G_2 = (X_2, \Sigma, \alpha_2, X_{02}, X_{m2})$, a *simulation relation* is a binary relation $\Phi \subseteq (X_1 \cup X_2)^2$ such that for $x_1, x_2 \in X_1 \cup X_2$, $(x_1, x_2) \in \Phi$ implies

1) $\sigma \in \overline{\Sigma}, x_1' \in \alpha_{\cup}^*(x_1, \sigma) \Rightarrow \exists x_2' \in \alpha_{\cup}^*(x_2, \sigma)$ such that $(x_1', x_2') \in \Phi$.
2) $x_1 \in X_{m1} \cup X_{m2} \Rightarrow x_2 \in X_{m1} \cup X_{m2}$.

G_1 *is said to be simulated by* G_2, denoted as $G_1 \sqsubseteq_{\Phi} G_2$, if there exists a simulation relation $\Phi \subseteq (X_1 \cup X_2)^2$ such that for all $x_{01} \in X_{01}$, exists $x_{02} \in X_{02}$ with $(x_{01}, x_{02}) \in \Phi$. This last fact is concisely written as $X_{01} \sqsubseteq_{\Phi} X_{02}$.

We write $x_1 \sqsubseteq_{\Phi} x_2$ to denote that there exists a simulation relation Φ with $(x_1, x_2) \in \Phi$, read as x_1 is simulated by x_2. We sometimes omit the subscript Φ from $\sqsubseteq_{\Phi}$ when it is clear from the context. Further, a simulation relation is called a *bisimulation* equivalence relation if it is symmetric. For a bisimulation equivalence relation Φ if $(x_1, x_2) \in \Phi$, then x_1 and x_2 are called *bisimilar*, written as $x_1 \simeq_{\Phi} x_2$ (or simply $x_1 \simeq x_2$ when Φ is clear from context). Two automata G_1 and G_2 are said to be bisimilar, denoted as

$G_1 \simeq_\Phi G_2$, if exists a bisimulation relation Φ such that for each $x_{0i} \in X_{0i}$ exists $x_{0j} \in X_{0j}$ $(i, j = 1, 2)$ such that $x_{0i} \simeq x_{0j}$, denoted for short as $X_{01} \simeq_\Phi X_{02}$.

III. Bisimilarity Control under Partial Observation

In this section, we study the control of a nondeterministic plant to ensure bisimilarity of the controlled plant and a given specification. A supervisor observes events through an *observation mask* M. If two events are observationally indistinguishable, and are enabled at a state, then a supervisor must perform identical state update on them. This is ensured by requiring the supervisor to be M-compatible, namely, whenever a pair of indistinguishable events are defined at a state, their successor states are the same. (Unobservable events are viewed as being ϵ-indistinguishable and so their successors must be the same as the successors of ϵ.) Events in a set $\Sigma_u \subseteq \Sigma$ are uncontrollable and must never be disabled by a supervisor. So besides being M-compatible (to accommodate limitations of partial observability) a supervisor must also be Σ_u-compatible, namely, all uncontrollable events be always enabled. We will use $G = (X, \Sigma, \alpha, X_0, X_m)$, $R = (Q, \Sigma, \delta, Q_0, Q_m)$, and $S = (Y, \Sigma, \beta, Y_0, Y_m)$ to denote the (nondeterministic) plant, specification, and supervisor, respectively.

Definition 2: Let $\Sigma_u \subseteq \Sigma$ be the set of uncontrollable events and $M : \overline{\Sigma} \to \overline{\Delta}$ be the observation mask, then

- S is called Σ_u-*compatible* if $\forall y \in Y$ and $\forall a \in \Sigma_u$, $\beta(y, a) \neq \emptyset$.
- S is called M-*compatible* if $\forall y \in Y$ and $\forall a, b \in \Sigma(y)$, if $M(a) = M(b)$, then $\beta(y, a) = \beta(y, b)$, where it is assumed that an ϵ-transition is implicitly defined as a self-loop.
- S is called (Σ_u, M)-*compatible* if S is Σ_u-compatible and M-compatible.

To motivate bisimilarity control under partial observation, we introduce the following manufacturing example, a solution to which is discussed latter.

Example 1: Consider a manufacturing system (shown in Figure 1) consisting of two workstations, one robot and three storage-stations. The robot moves among the workstations and storage-stations on guide rails. Initially, the robot departs from workstation 1 and nondeterministically travels on one of the rails (event a). On rail 1, the robot picks up a part from storage-station 1 (event b_1) and then delivers this part to workstation 2 for processing (event c). After the processing, robot returns the part to storage-station 1 (event b_1). On rail 2, the robot either picks up a part from storage-station 2 (event b_2) or from storage-station 3 (event b_3), and then delivers the part to workstation 2 for processing (event c). After the processing, the robot returns the part to either storage-station 2 or 3 (event b_2 or b_3). Not returning the part to its original storage-station is undesirable. After returning the part to the storage-station, the robot goes back to workstation 1 (event a) from where the entire process may be repeated. The state machine model G of the system is drawn in Figure 2.

The specification R, also drawn in Figure 2, shows the acceptable behavior. According to the specification, the robot returns any processed part to the same work-station from where it picked that part.

A part once picked must be delivered to workstation 2 for processing, i.e., the event c is uncontrollable. Only the events a and c are completely observable. Events b_1, b_2 and b_3 are observationally indistinguishable. Thus, we have $\Sigma = \{a, b_1, b_2, b_3, c\}$, $\Sigma_u = \{c\}$, and the observation mask M is given by, $M(a) = a$, $M(b_1) = M(b_2) = M(b_3) \neq \epsilon$ and $M(c) = c$. The control goal is to find a (Σ_u, M)-compatible supervisor S such that the controlled system $G\|S$ is bisimilar to the specification R.

We establish the main result that proves the decidability of bisimilarity enforcing control under partial observation by extending the "small model theorem" from the setting of complete observation [22] to the setting of partial observation. The small model theorem in [22] states that a bisimilarity enforcing Σ_u-compatible supervisor exists if and only if it exists over the state space $2^{X \times Q}$, where X is the state space of plant G and Q is the state space of specification R. The sufficiency is clearly obvious, while the key idea behind necessity is that given a Σ_u-compatible bisimilarity enforcing supervisor S (i.e., $G\|S \simeq R$), each state $y \in Y$ of S can be labeled by $lbl(y) \in 2^{X \times Q}$, and then states carrying identical labels can be merged to obtain state machine T with state space $2^{X \times Q}$ such that T is Σ_u-compatible and $G\|T \simeq R$. We recall from [22] that $(x, q) \in X \times Q$ belongs to $lbl(y)$ if and only if (x, y) is a state in $G\|S$ that is bisimilar to state q of R. We use the same labeling function for extending the small model theorem to the setting of partial observation.

Theorem 1: Given G and R, and a mask M, there exists a (Σ_u, M)-compatible supervisor S such that $G\|S \simeq R$ if and only if there exists a (Σ_u, M)-compatible state machine T with state space $2^{X \times Q}$ such that $G\|T \simeq R$.

Proof: (*Only If*) For necessity, suppose exists (Σ_u, M)-compatible S such that $G\|S \simeq_\Phi R$. Without loss of generality, all transitions of S participate in $G\|S$ (otherwise we can simply omit such transitions from S). Label each $y \in Y$ of S by $lbl(y) \subseteq X \times Q$ where $(x, q) \in lbl(y)$ if and only if (x, y) reachable in $G\|S$ and $q \in Q$ is such that $(x, y) \simeq_\Phi q$. Merge all states carrying the same label, and call the resulting state machine T. Then from the proof of [22, Theorem 2], $G\|T \simeq R$, T is Σ_u-compatible, and state space of T is $2^{X \times Q}$. We claim that T is also M-compatible, which will prove the necessity. Suppose $y_1, y_2 \in Y$ are such that $lbl(y_1) = lbl(y_2)$. Then y_1 and y_2 are merged to obtaining $\langle y_1, y_2 \rangle$. Also since there are no redundant transitions in S, it was shown in proof of [22, Theorem 2] that,

$$\Sigma(y_1) = \cup_{(x,q) \in lbl(y_1)} \Sigma(q) = \cup_{(x,q) \in lbl(y_2)} \Sigma(q) = \Sigma(y_2).$$

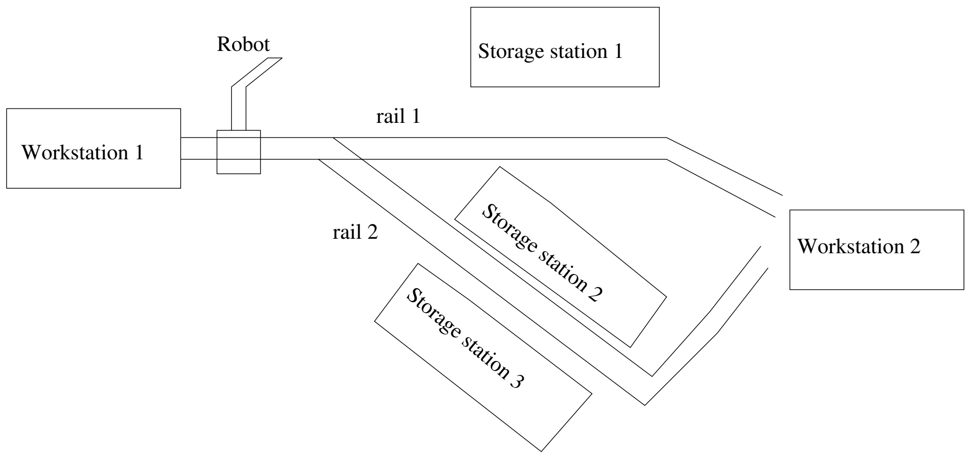

Fig. 1. A manufacturing system

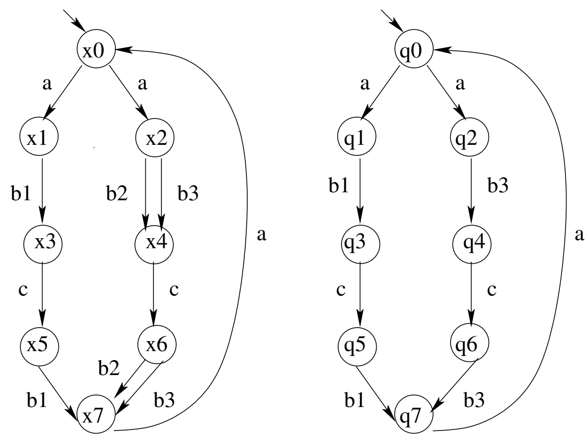

Fig. 2. G (left) and R (right)

So after merger, $\Sigma(\langle y_1, y_2\rangle) = \Sigma(y_1) = \Sigma(y_2)$. Since S is M-compatible, for any pair of indistinguishable events $a_1, a_2 \in \Sigma(y_1) = \Sigma(y_2)$, $\beta(y_1, a_1) = \beta(y_1, a_2)$ and $\beta(y_2, a_1) = \beta(y_2, a_2)$. So

$$\begin{aligned}\beta(\langle y_1, y_2\rangle, a_1) &= \beta(y_1, a_1) \cup \beta(y_2, a_1)\\ &= \beta(y_1, a_2) \cup \beta(y_2, a_2)\\ &= \beta(\langle y_1, y_2\rangle, a_2).\end{aligned}$$

Thus merger of states carrying the same label preserves M-compatibility, and so T is M-compatible.

(*If*) Set $S := T$, then S is (Σ_u, M)-compatible and $G\|S = G\|T \simeq R$. This completes the proof. ∎

Remark 1: From Theorem 1, an exhaustive search can be performed to determine the existence of a supervisor S over the state space $2^{X\times Q}$, the upper bound complexity of which is $O(2^{2^{|X|\times|Q|}})$. From this, the upper bound complexity of checking the existence of a supervisor under partial observation is same as the one under full observation. Better upper bounds may exist for the two cases, but are not known at this time.

Next we revisit the manufacturing example.

Example 2: Our goal is to find a (Σ_u, M)-compatible supervisor S such that $G\|S \simeq R$ (provided one exists). Such a supervisor is drawn in Figure 3. Since $\Sigma_u = \{c\}$, and c is defined at each state of S, S is Σ_u-compatible. Also state updates on indistinguishable pair of events b_1 and b_3 at states y_1, y_3, y_5, y_6, where they are both defined, are identical, implying that S is also M-compatible. The controlled system $G\|S$ is also drawn in Figure 3. The following bisimulation relation Φ exists between $G\|S$ and R:

$$\begin{aligned}\Phi \;=\; &\{(x_0y_0, q_0), (x_1y_1, q_1), (x_2y_1, q_2), (x_3y_2, q_3),\\ &(x_4y_2, q_4), (x_5y_3, q_5), (x_6y_3, q_6), (x_7y_4, q_7)\\ &(x_1y_5, q_1), (x_2y_5, q_2), (x_3y_6, q_3), (x_4y_6, q_4),\\ &(x_5y_6, q_5), (x_6y_6, q_6), (x_7y_6, q_7), (q_0, x_0y_0),\\ &(q_1, x_1y_1), (q_2, x_2y_1), (q_3, x_3y_2), (q_4, x_4y_2),\\ &(q_5, x_5y_3), (q_6, x_6y_3), (q_7, x_7y_4)(q_1, x_1y_5),\\ &(q_2, x_2y_5), (q_3, x_3y_6), (q_4, x_4y_6), (q_5, x_5y_6),\\ &(q_6, x_6y_6), (q_7, x_7y_6)\}.\end{aligned}$$

From Theorem 1, exists a (Σ_u, M)-compatible T with state space $2^{X\times Q}$ such that $G\|T \simeq R$. To obtain such a T, the

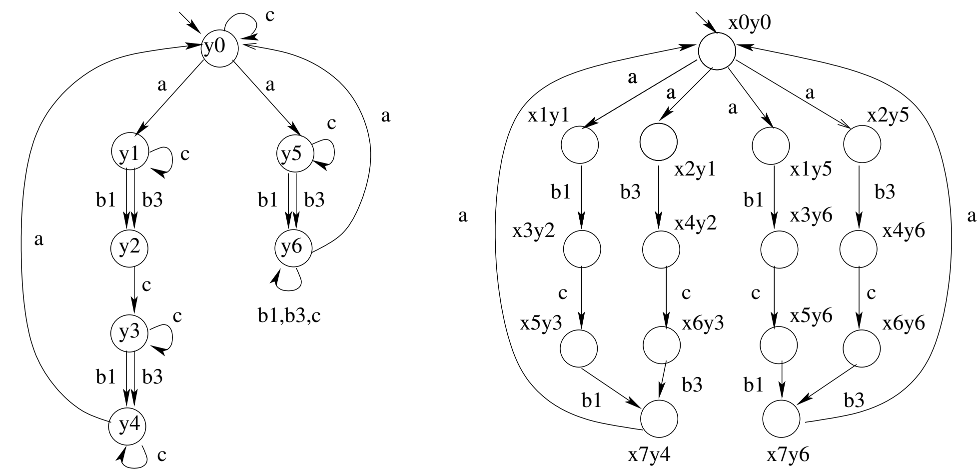

Fig. 3. S (left) and $G\|S$ (right)

	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	$lbl(\cdot)$
y_0	q_0	-	-	-	-	-	-	-	$\{(x_0, q_0)\}$
y_1	-	q_1	q_2	-	-	-	-	-	$\{(x_1, q_1), (x_2, q_2)\}$
y_2	-	-	-	q_3	q_4	-	-	-	$\{(x_3, q_3), (x_4, q_4)\}$
y_3	-	-	-	-	-	q_5	q_6	-	$\{(x_5, q_5), (x_6, q_6)\}$
y_4	-	-	-	-	-	-	-	q_7	$\{(x_7, q_7)\}$
y_5	-	q_1	q_2	-	-	-	-	-	$\{(x_1, q_1), (x_2, q_2)\}$
y_6	-	-	-	q_3	q_4	q_5	q_6	q_7	$\{(x_3, q_3), (x_4, q_4), (x_5, q_5), (x_6, q_6), (x_7, q_7)\}$

TABLE I

COMPUTATION OF LABELING FUNCTION FOR EXAMPLE 2

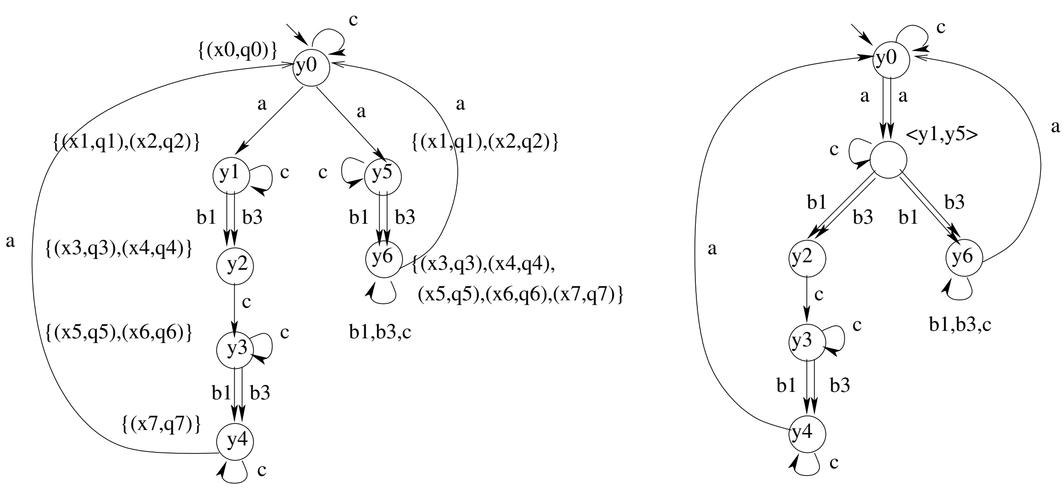

Fig. 4. The labeling of states in S (left) and T (right)

labeling of each state in S is shown in Figure 4, and is computed using Table I. State machine T is obtained by merging states in S carrying the same label. Since $lbl(y_1) = lbl(y_5)$, we merge y_1 and y_5. The resulting state machine T is drawn in Figure 4. T is (Σ_u, M)-compatible as expected. $G\|T$ is drawn in Figure 5. It can be easily checked that $G\|T \simeq G\|S \simeq R$.

IV. CONCLUSION

In this note we studied supervisory control of nondeterministic systems subject to a nondeterministic specification under partial observation, with the objective that the controlled system be bisimulation equivalent to the specification. We obtained a small model theorem showing that a control and observation compatible bisimilarity enforcing supervisor exists if and only if it exists over a certain finite state space, namely the power set of Cartesian product of the plant and the specification state spaces, thereby proving the decidability of bisimilarity enforcing control under partial observation of general nondeterministic systems and nondeterministic specifications. It was shown in [21], [22] that in the special case when plant is deterministic, the bisimilarity enforcing control under complete observation is polynomi-

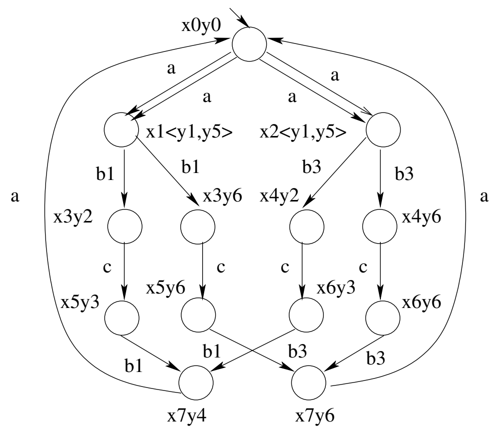

Fig. 5. $G \| T$

ally solvable. The existence can be verified linearly in size of plant and specification, and synthesis can be performed linearly in size of specification. It will be interesting to see if similar polynomial complexity results exist even for a partially observed plant when it is deterministic.

REFERENCES

[1] A. Arnold, A. Vincent, and I. Walukiewicz. Games for synthesis of controllers with partial observation. *Theoretical Computer Science*, 303:7–34, 2003.

[2] G. Barrett and S. Lafortune. Using bisimulation to solve discrete event control problem. In *1997 American Control Conference*, pages 2337–2341, Albuquerque, New Mexico, June 1997.

[3] M. Hennessey and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of ACM*, 32:137–161, 1985.

[4] M. Heymann and F. Lin. Discrete-event control of nondeterministic systems. *IEEE Transactions on Automatic Control*, 43(1):3–17, January 1998.

[5] K. Inan. Nondeterministic supervision under partial observations. In Guy Cohen and Jean-Pierre Quadrat, editors, *Lecture Notes in Control and Information Sciences 199*, pages 39–48. Springer-Verlag, New York, 1994.

[6] S. Jiang and R. Kumar. Supervisory control of nondeterministic discrete event systems with driven events via masked prioritized synchronization. *IEEE Transactions on Automatic Control*, 47(9):1438–1449, 2002.

[7] S. Jiang and R. Kumar. Supervisory control of discrete event systems with CTL* temporal logic specification. *SIAM Journal on Control and Optimization*, 2003. Accepted.

[8] J. Komenda. Coalgebra and supervisory control of discrete-event systems with partial observations. In *International Symposium (MTNS 2002)*, pages 12–16, Notre Dame, August 2002.

[9] J. Komenda. Computation of supremal sublanguages of supervisory control using coalgebra. In *Workshop on Discrete-Event Systems (WODES'02)*, pages 26–33, Zaragoza, October 2002.

[10] R. Kumar, S. Jiang, C. Zhou, and W. Qiu. Control using nondeterministic supervisors for partially observed discrete event systems. In *Proceedings of 2004 American Control Conference*, pages 4472–4476, Boston, MA, June 2004.

[11] R. Kumar and M. A. Shayman. Nonblocking supervisory control of nondeterministic systems via prioritized synchronization. *IEEE Transactions on Automatic Control*, 41(8):1160–1175, August 1996.

[12] R. Kumar and M. A. Shayman. Centralized and decentralized supervisory control of nondeterministic systems under partial observation. *SIAM Journal of Control and Optimization*, 35(2):363–383, March 1997.

[13] F. Lin and W. M. Wonham. On observability of discrete-event systems. *Information Sciences*, 44(3):173–198, 1988.

[14] P. Madhusudan and P.S. Thiagarajan. Branching time controllers for discrete event systems. *Theoretical Computer Science*, 274:117–149, 2002.

[15] H. Marchand and S. Pinchinat. Supervisory control problem using symbolic bisimulation techniques. In *2000 American Control Conference*, pages 4067–4071, Chicago, Illinois, USA, June 2000.

[16] H. Qin and P. Lewis. Factorization of finite state machines under observational equivalence. In *Lecture Notes In Computer Science*, volume 458. Springer-Verlag, 1990.

[17] P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal of Control and Optimization*, 25(1):206–230, 1987.

[18] J.J.M.M. Rutten. Coalgebra, concurrency, and control. In *5th Workshop on Discrete Event Systems*, pages 31–38, 2000.

[19] P. Tabuada. Open maps, alternating simulations and control synthesis. In *CONCUR*, pages 466–480, 2004.

[20] R. van Glabbeek. The linear time - branching time spectrum. In *In Proceedings CONCUR'90 of Lecture Notes in Computer Science*, volume 458, Amsterdam, 1990.

[21] C. Zhou, R. Kumar, and S. Jiang. Control of nondeterministic discrete event systems for bisimulation equivalence. In *Proceedings of 2004 American Control Conference*, pages 4488–4492, Boston, MA, June 2004.

[22] C. Zhou, R. Kumar, and S. Jiang. Control of nondeterministic discrete event systems for bisimulation equivalence. *IEEE Transactions on Automatic Control*, 2004. Accepted.