OUKA
The University of Osaka
Institutional Knowledge Archive

Title	Maximally Permissive Supervisors for Nonblocking Similarity Control of Nondeterministic Discrete-Event Systems
Author(s)	Li, Jinglun; Takai, Shigemasa
Citation	IEEE Transactions on Automatic Control. 2022, 68(6), p. 3529-3544
Version Type	AM
URL	https://hdl.handle.net/11094/106742
rights	© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

Maximally Permissive Supervisors for Nonblocking Similarity Control of Nondeterministic Discrete Event Systems

Jinglun Li and Shigemasa Takai, *Member, IEEE*

***Abstract*—This paper investigates a nonblocking similarity control problem for nondeterministic discrete event systems, which is a problem of synthesizing a nonblocking supervisor such that the supervised system is simulated by the given specification. In this research, the state of the system is not required to be observable, and the event occurrence is allowed to be partially observed. We propose an algorithm that computes a nonblocking supervisor from a possibly blocking one by iteratively removing certain states. Then, we identify two key properties of input supervisors, named state-unmergedness and strong maximal permissiveness, which together guarantee the maximal permissiveness of output nonblocking supervisors. The algorithm is applied to a supervisor with these two properties to obtain a maximally permissive nonblocking supervisor. In addition, we show that a nonblocking supervisor is generated by the algorithm if and only if there exists a solution to the nonblocking similarity control problem.**

***Index Terms*—Nondeterministic discrete event system, similarity control, nonblocking supervisor, maximal permissiveness, partial observation.**

I. Introduction

THE supervisory control problem for discrete event systems (DESs) arises in the study of automated event-driven systems [1]. Under the framework of supervisory control for deterministic DESs formulated by Ramadge and Wonham [1], a nonblocking supervisor is synthesized for the given system and the specification to achieve the equivalence of the marked language of the supervised system and the specification language. If such a supervisor does not exist, a nonblocking supervisor is synthesized for the supremal $L_m(G)$-closed and controllable sublanguage of the specification language [2].

In [3], [4], supervisory control was initially considered under the assumption that only a part of the events in the system can be observed. It was shown that there exists a nonblocking supervisor for the given specification that works under partial observation if and only if the specification language is $L_m(G)$-closed, controllable, and observable. As observability is not preserved under union in general, if the specification cannot be exactly achieved, the optimal nonblocking supervisor that achieves the supremal subspecification does not necessarily exist under partial observation. Varieties of works [3]–[8] have been done to synthesize a nonblocking supervisor for partially observed DESs. Especially, a maximally permissive supervisor was synthesized in [8] in the sense that there does not exist another supervisor that achieves a strictly larger subspecification.

Owing to unmodeled dynamics or abstraction, nondeterminism of transitions may arise in the system models [9]. Deterministic control for nondeterministic systems was studied in [10], [11], and [12], where the specification is given as a process, language, and trajectory, respectively. In [13], a notion of bisimulation equivalence, which is stronger than language equivalence, was introduced, and it is widely used for verification and control of dynamical systems. For the system and the specification modeled as nondeterministic automata, a bisimilarity control problem is a problem of synthesizing a supervisor such that the supervised system is bisimulation equivalent to the specification. This problem has been studied in the literature [9], [14]–[19].

If there does not exist a supervisor that achieves bisimulation equivalence of the supervised system and the specification, a possible approach is to relax the requirement of enforcing bisimilarity. One may consider a similarity control problem, which only requires us to synthesize a similarity-enforcing supervisor so that the supervised system is simulated by the specification. It is known that the simulation relation from the supervised system to the specification is stronger than the inclusion relations of their generated and marked languages. Similarity control was studied in [20]–[24] without addressing the nonblocking issue. In [20], a range similarity control problem was studied, where a supervisor is synthesized so that the supervised system simulates the lower bound specification and is simulated by the upper bound specification. In [21], a maximally permissive similarity-enforcing supervisor was synthesized under the assumption that the current state of the system is observable. The similarity control problem that does not require state observation was studied in [22]–[24]. In particular, maximally permissive supervisors which enforce similarity are synthesized for fully and partially observed systems in [23] and [24], respectively.

The nonblocking similarity control problem is a problem of synthesizing a nonblocking similarity-enforcing supervisor. Due to nondeterminism, imposing nonblockingness complicates the similarity control problem. To solve the problem, the assumption that the current state is observable was imposed in [25], [26], which is restrictive in general.

In this paper, we solve the nonblocking similarity control

This work was supported in part by JSPS KAKENHI Grant Number JP18K04201. This paper was presented in part at the 59th IEEE Conference on Decision and Control, Jeju Island, Republic of Korea, December 14–18, 2020.

J. Li and S. Takai are with the Division of Electrical, Electronic and Information Engineering, Osaka University, Suita, Osaka 565-0871, Japan (e-mail: lijl@is.eei.eng.osaka-u.ac.jp; takai@eei.eng.osaka-u.ac.jp).

problem without requiring state observation. Besides, (the event occurrence of) the system is allowed to be partially observed. We present an algorithm named NBSR, which computes a nonblocking supervisor from a possibly blocking one by iteratively removing certain states. We identify a class of supervisors as the inputs of NBSR, so that its output supervisors are maximally permissive.

The reminder of this paper is organized as follows. We introduce preliminary definitions in Section II. In Section III, we formulate the problem we consider in this paper. In Section IV, we provide an algorithm named NBSR that computes a nonblocking similarity-enforcing supervisor and show its validity. To investigate how to synthesize a maximally permissive supervisor using NBSR, we review some results of the existing work and synthesize a supervisor as a candidate for the input of NBSR in Section V. In section VI, we identify two properties of an input supervisor, named state-unmergedness and strong maximal permissiveness, which are the key for synthesizing a maximally permissive nonblocking supervisor by NBSR. Then, we derive a necessary and sufficient condition for the solvability of the nonblocking similarity control problem, and synthesize a maximally permissive supervisor when it is solvable. Finally, we conclude the paper in Section VII. Some important notations of this paper are listed in TABLE I. Their detailed explanations are given afterwards.

As compared with the authors' conference paper [27], this paper (i) contains the proofs of the technical results which were omitted in [27]; (ii) identifies the two key properties for synthesizing a maximally permissive supervisor by NBSR; (iii) extends the results to the case of partial observation; and (iv) provides a practical example to illustrate the motivation.

II. Preliminaries

A. Nondeterministic Automata

In this paper, DESs are modeled as nondeterministic finite-state automata. Let $\mathcal{A}(\Sigma)$ denote the set of all nondeterministic finite-state automata over the event set Σ. A nondeterministic finite-state automaton $G \in \mathcal{A}(\Sigma)$ is defined as a 5-tuple $G = (X, \Sigma, \alpha, X_0, X_m)$, where X is the finite set of states, $\alpha : X \times \Sigma \to 2^X$ is the nondeterministic state transition function, $X_0 \subseteq X$ is the nonempty set of initial states, and $X_m \subseteq X$ is the set of marked states, which represent, for example, the completion of a task. Σ^* denotes the set of all finite event strings of elements in Σ, including the empty string ε. The transition function α can be extended to $\alpha : X \times \Sigma^* \to 2^X$ inductively as

- $\alpha(x, \varepsilon) = \{x\}$, and
- $\forall s \in \Sigma^*, \forall \sigma \in \Sigma : \alpha(x, s\sigma) = \bigcup_{x' \in \alpha(x,s)} \alpha(x', \sigma)$

for any $x \in X$. Moreover, we let

$$\alpha(X', s) = \bigcup_{x' \in X'} \alpha(x', s)$$

for any $X' \subseteq X$ and any $s \in \Sigma^*$. The generated and marked languages of G are defined as $L(G) = \{s \in \Sigma^* \mid \alpha(X_0, s) \neq \emptyset\}$ and $L_m(G) = \{s \in L(G) \mid \alpha(X_0, s) \cap X_m \neq \emptyset\}$, respectively. $Reach(G) = \{x \in X \mid \exists s \in L(G) : x \in \alpha(X_0, s)\}$ denotes the set of all reachable states of G. G is said to be nonblocking if

$$\forall x \in Reach(G), \exists s \in \Sigma^* : \alpha(x, s) \cap X_m \neq \emptyset,$$

which guarantees that G can reach a marked state from any reachable state. This condition is stronger than the conventional nonblocking condition $\overline{L_m(G)} = L(G)$ defined for a deterministic automaton G, where $\overline{L_m(G)}$ denotes the prefix-closure of $L_m(G)$.

A subautomaton G' of G with respect to a subset of the state set X of G is defined as follows.

Definition 1: For an automaton $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and a subset $X' \subseteq X$ such that $X_0 \cap X' \neq \emptyset$, the subautomaton $G' \in \mathcal{A}(\Sigma)$ of G with respect to X' is defined as $G' = (X', \Sigma, \alpha', X_0', X_m') \in \mathcal{A}(\Sigma)$, where $X_0' = X_0 \cap X'$ and $X_m' = X_m \cap X'$. The nondeterministic transition function $\alpha' : X' \times \Sigma \to 2^{X'}$ is given as $\alpha'(x, \sigma) = \alpha(x, \sigma) \cap X'$ for each $x \in X' \subseteq X$ and each $\sigma \in \Sigma$.

A simulation relation is defined as follows.

Definition 2: [9] Let $G_1 = (X_1, \Sigma, \alpha_1, X_{10}, X_{1m}) \in \mathcal{A}(\Sigma)$ and $G_2 = (X_2, \Sigma, \alpha_2, X_{20}, X_{2m}) \in \mathcal{A}(\Sigma)$. A relation $\Phi \subseteq X_1 \times X_2$ is said to be a simulation relation from G_1 to G_2 if

- $\forall x_{10} \in X_{10}, \exists x_{20} \in X_{20} : (x_{10}, x_{20}) \in \Phi$,
- $\forall (x_1, x_2) \in \Phi, \forall \sigma \in \Sigma, \forall x_1' \in \alpha_1(x_1, \sigma), \exists x_2' \in \alpha_2(x_2, \sigma) : (x_1', x_2') \in \Phi$, and
- $\forall (x_1, x_2) \in \Phi : x_1 \in X_{1m} \Rightarrow x_2 \in X_{2m}$.

If there exists a simulation relation from G_1 to G_2, G_1 is said to be simulated by G_2, and we write $G_1 \sqsubseteq G_2$.

Remark 1: G_1 being simulated by G_2 means that the possible behavior in G_1 is also possible in G_2. For nondeterministic finite-state automata $G_1, G_2 \in \mathcal{A}(\Sigma)$, we only have

$$G_1 \sqsubseteq G_2 \Rightarrow L(G_1) \subseteq L(G_2) \wedge L_m(G_1) \subseteq L_m(G_2),$$

and the reverse relation does not hold in general. That is, in the nondeterministic setting, the simulation relation is stronger than the inclusion relations of the generated and marked languages.

B. Nondeterministic Supervisory Control

We model the system and the specification as automata $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$, respectively, which share the same event set Σ.

A supervisor is also modeled as an automaton $S = (Y, \Sigma, \beta, Y_0, Y) \in \mathcal{A}(\Sigma)$ whose states are all marked. The system G supervised by a supervisor S is modeled as their synchronous composition $S \| G$ [9], where

$$S \| G = (Y \times X, \Sigma, \xi, Y_0 \times X_0, Y \times X_m) \in \mathcal{A}(\Sigma).$$

The nondeterministic transition function $\xi : (Y \times X) \times \Sigma \to 2^{Y \times X}$ of $S \| G$ is defined as $\xi((y, x), \sigma) = \beta(y, \sigma) \times \alpha(x, \sigma)$ for each $(y, x) \in Y \times X$ and each $\sigma \in \Sigma$.

The event set Σ is partitioned into the controllable (respectively, observable) event set Σ_c (respectively, Σ_o) and the uncontrollable (respectively, unobservable) event set Σ_{uc} (respectively, Σ_{uo}). An uncontrollable event cannot be disabled by a supervisor. A supervisor observes the event occurrence through the natural projection $P : \Sigma^* \to \Sigma_o^*$ defined inductively as

- $P(\varepsilon) = \varepsilon$, and
- $\forall s \in \Sigma^*, \forall \sigma \in \Sigma : P(s\sigma) = \begin{cases} P(s)\sigma, & \text{if } \sigma \in \Sigma_o \\ P(s), & \text{if } \sigma \in \Sigma_{uo}. \end{cases}$

TABLE I
SOME IMPORTANT NOTATIONS

Notations	Meanings
$\Sigma_{uo}(y)$	The set of unobservable events defined at state y
$\mathcal{R}_y(S)$	The set of runs in S which reach state y
$L(r)$	The set of strings enabled via a run r
$\mathcal{S}(G,R)$	The set of similarity-enforcing supervisors for G and R
$\mathcal{MPS}(G,R)$	The set of maximally permissive supervisors in $\mathcal{S}(G,R)$
$\mathcal{NS}(G,R)$	The set of nonblocking similarity-enforcing supervisors for G and R
$\mathcal{MPNS}(G,R)$	The set of maximally permissive supervisors in $\mathcal{NS}(G,R)$
$\mathrm{NBSR}(S,G)$	The output of Algorithm 1 when it is applied to a supervisor S for G
$\min\mathcal{W}$	The set of minimal elements of a set $\mathcal{W}$
$\widehat{\mathcal{MPS}}(G,R)$	The set of state-unmerged and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisors in $\mathcal{S}(G,R)$ which are strongly more permissive than any $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisor in $\mathcal{S}(G,R)$

Then, a notion of admissibility is imposed on a supervisor to guarantee that: (i) the uncontrollable events are always enabled; and (ii) the supervisor does not react to the occurrence of any unobservable event.

Definition 3: For the system $G=(X,\Sigma,\alpha,X_0,X_m)\in\mathcal{A}(\Sigma)$, a supervisor $S=(Y,\Sigma,\beta,Y_0,Y)\in\mathcal{A}(\Sigma)$ is said to be admissible if, for any $(y,x)\in \mathit{Reach}(S\|G)$, it satisfies

- $\forall\sigma\in\Sigma_{uc}:\alpha(x,\sigma)\neq\emptyset\Rightarrow\beta(y,\sigma)\neq\emptyset$ [22], and
- $\forall\sigma\in\Sigma_{uo}:\beta(y,\sigma)\neq\emptyset\Rightarrow\beta(y,\sigma)=\{y\}$ [14].

Since the current state of the system is not required to be observed in this paper, we refer to full event observation and partial event observation as full observation and partial observation, respectively. For any state $y\in Y$ of a supervisor S, we let $\Sigma_{uo}(y)=\{\sigma\in\Sigma_{uo}\mid\beta(y,\sigma)\neq\emptyset\}$ be the set of unobservable events enabled at y. The control mechanism of an admissible supervisor S for the partially observed system G can be explained as follows. Let $(y,x)\in\mathit{Reach}(S\|G)$ be the current state of $S\|G$. An event $\sigma\in\Sigma$ is enabled by S if $\beta(y,\sigma)\neq\emptyset$. We suppose that an enabled event σ occurs in G and causes a transition from x to $x'\in\alpha(x,\sigma)$. If $\sigma\in\Sigma_{uo}$, S does not react to σ because of the second condition of admissibility, and the resulting state of $S\|G$ is $(y,x')\in\mathit{Reach}(S\|G)$. If $\sigma\in\Sigma_o$, S observes the occurrence of σ and chooses the next state y' nondeterministically from $\beta(y,\sigma)$. The resulting state of $S\|G$ is $(y',x')\in\mathit{Reach}(S\|G)$. A supervisor S is said to be nonblocking if $S\|G$ is nonblocking.

We also define a class of supervisors, named $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisors, in the following manner.

Definition 4: For the system $G=(X,\Sigma,\alpha,X_0,X_m)\in\mathcal{A}(\Sigma)$, a supervisor $S=(Y,\Sigma,\beta,Y_0,Y)\in\mathcal{A}(\Sigma)$ is said to be $\Sigma_{uc}\cap\Sigma_{uo}$-compatible if

$$\forall y\in Y,\forall\sigma\in\Sigma_{uc}\cap\Sigma_{uo}:\beta(y,\sigma)=\{y\}.$$

Note that a supervisor $S\in\mathcal{A}(\Sigma)$ that works under partial observation has to be admissible, but it need not to be $\Sigma_{uc}\cap\Sigma_{uo}$-compatible. Namely, for any state $y\in Y$ and any $\sigma\in\Sigma_{uc}\cap\Sigma_{uo}$, if there does not exist $(y,x)\in\mathit{Reach}(S\|G)$ such that $\alpha(x,\sigma)\neq\emptyset$, then $\beta(y,\sigma)$ need not to be nonempty. Moreover, by the definition of admissibility, the following proposition is trivial.

Proposition 1: For the system $G=(X,\Sigma,\alpha,X_0,X_m)\in\mathcal{A}(\Sigma)$ and an admissible supervisor $S=(Y,\Sigma,\beta,Y_0,Y)\in\mathcal{A}(\Sigma)$, we consider a supervisor $S_{com}=(Y,\Sigma,\beta_{com},Y_0,Y)\in\mathcal{A}(\Sigma)$, where

$$\beta_{com}(y,\sigma)=\begin{cases}\{y\}, & \text{if } \sigma\in(\Sigma_{uc}\cap\Sigma_{uo})-\Sigma_{uo}(y)\\ \beta(y,\sigma), & \text{otherwise}\end{cases}$$

for each $y\in Y$ and each $\sigma\in\Sigma$. Then, it holds that $S\|G=S_{com}\|G$ and S_{com} is admissible and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible.

Proposition 1 shows that any admissible supervisor S can be turned into an equivalent admissible and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible one S_{com} by complementing the transitions of each state y with self-loop transitions by events in $\Sigma_{uc}\cap\Sigma_{uo}$ which are not originally defined at y.

For any reachable state $y\in\mathit{Reach}(S)$ of an admissible supervisor $S\in\mathcal{A}(\Sigma)$, there exists an observable event string $s\in\Sigma_o^*\cap L(S)$ such that $y\in\beta(Y_0,s)$. Then, there exist states $y_0,y_1,\ldots,y_n\in\mathit{Reach}(S)$ with $y_0\in Y_0$, $y_n=y$, and $y_{i+1}\in\beta(y_i,\sigma_i)$ for each $i\in\{0,1,\ldots,n-1\}$, where $s=\sigma_0\sigma_1\ldots\sigma_{n-1}$. For a series of state transitions $y_0\xrightarrow{\sigma_0}y_1\xrightarrow{\sigma_1}\ldots\xrightarrow{\sigma_{n-1}}y_n$ that finally reaches $y_n=y$, we call $\gamma_0\sigma_0\ldots\gamma_{n-1}\sigma_{n-1}\gamma_n\in 2^{\Sigma_{uo}}(\Sigma_o 2^{\Sigma_{uo}})^*$ a run in S that reaches y, where $\gamma_j=\Sigma_{uo}(y_j)$ for each $j\in\{0,1,\ldots,n\}$. We let $\mathcal{R}_y(S)$ denote the set of all runs in S which reach y. Note that, when $y\in Y_0$, since $y\in\beta(Y_0,\varepsilon)$, the run $\Sigma_{uo}(y)\in 2^{\Sigma_{uo}}$ is an element of $\mathcal{R}_y(S)$. A run $r=\gamma_0\sigma_0\ldots\gamma_{n-1}\sigma_{n-1}\gamma_n\in\mathcal{R}_y(S)$ records not only the observable event string $\sigma_0\sigma_1\ldots\sigma_{n-1}\in\Sigma_o^*\cap L(S)$ that leads to y but also the control decisions $\gamma_0,\gamma_1,\ldots,\gamma_n\in 2^{\Sigma_{uo}}$ on unobservable events that S made. Here, the set of event strings that lead to y via r is given as $L(r,S)=\{s_0\sigma_0\ldots s_{n-1}\sigma_{n-1}s_n\in L(S)\mid\forall i\in\{0,1,\ldots,n\}:s_i\in\gamma_i^*\}\subseteq L(S)$, which is also the set of strings enabled by S when its current state is reached via r. Note that $L(r,S)$ is uniquely determined by r. We suppose that r is shared by another supervisor $S'\in\mathcal{A}(\Sigma)$, namely, there exists a state $y'\in\mathit{Reach}(S')$ such that $r\in\mathcal{R}_{y'}(S')$. Then, it holds that $L(r,S)=L(r,S')$. Hence, for the sake of simplicity, we write $L(r,S)$ as $L(r)$ with a slight abuse of the notation.

Example 1: Let $S\in\mathcal{A}(\Sigma)$ shown in Fig. 1 be a supervisor for some system $G\in\mathcal{A}(\Sigma)$, where $\Sigma=\{a,b,u_1,u_2\}$ and $\Sigma_{uo}=\{u_1,u_2\}$. We consider state $y_3\in\mathit{Reach}(S)$, which can be reached by $y_0\xrightarrow{a}y_1\xrightarrow{b}y_3$ or $y_0\xrightarrow{a}y_2\xrightarrow{a}y_3$. It follows that $\mathcal{R}_{y_3}(S)=\{r_1,r_2\}$, where $r_1=\{u_1\}a\emptyset b\{u_1,u_2\}$ and $r_2=\{u_1\}a\{u_2\}a\{u_1,u_2\}$. Then, we have $L(r_1)=\{s_1abs_1'\in L(S)\mid s_1\in\{u_1\}^*\wedge s_1'\in\{u_1,u_2\}^*\}$ and $L(r_2)=\{s_2as_2'as_2''\in L(S)\mid s_2\in\{u_1\}^*\wedge s_2'\in\{u_2\}^*\wedge s_2''\in\{u_1,u_2\}^*\}$.

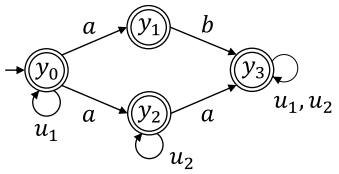


Fig. 1. Supervisor S.

In a special case where $\Sigma_{uo} = \emptyset$, namely, the system is fully observed, a run $r = \gamma_0\sigma_0\gamma_1\sigma_1\ldots\gamma_{n-1}\sigma_{n-1}\gamma_n = \emptyset\sigma_0\emptyset\sigma_1\ldots\emptyset\sigma_{n-1}\emptyset$ can be characterized by the string $\sigma_0\sigma_1\ldots\sigma_{n-1}$, and, since $\emptyset^* = \{\varepsilon\}$, it holds that $L(r) = \{\sigma_0\sigma_1\ldots\sigma_{n-1}\}$.

III. Problem Formulation

We first introduce some necessary terminologies. Given the (possibly partially observed) system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$, if a supervisor $S \in \mathcal{A}(\Sigma)$ satisfies $S\|G \sqsubseteq R$ and is admissible and nonblocking, then it is said to be a nonblocking similarity-enforcing supervisor. Let $\mathcal{NS}(G,R) \subseteq \mathcal{A}(\Sigma)$ be the set of all nonblocking similarity-enforcing supervisors for G and R. We evaluate the permissiveness of a supervisor with respect to the simulation relation. Then,

$$\mathcal{MPNS}(G,R) = \{S \in \mathcal{NS}(G,R) \mid \forall S' \in \mathcal{NS}(G,R) : S'\|G \sqsubseteq S\|G\}$$

denotes the set of all maximally permissive nonblocking similarity-enforcing supervisors.

Now, we formulate the problem to be solved in this paper.

Nonblocking similarity control problem : For the partially observed system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$, find a supervisor $S \in \mathcal{NS}(G,R)$. The nonblocking similarity control problem is said to be solvable if $\mathcal{NS}(G,R) \neq \emptyset$, and a supervisor $S \in \mathcal{NS}(G,R)$ is a solution to it. In particular, we synthesize a maximally permissive supervisor $\tilde{S} \in \mathcal{MPNS}(G,R)$ as a maximal solution.

Remark 2: By the definition of $\mathcal{MPNS}(G,R)$, a supervisor $S \in \mathcal{MPNS}(G,R)$ is maximally permissive in a global sense. This is achievable since a supervisor is allowed to be nondeterministic [28]. On the other hand, the order $\sqsubseteq$ on $\mathcal{A}(\Sigma)$ is a preorder, which is transitive and reflexive, but not antisymmetric [20]. Hence, for two $S_1, S_2 \in \mathcal{MPNS}(G,R)$, we cannot claim that $S_1 = S_2$ or $S_1\|G = S_2\|G$. Instead, we have $S_1\|G \sqsubseteq S_2\|G$ and $S_2\|G \sqsubseteq S_1\|G$, which imply that S_1 is as permissive as S_2 (with respect to the simulation relation). Note that $S_1\|G$ and $S_2\|G$ are language equivalent, that is, $L(S_1\|G) = L(S_2\|G)$ and $L_m(S_1\|G) = L_m(S_2\|G)$ hold.

For later use, we also introduce the following terminologies. A supervisor $S' \in \mathcal{A}(\Sigma)$ is said to be a similarity-enforcing supervisor if S' is admissible and $S'\|G \sqsubseteq R$ holds. (Namely, S' is possibly blocking.) Let $\mathcal{S}(G,R) \subseteq \mathcal{A}(\Sigma)$ be the set of all similarity-enforcing supervisors for G and R. Then, it holds that $\mathcal{NS}(G,R) \subseteq \mathcal{S}(G,R)$. Similarly, the set of all maximally permissive similarity-enforcing supervisors is given as

$$\mathcal{MPS}(G,R) = \{S \in \mathcal{S}(G,R) \mid \forall S' \in \mathcal{S}(G,R) : S'\|G \sqsubseteq S\|G\}.$$

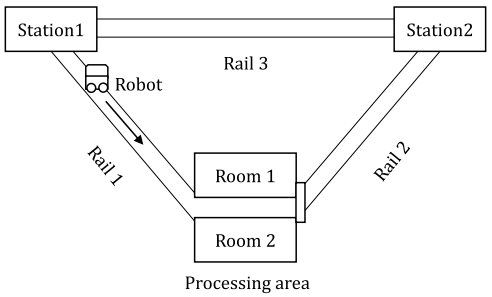


Fig. 2. Manufacturing system.

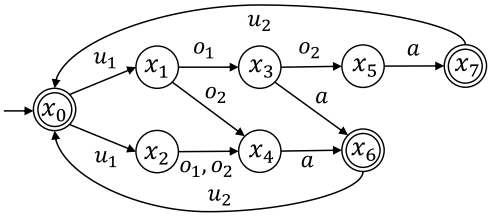


Fig. 3. System model G.

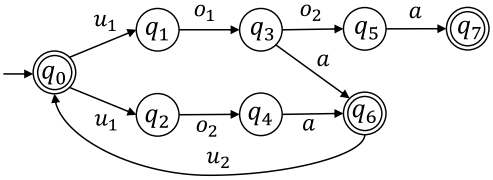


Fig. 4. Specification R.

Here, we provide an example to show the motivation of enforcing similarity. For the sake of simplicity, we assume that the system is fully observed.

Example 2: For a robot working in a manufacturing system shown in Fig. 2, which consists of two stations, two processing rooms, and three rails, we consider the nondeterministic model G shown in Fig. 3, where $\Sigma = \Sigma_o = \{o_1, o_2, a, u_1, u_2\}$. There are two kinds of parts A and B in station 1. Initially, the robot is in station 1, and moves to processing area through rail 1 after picking a part (event u_1), where the picked part cannot be identified. The transitions $x_0 \xrightarrow{u_1} x_1$ and $x_0 \xrightarrow{u_1} x_2$ denote that the robot moves to processing area with parts A and B, respectively. In processing area, the robot performs two kinds of operations o_1 and o_2 in rooms 1 and 2, respectively. After performing o_1 or o_2 (states x_3 and x_4), the robot can directly move to station 2 through rail 2 to unload the processed part (event a). Then the robot goes back to station 1 via rail 3 (event u_2). Besides, if o_1 is performed on part A (state x_3), the robot can further process the part by o_2. Then it unloads the processed part in station 2 and moves back to station 1. The events o_1, o_2, and a are controlled by opening or closing the entrances of room 1, room 2, and rail 2, respectively. The robot can freely travel back and forth in rails 1 and 3, that is, $\Sigma_{uc} = \{u_1, u_2\}$. G is marked when the robot is in a station (states x_0, x_6, and x_7).

As the specification, we consider the nondeterministic automaton R shown in Fig. 4. Suppose that some preparations are needed for supporting the operation of the robot, but the preparations for o_1 and o_2 cannot be done simultaneously.

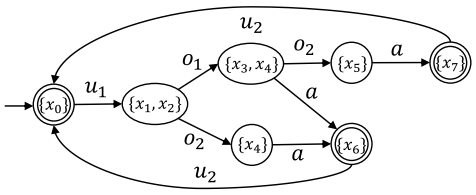


Fig. 5. Determinized system G_{det}.

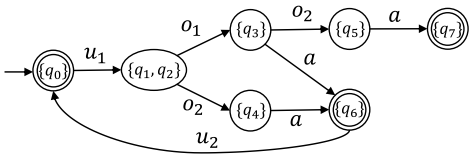


Fig. 6. Determinized specification R_{det}.

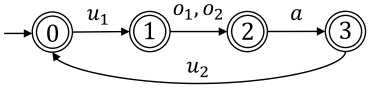


Fig. 7. Deterministic supervisor S_{det}.

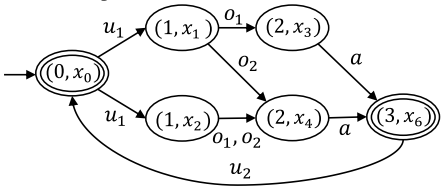


Fig. 8. Supervised system $S_{det}\|G$.

Then, after the occurrence of u_1, regardless of the kind of the picked part, the choice between the operations o_1 and o_2 is prohibited. Besides, if part A is processed by both the two operations o_1 and o_2, then the robot has to stay in station 2 after unloading.

Let us transform the nondeterministic models G and R into the language equivalent deterministic ones G_{det} and R_{det} shown in Figs. 5 and 6, respectively, and try to solve the problem using the conventional supervisory control theory for deterministic systems [2]. We can synthesize a maximally permissive deterministic supervisor S_{det} shown in Fig. 7, which achieves the supremal $L_m(G_{det})$-closed and controllable sublanguage of $L_m(R_{det})$. By applying S_{det} to the original nondeterministic model G, the supervised system $S_{det}\|G$ is obtained as shown in Fig. 8. In $S_{det}\|G$, although the language inclusion relations $L(S_{det}\|G) \subseteq L(R)$ and $L_m(S_{det}\|G) \subseteq L_m(R)$ are achieved, the requirement "o_1 and o_2 are not selectable after u_1" is not met. To satisfy this requirement, a similarity-enforcing supervisor is needed.

IV. Algorithm NBSR

In this section, we introduce an algorithm named NBSR (nonblocking state removal), which iteratively removes states of a supervisor that violate the first condition of admissibility or cause blockingness. The algorithm NBSR computes a nonblocking supervisor from a possibly blocking one. In the ith iteration, whether or not the supervisor $S^i = (Y^i, \Sigma, \beta^i, Y_0^i, Y^i)$ is nonblocking and satisfies the first condition of admissibility

Algorithm 1 NBSR

Require: $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma), S = (Y, \Sigma, \beta, Y_0, Y) \in \mathcal{A}(\Sigma)$

1: $S^1 = (Y^1, \Sigma, \beta^1, Y_0^1, Y^1) \leftarrow S, S^{NB} \leftarrow$ null, $i \leftarrow 1$
2: **while** $S^{NB} =$ null **do**
3: $\quad Y^{i+1} \leftarrow Y^i$
4: $\quad$ Compute $S^i\|G = (Y^i \times X, \Sigma, \xi^i, Y_0^i \times X_0, Y^i \times X_m)$
5: $\quad$ **for all** $(y,x) \in Reach(S^i\|G)$ **do**
6: $\quad\quad$ **if** $[\exists \sigma \in \Sigma_{uc} : \alpha(x,\sigma) \neq \emptyset \wedge \beta^i(y,\sigma) = \emptyset] \vee [\forall s \in \Sigma^* : \xi^i((y,x),s) \cap Y^i \times X_m = \emptyset]$ **then**
7: $\quad\quad\quad Y^{i+1} \leftarrow Y^{i+1} - \{y\}$
8: $\quad\quad$ **end if**
9: $\quad$ **end for**
10: $\quad$ **if** $Y^{i+1} = Y^i$ **then**
11: $\quad\quad S^{NB} \leftarrow S^i$
12: $\quad$ **else if** $Y^{i+1} \cap Y_0^i = \emptyset$ **then**
13: $\quad\quad S^{NB} \leftarrow$ undefined
14: $\quad$ **else**
15: $\quad\quad$ Compute the subautomaton $S^{i+1} = (Y^{i+1}, \Sigma, \beta^{i+1}, Y_0^{i+1}, Y^{i+1})$ of S^i with respect to Y^{i+1}
16: $\quad\quad i \leftarrow i+1$
17: $\quad$ **end if**
18: **end while**
19: **return** S^{NB}

is verified. The synchronous composition $S^i\|G$ of S^i and G is constructed in line 4, where, for each $(y,x) \in Y^i \times X$ and each $\sigma \in \Sigma$, $\xi^i((y,x),\sigma) = \beta^i(y,\sigma) \times \alpha(x,\sigma)$ is computed. Since $|Y^i \times X|$, $|\beta^i(y,\sigma)|$, and $|\alpha(x,\sigma)|$ are at most $|Y||X|$, $|Y|$, and $|X|$, respectively, the computational complexity of constructing $S^i\|G$ is $O(|Y|^2|X|^2|\Sigma|)$. In lines 5–9, each reachable state (y,x) of $S^i\|G$ is examined to decide whether y should be removed or not. All the states reachable from (y,x) are explored to verify if y violates the first condition of admissibility or causes blockingness. This requires to execute all the transitions from (y,x) in $S^i\|G$, whose number is bounded by $|X|^2|Y|^2|\Sigma|$. Then, the computational complexity in lines 5–9 is $O(|Y|^3|X|^3|\Sigma|)$. The set of remaining states can be written as

$$\begin{aligned} NB_G^{S^i}(Y^i) \;=\; & \{y \in Y^i \mid \forall (y,x) \in Reach(S^i\|G) : \\ & [\forall \sigma \in \Sigma_{uc} : \alpha(x,\sigma) \neq \emptyset \Rightarrow \beta^i(y,\sigma) \neq \emptyset] \wedge \\ & [\exists s \in \Sigma^* : \xi^i((y,x),s) \cap Y^i \times X_m \neq \emptyset]\}, \end{aligned} \quad (1)$$

namely, $Y^{i+1} = NB_G^{S^i}(Y^i)$. If $Y^{i+1} = Y^i$, which implies that S^i is nonblocking, then the algorithm outputs the supervisor S^i. If $Y^{i+1} \neq Y^i$ and $Y^{i+1} \cap Y_0^i = \emptyset$, the subautomaton of S^i with respect to Y^{i+1} cannot be constructed, so the algorithm outputs "undefined". In the case where $Y^{i+1} \neq Y^i$ and $Y^{i+1} \cap Y_0^i \neq \emptyset$, the subautomaton S^{i+1} of S^i with respect to Y^{i+1} is constructed, where, for each $y' \in Y^{i+1}$ and each $\sigma' \in \Sigma$, $\beta^{i+1}(y',\sigma')$ is computed. The computational complexity of constructing S^{i+1} is $O(|Y|^2|\Sigma|)$. Although the "bad" states in $Y^i - Y^{i+1}$ are removed from S^i to form S^{i+1}, such a state removal operation on S^i might eliminate transitions caused by uncontrollable events, and/or might block some path of $S^i\|G$ that leads to marked states. Hence, S^{i+1} does not necessarily satisfy the first condition of admissibility or is possibly blocking,

so the operation should be iteratively performed. Since the number of states of the supervisor S^i monotonically decreases, the algorithm NBSR terminates within at most $|Y|$ iterations. Therefore, the complexity of the algorithm NBSR is $O(|Y|^4|X|^3|\Sigma|)$.

For the sake of notational simplicity, we let $\mathrm{NBSR}(S,G)$ denote the output of the algorithm NBSR when it is applied to a supervisor $S \in \mathcal{A}(\Sigma)$. Note that $\mathrm{NBSR}(S,G)$ is either a supervisor $S^{NB} \in \mathcal{A}(\Sigma)$ or "undefined". By the definition of $NB_G^S(Y)$, we have the following lemma.

Lemma 1: For the system $G=(X,\Sigma,\alpha,X_0,X_m) \in \mathcal{A}(\Sigma)$, a supervisor $S=(Y,\Sigma,\beta,Y_0,Y) \in \mathcal{A}(\Sigma)$ satisfies the first condition of admissibility and is nonblocking if and only if $NB_G^S(Y)=Y$.

Proof: Let $\xi : (Y \times X) \times \Sigma \to 2^{Y \times X}$ be the transition function of $S\|G$.

($\Rightarrow$) By (1), we have $NB_G^S(Y) \subseteq Y$. To show $Y \subseteq NB(Y)_G^S$, for any $y \in Y$, we consider any $(y,x) \in Reach(S\|G)$. Since S satisfies the first condition of admissibility, for any $\sigma \in \Sigma_{uc}$ such that $\alpha(x,\sigma) \neq \emptyset$, we have $\beta(y,\sigma) \neq \emptyset$. Besides, there exists $s \in \Sigma^*$ such that $\xi((y,x),s) \cap Y \times X_m \neq \emptyset$ by the nonblockingness of S. It follows that $y \in NB_G^S(Y)$.

($\Leftarrow$) We consider any $(y,x) \in Reach(S\|G)$. Since $y \in Y = NB_G^S(Y)$, there exists $s \in \Sigma^*$ such that $\xi((y,x),s) \cap Y \times X_m \neq \emptyset$. It follows that S is nonblocking. Besides, for any $\sigma \in \Sigma_{uc}$ such that $\alpha(x,\sigma) \neq \emptyset$, we have $\beta(y,\sigma) \neq \emptyset$. Hence, S satisfies the first condition of admissibility. ∎

To show that the algorithm NBSR can be used for solving the nonblocking similarity control problem, we need the following proposition.

Proposition 2: [20] For any $G_1, G_1', G_2, G_2' \in \mathcal{A}(\Sigma)$ such that $G_1 \sqsubseteq G_2$ and $G_1' \sqsubseteq G_2'$, it holds that $G_1\|G_1' \sqsubseteq G_2\|G_2'$.

Then, we show in the following proposition that a solution to the nonblocking similarity control problem can be computed by applying NBSR to an appropriate supervisor.

Proposition 3: For the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$, we consider a supervisor $S \in \mathcal{A}(\Sigma)$ that satisfies the second condition of admissibility and $S\|G \sqsubseteq R$. If $\mathrm{NBSR}(S,G) = S^{NB} \in \mathcal{A}(\Sigma)$, then $S^{NB} \in \mathcal{NS}(G,R)$.

Proof: Since S^{NB} is a subautomaton of S, we have $S^{NB} \sqsubseteq S$. Since the order $\sqsubseteq$ on $\mathcal{A}(\Sigma)$ is reflexive, $G \sqsubseteq G$ holds. It follows from Proposition 2 that $S^{NB}\|G \sqsubseteq S\|G$. By $S\|G \sqsubseteq R$ and the transitivity of $\sqsubseteq$, we have $S^{NB}\|G \sqsubseteq R$. By Lemma 1, $S^{NB} = \mathrm{NBSR}(S,G) \in \mathcal{A}(\Sigma)$ satisfies the first condition of admissibility and is nonblocking. Since S satisfies the second condition of admissibility, so does S^{NB} by Definition 1. It follows that $S^{NB} \in \mathcal{NS}(G,R)$. ∎

We provide the following example to illustrate how the algorithm NBSR works.

Example 3: For the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$ shown in Figs. 9 (a) and (b), respectively, where $\Sigma = \{a,b,c,d,e,u\}$ and $\Sigma_{uc} = \Sigma_{uo} = \{u\}$, we consider a $\Sigma_{uc} \cap \Sigma_{uo}$-compatible supervisor $S \in \mathcal{A}(\Sigma)$ shown in Fig. 10, where at each state of S, a self-loop by u is defined but omitted for the sake of simplicity. The reachable part of the supervised system $S\|G$ is given in Fig. 11, from which we can verify that S satisfies $S \in \mathcal{S}(G,R)$, but it is blocking.

We apply the algorithm NBSR to S. In the first iteration, we let $S^1 = (Y^1,\Sigma,\beta^1,Y_0^1,Y^1) = S$. Since any marked state cannot be reached from (y_6,x_6), state y_6 of S^1 is removed. Then, the supervisor $S^2 \in \mathcal{A}(\Sigma)$ is constructed as the subautomaton of S^1 with respect to $Y^2 = Y^1 - \{y_6\}$. The supervised system $S^2\|G$ is obtained by eliminating states (y_6,x_6) and (y_6,x_7) from $S\|G$. Since the unique marked successor state (y_6,x_7) of state (y_3,x_4) is eliminated, the supervisor S^2 is still blocking. In the second iteration, state y_3 of S^2 is removed, and the subautomaton $S^3 \in \mathcal{A}(\Sigma)$ of S^2 with respect to $Y^3 = Y^2 - \{y_3\}$ is constructed. The supervised system $S^3\|G$ is shown in Fig. 12. In the third iteration, any state of S^3 is not removed, since no state of S^3 causes blockingness or violates the first condition of admissibility. Finally, the algorithm outputs the supervisor S^3, which solves the nonblocking similarity control problem for G and R.

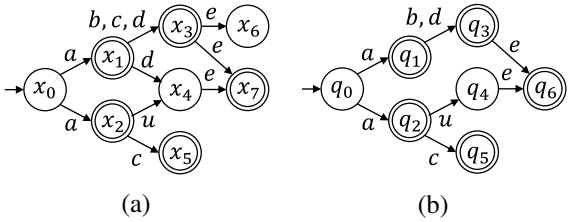


Fig. 9. (a) System G. (b) Specification R.

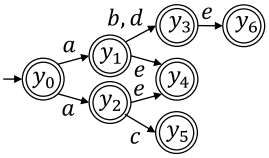


Fig. 10. Supervisor S.

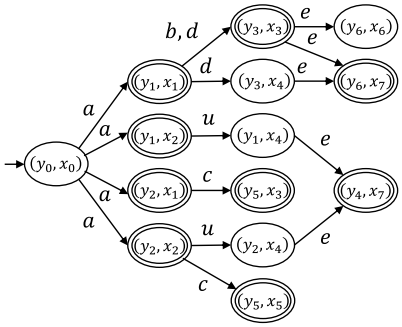


Fig. 11. Supervised system $S\|G$.

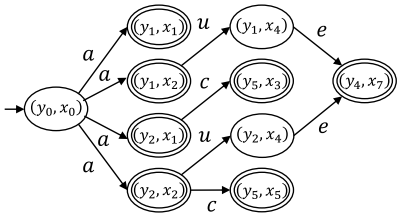


Fig. 12. Supervised system $S^3\|G$.

V. Synthesis of Supervisor without Imposing Nonblockingness

Proposition 3 has shown conditions on an input supervisor $S \in \mathcal{A}(\Sigma)$ of the algorithm NBSR, which guarantee that if the output $\mathrm{NBSR}(S,G)$ is $S^{NB} \in \mathcal{A}(\Sigma)$, then it is a solution to

the nonblocking similarity control problem for $G \in \mathcal{A}(\Sigma)$ and $R \in \mathcal{A}(\Sigma)$. Now we need to synthesize an appropriate candidate supervisor to which the algorithm NBSR is applied. The problem of synthesizing a supervisor $S \in \mathcal{S}(G,R)$ is studied in [24], where the marking of states is not considered. As the supervisor computed by the algorithm NBSR is expected to be maximally permissive, we synthesize an element of $\mathcal{MPS}(G,R)$ as the input of NBSR. For this purpose, we modify the results of [24] by taking marked states into account.

First, we present a necessary and sufficient condition for $\mathcal{S}(G,R) \neq \emptyset$. We extend the notion of Σ_{uc}-similarity of [21] by taking the presence of the marked state sets of G and R into account:

Definition 5: For the system $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and the specification $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$, a relation $\Phi \subseteq X \times Q$ is said to be a Σ_{uc}-simulation relation from G to R if

- $\forall x_0 \in X_0, \exists q_0 \in Q_0 : (x_0, q_0) \in \Phi$,
- $\forall (x,q) \in \Phi, \forall \sigma \in \Sigma_{uc}, \forall x' \in \alpha(x,\sigma), \exists q' \in \delta(q,\sigma) :$ $(x',q') \in \Phi$, and
- $\forall (x,q) \in \Phi : x \in X_m \Rightarrow q \in Q_m$.

If there exists a Σ_{uc}-simulation relation from G to R, G is said to be Σ_{uc}-simulated by R.

The first and second conditions of a Σ_{uc}-simulation relation mean that all the sequences of transitions by uncontrollable events from initial states of G are allowed by R. The third condition corresponds to that of a simulation relation. For any pair (x,q) of a Σ_{uc}-simulation relation, q should be marked if x is marked. The following proposition that extends Proposition 1 of [24] shows that Σ_{uc}-similarity is a necessary and sufficient condition for $\mathcal{S}(G,R) \neq \emptyset$.

Proposition 4: For the system $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and the specification $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$, $\mathcal{S}(G,R) \neq \emptyset$ if and only if G is Σ_{uc}-simulated by R.

Proof: ($\Leftarrow$) Let $S_u = (Y_u, \Sigma, \beta_u, Y_{u0}, Y_u) \in \mathcal{A}(\Sigma)$ be such that $Y_u = Y_{u0} = \{y_0\}$ and

$$\beta(y_0,\sigma) = \begin{cases} \{y_0\}, & \text{if } \sigma \in \Sigma_{uc} \\ \emptyset, & \text{otherwise} \end{cases}$$

for each $\sigma \in \Sigma$. Since S_u only has a single state y_0 at which self-loop transitions caused by uncontrollable events are defined, it is admissible. Since G is Σ_{uc}-simulated by R, there exists a Σ_{uc}-simulation relation $\Phi_u \subseteq X \times Q$ from G to R. We show that the relation $\Psi_u \subseteq (Y_u \times X) \times Q$ defined as

$$\Psi_u = \{((y_0,x),q) \in (Y_u \times X) \times Q \mid (x,q) \in \Phi_u\}$$

is a simulation relation from $S_u \| G$ to R.

- We consider any $(y_0,x_0) \in Y_{u0} \times X_0$. Since Φ_u is a Σ_{uc}-simulation relation from G to R, there exists $q_0 \in Q_0$ such that $(x_0,q_0) \in \Phi_u$. It follows that $((y_0,x_0),q_0) \in \Psi_u$.
- For any $((y_0,x),q) \in \Psi_u$ and any $\sigma \in \Sigma$, we consider any $(y_0,x') \in \xi_u((y_0,x),\sigma)$, where $\xi_u : (Y_u \times X) \times \Sigma \to 2^{Y_u \times X}$ is the transition function of $S_u \| G$. We have $(x,q) \in \Phi_u$, $y_0 \in \beta_u(y_0,\sigma)$ and $x' \in \alpha(x,\sigma)$. It follows from $\beta_u(y_0,\sigma) \neq \emptyset$ that $\sigma \in \Sigma_{uc}$. Since Φ_u is a Σ_{uc}-simulation relation from G to R, there exists $q' \in \delta(q,\sigma)$ such that $(x',q') \in \Phi_u$. It follows that $((y_0,x'),q') \in \Psi_u$.
- For any $((y_0,x),q) \in \Psi_u$ such that $(y_0,x) \in Y_u \times X_m$, since $(x,q) \in \Phi_u$ and Φ_u is a Σ_{uc}-simulation relation from G to R, we have $q \in Q_m$.

Hence, we have $S_u \in \mathcal{S}(G,R) \neq \emptyset$.

($\Rightarrow$) Let $S = (Y, \Sigma, \beta, Y_0.Y) \in \mathcal{A}(\Sigma)$ be any supervisor such that $S \in \mathcal{S}(G,R) \neq \emptyset$. Since $S \| G \sqsubseteq R$, there exists a simulation relation $\Psi \subseteq (Y \times X) \times Q$ from $S \| G$ to R. We show that the relation $\Phi \subseteq X \times Q$ defined as

$$\begin{aligned} \Phi \quad = \quad & \{(x,q) \in X \times Q \mid \exists y \in Y : \\ & ((y,x),q) \in \Psi \wedge (y,x) \in Reach(S\|G)\} \end{aligned}$$

is a Σ_{uc}-simulation relation from G to R.

- We consider any $x_0 \in X_0$. For any $y_0 \in Y_0$, we have $(y_0,x_0) \in Y_0 \times X_0 \subseteq Reach(S\|G)$. Since Ψ is a simulation relation from $S\|G$ to R, there exists $q_0 \in Q_0$ such that $((y_0,x_0),q_0) \in \Psi$. It follows that $(x_0,q_0) \in \Phi$.
- For any $(x,q) \in \Phi$ and any $\sigma \in \Sigma_{uc}$, we consider any $x' \in \alpha(x,\sigma)$. Since $(x,q) \in \Phi$, there exists $y \in Y$ such that $((y,x),q) \in \Psi$ and $(y,x) \in Reach(S\|G)$. As S is admissible, it follows from $x' \in \alpha(x,\sigma) \neq \emptyset$ that $\beta(y,\sigma) \neq \emptyset$. For any $y' \in \beta(y,\sigma)$, it holds that $(y',x') \in \beta(y,\sigma) \times \alpha(x,\sigma) = \xi((y,x),\sigma) \subseteq Reach(S\|G)$, where $\xi : (Y \times X) \times \Sigma \to 2^{Y \times X}$ is the transition function of $S\|G$. Moreover, since Ψ is a simulation relation from $S\|G$ to R, there exists $q' \in \delta(q,\sigma)$ such that $((y',x'),q') \in \Psi$. It follows that $(x',q') \in \Phi$.
- For any $(x,q) \in \Phi$ such that $x \in X_m$, there exists $y \in Y$ such that $((y,x),q) \in \Psi$. Since Ψ is a simulation relation from $S\|G$ to R, we have $q \in Q_m$.

Hence, G is Σ_{uc}-simulated by R. ■

Σ_{uc}-similarity is a reasonable condition. When it is not satisfied, the specification R is violated, even if all the controllable events are disabled.

To verify whether or not G is Σ_{uc}-simulated by R, we define a function $\tilde{F} : 2^{X \times Q} \to 2^{X \times Q}$ as

$$\begin{aligned} \tilde{F}(W) \quad = \quad & \{(x,q) \in W \mid [x \in X_m \Rightarrow q \in Q_m] \wedge [\forall \sigma \in \Sigma_{uc}, \\ & \forall x' \in \alpha(x,\sigma), \exists q' \in \delta(q,\sigma) : (x',q') \in W]\}, \quad (2) \end{aligned}$$

where $W \in 2^{X \times Q}$ is an arbitrary relation over $X \times Q$. $\tilde{F}$ is obtained by imposing the additional condition

$$x \in X_m \Rightarrow q \in Q_m$$

on the function $F : 2^{X \times Q} \to 2^{X \times Q}$ defined in [21]. It can be verified that $\tilde{F}$ is monotone, namely,

$$\forall W, W' \subseteq X \times Q : W \subseteq W' \Rightarrow \tilde{F}(W) \subseteq \tilde{F}(W').$$

Since $(2^{X \times Q}, \subseteq)$ is a complete lattice and $\tilde{F}$ is monotone, it follows from Tarski's fixpoint theorem [29] that there exists the unique maximal fixpoint $W^{\uparrow}$ of $\tilde{F}$. As the state sets X and Q of G and R, respectively, are finite, the unique maximal fixpoint $W^{\uparrow} \in 2^{X \times Q}$ of $\tilde{F}$ is computed by

$$W^{\uparrow} = \bigcap_{i \geq 0} \tilde{F}^i(X \times Q) \quad (3)$$

after at most $|X||Q|$ iterations, where $\tilde{F}^0(X \times Q) = X \times Q$ and $\tilde{F}^i(X \times Q) = \tilde{F}(\tilde{F}^{i-1}(X \times Q))$ for each $i > 0$.

The following proposition is obtained by straightforwardly extending Lemma 2 and Theorem 2 of [21], which indicates that whether or not G is Σ_{uc}-simulated by R can be checked by $W^{\uparrow}$.

Proposition 5: Let the system and the specification be given as $G=(X,\Sigma,\alpha,X_0,X_m)\in\mathcal{A}(\Sigma)$ and $R=(Q,\Sigma,\delta,Q_0,Q_m)\in\mathcal{A}(\Sigma)$, respectively. For the function $\tilde{F}:2^{X\times Q}\to 2^{X\times Q}$ defined by (2), the following statements hold.

- For any $\Phi\subseteq X\times Q$, Φ is a Σ_{uc}-simulation relation from G to R if and only if $\Phi=\tilde{F}(\Phi)$ and
$$\forall x_0\in X_0,\exists q_0\in Q_0:(x_0,q_0)\in\Phi.$$
- G is Σ_{uc}-simulated by R if and only if the unique maximal fixpoint $W^{\uparrow}$ of the function $\tilde{F}$ satisfies
$$\forall x_0\in X_0,\exists q_0\in Q_0:(x_0,q_0)\in W^{\uparrow}. \tag{4}$$

Since $\mathcal{NS}(G,R)\subseteq\mathcal{S}(G,R)$, by Proposition 4, Σ_{uc}-similarity is a necessary condition for $\mathcal{NS}(G,R)\neq\emptyset$. That is, if G is not Σ_{uc}-simulated by R, then the nonblocking similarity control problem is not solvable. Hereafter, we assume that G is Σ_{uc}-simulated by R, so that we have $\mathcal{S}(G,R)\neq\emptyset$. Besides, $W^{\uparrow}$ satisfies (4) and it is a Σ_{uc}-simulation relation from G to R.

The synthesis result of [24] can be directly applied to synthesize a maximally permissive similarity-enforcing supervisor in $\mathcal{MPS}(G,R)$. We define the set of admissible control patterns on unobservable events as $\Gamma_{uo}=\{\gamma\in 2^{\Sigma_{uo}}\mid \Sigma_{uc}\cap\Sigma_{uo}\subseteq\gamma\}$. For any $W\subseteq X\times Q$, we let $\pi_X(W)=\{x\in X\mid\exists q\in Q:(x,q)\in W\}$. Besides, for any $\mathcal{W}\subseteq 2^{W^{\uparrow}}$, $\min\mathcal{W}\subseteq\mathcal{W}$ denotes the set of minimal elements of $\mathcal{W}$, that is,

$$\min\mathcal{W}=\{W\in\mathcal{W}\mid\nexists W'\in\mathcal{W}:W'\subseteq W\wedge W'\neq W\}.$$

A state of the supervisor to be synthesized is represented as a 3-tuple $y=(W_1^y,\gamma_{uo}^y,W_2^y)\in 2^{W^{\uparrow}}\times\Gamma_{uo}\times 2^{W^{\uparrow}}$, whose elements can be informally explained as follows:

- $W_1^y\in 2^{W^{\uparrow}}$ is used to estimate the current state of G at the moment an observable event occurs. In each element (x,q) of W_1^y, x is a candidate for the current state of G, and q is a state of R, which is related to x by $W^{\uparrow}$ such that transitions from x should be simulated by those from q.
- $\gamma_{uo}^y\in\Gamma_{uo}$ is the set of unobservable events to be enabled at y, which contains all the uncontrollable and unobservable events.
- Some enabled unobservable events in γ_{uo}^y may occur in G before the next occurrence of an observable event. $W_2^y\in 2^{W^{\uparrow}}$ is used to estimate states that are reachable from those in $\pi_X(W_1^y)$ by enabled unobservable events. Similar to W_1^y, W_2^y consists of pairs of states x and q that are related by $W^{\uparrow}$.

To compute W_2^y from W_1^y and γ_{uo}^y, a function $U:2^{W^{\uparrow}}\times\Gamma_{uo}\to 2^{2^{W^{\uparrow}}}$ is defined as

$$\begin{aligned}U(W,\gamma) &= \{W'\in 2^{W^{\uparrow}}\mid W\subseteq W'\wedge[\forall(x,q)\in W',\\ &\quad\forall\sigma\in\gamma,\forall x'\in\alpha(x,\sigma),\exists q'\in\delta(q,\sigma):\\ &\quad(x',q')\in W']\}\end{aligned}$$

for each $W\in 2^{W^{\uparrow}}$ and each $\gamma\in\Gamma_{uo}$. Each element W' of $U(W,\gamma)$ is a certain invariant set that includes W. For any pair $(x,q)\in W'$, if there is a transition from x to $x'\in\alpha(x,\sigma)$ by an unobservable event $\sigma\in\gamma$, then a corresponding transition from q to $q'\in\delta(q,\sigma)$ such that $(x',q')\in W'$ must be defined, namely, all transitions from x by σ are allowed by the specification R. Hence, the nonemptiness of $U(W,\gamma)$ indicates that, for the estimated states in $\pi_X(W)$, γ is a "safe" control pattern on unobservable events. For each $x'\in\alpha(x,\sigma)$, $q'\in\delta(q,\sigma)$ such that $(x',q')\in W'$ is not unique in general. It is sufficient for W' to include one such pair (x',q'). In addition, state pairs that cannot be reached from any pair in W are irrelevant. These are the reasons that we only consider minimal elements of $U(W,\gamma)$.

Using $W^{\uparrow}$ obtained by (3), we can synthesize a nondeterministic supervisor

$$S_{W^{\uparrow}}=(Y_{W^{\uparrow}},\Sigma,\beta_{W^{\uparrow}},Y_{W^{\uparrow}0},Y_{W^{\uparrow}})\in\mathcal{A}(\Sigma) \tag{5}$$

as follows [24]:

- The state set $Y_{W^{\uparrow}}\subseteq 2^{W^{\uparrow}}\times\Gamma_{uo}\times 2^{W^{\uparrow}}$ is defined as
$$\begin{aligned}Y_{W^{\uparrow}} &= \{(W_1^y,\gamma_{uo}^y,W_2^y)\in 2^{W^{\uparrow}}\times\Gamma_{uo}\times 2^{W^{\uparrow}}\mid\\ &\quad W_1^y\neq\emptyset\wedge W_2^y\in\min U(W_1^y,\gamma_{uo}^y)\}.\end{aligned}$$
- The initial state set is defined as
$$\begin{aligned}Y_{W^{\uparrow}0} &= \{(W_1^{y_0},\gamma_{uo}^{y_0},W_2^{y_0})\in Y_{W^{\uparrow}}\mid|W_1^{y_0}|=|X_0|\wedge\\ &\quad[\forall x_0\in X_0,\exists q_0\in Q_0:(x_0,q_0)\in W_1^{y_0}]\}.\end{aligned}$$
- The nondeterministic transition function $\beta_{W^{\uparrow}}:Y_{W^{\uparrow}}\times\Sigma\to 2^{Y_{W^{\uparrow}}}$ is defined as
$$\beta_{W^{\uparrow}}(y,\sigma)=\begin{cases}\{(W_1^{y'},\gamma_{uo}^{y'},W_2^{y'})\in Y_{W^{\uparrow}}\mid & \\ W_1^{y'}\in\min N(W_2^y,\sigma)\}, & \text{if } \sigma\in\Sigma_{W^{\uparrow}}(y)\\ \{y\}, & \text{if } \sigma\in\gamma_{uo}^y\\ \emptyset, & \text{otherwise}\end{cases}$$
for each $y=(W_1^y,\gamma_{uo}^y,W_2^y)\in Y_{W^{\uparrow}}$ and each $\sigma\in\Sigma$, where $N:2^{W^{\uparrow}}\times\Gamma_o\to 2^{2^{W^{\uparrow}}}$ is defined as
$$\begin{aligned}N(W,\sigma) &= \{W'\in 2^{W^{\uparrow}}\mid\forall(x,q)\in W,\forall x'\in\alpha(x,\sigma),\\ &\quad\exists q'\in\delta(q,\sigma):(x',q')\in W'\}\end{aligned}$$
for each $W\in 2^{W^{\uparrow}}$ and each $\sigma\in\Sigma_o$, and
$$\begin{aligned}\Sigma_{W^{\uparrow}}(y) &= \{\sigma\in\Sigma_o\mid\alpha(\pi_X(W_2^y),\sigma)\neq\emptyset\wedge[\forall(x,q)\in W_2^y,\\ &\quad\forall x'\in\alpha(x,\sigma),\exists q'\in\delta(q,\sigma):(x',q')\in W^{\uparrow}]\}.\end{aligned}$$

Note that, by (4), there exists at least one candidate $W_0\in 2^{W^{\uparrow}}$ for the first element of an initial state of $S_{W^{\uparrow}}$. That is, $W_0\neq\emptyset$, $|W_0|=|X_0|$, and

$$\forall x_0\in X_0,\exists q_0\in Q_0:(x_0,q_0)\in W_0$$

hold. Since $W^{\uparrow}$ is a Σ_{uc}-simulation relation from G to R, we have $U(W_0,\Sigma_{uc}\cap\Sigma_{uo})\neq\emptyset$. Then, there exists a minimal element $W_0'\in\min U(W_0,\Sigma_{uc}\cap\Sigma_{uo})$ such that $(W_0,\Sigma_{uc}\cap\Sigma_{uo},W_0')\in Y_{W^{\uparrow}0}\neq\emptyset$. If $\mathcal{S}(G,R)\neq\emptyset$, which implies that $Y_{W^{\uparrow}0}\neq\emptyset$, then $S_{W^{\uparrow}}$ can be synthesized.

For any $y=(W_1^y,\gamma_{uo}^y,W_2^y)\in Y_{W^{\uparrow}}$, the set $\Sigma_{W^{\uparrow}}(y)$ is determined by W_2^y. $\Sigma_{W^{\uparrow}}(y)$ is the set of observable events such that

all transitions from $\pi_X(W_2^y)$ by these observable events are allowed by the specification R.

It is shown in the proof of Theorem 24 of [24] that $S_{W^\uparrow}$ is a maximally permissive similarity-enforcing supervisor.

Proposition 6: Let the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$ be such that G is Σ_{uc}-simulated by R. Then, the supervisor $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ defined by (5) satisfies $S_{W^\uparrow} \in \mathcal{MPS}(G,R)$.

Example 4: Recall the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$ shown in Figs. 9 (a) and (b), respectively. Let X and Q be the state sets of G and R, respectively. The unique maximal fixpoint $W^\uparrow$ of the function $\tilde{F}: 2^{X\times Q} \to 2^{X\times Q}$ defined by (2) is obtained as

$$\begin{aligned}
&\tilde{F}^0(X\times Q) = X\times Q;\\
&\tilde{F}^1(X\times Q) = \tilde{F}^0(X\times Q) - \{x_2\}\times\{q_0,q_1,q_3,q_4,q_5,q_6\} -\\
&\qquad\qquad \{x_1,x_2,x_3,x_5,x_7\}\times\{q_0,q_4\}\\
&\qquad\quad = \tilde{F}^2(X\times Q) = W^\uparrow.
\end{aligned}$$

With the above procedure, we synthesize $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ defined by (5), whose reachable part is shown in Fig. 13, where

$$\begin{aligned}
&\tilde{y}_0 = (\{(x_0,q_0)\},\{u\},\{(x_0,q_0)\}),\\
&\tilde{y}_1 = (\{(x_1,q_1),(x_2,q_2)\},\{u\},\{(x_1,q_1),(x_2,q_2),(x_4,q_4)\}),\\
&\tilde{y}_2 = (\{(x_1,q_2),(x_2,q_2)\},\{u\},\{(x_1,q_2),(x_2,q_2),(x_4,q_4)\}),\\
&\tilde{y}_3 = (\{(x_3,q_3)\},\{u\},\{(x_3,q_3)\}),\\
&\tilde{y}_4 = (\{(x_3,q_3),(x_4,q_3)\},\{u\},\{(x_3,q_3),(x_4,q_3)\}),\\
&\tilde{y}_5 = (\{(x_7,q_6)\},\{u\},\{(x_7,q_6)\}),\\
&\tilde{y}_6 = (\{(x_3,q_5),(x_5,q_5)\},\{u\},\{(x_3,q_5),(x_5,q_5)\}),\\
&\tilde{y}_7 = (\{(x_6,q_6),(x_7,q_6)\},\{u\},\{(x_6,q_6),(x_7,q_6)\}),
\end{aligned}$$

and the self-loop by u is omitted at each state. The supervised system $S_{W^\uparrow}\|G$ is shown in Fig. 14. We can verify that $S_{W^\uparrow}$ correctly estimates the current state of G by $\pi_X(W_2^{\tilde{y}_i})$ $(i = 0,1,\ldots,8)$. Besides, we have $S_{W^\uparrow}\|G \sqsubseteq R$ and $S_{W^\uparrow}$ is admissible. The maximal permissiveness of $S_{W^\uparrow}$ is guaranteed by Proposition 6. However, since $(\tilde{y}_7,x_6)$ is an unmarked deadlock state of $S_{W^\uparrow}\|G$, $S_{W^\uparrow}$ is blocking.

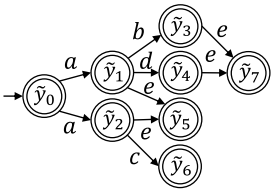


Fig. 13. Supervisor $S_{W^\uparrow}$.

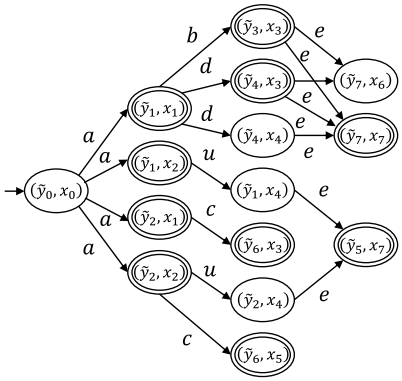


Fig. 14. Supervised system $S_{W^\uparrow}\|G$.

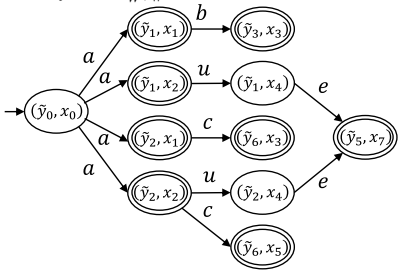


Fig. 15. Supervised system $S^{NB}_{W^\uparrow}\|G$.

VI. Synthesis of Maximally Permissive Nonblocking Supervisor

In this section, we show that the supervisor $S_{W^\uparrow}$ defined by (5) can be used as an input of the algorithm NBSR for synthesizing a maximally permissive nonblocking supervisor in $\mathcal{MPNS}(G,R)$. For any $S \in \mathcal{NS}(G,R) \subseteq \mathcal{S}(G,R)$, since $S_{W^\uparrow} \in \mathcal{MPS}(G,R)$ as shown in Proposition 6, we have $S\|G \sqsubseteq S_{W^\uparrow}\|G$. Then, our objective is to prove that $S\|G \sqsubseteq S^{NB}_{W^\uparrow}\|G$ still holds for the output supervisor $\mathrm{NBSR}(S_{W^\uparrow},G) = S^{NB}_{W^\uparrow}$. However, it is not straightforward.

$$\begin{aligned}
\forall S \in \mathcal{MPS}(G,R) : \mathrm{NBSR}(S,G) \in \mathcal{A}(\Sigma) \Rightarrow\\
\mathrm{NBSR}(S,G) \in \mathcal{MPNS}(G,R)
\end{aligned}$$

does not hold, which means that, even if an input S of NBSR is a maximally permissive similarity-enforcing supervisor, the output supervisor $\mathrm{NBSR}(S,G)$ is not necessarily a maximally permissive nonblocking one. This can be verified in the following example.

Example 5: We apply the algorithm NBSR to the supervisor $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ shown in Fig. 13, which is synthesized for the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$ shown in Figs. 9 (a) and (b), respectively. The output supervisor $S^{NB}_{W^\uparrow} \in \mathcal{A}(\Sigma)$ is obtained after three iterations, where the states $\tilde{y}_7$ and $\tilde{y}_4$ are removed in the first and second iterations, respectively. The supervised system $S^{NB}_{W^\uparrow}\|G$ is shown in Fig. 15, from which we can verify that $S^{NB}_{W^\uparrow} \in \mathcal{NS}(G,R)$.

By comparing the supervised systems $S_{W^\uparrow}\|G$ and $S\|G$ shown in Figs. 11 and 14, respectively, it holds that $S\|G \sqsubseteq S_{W^\uparrow}\|G$ and $S_{W^\uparrow}\|G \sqsubseteq S\|G$, which mean that $S \in \mathcal{A}(\Sigma)$ shown in Fig. 10 is as permissive as $S_{W^\uparrow}$. By Proposition 6 and the transitivity of the simulation relation, we have $S, S_{W^\uparrow} \in \mathcal{MPS}(G,R)$. However, after applying NBSR, the event string ab is disabled by S^3 as shown in Fig. 12, while it is possible in $S^{NB}_{W^\uparrow}\|G$. Since $S^3\|G \sqsubseteq S^{NB}_{W^\uparrow}\|G$ holds but $S^{NB}_{W^\uparrow}\|G$ is not simulated by $S^3\|G$, $S^3 = \mathrm{NBSR}(S,G)$ is not an element of $\mathcal{MPNS}(G,R)$.

In S, state y_3 can be reached via runs $\{u\}a\{u\}b\{u\}$ and $\{u\}a\{u\}d\{u\}$, by which the strings in $\{u\}^*\{a\}\{u\}^*\{b\}\{u\}^*$ and $\{u\}^*\{a\}\{u\}^*\{d\}\{u\}^*$ are enabled, respectively. However, for the system G, the occurrences of ab and ad lead to two different situations: (i) x_3 is the unique possible state when ab occurs; and (ii) both x_3 and x_4 are possible states when ad occurs. This means that state y_3 of S is in charge of these two different situations. As shown in Example 3, state y_3 is re-

moved in the second iteration of the algorithm NBSR. Hence, the supervisor $S^3 = \mathrm{NBSR}(S,G)$ disables the event strings in $\{u\}^*\{a\}\{u\}^*\{b\}\{u\}^*$ and $\{u\}^*\{a\}\{u\}^*\{d\}\{u\}^*$. On the other hand, in $S_{W\uparrow}$, two different states $\tilde{y}_3$ and $\tilde{y}_4$ deal with the situations (i) and (ii), respectively. As a result, the event strings in $\{u\}^*\{a\}\{u\}^*\{b\}\{u\}^*$ are enabled by $S^{NB}_{W\uparrow} = \mathrm{NBSR}(S_{W\uparrow}, G)$.

Example 5 suggests that an input of the algorithm NBSR should be carefully chosen from $\mathcal{MPS}(G,R)$. Now, we show that $S_{W\uparrow}$ is a proper input of NBSR for computing a maximally permissive nonblocking supervisor. In order to do so, we identify two properties, which guarantee the maximal permissiveness of the output $S^{NB}_{W\uparrow}$.

The first property is characterized in the following definition.

Definition 6: For the system $G = (X,\Sigma,\alpha,X_0,X_m) \in \mathcal{A}(\Sigma)$ and a supervisor $S = (Y,\Sigma,\beta,Y_0,Y) \in \mathcal{A}(\Sigma)$ that satisfies the second condition of admissibility, S is said to be state-unmerged (with respect to G) if

$$\forall y \in \mathit{Reach}(S), \forall r_1, r_2 \in \mathcal{R}_y(S): \bigcup_{s\in L(r_1)} \alpha(X_0,s) = \bigcup_{s\in L(r_2)} \alpha(X_0,s).$$

Each state of a state-unmerged supervisor only handles one situation of the system. The supervisor $S \in \mathcal{A}(\Sigma)$ shown in Fig. 10 is not state-unmerged, so that the string ab is disabled by the state removal procedure, although we can verify in Fig. 15 that enabling the strings in $\{u\}^*\{a\}\{u\}^*\{b\}\{u\}^*$ does not cause blockingness.

Note that, for any state-unmerged supervisor $S = (Y,\Sigma,\beta,Y_0,Y) \in \mathcal{A}(\Sigma)$, we have

$$\begin{aligned} \forall y \in \mathit{Reach}(S): C(y) &= \bigcup_{r\in\mathcal{R}_y(S)} \left(\bigcup_{s\in L(r)} \alpha(X_0,s) \right) \\ &= \bigcup_{s'\in L(r')} \alpha(X_0,s'), \end{aligned} \tag{6}$$

where $C(y) = \{x \in X \mid (y,x) \in \mathit{Reach}(S\|G)\}$ is the set of states of G which are in the charge of y, and r' is an arbitrary run in $\mathcal{R}_y(S)$.

To identify the second property, we consider the following example.

Example 6: We consider the system $G = (X,\Sigma,\alpha,X_0,X_m) \in \mathcal{A}(\Sigma)$ with $\Sigma = \Sigma_c = \Sigma_o = \{\sigma_1,\sigma_2,\sigma_3,\sigma_4\}$ shown in Fig. 16 (a) and the specification R which only requires that the event σ_2 is disabled. Two supervisors $S_1 = (Y_1,\Sigma,\beta_1,Y_{10},Y_1), S_2 = (Y_2,\Sigma,\beta_2,Y_{20},Y_2) \in \mathcal{A}(\Sigma)$ for G are given in Figs. 16 (b) and (c), respectively. The supervised systems $S_1\|G$ and $S_2\|G$ are shown in Figs. 17 (a) and (b), respectively. We can verify that $S_1, S_2 \in \mathcal{MPS}(G,R)$. However, there does not exist a state of S_2 that enables both σ_3 and σ_4 after the occurrence of σ_1. This results in blockingness of $S_2\|G$, as shown in Fig. 17 (b). When the algorithm NBSR is applied to S_2, states y_{21} and y_{22} are removed from S_2. Although S_2 is state-unmerged, the resulting supervisor is not an element of $\mathcal{MPNS}(G,R)$.

Example 6 suggests that a stronger version of a simulation relation is needed to evaluate the permissiveness of a supervisor when nonblockingness is required.

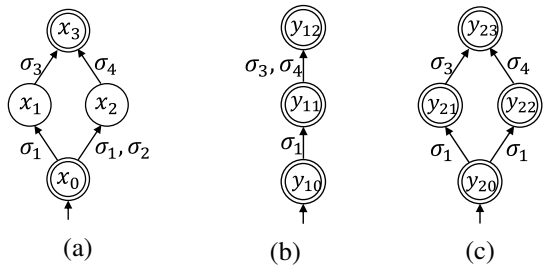


Fig. 16. (a) System G, supervisors (b) S_1 and (c) S_2.

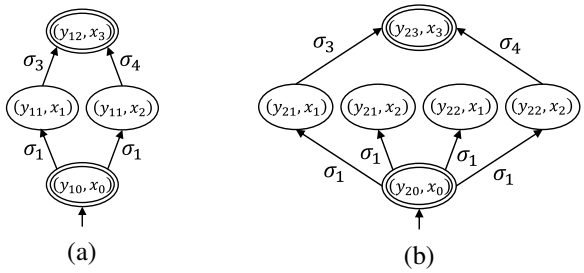


Fig. 17. Supervised systems (a) $S_1\|G$ and (b) $S_2\|G$.

Definition 7: For the system $G = (X,\Sigma,\alpha,X_0,X_m) \in \mathcal{A}(\Sigma)$ and two supervisor $S_1 = (Y_1,\Sigma,\beta_1,Y_{10},Y_1) \in \mathcal{A}(\Sigma)$ and $S_2 = (Y_2,\Sigma,\beta_2,Y_{20},Y_2) \in \mathcal{A}(\Sigma)$ that satisfy the second condition of admissibility, S_2 is said to be strongly more permissive than S_1 (with respect to G) if there exists a simulation relation $\Psi \subseteq (Y_1 \times X) \times (Y_2 \times X)$ from $S_1\|G$ to $S_2\|G$ such that, for any $((y_1,x_1),(y_2,x_2)) \in \Psi$, it holds that

- $x_1 = x_2$,
- $\mathcal{R}_{y_1}(S_1) \cap \mathcal{R}_{y_2}(S_2) \neq \emptyset$, and
- $\forall x \in X, \forall s' \in L(S_1\|G): [(y_1,x) \in \xi_1(Y_{10} \times X_0, s') \wedge (y_2,x) \in \xi_2(Y_{20} \times X_0, s')] \Rightarrow ((y_1,x),(y_2,x)) \in \Psi$,

where $\xi_1 : (Y_1 \times X) \times \Sigma \to 2^{Y_1 \times X}$ and $\xi_2 : (Y_2 \times X) \times \Sigma \to 2^{Y_2 \times X}$ are the transition functions of $S_1\|G$ and $S_2\|G$, respectively. We call such a simulation relation Ψ a strong simulation relation from $S_1\|G$ to $S_2\|G$.

Suppose that there exists a strong simulation relation Ψ from $S_1\|G$ to $S_2\|G$. For any $((y_1,x'),(y_2,x')) \in \Psi$, the second condition of a strong simulation relation requires that there exists a common run $r \in \mathcal{R}_{y_1}(S_1) \cap \mathcal{R}_{y_2}(S_2)$ that leads to y_1 and y_2 in S_1 and S_2, respectively. The third condition is imposed on y_1 and y_2 in $((y_1,x'),(y_2,x')) \in \Psi$. For any $x \in X$, if (y_1,x) and (y_2,x) can be reached by a common string $s' \in L(S_1\|G)$ as $(y_1,x) \in \xi_1(Y_{10} \times X_0, s')$ and $(y_2,x) \in \xi_2(Y_{20} \times X_0, s')$, respectively, then $((y_1,x),(y_2,x))$ has to be an element of Ψ. Since Ψ is a simulation relation from $S_1\|G$ to $S_2\|G$, for any $\sigma \in \Sigma$ with $\alpha(x,\sigma) \neq \emptyset$, if σ is enabled at y_1 by S_1, then it is also enabled at y_2 by S_2.

Remark 3: We can verify that, in Example 6, although $S_1\|G \sqsubseteq S_2\|G$ holds, there does not exist a strong simulation relation from $S_1\|G$ to $S_2\|G$. For example, a relation

$$\begin{aligned} \Phi = \{&((y_{10},x_0),(y_{20},x_0)),((y_{11},x_1),(y_{21},x_1)), \\ &((y_{11},x_2),(y_{22},x_2)),((y_{12},x_3),(y_{23},x_3))\} \end{aligned}$$

is a simulation relation from $S_1\|G$ to $S_2\|G$, where each element satisfies the first and second conditions in Definition 7. For the pair $((y_{11},x_1),(y_{21},x_1)) \in \Phi$, the state $x_2 \in X$, and

the string $\sigma_1 \in L(S_1\|G)$, we have $(y_{11},x_2) \in \xi_1(Y_{10}\times X_0,\sigma_1)$ and $(y_{21},x_2) \in \xi_2(Y_{20}\times X_0,\sigma_1)$, where $\xi_1 : (Y_1\times X)\times\Sigma \to 2^{Y_1\times X}$ and $\xi_2 : (Y_2\times X)\times\Sigma \to 2^{Y_2\times X}$ are the transition functions of $S_1\|G$ and $S_2\|G$, respectively. However, the pair $((y_{11},x_2),(y_{21},x_2))$ is not an element of Φ. Hence, the third condition in Definition 7 is not satisfied. Although all the elements of the relation

$$\hat{\Phi} = \Phi \cup \{((y_{11},x_2),(y_{21},x_2)),((y_{11},x_1),(y_{22},x_1))\}$$

meet the three conditions in Definition 7, for the pair $((y_{11},x_2),(y_{21},x_2)) \in \hat{\Phi}$ and the event $\sigma_4 \in \Sigma$, we have $(y_{12},x_3) \in \xi_1((y_{11},x_2),\sigma_4)$ and $\xi_2((y_{21},x_2),\sigma_4) = \emptyset$, which mean that the second condition of a simulation relation is not satisfied. Hence, $\hat{\Phi}$ is no longer a simulation relation from $S_1\|G$ to $S_2\|G$.

On the other hand, since the reverse relation

$$\hat{\Phi}^{-1} = \{((y_2,x),(y_1,x)) \in (Y_2\times X)\times(Y_1\times X) \mid ((y_1,x),(y_2,x)) \in \hat{\Phi}\}$$

is a strong simulation relation from $S_2\|G$ to $S_1\|G$, S_1 is strongly more permissive than S_2.

The following proposition shows that, for any two supervisors S_1 and S_2 which satisfy the second condition of admissibility, if S_2 is state-unmerged and strongly more permissive than S_1, then these two properties are preserved under each iteration of the algorithm NBSR.

Proposition 7: For the system $G = (X,\Sigma,\alpha,X_0,X_m) \in \mathcal{A}(\Sigma)$, we consider two supervisors $S_1 = (Y_1,\Sigma,\beta_1,Y_{10},Y_1) \in \mathcal{A}(\Sigma)$ and $S_2 = (Y_2,\Sigma,\beta_2,Y_{20},Y_2) \in \mathcal{A}(\Sigma)$ which satisfy the second condition of admissibility. We assume that S_2 is state-unmerged and strongly more permissive than S_1. If $Y_{10} \cap NB_G^{S_1}(Y_1) \neq \emptyset$, then the following two statements hold:

1) $Y_{20} \cap NB_G^{S_2}(Y_2) \neq \emptyset$, and
2) the subautomaton $S_2' \in \mathcal{A}(\Sigma)$ of S_2 with respect to $NB_G^{S_2}(Y_2)$ is state-unmerged and strongly more permissive than the subautomaton $S_1' \in \mathcal{A}(\Sigma)$ of S_1 with respect to $NB_G^{S_1}(Y_1)$.

The proof of Proposition 7 is given in the Appendix A.

Recall the supervisor $S_{W^\uparrow}$ defined by (5), which always enables all the uncontrollable and unobservable events. Since a strong simulation relation requires that the two supervisors being compared share the same run, hereafter, we only consider admissible and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisors without loss of generality. Note that it was shown in Proposition 1 that any admissible supervisor can be transformed into an equivalent admissible and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible one. We let $\widehat{\mathcal{MPS}}(G,R) \subseteq \mathcal{MPS}(G,R)$ be the set of all state-unmerged, admissible, and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisors which are strongly more permissive than any $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisor in $\mathcal{S}(G,R)$ (namely, strongly maximally permissive).

By Proposition 7, we can show that if the nonblocking similarity control problem is solvable, a maximally permissive nonblocking similarity-enforcing supervisor can be computed by applying the algorithm NBSR to a supervisor in $\widehat{\mathcal{MPS}}(G,R)$.

Theorem 1: We consider the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$. If $\mathcal{NS}(G,R) \neq \emptyset$, then, for any $\hat{S} = (\hat{Y},\Sigma,\hat{\beta},\hat{Y}_0,\hat{Y}) \in \widehat{\mathcal{MPS}}(G,R)$, it holds that $\mathrm{NBSR}(\hat{S},G) \in \mathcal{MPNS}(G,R)$.

Proof: We first show $\mathrm{NBSR}(\hat{S},G) \in \mathcal{A}(\Sigma)$. When the algorithm NBSR is applied to $\hat{S}$, it terminates after $n \leq |\hat{Y}|$ iterations. For each $i \in \{1,2,\ldots,n\}$, we let $\hat{S}^i$ be the result of the ith iteration. We consider any $S = (Y,\Sigma,\beta,Y_0,Y) \in \mathcal{NS}(G,R) \subseteq \mathcal{S}(G,R)$. By Proposition 1, there exists an equivalent admissible and $\Sigma_{uc}\cap\Sigma_{uo}$-compatible supervisor $S' = (Y',\Sigma,\beta',Y_0',Y') \in \mathcal{A}(\Sigma)$ such that $S\|G = S'\|G$. It follows that $S' \in \mathcal{NS}(G,R) \subseteq \mathcal{S}(G,R)$. Since S' satisfies the first condition of admissibility and is nonblocking, we have $NB_G^S(Y') = Y'$ by Lemma 1. It follows that $Y_0' \cap NB_G^S(Y') = Y_0' \cap Y' \neq \emptyset$, and S' itself is the subautomaton of S' with respect to $NB_G^S(Y')$. Since S' and $\hat{S}$ satisfy the second condition of admissibility, and $\hat{S}$ is state-unmerged and strongly more permissive than S', by Proposition 7, $\hat{S}^1 \in \mathcal{A}(\Sigma)$ holds, and $\hat{S}^1$ is state-unmerged and strongly more permissive than S'. By repeating the same argument, we can conclude that, for each $i \in \{1,2,\ldots,n\}$, $\hat{S}^i \in \mathcal{A}(\Sigma)$ holds, and $\hat{S}^i$ is state-unmerged and strongly more permissive than S'. Hence, $\mathrm{NBSR}(\hat{S},G) = \hat{S}^n \in \mathcal{A}(\Sigma)$ holds. In addition, we have $S\|G = S'\|G \sqsubseteq \hat{S}^i\|G$ for each $i \in \{1,2,\ldots,n\}$. Since $\hat{S} \in \widehat{\mathcal{MPS}}(G,R) \subseteq \mathcal{MPS}(G,R) \subseteq \mathcal{S}(G,R)$, it follows from Proposition 3 that $\mathrm{NBSR}(\hat{S},G) \in \mathcal{NS}(G,R)$. Hence, we have $\mathrm{NBSR}(\hat{S},G) \in \mathcal{MPNS}(G,R)$. ∎

The following proposition, whose proof is given in Appendix B, show that $S_{W^\uparrow}$ is an element of $\widehat{\mathcal{MPS}}(G,R)$.

Proposition 8: For the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$ such that G is Σ_{uc}-simulated by R, $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ defined by (5) satisfies $S_{W^\uparrow} \in \widehat{\mathcal{MPS}}(G,R)$.

Note that, since $S_{W^\uparrow} \in \widehat{\mathcal{MPS}}(G,R)$ can be synthesized if G is Σ_{uc}-simulated by R, by Propositions 4 and 8, we have

$$\mathcal{S}(G,R) \neq \emptyset \Leftrightarrow \widehat{\mathcal{MPS}}(G,R) \neq \emptyset. \tag{7}$$

By Theorem 1, Proposition 8, and the fact that $\mathcal{NS}(G,R) \subseteq \mathcal{S}(G,R)$, $\mathrm{NBSR}(S_{W^\uparrow},G)$ is a maximal solution to the nonblocking similarity control problem when it is solvable.

Corollary 1: We consider the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$. If $\mathcal{NS}(G,R) \neq \emptyset$, then G is Σ_{uc}-simulated by R, and it holds that $\mathrm{NBSR}(S_{W^\uparrow},G) = S_{W^\uparrow}^{NB} \in \mathcal{MPNS}(G,R)$ for the supervisor $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ defined by (5).

Finally, we show that the solvability of the nonblocking similarity control problem can be checked by applying the algorithm NBSR to $S_{W^\uparrow}$.

Theorem 2: For the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$, $\mathcal{NS}(G,R) \neq \emptyset$ if and only if G is Σ_{uc}-simulated by R, and there exists $\hat{S} \in \widehat{\mathcal{MPS}}(G,R)$ such that $\mathrm{NBSR}(\hat{S},G) \in \mathcal{A}(\Sigma)$.

Proof: ($\Leftarrow$) We consider any $\hat{S} \in \widehat{\mathcal{MPS}}(G,R)$ such that $\mathrm{NBSR}(\hat{S},G) \in \mathcal{A}(\Sigma)$. Since $\hat{S} \in \widehat{\mathcal{MPS}}(G,R) \subseteq \mathcal{S}(G,R)$, $\hat{S}$ satisfies the second condition of admissibility and $\hat{S}\|G \sqsubseteq R$ holds. By Proposition 3, we have $\mathrm{NBSR}(\hat{S},G) \in \mathcal{NS}(G,R) \neq \emptyset$.

($\Rightarrow$) Since $\emptyset \neq \mathcal{NS}(G,R) \subseteq \mathcal{S}(G,R)$, by Proposition 4, G is Σ_{uc}-simulated by R. Moreover, by (7), there exists $\hat{S} \in \widehat{\mathcal{MPS}}(G,R) \neq \emptyset$. Then, by Theorem 1, $\mathrm{NBSR}(\hat{S},G) \in \mathcal{MPNS}(G,R) \subseteq \mathcal{A}(\Sigma)$. ∎

Proposition 9: For the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$ such that G is Σ_{uc}-simulated by R, the following two statements are equivalent:

1) There exists $S \in \widehat{\mathcal{MPS}}(G,R)$ such that $\mathrm{NBSR}(S,G) \in \mathcal{A}(\Sigma)$.
2) For any $S' \in \widehat{\mathcal{MPS}}(G,R)$, $\mathrm{NBSR}(S',G) \in \mathcal{A}(\Sigma)$.

Proof: Since G is Σ_{uc}-simulated by R, it follows from Proposition 4 and (7) that $\widehat{\mathcal{MPS}}(G,R) \neq \emptyset$. It is sufficient to show 1) $\Rightarrow$ 2). For any $S \in \widehat{\mathcal{MPS}}(G,R)$ such that $\mathrm{NBSR}(S,G) \in \mathcal{A}(\Sigma)$, since $S \in \widehat{\mathcal{MPS}}(G,R) \subseteq \mathcal{S}(G,R)$, it follows from Proposition 3 that $\mathrm{NBSR}(S,G) \in \mathcal{NS}(G,R)$. Then, for any supervisor $S' \in \widehat{\mathcal{MPS}}(G,R)$, by Theorem 1, it holds that $\mathrm{NBSR}(S',G) \in \mathcal{A}(\Sigma)$. ■

By Theorem 2 and Propositions 8 and 9, we have the following corollary.

Corollary 2: For the system $G \in \mathcal{A}(\Sigma)$ and the specification $R \in \mathcal{A}(\Sigma)$, $\mathcal{NS}(G,R) \neq \emptyset$ if and only if G is Σ_{uc}-simulated by R, and $\mathrm{NBSR}(S_{W^\uparrow},G) = S^{NB}_{W^\uparrow} \in \mathcal{A}(\Sigma)$ holds for the supervisor $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ defined by (5).

By Corollary 2, whether or not the nonblocking similarity control problem is solvable can be verified using $S_{W^\uparrow}$. If G is Σ_{uc}-simulated by R and $\mathrm{NBSR}(S_{W^\uparrow},G) = S^{NB}_{W^\uparrow} \in \mathcal{A}(\Sigma)$ holds, then we have $\mathcal{NS}(G,R) \neq \emptyset$, and Corollary 1 guarantees the maximal permissiveness of the output supervisor $S^{NB}_{W^\uparrow}$; otherwise, it holds that $\mathcal{NS}(G,R) = \emptyset$. Since the number $|Y_{W^\uparrow}|$ of states of $S_{W^\uparrow}$ is at most $2^{2|X||Q|+|\Sigma_c \cap \Sigma_{uo}|}$, the computational complexity of computing $\mathrm{NBSR}(S_{W^\uparrow},G)$ is $O(|\Sigma||X|^3 2^{8|X||Q|+4|\Sigma_c \cap \Sigma_{uo}|})$. When the system is fully observed, namely, $\Sigma_{uo} = \emptyset$, $Y_{W^\uparrow}$ is reduced to

$$Y_{W^\uparrow} = \{(W,\emptyset,W) \in 2^{W^\uparrow} \times \{\emptyset\} \times 2^{W^\uparrow} \mid W \neq \emptyset\},$$

whose number is bounded by $2^{|X||Q|}$. In this case, $\mathrm{NBSR}(S_{W^\uparrow},G)$ can be computed in $O(|\Sigma||X|^3 2^{4|X||Q|})$.

Example 7: Let us recall Example 2. The system G in Fig. 3 is Σ_{uc}-simulated by the specification R shown in Fig. 4. For G and R, we synthesize the supervisor $S_{W^\uparrow}$ defined by (5). Its reachable part is shown in Fig. 18, where

$$\begin{aligned}
&\tilde{y}_0 = \{(x_0,q_0)\}, && \tilde{y}_1 = \{(x_1,q_1),(x_2,q_1)\},\\
&\tilde{y}_2 = \{(x_1,q_1),(x_2,q_2)\}, && \tilde{y}_3 = \{(x_1,q_2),(x_2,q_1)\},\\
&\tilde{y}_4 = \{(x_1,q_2),(x_2,q_2)\}, && \tilde{y}_5 = \{(x_3,q_3),(x_4,q_3)\},\\
&\tilde{y}_6 = \{(x_4,q_4)\}, && \tilde{y}_7 = \{(x_5,q_5)\},\\
&\tilde{y}_8 = \{(x_6,q_6)\},
\end{aligned}$$

and the first and second elements of each state are omitted for the sake of notational simplicity. Since $S_{W^\uparrow}$ is blocking, we apply the algorithm NBSR. After two iterations, NBSR outputs the supervisor $S^{NB}_{W^\uparrow}$ shown in Fig. 19. Then, by Corollary 2, the nonblocking similarity control problem is solvable. The supervised system $S^{NB}_{W^\uparrow}\|G$ is shown in Fig. 20. It can be verified that $S^{NB}_{W^\uparrow}\|G \sqsubseteq R$ holds, and $S^{NB}_{W^\uparrow}$ is admissible and nonblocking. We can confirm that the requirement "o_1 and o_2 are not selectable after u_1" is satisfied by $S^{NB}_{W^\uparrow}$. In addition, the maximal permissiveness of $S^{NB}_{W^\uparrow}$ is guaranteed by Corollary 1.

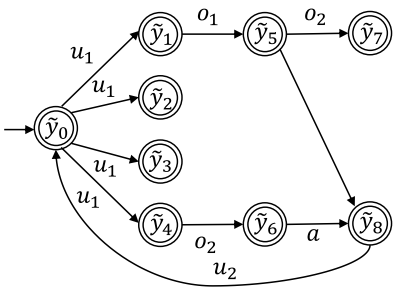


Fig. 18. Supervisor $S_{W^\uparrow}$.

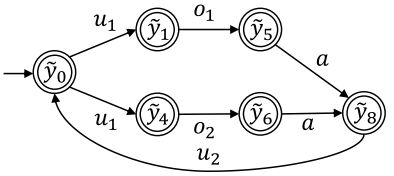


Fig. 19. Supervisor $S^{NB}_{W^\uparrow}$.

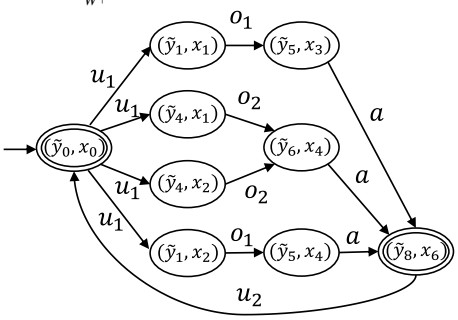


Fig. 20. Supervised system $S^{NB}_{W^\uparrow}\|G$.

VII. Conclusion

In this paper, we considered the nonblocking similarity control problem for the system and the specification modeled as nondeterministic automata subject to partial observation. We proposed an algorithm, named NBSR, for constructing a nonblocking supervisor from a blocking one. It was shown that state-unmergedness and strong maximal permissiveness are the key properties for synthesizing a maximally permissive nonblocking supervisor. We synthesized a supervisor with these two properties, which is blocking in general, as a candidate to which the algorithm NBSR is applied. Using the synthesized supervisor and NBSR, we showed how to check the solvability of the nonblocking similarity control problem and computed a maximal solution when it is solvable.

Appendix

A. Proof of Proposition 7

To prove Proposition 7, we need the following results.

Lemma 2: For the system $G = (X,\Sigma,\alpha,X_0,X_m) \in \mathcal{A}(\Sigma)$, we consider two supervisors $S_1 = (Y_1,\Sigma,\beta_1,Y_{10},Y_1) \in \mathcal{A}(\Sigma)$ and $S_2 = (Y_2,\Sigma,\beta_2,Y_{20},Y_2) \in \mathcal{A}(\Sigma)$ which satisfy the second condition of admissibility. If S_2 is state-unmerged and $\Psi \subseteq (Y_1 \times X) \times (Y_2 \times X)$ is a strong simulation relation from $S_1\|G$ to $S_2\|G$, then it holds that

$$\forall((y_1,x),(y_2,x)) \in \Psi : y_1 \in NB^{S_1}_G(Y_1) \Rightarrow y_2 \in NB^{S_2}_G(Y_2).$$

Proof: Let $\xi_1 : (Y_1 \times X) \times \Sigma \to 2^{Y_1 \times X}$ and $\xi_2 : (Y_2 \times X) \times \Sigma \to 2^{Y_2 \times X}$ denote the transition functions of $S_1\|G$ and $S_2\|G$, respectively. For any $((y_1,x),(y_2,x)) \in \Psi$ with $y_1 \in NB^{S_1}_G(Y_1)$, we consider any $(y_2,x') \in \mathit{Reach}(S_2\|G)$. Since S_2 is state-unmerged, it follows from (6) that $x' \in \bigcup_{s \in L(r)} \alpha(X_0,s)$ for

any run $r \in \mathcal{R}_{y_2}(S_2)$. By the second condition of a strong simulation relation, there exists a run r' such that $r' \in \mathcal{R}_{y_1}(S_1) \cap \mathcal{R}_{y_2}(S_2) \neq \emptyset$. Then, we have $x' \in \bigcup_{s' \in L(r')} \alpha(X_0, s')$. Hence, there exists an event string $s' \in L(r')$ such that $x' \in \alpha(X_0, s')$, $y_1 \in \beta_1(Y_{10}, s')$, and $y_2 \in \beta_2(Y_{20}, s')$. It follows that $s' \in L(S_1 \| G)$, $(y_1, x') \in \beta_1(Y_{10}, s') \times \alpha(X_0, s') = \xi_1(Y_{10} \times X_0, s') \subseteq \mathit{Reach}(S_1 \| G)$, and $(y_2, x') \in \beta_2(Y_{20}, s') \times \alpha(X_0, s') = \xi_2(Y_{20} \times X_0, s')$. By the third condition of a strong simulation relation, we have $((y_1, x'), (y_2, x')) \in \Psi$. Since $y_1 \in NB_G^{S_1}(Y_1)$ and $(y_1, x') \in \mathit{Reach}(S_1 \| G)$, there exists $s'' \in \Sigma^*$ such that $\xi_1((y_1, x'), s'') \cap Y_1 \times X_m \neq \emptyset$, that is, there exists $(y_1', x_m) \in Y_1 \times X_m$ such that $(y_1', x_m) \in \xi_1((y_1, x'), s'')$. Since $((y_1, x'), (y_2, x')) \in \Psi$ and Ψ is a simulation relation from $S_1 \| G$ to $S_2 \| G$, by repeatedly applying the second condition of Definition 2, there exists $(y_2', x_m) \in \xi_2((y_2, x'), s'')$ such that $((y_1', x_m), (y_2', x_m)) \in \Psi$. Since $x_m \in X_m$, we have $(y_2', x_m) \in \xi_2((y_2, x'), s'') \cap Y_2 \times X_m \neq \emptyset$. Besides, since $y_1 \in NB_G^{S_1}(Y_1)$, for any $\sigma \in \Sigma_{uc}$ such that $\alpha(x', \sigma) \neq \emptyset$, we have $\beta_1(y_1, \sigma) \neq \emptyset$. It follows that $\xi_1((y_1, x'), \sigma) \neq \emptyset$. Since Ψ is a strong simulation relation from $S_1 \| G$ to $S_2 \| G$, we have $\xi_2((y_2, x'), \sigma) \neq \emptyset$, which implies $\beta_2(y_2, \sigma) \neq \emptyset$. Thus, we have $y_2 \in NB_G^{S_2}(Y_2)$. ■

Lemma 2 just says, for two supervisors S_1 and S_2 such that S_2 is state-unmerged with respect to G and $((y_1, x), (y_2, x))$ is an element of a strong simulation relation Ψ from $S_1 \| G$ to $S_2 \| G$, if y_1 is not eliminated after an iteration of NBSR, neither is y_2.

Now, we can prove Proposition 7. We write $S_1' = (Y_1', \Sigma, \beta_1', Y_{10}', Y_1')$, where $Y_1' = NB_G^{S_1}(Y_1)$. Besides, we let $\xi_1 : (Y_1 \times X) \times \Sigma \to 2^{Y_1 \times X}$ and $\xi_2 : (Y_2 \times X) \times \Sigma \to 2^{Y_2 \times X}$ denote the transition functions of $S_1 \| G$ and $S_2 \| G$, respectively. Since S_2 is strongly more permissive than S_1, there exists a strong simulation relation $\Psi \subseteq (Y_1 \times X) \times (Y_2 \times X)$ from $S_1 \| G$ to $S_2 \| G$.

We first show that $Y_{20} \cap NB_G^{S_2}(Y_2) \neq \emptyset$. Since $Y_{10} \cap NB_G^{S_1}(Y_1) \neq \emptyset$, there exists $y_{10} \in Y_{10} \cap NB_G^{S_1}(Y_1)$. For any $x_0 \in X_0$, we have $(y_{10}, x_0) \in Y_{10} \times X_0$. Since Ψ is a strong simulation relation from $S_1 \| G$ to $S_2 \| G$, there exists $(y_{20}, x_0) \in Y_{20} \times X_0$ such that $((y_{10}, x_0), (y_{20}, x_0)) \in \Psi$. Since $y_{10} \in NB_G^{S_1}(Y_1)$, by Lemma 2, we have $y_{20} \in NB_G^{S_2}(Y_2)$. It follows that $y_{20} \in Y_{20} \cap NB_G^{S_1}(Y_2) \neq \emptyset$. Letting $Y_2' = NB_G^{S_2}(Y_2)$, we write the subautomaton S_2' of S_2 with respect to Y_2' as $S_2' = (Y_2', \Sigma, \beta_2', Y_{20}', Y_2')$.

We next show that S_2' is strongly more permissive than S_1'. We prove that the relation $\Psi' \subseteq (Y_1' \times X) \times (Y_2' \times X)$ given as

$$\Psi' = \{((y_1, x), (y_2, x)) \in \Psi \mid \mathcal{R}_{y_1}(S_1') \cap \mathcal{R}_{y_2}(S_2') \neq \emptyset\}$$

is a simulation relation from $S_1' \| G$ to $S_2' \| G$, where $\xi_1' : (Y_1' \times X) \times \Sigma \to 2^{Y_1' \times X}$ and $\xi_2' : (Y_2' \times X) \times \Sigma \to 2^{Y_2' \times X}$ are the transition functions of $S_1' \| G$ and $S_2' \| G$, respectively.

- We consider any $(y_{10}, x_0) \in Y_{10}' \times X_0 \subseteq Y_{10} \times X_0$. Since Ψ is a strong simulation relation from $S_1 \| G$ to $S_2 \| G$, there exists $(y_{20}, x_0) \in Y_{20} \times X_0$ such that $((y_{10}, x_0), (y_{20}, x_0)) \in \Psi$. Since $y_{10} \in Y_{10}' = Y_{10} \cap NB_G^{S_1}(Y_1) \subseteq NB_G^{S_1}(Y_1)$, by Lemma 2, we have $y_{20} \in NB_G^{S_2}(Y_2) = Y_2'$. It follows that $y_{20} \in Y_{20} \cap Y_2' = Y_{20}'$. In addition, by the second condition of a strong simulation relation, we have $\mathcal{R}_{y_{10}}(S_1) \cap \mathcal{R}_{y_{20}}(S_2) \neq \emptyset$, which implies $\Sigma_{uo}(y_{10}) = \Sigma_{uo}(y_{20})$. Since $y_{10} \in Y_{10}'$ and $y_{20} \in Y_{20}'$, we have $\Sigma_{uo}(y_{10}) = \Sigma_{uo}(y_{20}) \in \mathcal{R}_{y_{10}}(S_1') \cap \mathcal{R}_{y_{20}}(S_2') \neq \emptyset$. Then, we have $((y_{10}', x_0'), (y_{20}', x_0')) \in \Psi'$.
- For any $((y_1, x), (y_2, x)) \in \Psi' \subseteq \Psi$ and any $\sigma \in \Sigma$, we consider any $(y_1', x') \in \xi_1'((y_1, x), \sigma) \subseteq \xi_1((y_1, x), \sigma)$. Since Ψ is a strong simulation relation from $S_1 \| G$ to $S_2 \| G$, there exists $(y_2', x') \in \xi_2((y_2, x), \sigma) = \beta_2(y_2, \sigma) \times \alpha(x, \sigma)$ such that $((y_1', x'), (y_2', x')) \in \Psi$. Since $y_1' \in Y_1' = NB_G^{S_1}(Y_1)$, by Lemma 2, we have $y_2' \in NB_G^{S_2}(Y_2) = Y_2'$. It follows that $y_2' \in \beta_2(y_2, \sigma) \cap Y_2' = \beta_2'(y_2, \sigma)$, which further implies $(y_2', x') \in \beta_2'(y_2, \sigma) \times \alpha(x, \sigma) = \xi_2'((y_2, x), \sigma)$. By the definition of Ψ', we have $\mathcal{R}_{y_1}(S_1') \cap \mathcal{R}_{y_2}(S_2') \neq \emptyset$. In the case where $\sigma \in \Sigma_{uo}$, we have $y_1' = y_1$ and $y_2' = y_2$. It follows that $\mathcal{R}_{y_1'}(S_1') \cap \mathcal{R}_{y_2'}(S_2') \neq \emptyset$, which further implies $((y_1', x'), (y_2', x')) \in \Psi'$. In the case where $\sigma \in \Sigma_o$, we consider any run $r \in \mathcal{R}_{y_1}(S_1') \cap \mathcal{R}_{y_2}(S_2') \neq \emptyset$. Since $((y_1', x'), (y_2', x')) \in \Psi$, by the second condition of a strong simulation relation, we have $\Sigma_{uo}(y_1') = \Sigma_{uo}(y_2')$. Then, it follows that $r\sigma\Sigma_{uo}(y_1') = r\sigma\Sigma_{uo}(y_2') \in \mathcal{R}_{y_1'}(S_1') \cap \mathcal{R}_{y_2'}(S_2') \neq \emptyset$, which implies $((y_1', x'), (y_2', x')) \in \Psi'$.
- For any $((y_1, x), (y_2, x)) \in \Psi'$, if $(y_1, x) \in Y_1' \times X_m$, then $(y_2, x) \in Y_2' \times X_m$.

Thus, Ψ' is a simulation relation from $S_1' \| G$ to $S_2' \| G$.

In addition, we show that Ψ' is a strong simulation relation from $S_1' \| G$ to $S_2' \| G$. By the definition of Ψ', any $((\hat{y}_1, \hat{x}), (\hat{y}_2, \hat{x})) \in \Psi' \subseteq \Psi$ satisfies the first and second conditions of a strong simulation relation. To prove that it satisfies the third condition, we consider any $\hat{x}' \in X$ and any $\hat{s} \in L(S_1' \| G) \subseteq L(S_1 \| G)$ such that $(\hat{y}_1, \hat{x}') \in \xi_1'(Y_{10}' \times X_0, \hat{s})$ and $(\hat{y}_2, \hat{x}') \in \xi_2'(Y_{20}' \times X_0, \hat{s})$. We have $(\hat{y}_1, \hat{x}') \in \xi_1'(Y_{10}' \times X_0, \hat{s}) \subseteq \xi_1(Y_{10} \times X_0, \hat{s})$ and $(\hat{y}_2, \hat{x}') \in \xi_2'(Y_{20}' \times X_0, \hat{s}) \subseteq \xi_2(Y_{20} \times X_0, \hat{s})$. Since Ψ is a strong simulation relation from $S_1 \| G$ to $S_2 \| G$, we have $((\hat{y}_1, \hat{x}'), (\hat{y}_2, \hat{x}')) \in \Psi$. It follows that $((\hat{y}_1, \hat{x}'), (\hat{y}_2, \hat{x}')) \in \Psi'$. Thus, Ψ' is a strong simulation relation from $S_1' \| G$ to $S_2' \| G$, namely, S_2' is strongly more permissive than S_1'.

We finally show that S_2' is state-unmerged. For any $\bar{y} \in \mathit{Reach}(S_2')$, we consider any $r_1, r_2 \in \mathcal{R}_{\bar{y}}(S_2')$. By the definition of a subautomaton, we have $\bar{y} \in \mathit{Reach}(S_2') \subseteq \mathit{Reach}(S_2)$ and $r_1, r_2 \in \mathcal{R}_{\bar{y}}(S_2)$. Since S_2 is state-unmerged, we have

$$\bigcup_{s \in L(r_1)} \alpha(X_0, s) = \bigcup_{s \in L(r_2)} \alpha(X_0, s).$$

Hence, S_2' is state-unmerged.

B. Proof of Proposition 8

By proposition 6, the supervisor $S_{W^\uparrow} \in \mathcal{A}(\Sigma)$ defined by (5) satisfies $S_{W^\uparrow} \in \mathcal{MPS}(G, R) \subseteq \mathcal{S}(G, R)$. By the definition of $S_{W^\uparrow}$, $S_{W^\uparrow}$ is $\Sigma_{uc} \cap \Sigma_{uo}$-compatible. We need to show that $S_{W^\uparrow}$ is state-unmerged and strongly more permissive than any $\Sigma_{uc} \cap \Sigma_{uo}$-compatible supervisor $S \in \mathcal{S}(G, R)$.

We first prove that $S_{W^\uparrow}$ is state-unmerged. We show that

(i) $\forall y_0 \in Y_{W^\uparrow 0} : \pi_X(W_2^{y_0}) = \bigcup_{s \in \gamma_{uo}^{y_0 *}} \alpha(X_0, s)$, and

(ii) $\forall y, y' \in \mathit{Reach}(S_{W^\uparrow}), \forall \sigma \in \Sigma_o :$
$y' \in \beta_{W^\uparrow}(y, \sigma) \Rightarrow \pi_X(W_2^{y'}) = \bigcup_{s \in \gamma_{uo}^{y' *}} \alpha(\pi_X(W_2^y), \sigma s).$

where $y_0 = (W_1^{y_0}, \gamma_{uo}^{y_0}, W_2^{y_0})$, $y = (W_1^{y}, \gamma_{uo}^{y}, W_2^{y})$, and $y' = (W_1^{y'}, \gamma_{uo}^{y'}, W_2^{y'})$.

For this purpose, we need the following lemma.

Lemma 3: For the system $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and the specification $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$ such that G is Σ_{uc}-simulated by R, we consider the supervisor $S_{W^\uparrow} = (Y_{W^\uparrow}, \Sigma, \beta_{W^\uparrow}, Y_{W^\uparrow 0}, Y_{W^\uparrow}) \in \mathcal{A}(\Sigma)$ defined by (5). For any $y = (W_1^{y}, \gamma_{uo}^{y}, W_2^{y}) \in Y_{W^\uparrow}$, it holds that

$$\bigcup_{s \in \gamma_{uo}^{y*}} \alpha(\pi_X(W_1^{y}), s) = \pi_X(W_2^{y}).$$

Proof: Since $W_2^{y} \in \min U(W_1^{y}, \gamma_{uo}^{y})$ is a minimal element of $U(W_1^{y}, \gamma_{uo}^{y})$ we have

$$W_2^{y} \subseteq \bigcup_{(x,q) \in W_1^{y}} \bigcup_{s \in \gamma_{uo}^{y*}} \alpha(x,s) \times \delta(q,s),$$

which implies $\pi_X(W_2^{y}) \subseteq \bigcup_{s \in \gamma_{uo}^{y*}} \alpha(\pi_X(W_1^{y}), s)$. To prove $\bigcup_{s \in \gamma_{uo}^{y*}} \alpha(\pi_X(W_1^{y}), s) \subseteq \pi_X(W_2^{y})$, for any $(x,q) \in W_1^{y}$ and any $s \in \gamma_{uo}^{y*}$, we let $n = |s|$ and consider any $x' \in \alpha(x,s)$. Since $W_2^{y} \in \min U(W_1^{y}, \gamma_{uo}^{y}) \subseteq U(W_1^{y}, \gamma_{uo}^{y})$, it follows from the definition of U that $(x,q) \in W_1^{y} \subseteq W_2^{y}$. In the case where $n = 0$, we have $s = \varepsilon$ and $\{x\} = \alpha(x, \varepsilon)$, that is, $x' = x$. Since $(x,q) \in W_2^{y}$, we have $x' \in \pi_X(W_2^{y})$. In the case where $n > 0$, we write s as $s = \sigma_1 \sigma_2 \cdots \sigma_n$. Then, there exist $x_1, x_2, \cdots, x_n \in X$ such that $x_1 \in \alpha(x, \sigma_1)$, $x_i \in \alpha(x_{i-1}, \sigma_i)$ for $i = 2, 3, \cdots, n$, and $x_n = x'$. Since $(x,q) \in W_2^{y} \subseteq U(W_1^{y}, \gamma_{uo}^{y})$, by the definition of U, there exist $q_1, q_2, \cdots, q_n \in Q$ such that $q_1 \in \delta(q, \sigma_1)$, $q_i \in \delta(q_{i-1}, \sigma_i)$ for $i = 2, 3, \cdots, n$, and $(x_i, q_i) \in W_2^{y}$ for $i = 1, 2, \cdots, n$. That is, there exists $q_n \in \delta(q,s)$ such that $(x_n, q_n) = (x', q_n) \in W_2^{y}$. Namely, we have $x' \in \pi_X(W_2^{y})$. ■

To prove (i), we consider any $y_0 = (W_1^{y_0}, \gamma_{uo}^{y_0}, W_2^{y_0}) \in Y_{W^\uparrow 0}$. It follows from the definition of $Y_{W^\uparrow 0}$ that $\pi_X(W_1^{y_0}) = X_0$. By Lemma 3, we have

$$\pi_X(W_2^{y_0}) = \bigcup_{s \in \gamma_{uo}^{y_0 *}} \alpha(\pi_X(W_1^{y_0}), s) = \bigcup_{s \in \gamma_{uo}^{y_0 *}} \alpha(X_0, s).$$

To prove (ii), we consider any $y = (W_1^{y}, \gamma_{uo}^{y}, W_2^{y}), y' = (W_1^{y'}, \gamma_{uo}^{y'}, W_2^{y'}) \in Reach(S_{W^\uparrow})$ and any $\sigma \in \Sigma_o$ such that $y' \in \beta_{W^\uparrow}(y, \sigma)$. By the definition of $\beta_{W^\uparrow}$ and N, we have $\pi_X(W_1^{y'}) = \alpha(\pi_X(W_2^{y}), \sigma)$, which implies together with Lemma 3 that

$$\pi_X(W_2^{y'}) = \bigcup_{s \in \gamma_{uo}^{y' *}} \alpha(\pi_X(W_2^{y}), \sigma s).$$

The state-unmergedness of $S_{W^\uparrow}$ follows from the lemma below, which also indicates that $S_{W^\uparrow}$ correctly estimates the current state of G by the third element of its state.

Lemma 4: For the system $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and the specification $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$ such that G is Σ_{uc}-simulated by R, we consider the supervisor $S_{W^\uparrow} = (Y_{W^\uparrow}, \Sigma, \beta_{W^\uparrow}, Y_{W^\uparrow 0}, Y_{W^\uparrow}) \in \mathcal{A}(\Sigma)$ defined by (5). For any $y = (W_1^{y}, \gamma_{uo}^{y}, W_2^{y}) \in Reach(S_{W^\uparrow})$ and any $r \in \mathcal{R}_y(S_{W^\uparrow})$, it holds that

$$\pi_X(W_2^{y}) = \bigcup_{s \in L(r)} \alpha(X_0, s).$$

Proof: Any $r \in \mathcal{R}_y(S_{W^\uparrow})$ can be written as $r = \gamma_0$ or $r = \gamma_0 \sigma_0 \cdots \gamma_{n-1} \sigma_{n-1} \gamma_n$. First, we consider the case where $r = \gamma_0$. It follows that $y \in Y_{W^\uparrow 0}$, $\gamma_0 = \gamma_{uo}^{y}$, and $L(r) = \gamma_{uo}^{y*}$, which imply together with (i) that $\pi_X(W_2^{y}) = \bigcup_{s \in L(r)} \alpha(X_0, s)$.

Next, we consider the case where $r = \gamma_0 \sigma_0 \cdots \gamma_{n-1} \sigma_{n-1} \gamma_n$. There exist $y_0, y_1, \ldots, y_n \in Reach(S_{W^\uparrow})$ with $y_0 \in Y_{W^\uparrow 0}$, $y_n = y$, and $y_{i+1} \in \beta(y_i, \sigma_i)$ for each $i \in \{0, 1, \ldots, n-1\}$, where $\sigma_i \in \Sigma_o$, $y_i = (W_1^{y_i}, \gamma_{uo}^{y_i}, W_2^{y_i})$, and $\gamma_i = \gamma_{uo}^{y_i}$ for each $i \in \{0, 1, \ldots, n\}$. By (i), we have $\pi_X(W_2^{y_0}) = \bigcup_{s \in \gamma_{uo}^{y_0 *}} \alpha(X_0, s)$. For each $i \in \{0, 1, \ldots, n-1\}$, since $y_{i+1} \in \beta_{W^\uparrow}(y_i, \sigma_i)$, it follows from (ii) that $\pi_X(W_2^{y_{i+1}}) = \bigcup_{s \in \gamma_{uo}^{y_{i+1} *}} \alpha(\pi_X(W_2^{y_i}), \sigma_i s)$. Since $L(r) = \{s_0 \sigma_0 \ldots s_{n-1} \sigma_{n-1} s_n \in L(S_{W^\uparrow}) \mid \forall i \in \{0, 1, \ldots, n\} : s_i \in \gamma_{uo}^{y_i *}\}$, we have $\pi_X(W_2^{y}) = \pi_X(W_2^{y_n}) = \bigcup_{s \in L(r)} \alpha(X_0, s)$. ■

It follows from Lemma 4 that

$$\forall y \in Reach(S_{W^\uparrow}), \forall r_1, r_2 \in \mathcal{R}_y(S_{W^\uparrow}) : \bigcup_{s \in L(r_1)} \alpha(X_0, s) = \bigcup_{s \in L(r_2)} \alpha(X_0, s) = \pi_X(W_2^{y}),$$

namely, $S_{W^\uparrow}$ is state-unmerged.

We next prove that $S_{W^\uparrow}$ is strongly more permissive than any $\Sigma_{uc} \cap \Sigma_{uo}$-compatible supervisor in $\mathcal{S}(G,R)$.

We need the following lemma which can be proved in the same way as Theorem 1 of [22] by taking marked states into consideration.

Lemma 5: For the system $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and the specification $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$, if $S = (Y, \Sigma, \beta, Y_0, Y) \in \mathcal{A}(\Sigma)$ satisfies the first condition of admissibility and $S \| G \sqsubseteq R$ holds, then, for any simulation relation $\Psi \subseteq (Y \times X) \times Q$ from $S \| G$ to R, a relation $\Phi \subseteq X \times Q$ defined by

$$\begin{aligned} \Phi \;=\; & \{(x,q) \in X \times Q \mid \exists y \in Y : ((y,x),q) \in \Psi \wedge \\ & (y,x) \in Reach(S \| G)\} \end{aligned} \tag{8}$$

is a Σ_{uc}-simulation relation from G to R.

In addition, we have the following lemma.

Lemma 6: For the system $G = (X, \Sigma, \alpha, X_0, X_m) \in \mathcal{A}(\Sigma)$ and the specification $R = (Q, \Sigma, \delta, Q_0, Q_m) \in \mathcal{A}(\Sigma)$ such that G is Σ_{uc}-simulated by R, we consider any supervisor $S = (Y, \Sigma, \beta, Y_0, Y) \in \mathcal{S}(G,R)$ and any simulation relation $\Psi \subseteq (Y \times X) \times Q$ from $S \| G$ to R. For any $y \in Y$, if a relation $\Phi_y \subseteq X \times Q$ defined by

$$\begin{aligned} \Phi_y \;=\; & \{(x,q) \in X \times Q \mid ((y,x),q) \in \Psi \wedge \\ & (y,x) \in Reach(S \| G)\} \end{aligned} \tag{9}$$

is nonempty, then, for any nonempty subset $W \subseteq \Phi_y$, it holds that $\Phi_y \in U(W, \Sigma_{uo}(y)) \neq \emptyset$.

Proof: We consider any $y \in Y$ such that $\Phi_y \subseteq X \times Q$ defined by (9) is nonempty. It holds that $\Phi_y \subseteq \Phi$ for the relation $\Phi \subseteq X \times Q$ defined by (8), which is a Σ_{uc}-simulation relation from G to R by Proposition 5. Then, for the unique maximal fixpoint $W^\uparrow$ of the function $\tilde{F} : 2^{X \times Q} \to 2^{X \times Q}$ defined by (2), we have $\Phi_y \subseteq \Phi \subseteq W^\uparrow$. For any $W \subseteq \Phi_y$ with $W \neq \emptyset$, it holds that $W \in 2^{W^\uparrow}$. To show that $\Phi_y \in U(W, \Sigma_{uo}(y))$, we consider any $(x,q) \in \Phi_y$, any $\sigma \in \Sigma_{uo}(y)$, and any $x' \in \alpha(x, \sigma)$. By the definition of Φ_y, we have $((y,x),q) \in \Psi$ and $(y,x) \in Reach(S \| G)$. Since $\sigma \in \Sigma_{uo}(y) \subseteq \Sigma_{uo}$, it follows from the admissibility of S that $\beta(y, \sigma) = \{y\}$, which implies $(y,x') \in \beta(y, \sigma) \times \alpha(x, \sigma) = \xi((y,x), \sigma) \subseteq Reach(S \| G)$. Since Ψ is a simulation relation from $S \| G$ to R, there exists

$q' \in \delta(q,\sigma)$ such that $(y,x'),q') \in \Psi$, which implies together with $(y,x') \in Reach(S\|G)$ that $(x',q') \in \Phi_y$. It follows from $W \subseteq \Phi_y$ that $\Phi_y \in U(W,\Sigma_{uo}(y)) \neq \emptyset$. ■

Now we can prove the strong maximal permissiveness of $S_{W^\uparrow}$. We consider any $\Sigma_{uc} \cap \Sigma_{uo}$-compatible supervisor $S = (Y,\Sigma,\beta,Y_0,Y) \in \mathcal{S}(G,R)$. There exists a simulation relation $\Psi \subseteq (Y \times X) \times Q$ from $S\|G$ to R. Then, by Proposition 5, the relation $\Phi \subseteq X \times Q$ defined by (8) is a Σ_{uc}-simulation relation from G to R. For the unique maximal fixpoint $W^\uparrow$ of the function $\tilde{F}: 2^{X\times Q} \to 2^{X\times Q}$ defined by (2), we have $\Phi \subseteq W^\uparrow$. It follows that

$$\begin{aligned} &\forall (x,q) \in X \times Q : [\exists y \in Y : ((y,x),q) \in \Psi \wedge \\ &(y,x) \in Reach(S\|G)] \Rightarrow (x,q) \in \Phi \subseteq W^\uparrow. \end{aligned} \tag{10}$$

We let $\xi : (Y \times X) \times \Sigma \to 2^{Y\times X}$ and $\xi_{W^\uparrow} : (Y_{W^\uparrow} \times X) \times \Sigma \to 2^{Y_{W^\uparrow}\times X}$ denote the nondeterministic transition functions of $S\|G$ and $S_{W^\uparrow}\|G$, respectively. We prove that a relation $\tilde{\Phi} \subseteq (Y \times X) \times (Y_{W^\uparrow} \times X)$ defined as

$$\begin{aligned} \tilde{\Phi} \;=\; &\{((y,x),(\tilde{y},x)) \in (Y \times X) \times (Y_{W^\uparrow} \times X) \mid \\ &\mathcal{R}_y(S) \cap \mathcal{R}_{\tilde{y}}(S_{W^\uparrow}) \neq \emptyset \wedge [\forall (\tilde{x},\tilde{q}) \in W_2^{\tilde{y}} : \\ &((y,\tilde{x}),\tilde{q}) \in \Psi \wedge (y,\tilde{x}) \in Reach(S\|G)] \wedge \\ &[\exists s \in L(S\|G) : (y,x) \in \xi(Y_0 \times X_0, s) \wedge \\ &(\tilde{y},x) \in \xi_{W^\uparrow}(\tilde{Y}_{W^\uparrow 0} \times X_0, s)]\} \end{aligned}$$

is a strong simulation relation from $S\|G$ to $S_{W^\uparrow}\|G$, where $\tilde{y} = (W_1^{\tilde{y}}, \gamma_{uo}^{\tilde{y}}, W_2^{\tilde{y}})$.

- We consider any $(y_0,x_0) \in Y_0 \times X_0$. To prove that there exists $\tilde{y}_0 = (W_1^{\tilde{y}_0}, \gamma_{uo}^{\tilde{y}_0}, W_2^{\tilde{y}_0}) \in \tilde{Y}_{W^\uparrow 0}$ such that $((y_0,x_0),(\tilde{y}_0,x_0)) \in \tilde{\Phi}$, we first prove that there exists $W_1^{\tilde{y}_0} \in 2^{W^\uparrow}$ such that

$$|W_1^{\tilde{y}_0}| = |X_0| \wedge [\forall x_0' \in X_0, \exists q_0' \in Q_0 : (x_0',q_0') \in W_1^{\tilde{y}_0}] \tag{11}$$

and

$$\begin{aligned} &\forall (x_0',q_0') \in W_1^{\tilde{y}_0} : ((y_0,x_0'),q_0') \in \Psi \wedge \\ &(y_0,x_0') \in Reach(S\|G). \end{aligned} \tag{12}$$

For any $x_0' \in X_0$, we have $(y_0,x_0') \in Y_0 \times X_0 \subseteq Reach(S\|G)$. Since Ψ is a simulation relation from $S\|G$ to R, there exists $q_0' \in Q_0$ such that $((y_0,x_0'),q_0') \in \Psi$. By (10), we have $(x_0',q_0') \in \Phi \subseteq W^\uparrow$. Then, there exists $W_1^{\tilde{y}_0} \in 2^\Phi \subseteq 2^{W^\uparrow}$ that satisfies (11) and (12). Since $X_0 \neq \emptyset$ and $|W_1^{\tilde{y}_0}| = |X_0|$, we have $W_1^{\tilde{y}_0} \neq \emptyset$. For the relation $\Phi_{y_0} \subseteq X \times Q$ defined by (9), we have $W_1^{\tilde{y}_0} \subseteq \Phi_{y_0} \neq \emptyset$. Let $\gamma_{uo}^{\tilde{y}_0} = \Sigma_{uo}(y_0) \in \Gamma_{uo}$. It follows from Lemma 6 that $\Phi_{y_0} \in U(W_1^{\tilde{y}_0}, \gamma_{uo}^{\tilde{y}_0})$. Then, there exists $W_2^{\tilde{y}_0} \in \min U(W_1^{\tilde{y}_0}, \gamma_{uo}^{\tilde{y}_0}) \subseteq 2^{W^\uparrow}$ such that $W_2^{\tilde{y}_0} \subseteq \Phi_{y_0}$. Letting $\tilde{y}_0 = (W_1^{\tilde{y}_0}, \gamma_{uo}^{\tilde{y}_0}, W_2^{\tilde{y}_0})$, we have $\tilde{y}_0 \in Y_{W^\uparrow}$. By (11), it holds that $\tilde{y}_0 \in Y_{W^\uparrow 0}$. It follows from $\Sigma_{uo}(\tilde{y}_0) = \gamma_{uo}^{\tilde{y}_0} = \Sigma_{uo}(y_0)$ that $\Sigma_{uo}(y_0) = \Sigma_{uo}(\tilde{y}_0) \in \mathcal{R}_{y_0}(S) \cap \mathcal{R}_{\tilde{y}_0}(S_{W^\uparrow})$. Besides, we have $\varepsilon \in L(S\|G)$, $(y_0,x_0) \in \xi(Y_0 \times X_0, \varepsilon)$, and $(\tilde{y}_0,x_0) \in Y_{W^\uparrow 0} \times X_0 = \xi_{W^\uparrow}(Y_{W^\uparrow 0} \times X_0, \varepsilon)$. Moreover, by $W_2^{\tilde{y}_0} \subseteq \Phi_{y_0}$, we have

$$\begin{aligned} &\forall (\tilde{x}_0,\tilde{q}_0) \in W_2^{\tilde{y}_0} : ((y_0,\tilde{x}_0),\tilde{q}_0) \in \Psi \wedge \\ &(y_0,\tilde{x}_0) \in Reach(S\|G). \end{aligned}$$

It follows that $((y_0,x_0),(\tilde{y}_0,x_0)) \in \tilde{\Phi}$.

- For any $((y,x),(\tilde{y},x)) \in \tilde{\Phi}$ and any $\sigma \in \Sigma$, we consider any $(y',x') \in \xi((y,x),\sigma)$. It follows that $y' \in \beta(y,\sigma) \neq \emptyset$ and $x' \in \alpha(x,\sigma) \neq \emptyset$. Since $((y,x),(\tilde{y},x)) \in \tilde{\Phi}$, there exists $s \in L(S\|G)$ such that $(y,x) \in \xi(Y_0 \times X_0, s)$ and $(\tilde{y},x) \in \xi_{W^\uparrow}(Y_{W^\uparrow 0} \times X_0, s) \subseteq Reach(S_{W^\uparrow}\|G)$. It follows that $y \in \beta(Y_0,s)$, $\tilde{y} \in \beta_{W^\uparrow}(Y_{W^\uparrow 0},s)$, and $x \in \alpha(X_0,s)$. Then, there exists $r \in \mathcal{R}_{\tilde{y}}(S_{W^\uparrow})$ such that $s \in L(r)$. We write $\tilde{y} = (W_1^{\tilde{y}}, \gamma_{uo}^{\tilde{y}}, W_2^{\tilde{y}})$. By Lemma 4, we have $x \in \pi_X(W_2^{\tilde{y}})$, which further implies

$$x' \in \alpha(x,\sigma) \subseteq \alpha(\pi_X(W_2^{\tilde{y}}),\sigma) \neq \emptyset. \tag{13}$$

We consider the case where $\sigma \in \Sigma_o$. To prove that there exists $(\tilde{y}',x') \in \xi_{W^\uparrow}((\tilde{y},x),\sigma)$ such that $((y',x'),(\tilde{y}',x')) \in \tilde{\Phi}$, we first show $\sigma \in \Sigma_{W^\uparrow}(\tilde{y})$. For any $(\tilde{x},\tilde{q}) \in W_2^{\tilde{y}}$, we consider any $\tilde{x}' \in \alpha(\tilde{x},\sigma)$. By the definition of $\tilde{\Phi}$, we have $((y,\tilde{x}),\tilde{q}) \in \Psi$ and $(y,\tilde{x}) \in Reach(S\|G)$. Then, it holds that $(y',\tilde{x}') \in \beta(y,\sigma) \times \alpha(\tilde{x},\sigma) = \xi((y,\tilde{x}),\sigma) \subseteq Reach(S\|G)$. Since Ψ is a simulation relation from $S\|G$ to R, there exists $\tilde{q}' \in \delta(\tilde{q},\sigma)$ such that $((y',\tilde{x}'),\tilde{q}') \in \Psi$. It follows from (10) that $(\tilde{x}',\tilde{q}') \in \Phi \subseteq W^\uparrow$. By (13), we have $\sigma \in \Sigma_{W^\uparrow}(\tilde{y})$. Moreover, there exists a minimal element $W_1 \in 2^\Phi \subseteq 2^{W^\uparrow}$ of $N(W_2^{\tilde{y}},\sigma)$, that is, $W_1 \in \min N(W_2^{\tilde{y}},\sigma)$ such that

$$\forall (\tilde{x},\tilde{q}) \in W_2^{\tilde{y}}, \forall \tilde{x}' \in \alpha(\tilde{x},\sigma), \exists \tilde{q}' \in \delta(\tilde{q},\sigma) : (\tilde{x}',\tilde{q}') \in W_1,$$

and

$$\begin{aligned} &\forall (\tilde{x}',\tilde{q}') \in W_1 : ((y',\tilde{x}'),\tilde{q}') \in \Psi \wedge \\ &(y',\tilde{x}') \in Reach(S\|G). \end{aligned} \tag{14}$$

By (13), $W_1 \neq \emptyset$ holds. For the relation $\Phi_{y'} \subseteq X \times Q$ defined by (9), it follows from (14) that $W_1 \subseteq \Phi_{y'} \neq \emptyset$. Let $\gamma_{uo} = \Sigma_{uo}(y') \in \Gamma_{uo}$. By Lemma 6, we have $\Phi_{y'} \in U(W_1,\gamma_{uo})$. Then, there exists $W_2 \in \min U(W_1,\gamma_{uo})$ such that $W_2 \subseteq \Phi_{y'}$. Letting $\tilde{y}' = (W_1,\gamma_{uo},W_2)$, we have $\tilde{y}' \in Y_{W^\uparrow}$. Since $W_1 \in \min N(W_2^{\tilde{y}},\sigma)$, by the definition of $\beta_{W^\uparrow}$, we have $\tilde{y}' \in \beta_{W^\uparrow}(\tilde{y},\sigma)$. Then, it holds that $(\tilde{y}',x') \in \beta_{W^\uparrow}(\tilde{y},\sigma) \times \alpha(x,\sigma) = \xi_{W^\uparrow}((\tilde{y},x),\sigma)$. Since $(y',x') \in \xi((y,x),\sigma)$, we have $s\sigma \in L(S\|G)$, $(y',x') \in \xi(Y_0 \times X_0, s\sigma)$, and $(\tilde{y}',x') \in \xi_{W^\uparrow}(Y_{W^\uparrow 0} \times X_0, s\sigma)$. Besides, by the definition of $\tilde{\Phi}$, there exists $r \in \mathcal{R}_y(S) \cap \mathcal{R}_{\tilde{y}}(S_{W^\uparrow})$. It follows from $\Sigma_{uo}(y') = \gamma_{uo} = \Sigma_{uo}(\tilde{y}')$ that $r\sigma\Sigma_{uo}(y') = r\sigma\Sigma_{uo}(\tilde{y}') \in \mathcal{R}_{y'}(S) \cap \mathcal{R}_{\tilde{y}'}(S_{W^\uparrow}) \neq \emptyset$. Moreover, by $W_2 \subseteq \Phi_{y'}$, it holds that

$$\begin{aligned} &\forall (\tilde{x}'',\tilde{q}'') \in W_2 : ((y',\tilde{x}''),\tilde{q}'') \in \Psi \wedge \\ &(y',\tilde{x}'') \in Reach(S\|G). \end{aligned}$$

It follows that $((y',x'),(\tilde{y}',x')) \in \tilde{\Phi}$.

We next consider the case where $\sigma \in \Sigma_{uo}$. It follows from $\beta(y,\sigma) \neq \emptyset$ that $\sigma \in \Sigma_{uo}(y)$. Since $\mathcal{R}_y(S) \cap \mathcal{R}_{\tilde{y}}(S_{W^\uparrow}) \neq \emptyset$, we have $\Sigma_{uo}(y) = \Sigma_{uo}(\tilde{y}) = \gamma_{uo}^{\tilde{y}}$, which implies $\sigma \in \gamma_{uo}^{\tilde{y}}$. By the definition of $\beta_{W^\uparrow}$, we have $\{\tilde{y}\} = \beta_{W^\uparrow}(\tilde{y},\sigma)$, which further implies $(\tilde{y},x') \in \beta_{W^\uparrow}(\tilde{y},\sigma) \times \alpha(x,\sigma) = \xi_{W^\uparrow}((\tilde{y},x),\sigma)$. Since S is admissible, we have $\{y\} = \beta(y,\sigma)$, that is, $y = y'$. Hence, we have $s\sigma \in L(S\|G)$, $(y,x') \in \xi(Y_0 \times X_0, s\sigma)$, and $(\tilde{y},x') \in \tilde{\xi}_{W^\uparrow}(\tilde{Y}_{W^\uparrow 0} \times X_0, s\sigma)$.

Since $((y,x),(\tilde{y},x)) \in \tilde{\Phi}$, we have $\mathcal{R}_y(S) \cap \mathcal{R}_{\tilde{y}}(S_{W^\uparrow}) \neq \emptyset$ and

$$\forall (\tilde{x},\tilde{q}) \in W_2^{\tilde{y}} : ((y,\tilde{x}),\tilde{q}) \in \Psi \wedge (y,\tilde{x}) \in Reach(S\|G).$$

It follows that $((y,x'),(\tilde{y},x')) \in \tilde{\Phi}$.

- For any $((y,x),(\tilde{y},x)) \in \tilde{\Phi}$, if $(y,x) \in Y \times X_m$, then $(\tilde{y},x) \in Y_{W^\uparrow} \times X_m$.

Thus, $\tilde{\Phi}$ is a simulation relation from $S\|G$ to $S_{W^\uparrow}\|G$. By the definition of $\tilde{\Phi}$, any $((y,x),(\tilde{y},x)) \in \tilde{\Phi}$ satisfies the first and second conditions of a strong simulation relation. To prove the third condition, we consider any $\hat{x} \in X$ and any $\hat{s} \in L(S\|G)$ such that $(y,\hat{x}) \in \xi(Y_0 \times X_0,\hat{s})$ and $(\tilde{y},\hat{x}) \in \xi_{W^\uparrow}(Y_{W^\uparrow 0} \times X_0,\hat{s})$. Since $((y,x),(\tilde{y},x)) \in \tilde{\Phi}$, we have $\mathcal{R}_y(S) \cap \mathcal{R}_{\tilde{y}}(S_{W^\uparrow}) \neq \emptyset$ and

$$\forall (\tilde{x},\tilde{q}) \in W_2^{\tilde{y}} : ((y,\tilde{x}),\tilde{q}) \in \Psi \wedge (y,\tilde{x}) \in Reach(S\|G).$$

It follows that $((y,\hat{x}),(\tilde{y},\hat{x})) \in \tilde{\Phi}$. Hence, $\tilde{\Phi}$ is a strong simulation relation from $S\|G$ to $S_{W^\uparrow}\|G$.

REFERENCES

[1] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM J. Control Optim.*, vol. 25, no. 1, pp. 206–230, 1987.

[2] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language," *SIAM J. Control Optim.*, vol. 25, no. 3, pp. 637–659, 1987.

[3] R. Cieslak, C. Desclaux, A. S. Fawaz, and P. Varaiya, "Supervisory control of discrete-event processes with partial observations," *IEEE Trans. Autom. Control*, vol. 33, no. 3, pp. 249–260, 1988.

[4] F. Lin and W. Wonham, "On observability of discrete-event systems," *Inform. Sci.*, vol. 44, no. 3, pp. 173–198, 1988.

[5] H. Cho and S. I. Marcus, "On supremal languages of classes of sublanguages that arise in supervisor synthesis problems with partial observation," *Math. Control Sig. Syst.*, vol. 2, no. 1, pp. 47–69, 1989.

[6] S. Takai and T. Ushio, "Effective computation of an $L_m(G)$-closed, controllable, and observable sublanguage arising in supervisory control," *Syst. Control Lett.*, vol. 49, no. 3, pp. 191–200, 2003.

[7] K. Cai, R. Zhang, and W. Wonham, "Relative observability of discrete-event systems and its supremal sublanguages," *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 659–670, 2015.

[8] X. Yin and S. Lafortune, "Synthesis of maximally permissive supervisors for partially-observed discrete-event systems," *IEEE Trans. Autom. Control*, vol. 61, no. 5, pp. 1239–1254, 2016.

[9] C. Zhou, R. Kumar, and S. Jiang, "Control of nondeterministic discrete-event systems for bisimulation equivalence," *IEEE Trans. Autom. Control*, vol. 51, no. 5, pp. 754–765, 2006.

[10] A. Overkamp, "Supervisory control using failure semantics and partial specification," *IEEE Trans. Autom. Control*, vol. 42, no. 4, pp. 498–510, 1997.

[11] M. A. Shayman and R. Kumar, "Supervisory control of nondeterministic systems with driven events via prioritized synchronization and trajectory models," *SIAM J. Control Optim.*, vol. 33, no. 2, pp. 469–497, 1995.

[12] M. Heymann and F. Lin, "Discrete-event control of nondeterministic systems", *IEEE Trans. Autom. Control*, vol. 43, no. 1, pp. 3–17, 1998.

[13] R. Milner, *A Calculus of Communicating Systems*, Springer, 1980.

[14] C. Zhou and R. Kumar, "A small model theorem for bisimilarity control under partial observation," *IEEE Trans. Autom. Sci. & Eng.*, vol. 4, no. 1, pp. 93–97, 2007.

[15] C. Zhou and R. Kumar, "Bisimilarity enforcement for discrete event system using deterministic control," *IEEE Trans. Autom. Control*, vol. 56, no. 12, pp. 2986–2991, 2011.

[16] Y. Sun and H. Lin, "Bisimilarity enforcing supervisory control of nondeterministic discrete event systems," in *Proc. 2012 Amer. Control Conf.*, pp. 6102–6107, 2012.

[17] K. Kimura and S. Takai, "Bisimilarity control of nondeterministic discrete event systems under event and state observations," *IEICE Trans. Inf. & Syst.*, vol. E97-D, no. 5, pp. 1140–1148, 2014.

[18] S. Takai, "Bisimilarity enforcing supervisory control of nondeterministic discrete event systems with nondeterministic specifications," *Automatica*, vol. 108, article no. 108470, 2019.

[19] H. Farhat "Control of nondeterministic systems for bisimulation equivalence under partial information," *IEEE Trans. Autom. Control*, vol. 65, no. 12, pp. 5437–5443, 2020.

[20] C. Zhou and R. Kumar, "Control of nondeterministic discrete event systems for simulation equivalence," *IEEE Trans. Autom. Sci. & Eng.*, vol. 4, no. 3, pp. 340–349, 2007.

[21] K. Kimura and S. Takai, "Maximally permissive similarity enforcing supervisors for nondeterministic discrete event systems under event and state observations," *IEICE Trans. Fundamentals*, vol. 97-A, no. 7, pp. 1500–1507, 2014.

[22] N. Kushi and S. Takai, "Synthesis of similarity enforcing supervisors for nondeterministic discrete event systems," *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1457–1464, 2018.

[23] S. Takai, "Synthesis of maximally permissive supervisors for nondeterministic discrete event systems with nondeterministic specifications," *IEEE Trans. Autom. Control*, vol. 66, no. 7, pp. 3197–3204, 2021.

[24] J. Li and S. Takai, "Synthesis of maximally permissive supervisors for similarity control of partially observed nondeterministic discrete event systems," *Automatica*, vol. 135, article no. 109978, 2022.

[25] H. Yamada and S. Takai, "Nonblocking similarity control of nondeterministic discrete event systems under event and state observations," *IEICE Trans. Fundamentals*, vol. E101-A, no. 2, pp. 328–337, 2018.

[26] J. Li and S. Takai, "Maximally permissive nonblocking supervisors for similarity control of nondeterministic discrete event systems under event and state observations," *IEICE Trans. Fundamentals*, vol. E102-A, no. 2, pp. 399–403, 2019.

[27] J. Li and S. Takai, "Maximally permissive nonblocking similarity control of nondeterministic discrete event systems," in *Proc. 59th IEEE Conf. Decision Control*, pp. 73–78, 2020.

[28] R. Kumar, S. Jiang, C. Zhou, and W. Qiu, "Polynomial synthesis of supervisor for partially observed discrete-event systems by allowing nondeterminism in control," *IEEE Trans. Autom. Control*, vol. 50, no. 4, pp. 463–475, 2005.

[29] A. Tarski, "A lattice-theoretical fixpoint theorem and its applications," *Pacific J. Math.*, vol. 5, no. 2, pp. 285–309, 1955.

PLACE PHOTO HERE

Jinglun Li received the B.E. degree in electrical engineering from University of Shanghai for Science and Technology, Shanghai, China, in 2016 and the M.E. degree from Osaka University, Suita, Japan, in 2019, where he is currently pursuing the Ph.D degree. His current research interests include supervisory control theory of DESs and automata theory.

PLACE PHOTO HERE

Shigemasa Takai received the B.E. and M.E. degrees from Kobe University, Kobe, Japan, in 1989 and 1991, respectively, and the Ph.D degree from Osaka University, Suita, Japan, in 1995. From 1992 to 1998, he was a Research Associate at Osaka University. In 1998, he joined Wakayama University, Wakayama, Japan, as a Lecturer, and became an Associate Professor in 1999. From 2004 to 2009, he was an Associate Professor with Kyoto Institute of Technology, Kyoto, Japan. Since 2009, he has been a Professor with Osaka University, Suita, Osaka, Japan. His research interests include supervisory control and fault diagnosis of discrete event systems.