Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

MAS

Modelling, Analysis and Simulation

Modelling, Analysis and Simulation

Coinduction in Control of Partially Observed Discrete-Event Systems

J. Komenda

REPORT MAS-E0308 AUGUST 31, 2003

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3703

Coinduction in Control of Partially Observed Discrete-Event Systems

ABSTRACT
Coalgebra and coinduction provide new results and insights for the supervisory control of discrete-event systems (DES) with partial observations. In the case of full observations, coinduction has been used to define a new operation on languages called supervised product, which represents the language of the closed-loop system. The first language acts as a supervisor and the second as an open-loop system (plant). We show first that the supervised product is equal to the infimal controllable superlanguage of the supervisor's (specification) language with respect to the plant language. This can be generalized to the partial observation case, where the supervised product is shown to be equal to the infimal controllable and observable superlanguage. There are two different control laws for partially observed DES, that give the same closed-loop system if the specification is observable: permissive and antipermissive. A variation on the supervised product is presented, which corresponds to the control policy with the issue of of observability separated from the issue of controllability. It is shown to be equal to the infimal observable superlanguage. Similar idea for the antipermissive control law leads to a maximal observable sublanguage that contains the supremal normal sublanguage. We present an algorithm for its computation.

2000 Mathematics Subject Classification: 93C65
Keywords and Phrases: Discrete-Event Systems, Coalgebra, Coinduction, Supervisory control
Note: Also appeared in H. Peter Gumm (Ed.), Coalgebraic methods in Computer Science (CMCS'03), volume 82.1 of Electronic Notes in Theoretical Notes in Theoretical Computer Science, 2003.

Coinduction in Control of Partially Observed Discrete-Event Systems

Jan Komenda [1]

Centrum voor Wiskunde en Informatica (CWI)
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Abstract

Coalgebra and coinduction provide new results and insights for the supervisory control of discrete-event systems (DES) with partial observations. In the case of full observations, coinduction has been used to define a new operation on languages called supervised product, which represents the tuple of languages of the supervised system. The first language acts as a supervisor and the second as an open-loop system (plant). We show first that the supervised product is equal to the infimal controllable superlanguage of the supervisor's (specification) language with respect to the plant language. This can be generalized to the partial observation case, where the supervised product is shown to be equal to the infimal controllable and observable superlanguage. A modification on the supervised product is presented, which corresponds to the control policy for with the issue of observability is separated from the issue of controllability. The operation defined by coinduction is shown to be equal to the infimal observable superlanguage.

1 Introduction

Discrete-event (dynamical) systems (DES) have been studied using coalgebraic techniques [12], [8]. The reference model for DES are partial automata, which are coalgebras of a simple functor of the category of sets. They have been developed by J.J.M.M. Rutten in [12], i.e. partial automata as the model for control of DES with the partial automaton of (partial) languages as the final coalgebra. The main advantage of the use of coalgebra is the possibility to use the coinductive definitions and proofs that are shown to be pertinent in many situations. [8] presents a formulation of control of DES with partial observations in terms of coalgebra. The generalizations to partially observed DES is not straighforward, but requires the development of new concepts.

[1] Email: Jan.Komenda@cwi.nl

This is a preliminary version. The final version will be published in Electronic Notes in Theoretical Computer Science
URL: www.elsevier.nl/locate/entcs

Coinduction is used to define a binary operation on partial languages called supervised product, which represents the tuple of languages of the closed-loop system, where the first language is the language of the supervisor and the second language is that of the open-loop system. The supervised product corresponds to the infimal controllable and observable superlanguage. A minor modification is shown to be equal to the infimal observable superlanguage. We have obtained as a by-product coinductive definitions of these important languages. These results have been submitted for publication elsewhere [8], but here we focuse only on these coinductive techniques and present some new results as well (e.g. a coinductive definition of the supremal controllable sublanguage.)

The paper is organized as follows. Section 2 recalls the partial automata from [12] as the coalgebraic framework for DES represented by automata. The reader interested in more details about the key notions like bisimulation, coinduction, and finality should consult [13] or [12]. In Section 3 we introduce the reader into the supervisory control of DES. Section 4 presents a coalgebraic treatment of partially observed DES, where observability relations from [8] are recalled. In section 5 we define by coinduction the language of the closed-loop system as supervised product of the supervisor (specification) language and the open-loop (plant) language. Section 6 shows the power and flexibility of coinductive definitions: in the full observation case both the infimal controllable superlanguage and the supremal controllable sublanguage are defined by coinduction. In the partial observation case, by modifying the supervised product we obtain a coinductive definition of the infimal observable superlanguage.

2 Partial automata

In this section we recall from [12] partial automata as coalgebras with a special emphasis on the final coalgebra of partial automata, i.e. partial automaton of partial languages.

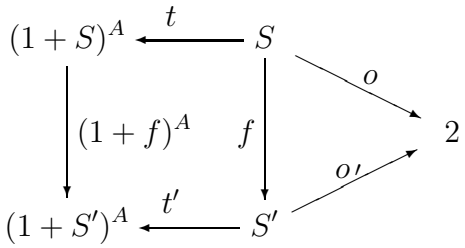
Let A be an arbitrary set (usually finite and referred to as the set of inputs or events). The empty string will be denoted by ε. Denote by $1 = \{\emptyset\}$ the one element set and by $2 = \{0, 1\}$ the set of Booleans. A partial automaton is a pair $S = (S, \langle o, t\rangle)$, where S is a set of states, and a pair of functions $\langle o, t\rangle \ : S \to 2 \times (1 + S)^A$, consists of an output function $o \ : \ S \to 2$ and a transition function $S \to (1 + S)^A$. The output function o indicates whether a state $s \in S$ is accepting (or terminating) : $o(s) = 1$, denoted also by $s \downarrow$, or not: $o(s) = 0$, denoted by $s \uparrow$. The transition function t associates to each state s in S a function $t(s) \ : \ A \to (1 + S)$. The set $1 + S$ is the disjoint union of S and 1. The meaning of the state transition function is that $t(s)(a) = \emptyset$ iff $t(s)(a)$ is undefined, which means that there is no $a-$transition from the state $s \in S$. $t(s)(a) \in S$ means that the $a-$transition from s is possible and we define in this case $t(s)(a) = s_a$, which is denoted mostly by $s \xrightarrow{a} s_a$. This

notation can be extended by induction to arbitrary strings in A^*. Assuming that $s \xrightarrow{w} s_w$ has been defined, define $s \xrightarrow{wa}$ iff $t(s_w)(a) \in S$, in which case $s_{wa} = t(s_w)(a)$ and $s \xrightarrow{wa} s_{wa}$.

A *homomorphism* between partial automata $S = (S, \langle o, t\rangle)$ and $S' = (S', \langle o', t'\rangle)$ is a function $f : S \to S'$ with, for all $s \in S$ and $a \in A$:

$$o'(f(s)) = o(s) \text{ and } s \xrightarrow{a} s_a \text{ iff } f(s) \xrightarrow{a} f(s)_a,$$

in which case: $f(s)_a = f(s_a)$.



A partial automaton $S' = (S', \langle o', t'\rangle)$ is a *subautomaton* of $S = (S, \langle o, t\rangle)$ if $S' \subseteq S$ and the inclusion function $i : S' \to S$ is a homomorphism.

A *simulation* between two partial automata $S = (S, \langle o, t\rangle)$ and $S' = (S', \langle o', t'\rangle)$ is a relation $R \subseteq S \times S'$ with, for all $s \in S$ and $s' \in S'$:

$$\text{if } \langle s, s'\rangle \in R \text{ then } \begin{array}{ll} (i) & o(s) \leq o(s'), \text{ i.e. } s\downarrow \Rightarrow s'\downarrow, \text{ and} \\ (ii) & \forall a \in A : \ s \xrightarrow{a} \Rightarrow (s' \xrightarrow{a}) \text{ and } \langle s_a, s'_a\rangle \in R, \end{array}$$

A *bisimulation* between two partial automata $S = (S, \langle o, t\rangle)$ and $S' = (S', \langle o', t'\rangle)$ is a relation $R \subseteq S \times S'$ with, for all $s \in S$ and $s' \in S'$:

$$\text{if } \langle s, s'\rangle \in R \text{ then } \begin{array}{ll} (i) & o(s) = o(s'), \text{ i.e. } s\downarrow \text{ iff } s'\downarrow \\ (ii) & \forall a \in A : \ s \xrightarrow{a} \Rightarrow (s' \xrightarrow{a}) \text{ and } \langle s_a, s'_a\rangle \in R, \text{ and} \\ (iii) & \forall a \in A : \ s' \xrightarrow{a} \Rightarrow (s \xrightarrow{a}) \text{ and } \langle s_a, s'_a\rangle \in R. \end{array}$$

We write $s \sim s'$ whenever there exists a bisimulation R with $\langle s, s'\rangle \in R$. This relation is the union of all bisimulations, i.e. the greatest bisimulation also called *bisimilarity.*

2.1 Final automaton of partial languages

Below we define the partial automaton of partial languages over an alphabet (input set) A, denoted by $\mathcal{L} = (\mathcal{L}, \langle o_{\mathcal{L}}, t_{\mathcal{L}}\rangle)$. More formally, $\mathcal{L} = \{\Phi : A^* \to (1+2) \mid dom(\Phi) \neq \emptyset \text{ is prefix-closed}\}$. To each partial language Φ a pair $\langle V, W\rangle$ can be assigned: $W = dom(\Phi)$ and $V = \{w \in A^* \mid \Phi(w) = 1 (\in 2)\}$. Conversely, to a pair $\langle V, W\rangle \in \mathcal{L}$, a function Φ can be assigned : $\Phi(w) = 1$ if $w \in V$, $\Phi(w) = 0$ if $w \in W$ and $w \notin V$, and $\Phi(w)$ is undefined if $w \notin W$.

Therefore we can write :

$$\mathcal{L} = \{(V, W) \mid V \subseteq W \subseteq A^*,\ W \neq \emptyset, \text{ and W is prefix-closed}\}.$$

The transition function $t_{\mathcal{L}} : \mathcal{L} \to (1+\mathcal{L})^A$ is defined using input derivatives. Recall that for any partial language $L = (L^1, L^2) \in \mathcal{L}$, $L_a = (L_a^1, L_a^2)$, where $L_a^i = \{w \in A^* \mid aw \in L^i\}$, $i = 1, 2$. If $a \notin L^2$ then L_a is undefined. Given any $L = (L^1, L^2) \in \mathcal{L}$, the partial automaton structure of $\mathcal{L}$ is given by:

$$o_{\mathcal{L}}(L) = \begin{cases} 1 & \text{if } \varepsilon \in L^1 \\ 0 & \text{if } \varepsilon \notin L^1 \end{cases}$$

and

$$t_{\mathcal{L}}(L)(a) = \begin{cases} L_a & \text{if } L_a \text{ is defined} \\ \emptyset & \text{otherwise} \end{cases}.$$

Notice that if L_a is defined, then $L_a^1 \subseteq L_a^2$, $L_a^2 \neq \emptyset$, and L_a^2 is prefix-closed. The following notational conventions will be used: $L \downarrow$ iff $\varepsilon \in L^1$, and $L \xrightarrow{w} L_w$ iff L_w is defined iff $w \in L^2$.

Recall from [12] that $\mathcal{L} = (\mathcal{L}, \langle o_{\mathcal{L}}, t_{\mathcal{L}} \rangle)$ is final among all partial automata: for any partial automaton $S = (S, \langle o, t \rangle)$ there exists a unique homomorphism $l : S \to \mathcal{L}$. Another characterization of finality of $\mathcal{L}$ is that it satisfies the principle of coinduction: for all K and L in $\mathcal{L}$, if $K \sim L$ then $K = L$. Definition by coinduction of an operation on elements of a final coalgebra consists in defining the same coalgebraic structure on the operation (for instance we define binary operations on partial languages by defining derivatives and output functions further in this paper). More details about coinduction and finality can be found in [13] or [12]. Recall that the unique homomorphism l given by finality of $\mathcal{L}$ maps a state $s \in S$ to the partial language $l(s) = (L_s^1, L_s^2) = (\{w \in A^* \mid s \xrightarrow{w} \text{ and } s_w \downarrow\}, \{w \in A^* \mid s \xrightarrow{w}\})$.

We adopt the notation from [11], page 9, easily extended from automata to partial automata, and denote the minimal (in size of the state set) representation of a partial language L by $\langle L \rangle$. Hence, $\langle L \rangle = (DL, \langle o_{\langle L \rangle}, t_{\langle L \rangle} \rangle)$ is a subautomaton of $\mathcal{L}$ generated by L. This means that $o_{\langle L \rangle}$ and $t_{\langle L \rangle}$ are uniquely determined by the corresponding structure of $\mathcal{L}$. The carrier set of this minimal representation of L is denoted by DL, where $DL = \{L_u \mid u \in L^2\}$. Let us call this set the *set of derivatives* of L. Inclusion of partial languages that corresponds to a simulation relation is meant componentwise. Some further notation from [12] is used, e.g. 'zero' (partial) language is denoted by 0, i.e. $0 = (\emptyset, \{\varepsilon\})$.

There is yet another important concept that will be needed in this paper. Namely, given an (ordinary) language L, the *suffix closure* of L is defined by $\text{suffix}(L) = \{s \in A^* \mid \exists u \in A^* \text{ with } us \in L\}$. For partial languages, the suffix closure is defined in the same way as the prefix closure, i.e. componentwise.

There is the following relation between the transition structure of L and its suffix closure operator.

Lemma 2.1 *For each (partial) language L:* $\text{suffix}(L) = \cup_{u \in L^2} L_u$.

Proof. Immediate from $L_u = (\{s \in A^* \mid us \in L^1\},\ \{s \in A^* \mid us \in L^2\})$. □

3 Introduction to supervisory control of partially observed DES

In this section we introduce the reader to the supervisory control of DES formulated within the algebraic framework.

In supervisory control of DES represented by generators in the form of a partial automaton S, the aim is to restrict the behavior (i.e. partial language $L(S) = (L(S)^1, L(S)^2)$) of a partial automaton to an admissible behavior that is given by a specification language K. The specification language encodes that certain strings are not legal, liveness of the system, and possibly fairness in regard to two or more agents. We assume that $A = A_c \cup A_{uc}$ is a partition of A into controllable events (A_c) and uncontrollable (A_{uc}) events. Only controllable events can be disabled. A_{uc} are uncontrollable events that cannot be disabled by any supervisor. We suppose that $K \subseteq L(S)$.

In the case of full observations a controller can monitor every event that is executed by the system, hence it knows exactly the state of the system. A supervisory controller V interacting with S is adjoined to S. V then influences the DES by specifying after observing an event a subset of the controllable events which are then possible in the uncontrolled system and are enabled/disabled by V.

The action of the supervisor is to enable only a subset of events that are possible in the uncontrolled system based on the observed past behavior of the plant. Events that are not enabled by the supervisor and are in the plant are disabled in the closed-loop system that is formed of the plant and the supervisor. The map that associates to a possible observed word a set of enabled events when the supervisor observes this string from the plant is called a *control law*. Supervisory control of DES consists in restricting the behavior of a partial automaton S to a sublanguage of $L(S)$ called a *legal* (or admissible) *behaviour* given by a specification language.

The set of all control patterns, i.e. the set of events that are enabled is $\Gamma = \{T \subseteq A \mid T \supseteq A_{uc}\}$.
Formally, a control law of the supervisor is a function $\gamma_V : L(S)^2 \to \Gamma$.

The resulting (closed-loop) system is denoted by V/S and the uncontrolled system has language $(L(S)^1, L(S)^2)$. The (partial) language of V/S is the language $(L(V/S)^1, L(V/S)^2)$ defined by induction as follows:

(i) $\varepsilon \in L(V/S)^2$

(ii) $sa \in L(V/S)^2$ iff $s \in L(V/S)$ and $a \in \gamma_V(s)$ and $sa \in L(S)^2$.

The *marked behavior* of V/S is $L(V/S)^1 = L(V/S)^2 \cap L(S)^1$. We have therefore $\emptyset \subseteq L(V/S)^1 \subseteq L(S)^1$. Supervisor V is said to be *nonblocking* if $L(\bar{V/S})^1 = L(V/S)^2$.

Definition 3.1 A (partial) language K is said to be *controllable* with respect to $L = L(S)$ and A_{uc} if $\forall s \in K^2$ and $\forall a \in A_{uc}$ such that $sa \in L^2$ we have $sa \in K^2$. An equivalent statement is: $K^2 A_{uc} \cap L^2 \subseteq K^2$.

Theorem 3.2 *[14] There exists a nonblocking supervisory controller V for partial automaton S such that $L(V/S)^1 = K^1$ and $L(V/S)^2 = K^2$ iff*

(i) *K is controllable with respect to $L(S)$ and A_{uc} and*

(ii) *$K^1 = K^2 \cap L(S)^1$. (K is $L(S)^1-$closed.)*

Corollary 3.3 *Let $K \subseteq L(S)$ be nonempty. There exists a supervisory controller V for S such that $L(V/S)^1 = K^1$ and $L(V/S)^2 = K^2$ iff K is controllable with respect to $L(S)$ and A_{uc}.*

The underlying control law is:

$$\gamma_V(s) = A_{uc} \cup \{a \in A_c : \ sa \in K^2\}.$$

Under the assumption of partial observations, a controller can monitor only a subset of all events, i.e. only a subset of events are observed. For example, an unobservable event is a failure of an element in a network. Let $A = A_o \cup A_{uo}$ be a partition of A into observable events (A_o) and unobservable (A_{uo}) events with the natural projection $P : A^* \to A_o^*$. Recall that $P(a) = \varepsilon$ for any $a \in A_{uo}$, $P(a) = a$ for $a \in A_o$, and P is catenative. This means that the supervisor must decide what action to take regarding an event based only on the projected past behavior. Formally, the action of the supervisory controller (the events enabled after supervisor observes $s \in A_o^*$) is defined as: $\gamma_V^P \ : \ P(L(S))^2 \to \Gamma$.

Let $K = (K^1, K^2)$ be the desired behavior (partial language) and V be the supervisory controller.

The control law under partial observations is ($\forall s \in A_o^*$ the events enabled after supervisor observes $s \in A_o^*$):

$$\gamma_V^P(s) = A_{uc} \cup \{a \in A_c : \ \exists s' \in K^2 \text{ with } P(s') = P(s) \text{ and } s'a \in K^2 \ \ \}.$$

It has been shown [10] that for partially-observed DES in addition to the above mentioned controllability, and $L(S)^1-$closedness of K, yet another condition called *observability* is necessary for achieving this language as a desirable behavior of the resulting (closed-loop) system. It will be studied in the next section using the coalgebraic framework of the preceding section. The intuition is that two strings in language K^2 with the same projection must behave the same with respect to the membership in K^2 and L^2 of their prolongation by any controllable event.

Definition 3.4 (Observability.) A partial language K is said to be *observable* with respect to another partial language L (with $K \subseteq L$) and projection P if for all $s \in K^2$ and $a \in A$ the following implication holds true :

$$sa \in L^2,\ s'a \in K^2, \text{ and } P(s) = P(s') \ \Rightarrow \ sa \in K^2.$$

The following theorem gives necessary and sufficient conditions for a given partial language K to be exactly achievable by a supervisor.

Theorem 3.5 *[10] There exists a nonblocking supervisory controller with partial observations V for partial automaton S such that $L(V/S)^1 = K^1$ and $L(V/S)^2 = K^2$ if and only if*

(i) *K is controllable with respect to $L(S)$ and A_{uc}*

(ii) *K is observable with respect to L and P,and*

(iii) *$K^1 = K^2 \cap L(S)^1$. (K is $L(S)^1-$closed.)*

4 Observability and coalgebra

In the following definition we introduce the notion of weak derivative (transition). Roughly speaking it disregards unobservable steps, which correspond to so called internal moves in the framework of process algebras.

Definition 4.1 (Nondeterministic weak transitions.) For an event $a \in A$ and a state s in partial automaton $S = (S, \langle o, t\rangle)$ we define $s \overset{P(a)}{\Rightarrow} s'$ if there exists $u \in A^*$ such that $P(u) = P(a)$ and $s \overset{u}{\rightarrow} s' = s_u$).

Remark 4.2 According to this notation for unobservable events $s \overset{\varepsilon}{\Rightarrow} s'$ is an abbreviation for $\exists \tau \in A_{uo}^*$ such that $s \overset{\tau}{\rightarrow} s' = s_\tau$. For $a \in A_o$ our notation means that there exist $\tau, \tau' \in A_{uo}^*$ such that $s \overset{\tau a \tau'}{\rightarrow} s' = s_{\tau a \tau'}$. This definition can be extended to strings (words in A^*) in the following way:
$s \overset{P(w)}{\Rightarrow} s'$ iff $\exists t \in A^* \ : \ P(w) = P(t)$ and $s \overset{t}{\rightarrow} s' = s_t$.

There may exist two or more $u \in A^*$ satisfying the condition in the definition of weak transition. Hence, the weak transition structure introduced above is not deterministic. We have introduced deterministic weak transition structure on $\mathcal{L}$ in [8], but it will not be needed in this paper.

Let us recall first the concept of control relation introduced in [12].

Definition 4.3 (Control relation.) Given two partial automata $S' = (S', \langle o', t'\rangle)$ and $S = (S, \langle o, t\rangle)$ as above, a binary relation C on $S' \times S$ is called a *control relation* if for any $\langle s, t\rangle \in C$ the following items hold:

(i) $\forall a \in A \ : \ s \overset{a}{\rightarrow}\!\!\!\!\!\cdot\ s_a$ and $t \overset{a}{\rightarrow} t_a \ \Rightarrow \ \langle s_a, t_a\rangle \in C$

(ii) $\forall u \in A_{uc} : \ t \overset{u}{\rightarrow} t_u \ \Rightarrow \ s \overset{u}{\rightarrow}\!\!\!\!\!\cdot\ s_u$ and $\langle s_u, t_u\rangle \in C$.

Control relation corresponds to controllability of a language K with respect to L and A_{uc}. It is also necessary for achieving K. Recall that K is

controllable with respect to L and A_{uc} iff there exists a control relation C such that $(K, L) \in C$. Our aim is to find a relational characterization of observability. The following auxiliary relation is needed, where s_0 denotes the initial state of S.

Definition 4.4 (Observational indistinguishability relation on S.) A binary relation $Aux(S)$ on S is called an *observational indistinguishability relation* if the following two conditions hold:

(i) $\langle s_0, s_0\rangle \in Aux(S)$

(ii) If $\langle s,t\rangle \in Aux(S)$ then $\forall a \in A : \quad (s \overset{P(a)}{\Rightarrow} s'$ for some s' and $t \overset{P(a)}{\Rightarrow} t'$ for some t' $) \Rightarrow \langle s', t'\rangle \in Aux(S)$

From the definition of weak transitions it follows that (ii) is equivalent to (ii)' and (iii)' below:
(ii)' If $\langle s,t\rangle \in Aux(S)$ then : $(s \overset{\varepsilon}{\Rightarrow} s'$ for some s' and $t \overset{\varepsilon}{\Rightarrow} t'$ for some t' $) \Rightarrow \langle s', t'\rangle \in Aux(S)$
(iii)' If $\langle s,t\rangle \in Aux(S)$ then $\forall a \in A_o$: $(s \overset{a}{\rightarrow} s_a$ and $t \overset{a}{\rightarrow} t_a$ $) \Rightarrow \langle s_a, t_a\rangle \in Aux(S)$. $Aux(S)$ can be characterized by the following lemma.

Lemma 4.5 *For any $s, s' \in S$: $\langle s, s'\rangle \in Aux(S)$ iff there exist two strings $w, w' \in K^2$ such that $P(w) = P(w')$, $s = (s_0)_w$, and $s' = (s_0)_{w'}$.*

Proof. ($\Leftarrow$) Let $s, s' \in S$ such that there exist two strings $w, w' \in K^2$ with $P(w) = P(w')$ and $s = (s_0)_w$ and $s' = (s_0)_{w'}$. Let $w = w_1 \dots w_n$, $w' = t_1 \dots t_m$, and $P(w) = P(w') = a_1 \dots a_k$. Then $n \geq k$ and $m \geq k$ and there exists two increasing sequences of integers (indices) $u_i \geq i,\ i = 1, \dots, k$ and $v_i \geq i,\ i = 1, \dots, k$ such that $a_i = w_{u_i} = t_{v_i}$. Since all a_i are observable events we can write $s_0 \overset{P(a_1)\dots P(a_k)}{\Longrightarrow} s$ and $s_0 \overset{P(a_1)\dots P(a_k)}{\Longrightarrow} s'$, whence by (ii) inductively applied $\langle s, s'\rangle \in Aux(S)$.
($\Rightarrow$) Let $\langle s, s'\rangle \in Aux(S)$. Then by the construction of $Aux(S)$ there exist $a_1, \dots, a_k \in A$ such that $s_0 \overset{P(a_1)\dots P(a_k)}{\Longrightarrow} s$ and $s_0 \overset{P(a_1)\dots P(a_k)}{\Longrightarrow} s'$. Therefore there exist by definition of nondeterministic weak transitions two strings w, w' with the same projection such that $s = (s_0)_w$ and $s' = (s_0)_{w'}$. □

Our aim now is to provide a coalgebraic characterization of observability. Since observability is a property of the second (closed) components of K and L, we can assume that $S_1 = (S_1, \langle o_1, t_1\rangle)$ is a partial automaton with initial state $s_0 \in S$ that represents K in the sense $K = l_1(s_0)$, $l_1 : S_1 \to \mathcal{L}$ being the unique behavior homomorphism defined by finality of $\mathcal{L}$. Moreover, since $K \subseteq L$, we can assume that S_1 is a subautomaton of $S = (S, \langle o, t\rangle)$ with $L = l(s_0)$ ($l : S \to \mathcal{L}$ is the behavior homomorphism) and s_0 their common initial state. Let the transition function of S be denoted by $\rightarrow$, i.e. $s \overset{a}{\rightarrow} s_a$ means $s_a = t(s)(a)$ and similarly the transition function t_1 of S_1 is denoted by $\rightarrow_1$, i.e. $s \overset{a}{\rightarrow}_1 s^1_a$ means $s^1_a = t_1(s)(a)$. Notice also that due to the requirement that S_1 is a subautomaton of S, we have in fact $s^1_a = s_a \in S_1$. It means

that the superscript ^{1} can be dropped here. Let us introduce observability relations, in which the observational indistinguishability relation is involved.

Definition 4.6 (Observability relation.) A binary relation $O(S_1, S)$ on $S_1 \times S$ is called the *observability relation* if for any $\langle s, t\rangle \in O(S_1, S)$ the following items hold:

(i) $\forall a \in A:\ s \xrightarrow{a}_1 s_a \ \Rightarrow\ t \xrightarrow{a} t_a$ and $\langle s_a, t_a\rangle \in O(S_1, S)$

(ii) $\forall a \in A:\ t \xrightarrow{a} t_a$ and $(\exists s' : \langle s, s'\rangle \in Aux(S_1) : s' \xrightarrow{a}_1 s'_a)$
$\Rightarrow\ s \xrightarrow{a}_1 s_a$ and $\langle s_a, t_a\rangle \in O(S_1, S)$.

For $s \in S_1$ and $s' \in S$ we write $s \approx_{O(S_1,S)} s'$ whenever there exists an observability relation $O(S_1, S)$ on $S_1 \times S$ such that $\langle s, s'\rangle \in O(S_1, S)$. It has been proven in [6] that:

Theorem 4.7 *A (partial) language K is observable with respect to L ($K \subseteq L$) and P iff $s_0 \approx_{O(S_1,S)} s_0$.*

5 Coinductive definition of supervised product and partial bisimulation under partial observations.

In this section we present the definition of a supervised product of languages that describes the behavior of a supervised DES under partial observations. Assume throughout this section that the specification K and the open-loop partial language L ($K \subseteq L$) are specified.

In the last section we have recalled from [6] relations on automata representations and we have formulated the property of observability using these relations. Here we aim at using the coinductive definitions. For this reason we must work with the final automaton of partial languages, where the coinductive definitions can be used. Note that observability relations can be defined on the final automaton $\mathcal{L}$. However, there is a difficulty with the fact that once we use the minimal representations $\langle K\rangle$, $\langle L\rangle \in \mathcal{L}$ as the subautomata of $\mathcal{L}$ generated by K and L, respectively, it is not true in general that for $K \subseteq L$, $\langle K\rangle$ is a subautomaton of $\langle L\rangle$. Therefore some additional technicalities are involved. In particular, $Aux(S_1)$ is replaced by $Aux(K, L)$ to stress the fact that both $\langle K\rangle$ and $\langle L\rangle$ are involved. Its definition has been presented in [7].

In order to characterize the observability we first need to introduce the following auxiliary relation defined on $DK \times DL$. Note that any relation $R \subseteq (DK \times DL)^2$ can be endowed with the following transition structure: for $a \in A$ $(M, N) \xrightarrow{a} (M', N')$ iff $M \xrightarrow{a} M_a$ and $N \xrightarrow{a} N_a$ with $M' = M_a$ and $N' = N_a$. We write also $(M, N) \overset{P(a)}{\Rightarrow} (M', N')$ iff $\exists s \in M^2 \cap N^2$: $P(s) = a$, $M' = M_s$, and $N' = N_s$.

Definition 5.1 A binary relation $Aux(K, L) \subseteq (DK \times DL)^2$ is called *observational indistinguishability relation* if the following conditions hold:

(i) $\langle (K, L), (K, L)\rangle \in Aux(K, L)$

(ii) If $\langle (M,N),(Q,R)\rangle \in Aux(K,L)$ then $\forall a \in A:$ if $(M,N) \stackrel{P(a)}{\Rightarrow} (M',N')$ and $(Q,R) \stackrel{P(a)}{\Rightarrow} (Q',R') \Rightarrow \langle (M',N'),(Q',R')\rangle \in Aux(K,L)$

For $\langle (M,N),(Q,R)\rangle \in DK \times DL$ we write $(M,N) \approx_{Aux}^{K,L} (Q,R)$ whenever $\langle (M,N),(Q,R)\rangle \in Aux(K,L)$. Similarly as Lemma 4.5 it is easy to show that

Lemma 5.2 *For given partial languages K, L: $\langle (M,N),(Q,R)\rangle \in Aux(K,L)$ iff there exist two strings $s, s' \in K^2$ such that $P(s) = P(s')$, $M = K_s$, $N = L_s$, $Q = K_{s'}$, and $R = L_{s'}$.*

Now we repeat the definition of observability relation used in [7].

Definition 5.3 (Observability relation.) Given two (partial) languages K and L, a binary relation $O(K,L) \subseteq DK \times DL$ is called an *observability relation* if for any $\langle M,N\rangle \in O(K,L)$ the following items hold:

(i) $\forall a \in A:\ M \stackrel{a}{\rightarrow} \Rightarrow \quad N \stackrel{a}{\rightarrow}$ and $\langle M_a, N_a\rangle \in O(K,L)$

(ii) $\forall a \in A:\ N \stackrel{a}{\rightarrow}$ and $(\exists M' \in DK, N' \in DL:\ (M',N') \approx_{Aux}^{K,L} (M,N)$ and $M' \stackrel{a}{\rightarrow}) \Rightarrow M \stackrel{a}{\rightarrow}$

For $M \in DK$ and $N \in DL$ we write $M \approx_{O(K,L)} N$ whenever there exists an observability relation $O(K,L)$ on $DK \times DL$ such that $\langle M,N\rangle \in O(K,L)$. We have shown in [7]:

Theorem 5.4 *A (partial) language K is observable with respect to L (with $K \subseteq L$) and P iff $K \approx_{O(K,L)} L$.*

In order to check whether for a given pair of (partial) languages (K and L), K is observable with respect to L, it is sufficient to establish an observability relation $O(K,L)$ on $DK \times DL$ such that $\langle K,L\rangle \in O(K,L)$.

Remark 5.5 In the sequel we need also another type of auxiliary relations $Aux(S)$ for the special case $S = \langle K\rangle$. We will write $Aux(K)$ instead of $Aux(\langle K\rangle)$. Notice that it is possible to extend the definition of $Aux(K)$ from DK to Pwr(suffix(K)) with the only difference, that the propagation of this relation is realized by unions of nondeterministic transitions. The following definition will consider arguments from Pwr(suffix(K)) and Pwr(suffix(L)) rather than from DK and DL. In fact we will work with unions of the form $\cup_{i=1}^{k} K_{s_i} \in$ Pwr(suffix(K)), where $P(s_1) = \ldots = P(s_k)$. In order to keep the notation simple, we will use an extension of $Aux(K)$ to such unions of derivatives. In the definition of supervised product this will be needed.

Now we give a formal definition of $Aux(K)$ extended to Pwr(suffix(K)).

Definition 5.6 (Extension of $Aux(K)$ from DK to Pwr(suffix(K)).) A binary relation $Aux(K) \subseteq (\text{Pwr}(\text{suffix}(K)))^2$ is called an *observational indistinguishability relation* if the following two conditions hold :

(i) $\langle (K,K) \in Aux(K)$

(ii) If $\langle M,N\rangle \in Aux(K)$ then $\forall a \in A$: if $M \xrightarrow{a}$ M_a and $N \xrightarrow{a}$ N_a $\Rightarrow$ $\langle M_a, N_a\rangle \in Aux(K)$

(iii) If $\langle M,N\rangle \in Aux(K)$ then if $M \stackrel{\varepsilon}{\Rightarrow} M_1, M \stackrel{\varepsilon}{\Rightarrow} M_2, \ldots, M \stackrel{\varepsilon}{\Rightarrow} M_n$, and $N \stackrel{\varepsilon}{\Rightarrow} N_1, \ldots, N \stackrel{\varepsilon}{\Rightarrow} N_m$, then $\langle \cup_{i=1}^n M_i, \cup_{j=1}^m N_j\rangle \in Aux(K)$.

Clearly, an extension of Lemma 4.5 holds. Namely, $\langle \cup_{i=1}^k K_{s_i}, \cup_{j=1}^l L_{t_j}\rangle \in Aux(K)$, where $P(s_1) = \ldots = P(s_k)$ and $P(t_1) = \ldots = P(t_l)$ iff $P(s_1) = P(t_1)$, which implies naturally $P(s_i) = P(t_j)\ \forall i,j$. The notation $\cup_{i=1}^k K_{s_i} \approx_{Aux}^K \cup_{j=1}^l L_{t_j}$ is also used. Now we are ready for the coinductive definition of supervised product.

Definition 5.7 (Supervised product under partial observations.) Define the following binary operation on (partial) languages called supervised product under partial observations for all $M \in \mathrm{Pwr}(\mathrm{suffix}(K))$ and $N \in \mathrm{Pwr}(\mathrm{suffix}(L))$:

$$(M/_U^O N)_a =$$

(1) $M_a/_U^O N_a$ if $M \xrightarrow{a}$ and $N \xrightarrow{a}$;

(2) $(\cup_{\{M':\langle M',M\rangle \in Aux(K)\}} M'_a)/_U^O N_a$ if $M \not\xrightarrow{q}$ and $\exists M' \in DK : M' \approx_{Aux}^K M$ such that $M' \xrightarrow{a}$ and $N \xrightarrow{a}$ and $a \in A_c \cup A_o$;

(3) $0/_U^O N_a$ if $M \not\xrightarrow{q}$ and $(\forall M' \in DK : M' \approx_{Aux}^K M)$ $M' \not\xrightarrow{q}$ and $N \xrightarrow{a}$ and $a \in A_{uc} \cap A_o$;

(4) $M/_U^O N_a$ if $M \not\xrightarrow{q}$ and $N \xrightarrow{a}$ and $a \in A_{uc} \cap A_{uo}$;

(5) $\emptyset$ otherwise

and $(M/_U^O N)\downarrow$ iff $N\downarrow$.

Remark 5.8 1. According to Lemma 2.1, $DL \subseteq \mathrm{Pwr}(\mathrm{suffix}(L))$ and since $K \subseteq L$ also $DK \subseteq \mathrm{Pwr}(\mathrm{suffix}(L))$.
2. It follows from the definition of supervised product that $K \subseteq (K/_U^O L) \subseteq L$. Both inclusions can be verified by construction of the corresponding simulation relations. Let us show that $K \subseteq (K/_U^O L)$. Consider the following relation:

$$R(K,L) = \{\langle K_w, (K/_U^O L)_w\rangle \mid w \in K^2\} \subseteq DK \times D(K/_U^O L).$$

It is easy to see that $R(K,L)$ is a simulation relation proving the claimed inclusion. Take $w \in K^2$.
(i) If $K_w\downarrow$, then $w \in K^1$, i.e. $w \in L^1$, which means $L_w\downarrow$. Furthermore, it follows from the definition of 5.7 that $(K/_U^O L)_w = K_w/_U^O L_w$. Therefore, $(K/_U^O L)_w\downarrow$.
(ii) if for $a \in A$: $K_w \xrightarrow{a}$, then $(K/_U^O L)_{wa} = (K_w/_U^O L_w)_a = K_{wa}/_U^O L_{wa}$, i.e. $(K/_U^O L)_w \xrightarrow{a}$ and $\langle K_{wa}, (K/_U^O L)_{wa}\rangle \in R(K,L)$.

As a consequence we conclude that the range of supervised product is again $\mathrm{Pwr}(\mathrm{suffix}(L))$. Therefore, supervised product can be also viewed as a (partial) binary operation on $\mathrm{Pwr}(\mathrm{suffix}(L))$.

The definition of supervised product under partial observations is quite complicated due to the interconnections between observability and controllability that must be taken into account. It deserves additional comments. Notice that several cases must be distinguished. First of all, by (1) the controller allows any event that does not exit from its (supervisor) language. A controllable event is enabled when the supervisor observes $s \in A_o^*$ iff there exists a string with the same projection as s that can be prolongated by this event within the supervisor's language, which is included in (2). The controller also enables all uncontrollable events that are possible in the plant, but the future actions depend on whether the occured uncontrollable event is observable or not. If the uncontrollable event is unobservable then the first component of the supervised product need not to move, but only the second component is updated as is seen from (4) above. In the case that the uncontrollable event a is observable, there must be further specified whether there exists a derivative indistinguishable from the derivative currently considered that can make an $a-$transition (i.e. there exists a string that has the same projection as s that can be prolongated by a within the supervisor's language), in which case the action is the same as for controllable events (i.e. this case is included in (2) above), or whether there is no such a derivative, which means that only uncontrollable events that are possible in the plant are allowed in the future. The latter case corresponds to the term containing the zero partial language and is labelled by (3) above. In any other case (5) the controllable events are disabled by the supervisor. We have therefore the coinductive definition of the closed-loop language that gives a clear picture of what is the mechanism of discrete-event control under partial observations. It is a coalgebraic formulation of theorem 3.5.

Recall from [7] the concept of partial bisimulation under partial observations.

Definition 5.9 (Partial bisimulation.) A binary relation $R(K,L) \subseteq DK \times DL$ is called a *partial bisimulation under partial observations* if for all $\langle M, N\rangle \in R(K,L)$:
(i) $o(M) = o(N)$ $(M\downarrow \text{ iff } N\downarrow)$
(ii) $\forall a \in A:\ M \xrightarrow{a} \Rightarrow N \xrightarrow{a}$ and $\langle M_a, N_a\rangle \in R(K,L)$
(iii) $\forall u \in A_{uc}:\ N \xrightarrow{u} \Rightarrow M \xrightarrow{u}$
(iv) $\forall a \in A_c:\ N \xrightarrow{a}$ and $(\exists (M',N') \approx^{K,L}_{Aux} (M,N):\ M' \xrightarrow{a})\ \Rightarrow\ M \xrightarrow{a}$.

For $M \in DK$ and $N \in DL$ we write $M \approx_U^{O(K,L)} N$ whenever there exists a partial bisimulation under partial observations $R(K,L)$ such that $\langle M, N\rangle \in R(K,L)$. This relation is called partial bisimilarity under partial observations.

Remark 5.10 Notice that (i) relates the marking components of the languages involved and (ii) corresponds to the language simulation (inclusion), while (iii) to the controllability and (iv) to the observability condition. The condition for observability is required only for $a \in A_c$, because by (iii) a stronger condition (controllability) is required for $a \in A_{uc}$.

We have formulated in [8] a coalgebraic formulation of the controllability and observability theorem for supervisory control of DES with partial observations.

Theorem 5.11 *Let $K \subseteq L$ are given partial languages. Then $K \approx_U^{O(K,L)} L$ iff $K = K/_U^O L$. The supervised product under partial observations of the languages K and L equals K iff K and L are partially bisimilar in the sense of Definition 5.9.*

6 Infimal observable superlanguages and supremal controllable sublanguages

In the last section we have introduced an operation on partial languages called supervised product under partial observations. This operation corresponds to the behavior of the supervised (closed-loop) discrete-event system modelled by a partial automaton.

Remark 6.1 We consider from now on an order relation on partial languages induced by their second components only, i.e. we write $K \subseteq L$ iff $K^2 \subseteq L^2$. The same applies for infimum and supremum operations. Note that only the second condition of simulation relations [12] must be checked to prove such defined inclusion of partial languages.

Let us recall the coinductive definition of supervised product in the case of full observations from [12]. It turns out that the supervised product defined therein provides the infimal controllable superlanguage. As a by-product we have its coinductive definition.

Definition 6.2 ([12]) Define the following binary operation on (partial) languages for all $K, L \in \mathcal{L}$ and $\forall a \in A$:

$$(K/_U L)_a = \begin{cases} K_a/_U L_a & \text{if } K \xrightarrow{a} \text{ and } L \xrightarrow{a} \\ 0/_U L_a & \text{if } K \overset{a}{\not\to} \text{ and } L \xrightarrow{a} \text{ and } a \in A_{uc} \\ \emptyset & \text{otherwise} \end{cases}$$

and $(K/_U L)\downarrow$ iff $L\downarrow$.

Theorem 6.3 $(K/_U L) = \inf\{M \supseteq K : M \text{ is controllable with respect to } L \text{ and } A_{uc}\}$, *i.e. $K/_U L$ equals the infimal controllable superlanguage of K.*

Proof. Let us show that $K/_U L$ is a superlanguage of K that is controllable with respect to L and A_{uc}. It is clear from the definition of supervised product that $K \subseteq (K/_U L)$ in the sense of Remark 6.1. Let us show that $K/_U L$ is controllable with respect to L and A_{uc}. It is sufficient to prove that the following relation is a control relation.

$$C = \{\langle (K/_U L), L\rangle \mid K, L \in \mathcal{L}\}.$$

(i) Let $(K/_U L) \xrightarrow{a}$ and $L \xrightarrow{a}$ for $a \in A$. Then by coinductive definition of $K/_U L$ either $(K/_U L)_a = (K_a/_U L_a)$ or $(K/_U L)_a = (0/_U L_a)$. However, by definition of C in both cases we have $\langle (K/_U L)_a, L_a \rangle \in C$.
(ii) If $L \xrightarrow{u}$ for $u \in A_{uc}$, then either $K \xrightarrow{u}$ and hence $(K/_U L) \xrightarrow{u}$ or $K \not\xrightarrow{u}$, but according to the definition of $K/_U L$ we have still $(K/_U L) \xrightarrow{u} (0/_U L_u)$.

It remains to show the infimality. Let $M \supseteq K$ be controllable with respect to L and A_{uc}. Then

$$R = \{\langle (K/_U L), M\rangle \mid K, L, M \in \mathcal{L} :\ K \subseteq M \subseteq L,\ \text{ and } M^2 A_{uc} \cap L^2 \subseteq M^2\}.$$

satisfies (ii) of simulation relations. Let $(K/_U L) \xrightarrow{a}$ for $a \in A$. According to the definition of $K/_U L$ we have two possibilities: either $K \xrightarrow{a}$ and $L \xrightarrow{a}$, in which case $(K/_U L)_a = K_a/_U L_a$ or $K \not\xrightarrow{a}$ and $L \xrightarrow{a}$ and $a \in A_{uc}$. In the first case we have $M \xrightarrow{a}$ simply because $K \xrightarrow{a}$ and $K \subseteq M$, while in the latter case we have $M \xrightarrow{a}$ because of the controllability of M with respect to L and A_{uc} (by definition 4.3 of control relations for $a \in A_{uc}$: $L \xrightarrow{a} \Rightarrow M \xrightarrow{a}$). Moreover in both cases $\langle (K/_U L)_a, M_a\rangle \in R$. Thus (ii) of simulation relations (section 2) holds and $K/_U L \subseteq M$. □

Although the infimal controllable superlanguages are important [9], supremal controllable sublanguages are even more interesting. In [12] an algorithm for their computation, based on control relations, has been presented. It turns out that it is also possible to define the supremal controllable sublanguage by coinduction.

Definition 6.4 Define the following binary operation on (partial) languages for all $K, L \in \mathcal{L}$ and $\forall a \in A$:

$$(K/^S_C L)_a = \begin{cases} K_a/^S_C L_a & \text{if } K \xrightarrow{a} \text{ and } L \xrightarrow{a} \\ & \text{and if } \forall u \in A^*_{uc} :\ L_a \xrightarrow{u} \Rightarrow\ K_a \xrightarrow{u} \\ \emptyset & \text{otherwise} \end{cases}$$

and $(K/^S_C L)\downarrow$ iff $L\downarrow$.

Theorem 6.5 $(K/^S_C L) = \sup\{M \subseteq K :\ M$ *is controllable with respect to* L *and* $A_{uc}\}$*, i.e.* $K/^S_C L$ *equals the supremal controllable sublanguage of* K*.*

Proof. First we show that $K/^S_C L$ is a sublanguage of K that is controllable with respect to L and A_{uc}. It is clear from the definition of $K/^S_C L$ that $(K/^S_C L) \subseteq K$ in the sense of Remark 6.1. Indeed, we take $U = (K/^S_C L)_w = K_w/^S_C L_w$ and $V = K_w$ for some $w \in (K/^S_C L)^2$ and then $U \xrightarrow{a} \Rightarrow\ V \xrightarrow{a}$. Let us show that $K/^S_C L$ is controllable with respect to L and A_{uc}. It is sufficient to prove that the following relation is a control relation (definition 4.3).

$$C = \{\langle (K/^S_C L)_w, L_w\rangle \mid w \in (K/^S_C L)^2\}.$$

Take a pair $M = (K/^S_C L)_s$ and $N = L_s$ for some $s \in (K/^S_C L)^2$.
(i) Let $(K/^S_C L)_s \xrightarrow{a}$ and $L_s \xrightarrow{a}$ for $a \in A$. Then by coinductive definition of $K/^S_C L$ we have $(K/^S_C L)_{sa} = (K_{sa}/^S_C L_{sa})$, which by definition of C means that $\langle (K/^S_C L)_{sa}, L_{sa}\rangle \in C$.
(ii) Let $L_s \xrightarrow{u}$ for $u \in A_{uc}$. Since $(K/^S_C L) \xrightarrow{s}$, we have by definition 6.4 that $K \xrightarrow{s}$ and $L \xrightarrow{s}$
and $\forall u \in A^*_{uc} : \; L_s \xrightarrow{u} \Rightarrow \; K_s \xrightarrow{u}$. Therefore we deduce $K_s \xrightarrow{u}$. Furthermore, $\forall v \in A^*_{uc}$: $L_{su} \xrightarrow{v} \Rightarrow L_s \xrightarrow{uv} \Rightarrow \; K_s \xrightarrow{uv} \Rightarrow \; K_{su} \xrightarrow{v}$, because $uv \in A^*_{uc}$ and $(K/^S_C L) \xrightarrow{s}$. Hence $(K/^S_C L)_s \xrightarrow{u}$, which proves that C is a control relation, i.e. $K/^S_C L$ is controllable with respect to L and A_{uc}.

It remains to show the supremality. Let $M \subseteq K$ be controllable with respect to L and A_{uc}. In order to show that $M^2 \subseteq (K/^S_C L)^2$, we consider

$$R = \{\langle M_w, (K/^S_C L)_w\rangle \mid w \in M^2\}.$$

Take $\langle M_s, (K/^S_C L)_s\rangle \in R$ for some $s \in M^2$. Let $M_s \xrightarrow{a}$ for $a \in A$. Then $K_s \xrightarrow{a}$, and $L_s \xrightarrow{a}$, since $M \subseteq K \subseteq L$. In order to prove that $(K/^S_C L)_s \xrightarrow{a}$, it remains to show that $\forall u \in A^*_{uc} : \; L_{sa} \xrightarrow{u} \Rightarrow \; K_{sa} \xrightarrow{u}$. But this is straightforward: if $L_{sa} \xrightarrow{u}$, then by controllability of M we deduce $M_{sa} \xrightarrow{u}$, thus from $M \subseteq K$ it follows that $K_{sa} \xrightarrow{u}$. It follows that R satisfies (ii) of simulation relations, i.e. $M \subseteq K/^S_C L$. □

In the case of partial observations, we now separate the issue of controllability from observability and introduce the following modification of the supervised product.

Definition 6.6 Define the following binary operation on (partial) languages for all $M \in \mathrm{Pwr}(\mathrm{suffix}(K))$ and $N \in \mathrm{Pwr}(\mathrm{suffix}(L))$ and $\forall a \in A$:

$$(M/^O N)_a = \begin{cases} M_a/^O N_a & \text{if } M \xrightarrow{a} \text{ and } N \xrightarrow{a} \\ \cup_{\{M' : \langle M', M\rangle \in Aux(K)\}} M'_a/^O N_a & \text{if } M \not\xrightarrow{a} \text{ and } \exists M' \in DK : \\ & M' \approx^K_{Aux} M \text{ such that } M' \xrightarrow{a} \\ & \text{and } N \xrightarrow{a} \\ \emptyset & \text{otherwise} \end{cases}$$

and $(M/^O N)\downarrow$ iff $N\downarrow$.

The new operation has the following pleasant property:

Theorem 6.7 $(K/^O L) = \inf\{M \supseteq K : \; M \textit{ is observable with respect to } L \textit{ and } P\}$. *The infimal observable superlanguage of K equals* $(K/^O L)$.

Proof. We show that $K/^O L$ is an observable partial language containing K that is smaller then any other observable superlanguage of K.

Let us show that $K/^{O}L$ is a superlanguage of K that is observable with respect to L. It is clear from the definition 6.6 that $(K/^{O}L)^2$ is a superlanguage of K^2. Formally it can be checked by constructing an obvious simulation relation. Let us show that $K/^{O}L$ is observable with respect to L. For this purpose denote

$$O = \{\langle (K/^{O}L)_u, L_u\rangle \mid u \in (K/^{O}L)^2 \}.$$

We show that O is an observability relation on $D(K/^{O}L) \times DL$. Take a pair $\langle U, V\rangle \in R$. We can assume that $U = (K/^{O}L)_s$ and $V = L_s$ for some $s \in (K/^{O}L)^2$.
(i) Let $a \in A$ such that $(K/^{O}L)_s \xrightarrow{a}$. It follows from the definition 6.6 that $L_s \xrightarrow{a}$ and from the definition of O that $\langle (K/^{O}L)_{sa}, L_{sa}\rangle \in O$.
(ii) Let $a \in A$ be such that $L_s \xrightarrow{a}$ and there exist $M \in D(K/^{O}L)$ and $N \in DL$: $(M, N) \approx^{K/^{O}L,L}_{Aux} ((K/^{O}L)_s, L_s)$ with $M \xrightarrow{a}$. Using Lemma 5.2 there exist $w, w' \in A^*$ such that $P(w) = P(w')$,$(K/^{O}L)_s = (K/^{O}L)_w$, $L_s = L_w$, and $M = (K/^{O}L)_{w'} \xrightarrow{a}$. According to definition 6.6 inductively applied there exist $s_i \in A^*,\ i \in I$ for some index set I such that $(K/^{O}L)_{w'} = (\cup_{i\in I} K_{s_i})/^{O}L_{w'}$, where $P(s_i) = P(w')\ \forall i \in I$. Notice, that it can be that $I = \{1\}$ and $s_1 = w'$. From $L_s = L_w$ we have $L_w \xrightarrow{a}$. Since $M \xrightarrow{a}$, by definition 6.6 either there exist $s_j,\ j \in J \subseteq I$ such that $K_{s_j} \xrightarrow{a}$ for $j \in J$ and $M_a = (\cup_{j\in J} K_{s_j a})/^{O}L_{w'a}$, or there exist $w_k,\ k \in N$ for some index set N such that $K \xrightarrow{w_k a}$, $P(w_k) = P(w')$ and $M_a = (\cup_{k\in K} K_{w_k a})/^{O}L_{w'a}$. Since also $P(w_k) = P(w)$ for all $k \in K$, we deduce finally that according to definition 6.6 in both cases there must be $(K/^{O}L)_s = (K/^{O}L)_w \xrightarrow{a}$, which proves that O is an observability relation.

The last step of the proof is to show that if $M \supseteq K$ is a language which is observable with respect to L and P, then $(K/^{O}L) \subseteq M$. It is sufficient to prove that

$$R = \{\langle (K/^{O}L)_u, M_u\rangle \mid u \in (K/^{O}L)^2 \text{ and } K \subseteq M \approx_{O(M,L)} L\}$$

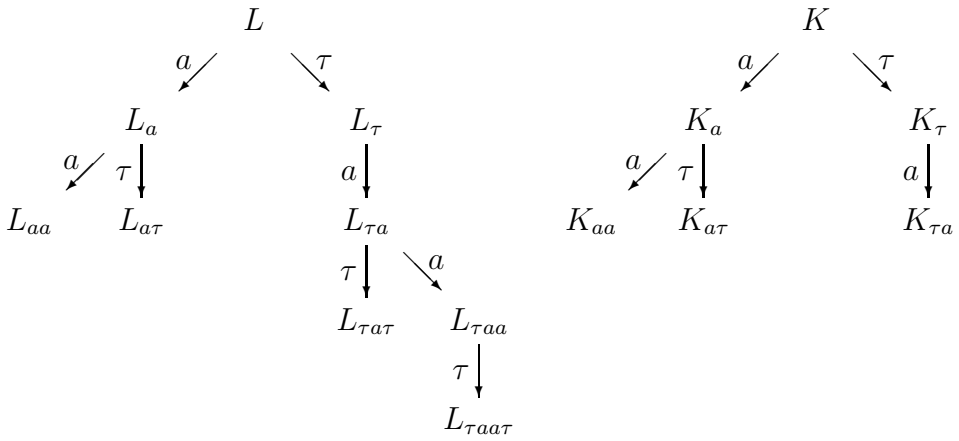
satisfies (ii) of simulation relation.

Take a pair $\langle U, V\rangle \in R$. We can assume that $U = (K/^{O}L)_w$ and $V = M_w$ for some $w \in (K/^{O}L)^2$. Let $U \xrightarrow{a}$. There exist $s_i \in K^2$ for i in some index set I such that $P(s_i) = P(w)\ \forall i \in I$ and $U = (\cup_{i\in I} K_{s_i})/^{O}L_w$. Now, $U \xrightarrow{a}$ implies that either there exists $J \subseteq I$ such that $K_{s_j} \xrightarrow{a}$ for $j \in J$ and $U_a = (\cup_{j\in J} K_{s_j a})/^{O}L_{wa}$ or there exist $w_k \in A^*,\ k \in N$ for some index set N such that $P(w_k) = P(w)$ and $U_a = (\cup_{k\in N} K_{w_k a})/^{O}L_{wa}$. In the first case we have $w \in M^2$ (because $V = M_w$), $s_j \in M^2$, because $s_j \in K^2 \subseteq M^2$, $s_j a \in M^2$, $wa \in L^2$ and $P(s_j) = P(w)$. Therefore $wa \in M^2$, because M is observable with respect to L and P. In the second case we have similarly $w \in M^2$, $w_k \in M^2$, $w_k a \in M^2$, $wa \in L^2$ and $P(w_k) = P(w)$, which gives also $wa \in M^2$. Hence $V = M_w \xrightarrow{a}$, and trivially $\langle U_a, V_a\rangle \in R$, which was to be shown. □

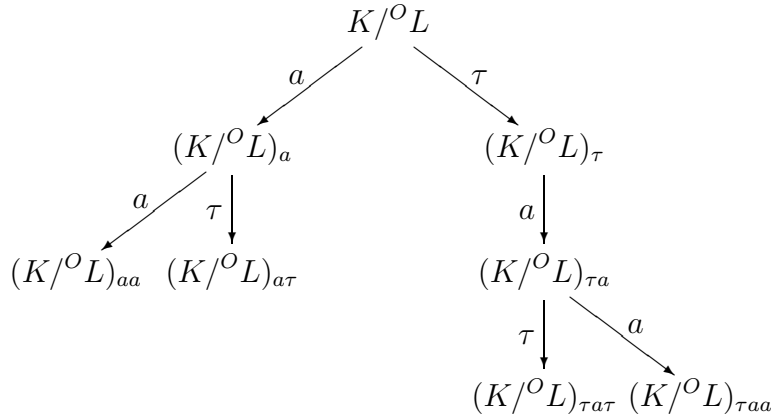
As an illustration of the new operation, let us consider the following ex-

ample.

Example 6.8 We consider prefix-closed languages K^2 and L^2 given by the following tree automata, different from $\langle K\rangle$, resp. $\langle L\rangle$ from $\mathcal{L}$! The marked components are not considered, $A = \{a, \tau\}$, and $A_o = \{a\}$.



Then



We have for instance $(K/^OL)_{\tau a\tau} = (K_{\tau a}/^OL_{\tau a})_\tau = K_{a\tau}/^OL_{\tau a\tau}$ according to the definition 6.6, because $K_{\tau a} \stackrel{\tau}{\not\to}$, but there exists $K_a \approx^K_{Aux} K_{\tau a}$ with $K_a \stackrel{\tau}{\to} K_{a\tau}$. Also, $K/^OL$ is indeed the infimal observable superlanguage of K as stated in theorem 6.7.

As for the original definition of supervised product it can be shown in a similar way that

Theorem 6.9 $(K/^O_UL) = \inf\{M \supseteq K :\ M$ *is controllable with respect to* L *and* A_{uc} *and observable with respect to* L *and* $P\}$. *Thus,* $(K/^O_UL)$ *equals the infimal controllable and observable superlanguage of* K.

The proof is somewhat more complicated because of the technicalities related to considering different cases separately and hence is omitted.

Note that the infimality of the above defined operations is in both cases only with respect to the second (closed) components of the partial languages

involved. The following example shows that the infimality with respect to the marking component cannot hold.

Example 6.10 Take $K = (\{a\}, \{\varepsilon, a, \tau, \tau a, \tau ab\})$,
$L = (\{a, ab\}, \{\varepsilon, a, ab, ab\tau, \tau, \tau a, \tau ab\})$, $A_{uo} = \{\tau\}$, and
$M = (\{a, \tau\}, \{\varepsilon, a, ab, \tau, \tau a, \tau ab\})$. Then $K/^O L = (\{a, ab\}, \{\varepsilon, a, ab, \tau, \tau a, \tau ab\})$. Hence $M \supseteq K$, M is observable with respect to L and P, but $(K/^O L)^1 \not\subseteq M^1$, because $ab \in (K/^O L)^1 \setminus M^1$.

Similar examples can be constructed for $K/_U L$ or $K/^O_U L$.

7 Conclusion

Supervisory control of DES with partial observations has been studied by coalgebraic techniques. Finality of the automaton of partial languages can be used for coinductive proofs as well as for coinductive definitions. We have for instance described the control policy under partial observations by defining the closed-loop languages as corresponding supervised products. A minor modification of the coinductive definition of supervised product leads to the infimal closed observable superlanguage. Future research might include a study of the computational complexity of coalgebraic algorithms or a study of timed DES using coalgebra.

Acknowledgement

The author is grateful to Jan H. van Schuppen for his numerous comments, suggestions, and other help.

References

[1] Brandt,R.D., V. Garg, R. Kumar, F. Lin, S.I. Marcus, W.M. Wonham, *Formulas for Calculating Supremal Controllable and Normal Sublanguages*, Systems & Control Letters 15: 111-117, 1990.

[2] Cieslak R., C. Desclaux, A. Fawaz, and P. Varaiya, *Supervisory Control of a Class of Discrete Event Processes*, IEEE Trans. Automatic Control, 33:249-260, 1988.

[3] Cassandras S.G. and S. Lafortune, Introduction to Discrete Event Systems, Kluwer Academic Publishers, Dordrecht 1999.

[4] Cho M. and S.I. Marcus, *On Supremal Languages of Classes of Sublanguages that Arise in Supervisor Synthesis Problems with Partial Observations*, Mathematics of Control, Signal, and Systems, 2:47-69, 1989.

[5] Cho M. and S.I. Marcus, *Supremal and Maximal Sublanguages Arising in Supervisor Synthesis Problems with Partial Observations*, Math. Systems Theory, 22:171-211, 1989.

[6] Komenda J., *Computation of Supermal Sublanguages of Supervisory Control Using Coalgebra*, Proceedings WODES'02, Workshop on Discrete-Event Systems, Zaragoza, p. 26-33, October 2-4, 2002.

[7] Komenda J., *Coalgebra and Supervisory Control of Discrete-Event Systems with Partial Observations*, Proceedings of MTNS 2002, Notre Dame (IN), August 2002.

[8] Komenda J., Coalgebra and Coinduction in Discrete-Event Control with Partial Observations, Submitted for publication.

[9] Lafortune S. and E. Chen, *The Infimal Closed and Controllable Superlanguage and its Applications in Supervisory Control*, IEEE Trans. on Automatic Control, Vol. 35,N 4, p. 398-405, 1990.

[10] Lin F. and W.M. Wonham, *On Observability of Discrete-Event Systems*, Information Sciences,44: 173-198, 1988.

[11] Rutten J.J.M.M., Automata and Coinduction (an Exercise in Coalgebra), Research Report CWI, SEN-R9803, Amsterdam, May 1998. Avaiable also at `http://www.cwi.nl/~janr`.

[12] Rutten J.J.M.M., Coalgebra, Concurrency, and Control, Research Report CWI, SEN-R9921, Amsterdam, November 1999. Avaiable also at `http://www.cwi.nl/~janr`.

[13] Rutten J.J.M.M. , *Universal Coalgebra: A Theory of Systems*, Theoretical Computer Science 249(1):3-80, 2000.

[14] Ramadge P.J. and W.M. Wonham, *The Control of Discrete-Event Systems*, Proc. IEEE, 77:81-98, 1989.

[15] Rudie K. and W.M. Wonham, *The Infimal Prefix-Closed and Observable Superlanguage of a Given Language*, Systems & Control Letters 15 (1990), 361-371.

[16] Wonham W.M. and P.J. Ramadge, *On the Supremal Controllable Sublanguage of a Given Language*, SIAM J. Control Optim., 25:637-659, 1987.

[17] Yoo, T.S., S. Lafortune, and F.Lin, A Uniform Approach for Computing Supremal Sublanguages Arising in Supervisory Control theory, Preprint, Dept. of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor 2001.

[18] Yoo, T.S. and S. Lafortune, *General Architecture for Decentralized Supervisory Contol of Discrete-Event Systems*, Discrete Event Dynamic Systems: Theory and Applications, 12, 335-377, 2002.