Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

Decentralized supervisory control of discrete-event systems with communication

K.C. Wong and J.H. van Schuppen

Department of Operations Reasearch, Statistics, and System Theory

BS-R9606 1996

Report BS-R9606
ISSN 0924-0659

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

Decentralized Supervisory Control of Discrete-Event Systems with Communication

K.C. Wong and J.H. van Schuppen

CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
{kcwong,J.H.van.Schuppen}@cwi.nl

Abstract
This paper formulates a problem of communication channels in decentralized control. A necessary and sufficient condition is obtained for the existence of such a channel and is illustrated with an example of communication protocols. The existence of minimal communication channels is also investigated.

AMS Subject Classification (1991): 93B50, 93C30, 68Q60, 68Q68.
Keywords and Phrases: Discrete event system, decentralized control, communication.
Note: Paper is included in the Proceedings of the Workshop on Discrete Event Systems (WODES96), which workshop was held August 19-21, 1996 at Edinburgh, Scotland.

1 Introduction

This paper formulates and solves a problem of existence of communication channels between supervisors in decentralized control of discrete event systems (DES).

In decentralized supervisory control of discrete event systems, two or more supervisors with different partial observations control a system. Existence of decentralized supervisors for a given specification has been studied in [2, 6].

In the design of decentralized control systems, communication channels between supervisors can often be incorporated. An example of such a channel is the common-channel signaling used in communication networks, see [7, Section 12-2].

For a given control task and an observation channel, a question is whether there is sufficient information obtained through the channel for a supervisor to synthesize the given control task. The notion of observable languages [3] addresses this question. In the situation where the channel does not provide sufficient information but there is another supervisor observing the same system through another channel, the problem might be overcome by communication from the other supervisor. A problem of existence of such a communication channel for resolving the ambiguity due to the insufficiency of the first channel is formulated. A necessary and sufficient condition is presented. As communication always comes at a cost, existence of 'minimal' communication channels is also investigated.

The paper is organized as follows. Section 2 contains the problem formulation. A necessary and sufficient condition is presented in Section 3 and is illustrated by an example in Section 4. In Section 5, the existence of minimal channels is studied. Conclusions are presented in Section 6.

2 Problem formulation

In this section we formulate a problem of decentralized supervisory control with communication. First we recall the basic notations and definitions of supervisory control of discrete-event systems. For a more detailed introduction to the approach the reader is referred to [4].

In [4], a DES is modelled by an automaton $G = (Q, \Sigma, \delta, q_0, Q_m)$, where Q is a nonempty state set, Σ is a set of event symbols, $\delta : Q \times \Sigma \longrightarrow Q$ is a (partial) state transition function, q_0 is the initial state, and $Q_m \subseteq Q$ is a set of distinguished or 'marked' states. In this approach G is interpreted as a device that starts in the state q_0 and generates a sequence of events, and is referred to as a *process* or a *generator*.

The 'external' behaviors of a process are modelled by sets of finite sequences of events generated by G. Formally they are languages. Let Σ^* be the set of all finite strings of event symbols Σ, including the empty string $\epsilon \notin \Sigma$. A *language* is a subset of Σ^*. Extend the transition δ to $\delta : Q \times \Sigma^* \longrightarrow Q$ in the standard fashion. The *closed language* and *marked language generated* by G are respectively

$$L(G) = \{s \in \Sigma^* \mid \delta(q_0, s) \text{ is defined } \},$$

$$L_m(G) = \{s \in L(G) \mid \delta(q_0, s) \in Q_m\}.$$

For $H \subseteq \Sigma^*$, the *prefix closure* of H is given by

$$\overline{H} = \{s \in \Sigma^* \mid su \in H \text{ for some } u \in \Sigma^*\}.$$

The language H is *prefix-closed* if $\overline{H} = H$. The language $L(G)$ is prefix-closed.

Control is introduced to the process G by assuming a subset of events can be disabled (by external supervisor(s)) and thus be prevented from occurring. These events $\Sigma_c \subseteq \Sigma$ are *controllable* events. The remaining events $\Sigma_u = \Sigma - \Sigma_c$ are *uncontrollable* events.

The set $\Gamma = \{\gamma \subseteq \Sigma \mid \Sigma_u \subseteq \gamma\}$ is called the set of *control patterns* for G. A *supervisory control* for G is any map $v : L(G) \longrightarrow \Gamma$. Thus a supervisory control is a map that specifies the set of events to be enabled for each strings in the closed behavior of G. A supervisory control can be concretely represented by a *supervisor* $S = (T, \varphi)$, where $T = (X, \Sigma, \xi, x_0)$ is an automaton and $\varphi : X \longrightarrow \Gamma$. A supervisor starts at the initial state x_0 and executes state transition $\xi : X \times \Sigma \longrightarrow X$ in response to events $\sigma \in \Sigma$ generated by G. At each state the map φ selects a subset of events to be enabled and thus exerts some control over the future behavior of G. The closed behavior of G under the supervision of a supervisory control v, denoted $L(v/G)$, is defined as follows:

$$\begin{array}{ll} (i) & \epsilon \in L(v/G), \\ (ii) & s \in L(v/G),\ \sigma \in v(s),\ \&\ s\sigma \in L(G) \Longleftrightarrow s\sigma \in L(v/G). \end{array}$$

The family of sublanguages of $L(G)$ that can be realized with a supervisory control as above can be described with the notion of controllable languages. A language $K \subseteq \Sigma^*$ is *controllable* with respect to $L(G)$ if $\overline{K}\Sigma_u \cap L(G) \subseteq \overline{K}$.

For a given plant G and a generator E with its behavior representing the desired closed-loop behavior, the basic control problem is to find a supervisory control v so that the closed-loop behavior of G and v is a subset of the specification language generated by E.

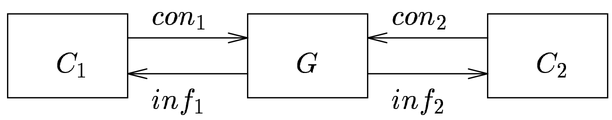


Figure 1: Decentralized supervisory control.

Decentralized control has been studied in the framework of [4]. The 'architecture' is displayed in Figure 1, where the C_i's are local supervisors, each observing through an information channel inf_i and issuing control signals through con_i. For an example of decentralized control, the reader can glance ahead to Section 4 where a communication protocol is discussed. In this example, C_1 might be a supervisor at the site of the sender and C_2 a supervisor at the receiver site. The plant G consists of the sender, the channel, and the receiver.

In [6], necessary and sufficient conditions are obtained for the existence of two local supervisors such that their concurrent supervision implements a solution to a single, global specification. In [6], the channel inf_i is modelled by projections. Let $\Sigma_o \subseteq \Sigma$. A *projection* is a map $p : \Sigma^* \longrightarrow \Sigma_o^*$ defined inductively according to

$$\begin{array}{ll} p(\epsilon) = \epsilon, & \\ p(\sigma) = \epsilon, & \text{if } \sigma \in \Sigma - \Sigma_o, \\ p(\sigma) = \sigma, & \text{if } \sigma \in \Sigma_o, \\ p(s\sigma) = p(s)p(\sigma), & \text{for } s \in \Sigma^*, \sigma \in \Sigma. \end{array}$$

Thus a projection erases events in $\Sigma - \Sigma_o$ along a string. Control decision of a local supervisor is made based on the observed strings $p(L(G))$, i.e., $v' : p(L(G)) \longrightarrow \Gamma$. In the example in Section 4, the local supervisor at the sender observes only events of the sender. Thus $v' \circ p : L(G) \longrightarrow \Gamma$ is a supervisory control for G. Conversely we can say that a supervisory control $v : L(G) \longrightarrow \Gamma$ can be *implemented through* p if there is a map $v' : p(L(G)) \longrightarrow \Gamma$ such that $v = v' \circ p$. This situation can be displayed in Figure 2. The

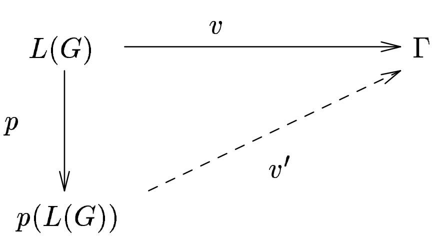


Figure 2: Supervisory control v implemented through p.

factorization of v through p in Figure 2 is possible precisely when $ker\, p \leq ker\, v$, where $ker\, p$ denotes the equivalence kernel of the map p. For a function $f : X \longrightarrow Y$, $ker\, f$ is an equivalence relation on X such that for all $x, x' \in X$, $(x, x') \in ker\, f$ iff $f(x) = f(x')$. Thus the factorization in Figure 2 is possible if p is 'detailed' enough to distinguish all different 'control situations' of v. The notion of observable languages in [3] provides a necessary and sufficient condition for the existence of a supervisory control that synthesizes a given language and can be implemented through a given projection map.

In the protocol example of Section 4, it is found in [5] that the channels p_1 and p_2 are not detailed enough to implement a decentralized solution. In particular, p_1 is not detailed enough for the subtask of the sender. In case where strings that are indistinguishable by p_1 can be distinguish by p_2, communication can be made from the receiver to the sender so that control decision can be made. This is indeed the case for the example in Section 4; with additional communication from the receiver to the sender a correct protocol (stop and wait protocol) can be constructed (see [1, Section 2.4.1]).

The above discussion motives an extension of the decentralized control in Figure 1 and can be displayed as in Figure 3, where inf_{ij} models a communication channel from

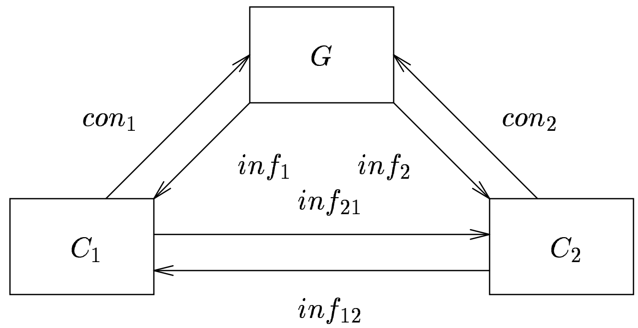


Figure 3: Decentralized supervisory control with communication.

supervisor j to supervisor i. Thus supervisor i bases its decision on information obtained through inf_i and through communication from supervisor j, $inf_{ij} \circ inf_j$. In the case where it is not possible to achieve decentralized control due to the insufficiency of channels inf_1 and inf_2, the problem might be overcome by allowing communication as described. Thus the question is that, for given fixed channels inf_1 and inf_2 and a language to be synthesized, when is it possible to achieve decentralized control with communication as displayed in Figure 3 for the given language? In this article we consider a formulation of a simpler, asymmetric version of this problem.

Since information communicated might be a summary of the information obtained by the other local supervisor, we model the information channels with the more general prefix-preserving or causal maps of [8]. Let T be another event set. A map $\theta : L(G) \longrightarrow T^*$ is *prefix-preserving* or *causal* [8] if

$$\theta(\epsilon) = \epsilon,$$
$$\theta(s\sigma) = \begin{cases} \theta(s) & \text{or,} \\ \theta(s)\tau & \text{for some } \tau \in T, \end{cases}$$

for $s \in L(G)$, $\sigma \in \Sigma$, and $s\sigma \in L(G)$. Clearly projections are causal.

We also need the following notation: Let $f_i : X \longrightarrow Y_i$ be a map, for $i = 1, 2$. Then $f_1 \times f_2 : X \longrightarrow Y_1 \times Y_2 : x \longmapsto (f_1(x), f_2(x))$. We are interested in the following problem:

Problem 1 *Let G be a generator and K be a nonempty, prefix-closed, and controllable sublanguage of $L(G)$. Let $\theta_i : \Sigma^* \longrightarrow T_i^*$, where $i = 1, 2$, be causal maps. Then find a causal map $\theta : T_2^* \longrightarrow T_3^*$, where T_3 is another event alphabet, if possible, such that there is a supervisory control v for G with $L(v/G) = K$ and $ker((\theta_1 \times (\theta \circ \theta_2))|_{L(G)}) \leq ker\, v$.*

Intuitively we can think of θ_1 and θ_2 as models of fixed observation channels inf_1 and inf_2 in Figure 3. Suppose that we are to implement a supervisory control for a prefix-closed language $K \subseteq L(G)$ based on the observation through the θ_1 channel. In the case that θ_1 is not sufficient, information about the plant G obtained through another channel θ_2 could be communicated to the agent observing through channel θ_1 for, if possible, resolving the ambiguity due to the insufficiency of channel θ_1. In $\theta_1 \times (\theta \circ \theta_2)$, θ_1 represents the information obtained through observation and $\theta \circ \theta_2$ the information received via channel θ from an agent observing channel θ_2.

3 Supervisory control with observation and communication

We consider the solvability of Problem 1.

First we recall the notion of observable languages in [3], stated here in a slight more general form. Let G be a generator and $K \subseteq \Sigma^*$. Define a binary relation on Σ^* as follows: $(s, s') \in act_K$ iff

- *i)* $(\forall \sigma \in \Sigma)\, s\sigma \in \overline{K}, s' \in \overline{K}, s'\sigma \in L(G) \Longrightarrow s'\sigma \in \overline{K}$;
- *ii)* $s \in K \cap L_m(G), s' \in \overline{K} \cap L_m(G) \Longrightarrow s' \in K$;
- *iii)* conditions (i) and (ii) hold with s and s' interchanged.

Let $g : \Sigma^* \longrightarrow Y$ be an (arbitrary) map, where Y is a set. A language $K \subseteq \Sigma^*$ is *observable* (with respect to (G, g)) [3] if $ker\, g \leq act_K$. A characterization is given in Appendix A.

The following theorem provides a necessary and sufficient condition for Problem 1.

Theorem 3.1 *Let G be a generator with its event set Σ partitioned into controllable and uncontrollable events. Let K be a nonempty, controllable, and prefix-closed sublanguage of $L(G)$. Let $\theta_i : \Sigma^* \longrightarrow T_i^*$, where $i = 1, 2$, be causal maps. Then Problem 1 is solvable if and only if $ker\, \theta_1 \wedge ker\, \theta_2 \leq act_K$.* □

Here the equivalence relation $ker\, \theta_1 \wedge ker\, \theta_2$ is the meet of $ker\, \theta_1$ and $ker\, \theta_2$ (for all $s, s' \in \Sigma^*$, $(s, s') \in ker\, \theta_1 \wedge ker\, \theta_2$ iff $(s, s') \in ker\, \theta_1$ and $(s, s') \in ker\, \theta_2$). Thus $ker\, \theta_1 \wedge ker\, \theta_2$ can be interpreted to represent the 'information obtained when θ_1 and θ_2 are used in cooperation'. Thus Problem 1 is solvable if and only if the combined 'resolution' of θ_1 and θ_2 is fine enough to distinguish all different control situations for the language K in the plant G.

If ker $\theta_1 \leq act_K$, then Problem 1 is certainly solvable as $ker\, \theta_1 \wedge ker\, \theta_2 \leq ker\, \theta_1 \leq act_K$. Furthermore, we reduce back to a situation considered in [3]: For a nonempty, closed, and controllable sublanguage K of $L(G)$, there exists a supervisory control v for G such that L(v/G) = K and $ker\, (\theta_1|_{L(G)}) \leq ker\, v$ if and only if $ker\, \theta_1 \leq act_K$. In this case, to solve Problem 1 one could simply take $\theta : T_2^* \longrightarrow \{\epsilon\}$, i.e., no communication is necessary.

4 Example

In this section, we illustrate the use of Theorem 3.1 with an example of the alternating bit protocol, borrowed from [5]. As shown in [5] the protocol formulated in that paper is incorrect. Here we show that with more detailed communication between the sender and the receiver a correct protocol can be constructed.

The problem is the reliable transmission of data over an unreliable channel. Data frames may be either damaged or lost in the transmission. It is assumed that there are mechanisms for detecting damaged data frames and for transmitting damaged or lost frames. The models of the sender and the receiver in [5] and the models of the channel with capacity of 1, consisting of the forward path from the sender to the receiver ($Channel_{sr}$) and the backward path from the receiver to the sender ($Channel_{rs}$),[1] are presented in Figures 4, 5, and 6. In the Sender, $getframe$ indicates the sender gets a new frame of data; the event $send_i$ $(i = 0, 1)$ sends the most recent frame to the receiver with a sequence number i; $timer$ starts a timer; $rcvack$ means that an acknowledgement is received; $timeout$ indicates that the timer has timed out; $cksumerrack$ means that an acknowledgement has arrived

[1] This is a slight modification of the channel given in [5].

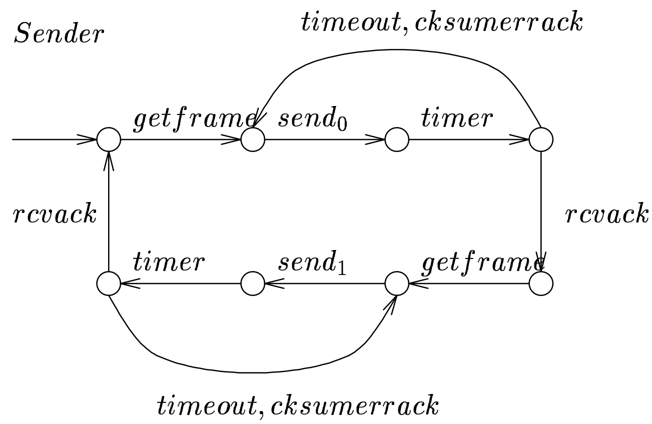


Figure 4: Model of the sender

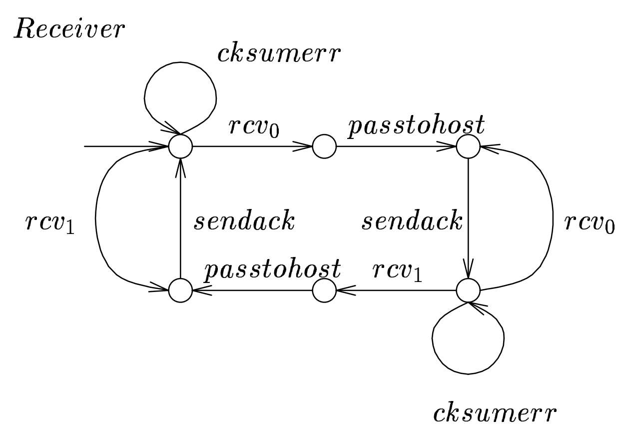


Figure 5: Model of the receiver

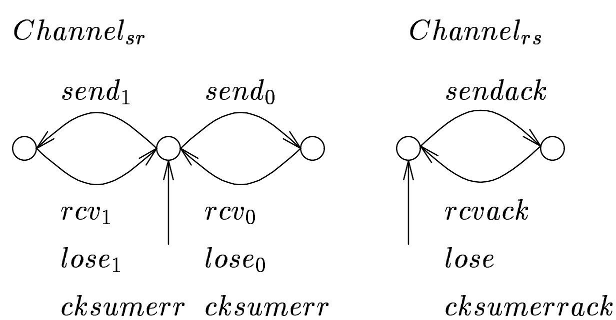


Figure 6: Model of the channel.

damaged. In the Receiver, the event rcv_i means that the receiver receives a frame with i as its sequence number; *passtohost* indicates that a frame is passed from the receiver to a host; *sendack* sends an acknowledgement; *cksumerr* means that a frame has arrived damaged. In the channel, the events *lose* and $lose_i$, where $i = 0, 1$, indicate that a frame is lost in the channel. The set of all possible events is

$$\begin{aligned}\Sigma &= \{getframe, send_0, send_1, rcv_0, rcv_1, \\ &\quad sendack, rcvack, passtohost, timer, timeout, \\ &\quad cksumerr, cksumerrack, lose, lose_0, lose_1\}.\end{aligned}$$

The plant G is given as the synchronous product of *Sender*, *Receiver*, $Channel_{sr}$, and $Channel_{rs}$.

It is assumed in [5] that each agent observes and controls events at its physical location. The observable and controllable sets of events of the sender are respectively

$$\begin{aligned}\Sigma_{1,o} &= \{getframe, send_0, send_1, timer, timeout, \\ &\quad rcvack, cksumerrack\},\end{aligned}$$

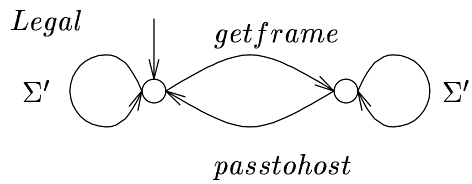
$$\Sigma_{1,c} = \{getframe, send_0, send_1, timer\}.$$

The observable and controllable sets of events of the receiver are respectively

$$\Sigma_{2,o} = \{rcv_0, rcv_1, passtohost, sendack, cksumerr\},$$

$$\Sigma_{2,c} = \{passtohost, sendack\}.$$

The safety requirement of the alternating bit protocol may be described as in [5]: $E = L(G) \cap L(Legal)$, where *Legal* is given in Figure 7. The legal language E can be expressed



where $\Sigma' := \Sigma - \{getframe, passtohost\}$.

Figure 7: Legal language.

as an intersection of E_1 and E_2 with $E_i := L(G) \cap L(Legal_i)$, where the $Legal_i$'s are displayed in Figure 8. The language $L(Legal_i)$ is controllable with respect to $L(G)$ and $\Sigma_{i,c}$, namely $L(Legal_i)\Sigma_{i,u} \cap L(G) \subseteq L(Legal_i)$. It follows that E_i is controllable with respect to $L(G)$ and $\Sigma_{i,c}$. The language E_1 represents the control subtask to be synthesized by the supervisor at the sender and E_2 the control subtask for the supervisor at the receiver.

We now focus on E_1. Let p_i be the projection from Σ^* to $\Sigma_{i,o}^*$. It turns out that E_1 is not observable with respect to p_1. Thus the supervisory controls for E_1 cannot be implemented through p_1. To see this, let t and t', and σ be given as in [5]:

$$\begin{aligned}t &= getframe\ send_0\ timer\ rcv_0\ passtohost \\ &\quad sendack\ timeout\ send_0\ timer\ rcvack \\ &\quad rcv_0\ sendack\ getframe\ send_1\ timer \\ &\quad lose_1\ rcvack \\ t' &= getframe\ send_0\ timer\ rcv_0\ passtohost \\ &\quad sendack\ timeout\ send_0\ timer\ rcvack \\ &\quad rcv_0\ sendack\ getframe\ send_1\ timer \\ &\quad rcv_1\ passtohost\ rcvack \\ \sigma &= getframe\end{aligned}$$

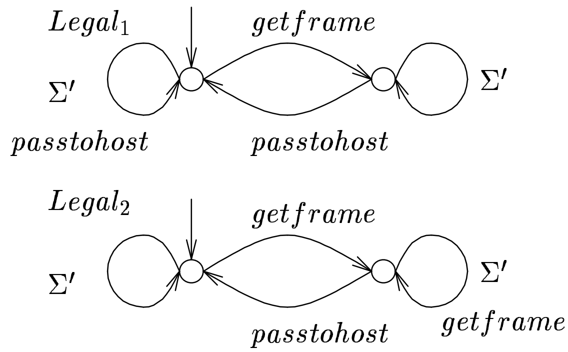


Figure 8: Component legal languages.

Then $p_1(t) = p_1(t')$ but $t \in E_1$, $t\sigma \in L(G)$, $t\sigma \notin E_1$, and $t'\sigma \in E_1$, i.e., $(t, t') \notin act_{E_1}$.

However, it is possible to implement a supervisory control for E_1 at the sender with additional information communicated by the receiver, as we see in the following claim.

Claim 1 $ker\, p_1 \wedge ker\, p_2 \leq act_{E_1}$.

Proof: Let $s, s' \in \Sigma^*$ such that $(s, s') \in ker\, p_1 \wedge ker\, p_2$. Thus $p_1(s) = p_1(s')$ and $p_2(s) = p_2(s')$. We need to show $(s, s') \in act_{E_1}$. Since E_1 is closed, (ii) in the definition of *act* is automatic. To show (i) in the definition of *act*, let $\sigma \in \Sigma$ such that $s\sigma \in E_1$, $s' \in E_1$, and $s'\sigma \in L(G)$. Need to show $s'\sigma \in E_1$. Suppose that $s'\sigma \notin E_1$. Since $s'\sigma \in L(G)$ and $E_1 = L(G) \cap L(Legal_1)$, $s'\sigma \notin L(Legal_1)$. From $Legal_1$ and that $s' \in E_1$, $\sigma = getframe$ and $\delta(q_0, s') = q_1$ where δ is the state transition function of $Legal_1$, q_0 the initial state, and q_1 the other state. Since $s' \in E_1 = L(G) \cap L(Legal_1)$, it can be verified that $|s'|_{getframe} = |s'|_{passtohost} + 1$, where $|s'|_\sigma$ denotes the number of the event σ in s'. Since $s\sigma = s\ getframe \in E_1 \subseteq L(Legal_1)$, $\delta(q_0, s) = q_0$. Since $s \in E_1$, again it can be verified that $|s|_{getframe} = |s|_{passtohost}$. Since $p_1(s) = p_1(s')$, $|s|_{getframe} = |s'|_{getframe}$. Thus $|s|_{passtohost} = |s'|_{passtohost} + 1$. This implies $p_2(s) \neq p_2(s')$, a contradiction. Hence $s'\sigma \in E_1$. It is similar to show (i) with s and s' interchanged. □

From the proof of Claim 1, it is sufficient to introduce additional information in the acknowledgement so that for all $s, s' \in E_1$ such that $s\ getframe, s'\ getframe \in L(G)$ and $p_1(s) = p_1(s')$ imply $|s|_{passtohost} = |s'|_{passtohost}$. This could be done by attaching the identification number of the message to which the receiver is acknowledging, i.e., changing *sendack* to $sendack_i$ $(i = 0, 1)$ and *rcvack* to $rcvack_i$. To see this, we note that $p_1(s)$ and $p_1(s')$ must end with $rcvack_0$ or $rcvack_1$. Initially $p_1(s) = p_1(s') = getframe\ u\ rcvack_0$, where $u \in \{send_0, timer, timeout, cksumerrack,\ rcvack_1\}^*$. Since $rcvack_0$ must follow $sendack_0$ which in turn must follow *passtohost*, both s and s' must contain one *passtohost*. The same agurment applies for longer strings by induction. The problematic strings t and t' would now end with $rcvack_0$. Since the most recent frame sent has the sequence number of 1, a new frame would not be obtained until an $rcvack_1$ is received.

5 Minimal channels

When Problem 1 is solvable, we consider if there is a 'minimal' communication map θ that solves the same problem.

Let $\mathcal{E}_{pre}(T_2^*)$ be the set of equivalence kernels of causal maps with T_2^* as their domain. Let $f : X \longrightarrow Y$ be a function and π be an equivalence relation on Y. Define $\pi \circ f$

as an equivalence relation on X as follows: For $x, x' \in X$, $(x, x') \in \pi \circ f$ if and only if $(f(x), f(x')) \in \pi$. We say that $\pi \in \mathcal{E}_{pre}(T_2^*)$ *solves* Problem 1 if $ker\, \theta_1 \wedge (\pi \circ \theta_2) \leq act_K$. With K, $L(G)$, and θ_i in Problem 1 fixed, a partition π in $\mathcal{E}_{pre}(T_2^*)$ is *maximal* with respect to Problem 1 if π solves Problem 1 and for any $\pi' \in \mathcal{E}_{pre}(T_2^*)$ that solves the same problem and $\pi \leq \pi'$ we have $\pi = \pi'$. Here a maximal π corresponds to a channel that transmits minimal amount of information.

Theorem 5.1 *If Problem 1 is solvable, then there exists at least one maximal π in $\mathcal{E}_{pre}(T_2^*)$ that solves the same problem.* □

Thus if Problem 1 admits a solution, then there is a 'minimal' causal channel that solves the same problem.

Next we consider Problem 1 restricting to a proper subclass of causal maps, namely projections. Let K and $L(G)$ be given as before and let $\Sigma_1, \Sigma_2 \subseteq \Sigma$ and $p_i : \Sigma^* \longrightarrow \Sigma_i^*$ be the projection, for $i = 1, 2$. Thus p_i and Σ_i replace θ_i and T_i respectively. Let

$$\mathcal{E}_P(\Sigma^*) := \{ker\, p \in \mathcal{E}(\Sigma^*) \,|\, p \text{ is the projection from } \Sigma^* \text{ to } (\Sigma')^*, \text{ where } \Sigma' \subseteq \Sigma\}.$$

Corollary 5.2 *For Problem 1 with projections as described above, if Problem 1 is solvable (i.e., $ker\, p_1 \wedge ker\, p_2 \leq act_K$), then there exists at least one maximal π in $\mathcal{E}_P(\Sigma_2^*)$ that solves the same problem.* □

For projections, a 'minimal' projection corresponds to the one with 'minimal' number of events.

The problem of computing a minimal channel for Problem 1 remains to be studied. The following is a related result. First we need the following notion: Let $\Sigma_3 \subseteq \Sigma_2$ and $p : \Sigma_2^* \longrightarrow \Sigma_3^*$ be the projection. Let v be a supervisory control for G such that $L(v/G) = K$. Then p is said to *solve the Problem 1 for v* if $ker\,((p_1 \times (p \circ p_2))|_{L(G)}) \leq ker\, v$.

Proposition 5.3 *Let K, $L(G)$, p_i, and Σ_i be as in the foregoing discussion. Let v be a supervisory control for G such that $L(v/G) = K$. Let $\Sigma_a, \Sigma_b \subseteq \Sigma_2$ and p_a and p_b be the corresponding projections. Suppose that $(\Sigma_1 \cup \Sigma_2)^* \subseteq L(G)$. Then, if p_a and p_b solve Problem 1 for the supervisory control v, so does the projection $p : \Sigma_2^* \longrightarrow (\Sigma_a \cap \Sigma_b)^*$.* □

Thus under the assumptions in Proposition 5.3 if two communication channels solve the problem with respect to the same supervisory control, then a simpler channel with the common events of the previous two channels is sufficient for solving the same problem.

6 Conclusion

We have formulated and solved in principle a problem of existence of communication channels between two decentralized supervisors for implementing a supervisory control. The existence of minimal channels is also investigated.

One possible direction for future work would be to investigate computation of minimal channels. Our study here focuses on a simple, asymmetric version of the general problem of decentralized control with communication. Another direction would be to consider the full, symmetric version of this problem.

A Observable languages

We give a characterization of observable languages of [3], given in Section 3 in a slightly more general form. For a language H over Σ let

$$Elig_H : \Sigma^* \longrightarrow \mathcal{P}(\Sigma) : s \longmapsto \{\sigma \in \Sigma \mid s\sigma \in \overline{H}\},$$

where $\mathcal{P}(X)$ denotes the power set of the set X. For a sublanguage K of $L(G)$, let

$$F_K := \{f : L(G) \longrightarrow \mathcal{P}(\Sigma) \mid (\forall s \in \overline{K})\, Elig_K(s) = Elig_{L(G)}(s) \cap f(s)\}.$$

Proposition A.1 *Let $K \subseteq L_m(G)$. Then K is observable with respect to (G, g) if and only if there exists $f \in F_K$ such that $ker\,(g|_{L(G)}) \leq ker\, f$, and $ker\,(g|_{\overline{K} \cap L_m(G)}) \leq \{K, \overline{K} \cap L_m(G) - K\}$.* □

Let $S_K := \{f : L(G) \longrightarrow \mathcal{P}(\Sigma) \mid (\exists g \in F_K)(\forall s \in L(G)) f(s) = g(s) \cup \Sigma_u\}$.

Proposition A.2 *Let K be a nonempty, controllable sublanguage of $L(G)$. Then S_K is the set of all supervisory control v for G such that $L(v/G) = \overline{K}$.* □

References

[1] D. Bertsekas and R. Gallager. *Data Networks.* Prentice Hall, Englewood Cliffs, New Jersey, 1992.

[2] R. Cieslak, C. Desclaux, A. S. Fawaz, and P. Varaiya. Supervisory control of discrete-event processes with partial observations. *IEEE Transactions on Automatic Control*, 33(3):249–260, March 1988.

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

[4] P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proc. IEEE, Special Issue on Discrete Event Dynamic Systems*, 77(1):81–98, January 1989.

[5] K. Rudie and W. M. Wonham. Protocol verification using discrete-event systems. In *Proc. of the 31st Conference on Decision and Control*, pages 3770–3777, Tucson, Arizona, December 1992.

[6] K. Rudie and W. M. Wonham. Think globally, act locally: decentralized supervisory control. *IEEE Transactions on Automatic Control*, 37(11):1692–1708, 1992.

[7] M. Schwartz. *Telecommunication networks.* Addison-Wesley Publishing Co., Reading, 1987.

[8] H. Zhong and W. M. Wonham. On the consistency of hierarchical supervision in discrete-event systems. *IEEE Transactions on Automatic Control*, 35(10):1125–1134, October 1990.