On the Computation of Observers in Discrete-Event Systems

K. C. Wong and W. M. Wonham
Systems Control Group
Department of Electrical and Computer Engineering
University of Toronto
Toronto, Canada M5S 1A4
kcwong, wonham@control.utoronto.ca

March 8, 2003

Keywords: discrete-event systems, causal reporter maps, natural projections, observers.

Abstract

The concept of observer was introduced in previous work by the authors on a hierarchical control theory of discrete-event systems (DES). It was shown that the observer property of the 'causal reporter' map, which in this theory models information flow in a hierarchical DES, plays a role in ensuring that a nonblocking supervisor at a given level of a hierarchy does not cause blocking in the level below. In this paper we investigate the following problem: *Given a causal reporter map that is not an observer, how can we design an observer by modifying this map?* In case the latter is represented by a finite Mealy automaton, an effective computational algorithm is developed for computing an observer with the coarsest possible equivalence kernel that is finer than that of the given map. Three examples are provided for illustration.

1 Introduction

The concept of observer was introduced in [WW96], where it was shown to play a useful role in the nonblocking hierarchical control theory of [ZW90]. In the two-level hierarchy of [ZW90], the information channel which summarizes low-level behavior and reports to the high-level system is modeled by a *causal reporter* map. An observer is a reporter map with a certain additional property. It was shown in [WW96] that the observer property of the reporter map plays a role in ensuring that the low-level implementation of a nonblocking high-level supervisor is also nonblocking.

To ensure nonblocking hierarchical control, sufficient conditions [WW96] are (1) $\mathcal{C}_{hi}(M) = \theta\mathcal{C}_{lo}(L)$, (2) $\theta^{-1}(M_m) = L_m$, (3) partner-freeness, (4) the observer property for θ. Here L_m and L are the marked and closed behaviours of the low-level system, M_m and M are the marked and closed behaviours of the high-level system, θ is the reporter map, and $\mathcal{C}$ denotes the appropriate family of controllable sublanguages. In the setting of [ZW90], with output control consistency established, the observer and partner-freeness conditions imply condition (1), as shown in [WW98] (Proposition 11). Condition (2) is easy to arrange directly. An algorithm for modifying a hierarchical structure to satisfy the partner-freeness condition was developed in [ZW90]. The remaining question is how to modify the hierarchical structure so that the reporter map is an observer. The present paper addresses this question. With procedures for arranging all the conditions at hand, one can then consider the arrangement for nonblocking hierarchical control, which might involve an iterative application of the procedures until all the conditions are established.

More recently, in [HC02], the topic of nonblocking hierarchical control is investigated in a state (rather than language) framework. To ensure nonblocking hierarchical control, a so-called nonblocking in-block controllable partition automaton is introduced, the definition (Definition III.4 of [HC02]) of which consists of two parts. The first condition of part 1 requires that any high-level disablement at a high-level state can be realized or implemented by a set of low-level disablements. This is akin to the partner-freeness condition of [ZW90]. The second condition of part 1 states essentially that no high-level disablement causes blocking in the low-level, in transitions from the current state to the next state subset. In our setting this is a *consequence* of the partner-freeness and observer conditions, whereas in [HC02], the desired result is *defined* into the structure. The second part of the definition of nonblocking in-block controllable partition automaton states that, when a marker state in the high level is reached, it is guaranteed that the low-level automaton can always find a path to a low-level marker state. In our setting, this is a *consequence* of the observer property with respect to the marked language.

In other works of hierarchical and decentralized control of discrete-event systems, the concept of observer also provides a useful property. It is well known that a 'natural' (zero-memory) projection of a DES can produce a more complex, even exponentially more complex, projected model. In [Won98], it was shown that a natural projection with the observer property always yields a projected model that is simpler than the original DES. In [Pu00], a weaker notion of observer is shown to be equivalent to the existence of a consistent high-level system. Furthermore, it is shown in the same work that a "reliable" abstraction of a low-level system exists if and only if the reporter map is an observer. In [Wnhm02] (Sect. 5.7), a locally output controllable hierarchical structure is shown to be globally output controllable if the reporter map is an observer.

In this paper we investigate the following problem: *Given a causal reporter map that is not an observer, how can we compute an observer by modifying this map?*

The problem of computing an observer from a given causal reporter map has been studied in [Gua97]. However, the procedure there differs from ours and, unlike ours, it does not generally produce an observer with the coarsest possible equivalence kernel. In [Gua97], a finite reachability tree is first constructed and computation is carried out on this tree. An automaton is then constructed from this modified tree to represent the new map. In our approach, the computation is carried out on the structure of the automaton. A comparison of complexity of the two approaches will be given later. In [Pu00], there is a procedure for checking if a reporter map is an observer. However, if the given map is not an observer, there is no algorithm provided in [Pu00] to modify the map to arrange the observer property.

We outline our solution to this problem. First we note that a causal reporter map is a function from one formal language to another. It turns out that the equivalence kernel of a causal reporter map can be characterized as a *quasi-congruence* for a prefix-closure function. The formal definition of a quasi-congruence will be given in Section 2. Our terminology mildly extends that of [Wnhm76]; in computer science a quasi-congruence is called a *bisimulation* [Mil89, Arn94]. An observer is a causal reporter map whose equivalence kernel is a quasi-congruence for a postfix-closure function; the formal definition will be given in Section 2. Thus, observers are characterized by the set of quasi-congruences (for both the prefix and postfix closure functions). It turns out that these form a complete upper semilattice. For a given causal map, the coarsest quasi-congruence (for the prefix and postfix closure functions) that is finer than the equivalence kernel of the map exists. Call this quasi-congruence π^*. A natural solution to our problem is to modify the given map in such a way that the equivalence kernel of the new function becomes π^*.

The computation is carried out on finite Mealy automata, as presented in Section 3. Let $\mathbf{G}$ be a finite Mealy automaton representing a given causal reporter map θ. First we relabel the transitions of $\mathbf{G}$ to form $r(\mathbf{G})$, encoding the output behavior of $\mathbf{G}$ into its transition structure. Then we compute the coarsest quasi-congruence ρ on $r(\mathbf{G})$ using the partition refinement algorithm of [Fer90]. Because $r(\mathbf{G})$ is also finite, ρ can be computed in a finite number of iterations of a monotone operator. A modification of the map is carried out by relabeling the transitions of $r(\mathbf{G})$ in accordance with ρ. The process is then repeated.

As a termination criterion for the iterative computation, a characterization of the observer property is given in Section 4. A causal map is an observer if and only if the quotient automaton $\overline{r(\mathbf{G})}$ constructed from $r(\mathbf{G})$ and ρ is deterministic and contains no 'unobservable' transitions. After computing ρ on $r(\mathbf{G})$ and constructing $\overline{r(\mathbf{G})}$, if $\overline{r(\mathbf{G})}$ is deterministic and free of unobservable transitions, then the given map is an observer and the computation terminates. If not, then relabeling is needed on $r(\mathbf{G})$, as explained in Section 5. From the above characterization, a natural relabeling scheme is to relabel the events so that the quotient becomes deterministic and free of unobservable transitions. However, a relabeling of $r(\mathbf{G})$ might destroy the quasi-congruence property of ρ. Thus, the coarsest quasi-congruence for the new Mealy automaton is again computed, and the process repeated.

It is shown in Section 6 that this iterative computation converges finitely. Furthermore, upon convergence the equivalence kernel of the new causal reporter map is π^*.

It should be emphasized that, although in our procedure we use an algorithm (that of [Fer90]) for computing bisimulations, our objective is not the computation of bisimulations, but the computation of a function whose equivalence kernel is π^*.

The overall complexity of this computation is $O(k^4 n)$, where k is the number of states of $\mathbf{G}$, and n the number of transitions. The complexity of the algorithm in [Gua97] is $O(k^7)$, where k again is the number of states of the automaton representing the reporter map.

Three examples are provided in Section 7 and conclusions stated in Section 8.

The observer computation algorithm was first reported in [WW00].

2 Preliminaries

In this section we recall some basic definitions and concepts to be used later.

2.1 Formal languages and causal reporter maps

Let Σ be a nonempty, finite alphabet of event labels and Σ^+ the set of all finite strings of elements in Σ. Let $\Sigma^* := \Sigma^+ \cup \{\epsilon\}$, where $\epsilon (\notin \Sigma)$ denotes the empty string. A subset $L \subseteq \Sigma^*$ is called a *language* over Σ. A string $s \in L$ is a *word* of L. For $s, t \in \Sigma^*$, s is a *prefix* of t, denoted $s \leq t$, if there is a string u such that $su = t$. The closure of L is the language $\overline{L}$ consisting of all the prefixes of words in L: $\overline{L} := \{s \in \Sigma^* \mid (\exists t \in L)\, s \leq t\}$. A language L is *(prefix) closed* if $L = \overline{L}$.

Let L be a language, not necessarily closed, over Σ and let T be a second event alphabet. A function $\theta : \overline{L} \longrightarrow T^*$ is a *causal reporter* map [ZW90] if it satisfies the following properties:

$$\begin{aligned} &\theta(\epsilon) = \epsilon, \\ &\theta(s\sigma) = \begin{cases} \text{either } \theta(s) \\ \text{or } \theta(s)\tau, \text{ for some } \tau \in T \end{cases} \end{aligned} \tag{1}$$

for $s \in \overline{L}$ and $\sigma \in \Sigma$. A causal reporter map is prefix-preserving, namely if $s \leq s'$ then $\theta(s) \leq \theta(s')$, although the converse need not be true. In the two-level hierarchy of [ZW90], the information channel which summarizes behaviors in the low-level process and reports relevant information to the high-level system is modeled by a causal reporter map.

We will make the following assumption concerning the event labels in Σ and T. Let

$$\Sigma_l := \{\sigma \in \Sigma \,|\, (\exists s \in \Sigma^*)\, s\sigma \in \overline{L} \text{ and } \theta(s\sigma) = \theta(s)\}$$

From (1), we see that the occurrence of an event σ at a string s might or might not generate an event τ in T. The set Σ_l is the set of events which generate no event in T for at least one string in $\overline{L}$. We will assume for this paper that

$$\Sigma_l \cap T = \emptyset \tag{2}$$

The reason for this assumption will be clear later; the assumption is not restrictive, as Σ and T are generally disjoint. Also, a natural projection [LW88] satisfies this assumption: Let $\Sigma_o \subseteq \Sigma$ be the set of *observable* events. The *natural projection* p from Σ^* to Σ_o^* is defined according to:

$$\begin{aligned} p(\epsilon) &= \epsilon, \\ p(s\sigma) &= \begin{cases} p(s) & \text{if } \sigma \notin \Sigma_o \\ p(s)\sigma & \text{if } \sigma \in \Sigma_o \end{cases} \end{aligned}$$

for $s\sigma \in \Sigma^*$. Thus, $\Sigma_l = \Sigma - \Sigma_o$, and our assumption is clearly satisfied, as in this case $T = \Sigma_o$.

2.2 Quasi-congruences

It turns out that the equivalence kernel of a causal reporter map can be characterized as a 'quasi-congruence' for the prefix-closure function.

Before introducing the notion of quasi-congruence, we recall some basic concepts. Let X be a set. We denote by $\mathcal{E}(X)$ the set of all equivalence relations (a binary relation that is reflexive, symmetric and transitive; for a formal definition, see, for instance, [BS81]) on X. For $\omega \in \mathcal{E}(X)$, let $[x]_\omega$ denote the equivalence class of ω containing x, and let X/ω denote the set of equivalence classes of ω, i.e., $X/\omega := \{[x]_\omega \,|\, x \in X\}$. Let $proj_\omega : X \longrightarrow X/\omega$ be the canonical projection associated with ω, mapping an element x in X to its equivalence class $[x]_\omega$. Let $f : X \longrightarrow Y$ be a function. The *equivalence kernel* of f, denoted $ker\, f$, is an equivalence relation on X defined as follows: for $x, x' \in X$, $x \equiv x' (mod\,(ker\, f))$ iff $f(x) = f(x')$.

For a set X, let $\mathcal{P}(X)$ denote the power set of X. For a given function $f : X \longrightarrow Y$, a function $\widehat{f} : \mathcal{P}(X) \longrightarrow \mathcal{P}(Y)$ can be induced as follows: for $X' \in \mathcal{P}(X)$, $\widehat{f}(X') = \{f(x) \,|\, x \in X'\}$. We shall drop the hat on the induced function, as context will determine the intended meaning. The *inverse image* function is defined as follows: $f^{-1} : \mathcal{P}(Y) \longrightarrow \mathcal{P}(X) : Y' \longmapsto \{x \in X \,|\, f(x) \in Y'\}$. We shall also use the notation f^{-1} to denote the *inverse* of f when f is bijective. Again, the meaning will be clear from context.

Let X and Y be sets and $f : X \longrightarrow \mathcal{P}(Y)$ be a function. Let $\rho \in \mathcal{E}(Y)$. Define $\rho \circ f \in \mathcal{E}(X)$ as follows: for $x, x' \in X$,

$$x \equiv x' \,(mod\, \rho \circ f) \Longleftrightarrow proj_\rho(f(x)) = proj_\rho(f(x'))$$

Thus, $\rho \circ f$ is the equivalence kernel of the composition of $proj_\rho$ with f. Note that, if g is a function from X to Y, then $x \equiv x' \,(mod\, \rho \circ g)$ iff $g(x) \equiv g(x') \,(mod\, \rho)$.

For two equivalence relations π and ω on X, π *refines* ω, denoted $\pi \leq \omega$, if for all $x, x' \in X$, $x \equiv x' \,(mod\, \pi)$ implies $x \equiv x' \,(mod\, \omega)$. We denote the finest and coarsest possible equivalence relations on X by $\perp$ and $\top$ respectively: for $x, x' \in X$, $x \equiv x' \,(mod\, \perp)$ iff $x = x'$, and $x \equiv x' \,(mod\, \top)$ for all x, x'. Let $\vee$ and $\wedge$ be respectively the *join* and *meet* operations of the lattice $\mathcal{E}(X)$.

Let X be a set and $\alpha_i : X \longrightarrow \mathcal{P}(X)$ be a function where i ranges over an index set I. Let

$$\mathcal{S} := (X, \{\alpha_i \,|\, i \in I\})$$

We call $\mathcal{S}$ a *dynamic system*, with the following interpretation. The elements $x \in X$ are the system "states", and $\{\alpha_i \,|\, i \in I\}$ is the "state transition function", representing the system dynamics. The function α_i is extended from a state to a set of states as follows: for $X' \subseteq X$, $\alpha_i(X') := \bigcup\{\alpha_i(x) \,|\, x \in X'\}$. This convention of extending a 'transition' function is used throughout the paper. We say that $\omega \in \mathcal{E}(X)$ is a *quasi-congruence* for $\mathcal{S}$ [Wnhm02] if

$$\omega \leq \bigwedge_{i \in I} (\omega \circ \alpha_i)$$

A quasi-congruence is an equivalence relation on X which is compatible with α_i, for all $i \in I$, in the following sense. If x and x' are related by ω, then x and x' are related by $\omega \circ \alpha_i$, for all $i \in I$. That is, $proj_\omega(\alpha_i(x)) = proj_\omega(\alpha_i(x'))$. In other words, for all $x_1 \in \alpha_i(x)$ there is $x_1' \in \alpha_i(x')$ such that x_1 and x_1' are related by ω, and vice versa.

The equivalence kernel of a causal reporter map can be characterized as a quasi-congruence for the prefix-closure function. Define $pre : \Sigma^* \longrightarrow \mathcal{P}(\Sigma^*) : s \longmapsto \overline{\{s\}}$.

Proposition 2.1 ([WTMH00]). *If θ is a causal reporter map from $\overline{L}$ to T^*, then $ker\,\theta$ is a quasi-congruence for $(\overline{L}, pre)$. Conversely, given a quasi-congruence ω for $(\overline{L}, pre)$, there are an event alphabet T and a causal reporter map $\theta : \overline{L} \longrightarrow T^*$ such that $ker\,\theta = \omega$.* □

We repeat from [Wnhm02] some basic properties of quasi-congruences. Let $\mathcal{S}$ be as above. The equivalence relation $\bot$ is a quasi-congruence for $\mathcal{S}$, and the quasi-congruences for $\mathcal{S}$ form a complete upper semilattice under the *join* operation $\vee$ of the lattice $\mathcal{E}(X)$. In general, the set of quasi-congruences for $\mathcal{S}$ is not closed under the *meet* operation $\wedge$ of the lattice $\mathcal{E}(X)$. Let Y be a set and $\gamma : X \longrightarrow Y$ be a function. Then (cf. [Wnhm76])

$$\omega^* := sup\{\omega \in \mathcal{E}(X) \,|\, \omega \leq (ker\,\gamma) \wedge \bigwedge_{i \in I} (\omega \circ \alpha_i)\}$$

exists. To compute ω^*, define $\psi : \mathcal{E}(X) \longrightarrow \mathcal{E}(X)$ according to

$$\psi(\omega) = (ker\,\gamma) \wedge \bigwedge_{i \in I} (\omega \circ \alpha_i) \tag{3}$$

The function ψ is monotone and ω^* is the greatest fixpoint of ψ. An operator $\varphi : \mathcal{E}(X) \longrightarrow \mathcal{E}(X)$ is said to be *downward continuous* [Wnhm02] if, for all nonincreasing sequences $\omega_n \in \mathcal{E}(X)$, $n = 0, 1, 2, \cdots$, (i.e., $\omega_n \geq \omega_{n+1}$ for $n \geq 0$), $inf_n(\varphi(\omega_n)) = \varphi(inf_n(\omega_n))$. The operator ψ defined in (3) is generally not downward continuous: in general $\omega^* < inf\{\psi^n(ker\,\gamma) \,|\, n = 0, 1, 2, \cdots\}$ (where the inequality is strict). On the other hand, it is well known (e.g. [Sif82]) that if α_i is *image-finite* for all i (i.e., $(\forall x \in X)\, |\alpha_i(x)| < \infty$), then ψ is downward continuous and $\omega^* = inf\{\psi^n(ker\,\gamma) \,|\, n = 0, 1, 2, \cdots\}$.

Next we define quotient and homomorphism. Let $\mathcal{S} = (X, \{\alpha_i \,|\, i \in I\}, x_0, X_m)$ be a dynamic system with initial state x_0 and set of *marker* states X_m. Let $\omega \in \mathcal{E}(X)$ be a quasi-congruence for $\mathcal{S}$ and let $proj_\omega : X \longrightarrow X/\omega$ be the canonical projection. The *quotient* dynamic system of $\mathcal{S}$ by ω, written $\mathcal{S}/\omega$, is given by $(Y, \{\beta_i \,|\, i \in I\}, y_0, Y_m)$, where $Y := X/\omega$, $\beta_i : Y \longrightarrow \mathcal{P}(Y) : y \longmapsto proj_\omega(\alpha_i(proj_\omega^{-1}(y)))$, $y_0 := proj_\omega(x_0)$ and $Y_m := proj_\omega(X_m)$.

Let $\mathcal{S} = (X, \{\alpha_i \,|\, i \in I\}, x_0, X_m)$ and $\mathcal{R} = (Y, \{\beta_i \,|\, i \in I\}, y_0, Y_m)$ be dynamic systems. A function $f : X \longrightarrow Y$ is a *homomorphism* from $\mathcal{S}$ to $\mathcal{R}$ if $y_0 = f(x_0)$, $Y_m = f(X_m)$ and, for $i \in I$, $f \circ \alpha_i = \beta_i \circ f$. If f is surjective, it is called an *epimorphism.* The canonical projection $proj_\omega$ in the definition of quotients above is an epimorphism from $\mathcal{S}$ to $\mathcal{S}/\omega$. Also, it is straightforward to verify that if f is a homomorphism from $\mathcal{S}$ to $\mathcal{R}$ then the equivalence kernel of f is a quasi-congruence for $\mathcal{S}$.

We now present a technical result to be used later. As preparation, let X and Y be sets and let $f : X \longrightarrow Y$ be a surjective function. Define

$$f^\times : \mathcal{E}(Y) \longrightarrow \mathcal{E}(X) : \rho \longmapsto f^\times(\rho) := \rho \circ f \tag{4}$$

Let $\mathcal{E}_f(X) := \{\omega \in \mathcal{E}(X) \,|\, ker\, f \leq \omega\}$. The set $\mathcal{E}_f(X)$ is a complete sublattice of $\mathcal{E}(X)$. Let $\pi, \omega \in \mathcal{E}(X)$ such that $\pi \leq \omega$. Following [BS81], define $\omega/\pi \in \mathcal{E}(X/\pi)$ as follows: for $x, x' \in X$,

$$[x]_\pi \equiv [x']_\pi \ (mod\, \omega/\pi) \text{ iff } x \equiv x' \ (mod\, \omega) \tag{5}$$

Define

$$f_\times : \mathcal{E}_f(X) \longrightarrow \mathcal{E}(Y) : \omega \longmapsto f_\times(\omega) := \omega/(ker\, f) \tag{6}$$

It is easy to see that $f^\times(\mathcal{E}(Y)) \subseteq \mathcal{E}_f(X)$, so we restrict the codomain of $f^\times$ to $\mathcal{E}_f(X)$. It is also clear that $f_\times$ is a lattice isomorphism from $\mathcal{E}_f(X)$ to $\mathcal{E}(Y)$, with $f^\times$ being its inverse.

Fact 2.1. *Let $\mathcal{S} = (X, \{\alpha_i \,|\, i \in I\})$ and $\mathcal{R} = (Y, \{\beta_i \,|\, i \in I\})$ be dynamic systems and f an epimorphism from $\mathcal{S}$ to $\mathcal{R}$. Then*

1. *If $\omega \in \mathcal{E}_f(X)$ is a quasi-congruence for $\mathcal{S}$, then $f_\times(\omega) \in \mathcal{E}(Y)$ is a quasi-congruence for $\mathcal{R}$.*
2. *If $\rho \in \mathcal{E}(Y)$ is a quasi-congruence for $\mathcal{R}$, then $f^\times(\rho) \in \mathcal{E}_f(X)$ is a quasi-congruence for $\mathcal{S}$.*
3. *Let*

$$\omega_\mathcal{S} := sup\{\omega \in \mathcal{E}(X) \,|\, \omega \text{ is a quasi-congruence for } \mathcal{S}\}$$
$$\rho_\mathcal{R} := sup\{\rho \in \mathcal{E}(Y) \,|\, \rho \text{ is a quasi-congruence for } \mathcal{R}\}$$

Then $\omega_\mathcal{S} = f^\times(\rho_\mathcal{R})$ and $\rho_\mathcal{R} = f_\times(\omega_\mathcal{S})$.

□

2.3 Observers

We recall the definition of observers. For a given language H (not necessarily closed) over Σ and a string $s \in \Sigma^*$, the *postfix closure* of s *in* H, denoted $pos_H(s)$, consists of all extensions of s in H:

$$pos_H(s) = \{u \in H \,|\, s \leq u\}$$

For a fixed language H, pos_H is a function from Σ^* to $\mathcal{P}(H)$. Let $H \subseteq \overline{L}$ and θ be a causal reporter map from $\overline{L}$ to T^*. We say that θ is an *H-observer* [WTMH00] if $ker\,\theta$ is a quasi-congruence for $(\overline{L}, pos_H)$. For an H-observer, if two strings s and s' are related by $ker\,\theta$, that is, $\theta(s) = \theta(s')$, then s and s' are related by $(ker\,\theta \circ pos_H)$, that is, for all $u \in pos_H(s)$ there is $u' \in pos_H(s')$ such that u and u' are related by $ker\,\theta$, and vice versa. In other words, for an extension of s to a string in H, say $sv \in H$, there is an extension of s' to a string in H, say $s'v' \in H$, such that $\theta(sv) = \theta(s'v')$. Thus, for an H-observer, if one examines the behavior of the high-level system and finds that $\theta(s)$ can be extended to a string $\theta(s)t \in \theta(H)$, then no matter where the low-level system is, as long as its θ-image is $\theta(s)$, it can be extended to a string in H with its θ-image equal to $\theta(s)t$, as stated in Lemma 2.1 below. Consider the following example. Let $H = \{\alpha_1\alpha_2, \alpha_3\alpha_4\}$ and $\theta : \epsilon \longmapsto \epsilon;\ \alpha_1 \longmapsto \epsilon;\ \alpha_3 \longmapsto \epsilon;\ \alpha_1\alpha_2 \longmapsto \tau_1;\ \alpha_3\alpha_4 \longmapsto \tau_2$. Note that α_1 and α_3 are related by $ker\,\theta$. But $pos_H(\alpha_1) = \{\alpha_1\alpha_2\}$, $pos_H(\alpha_3) = \{\alpha_3\alpha_4\}$ and $\theta(\alpha_1\alpha_2) \neq \theta(\alpha_3\alpha_4)$. Thus α_1 and α_3 are not related by $ker\,\theta \circ pos_H$, and so the θ in this example is not an H-observer.

Lemma 2.1 ([WTMH00]). *Let θ be a causal reporter map from $\overline{L}$ to T^* and $H \subseteq \overline{L}$. Then θ is an H-observer iff $(\forall s \in \overline{L})(\forall t \in T^*)\, \theta(s)t \in \theta(H) \Longrightarrow (\exists u \in \Sigma^*)\, su \in H$ and $\theta(su) = \theta(s)t$.* □

If $H = \overline{L}$, we say that θ is an *observer*. In this case, from Lemma 2.1 and Propositions 7 and 8 in [WW96], our definition of an observer is logically equivalent to that in [WW96].

Let L be a language over Σ, not necessarily closed, and let $\theta : \overline{L} \longrightarrow T^*$ be a causal reporter map. From now on L and θ are fixed. If θ is not an L-observer, our problem is to modify θ such that the resultant θ' is an L-observer. From the facts on quasi-congruences, we see that

$$\pi^* := sup\{\pi \in \mathcal{E}(\overline{L}) \mid \pi \leq (ker\,\theta) \wedge (\pi \circ pre) \wedge (\pi \circ pos_L)\}$$

exists. We shall modify θ iteratively until the equivalence kernel of the resultant map θ' is π^*, and thus obtain a natural solution to our problem.

In this approach, we obtain an observer with the coarsest equivalence kernel that is finer than that of the given map. One might ask if one could obtain an observer with the finest equivalence kernel that is coarser than that of the given map. It turns out that in general such an observer does not exist. This is because the set of quasi-congruences for *pre* and pos_L is generally not closed under the meet operation. The reader is invited to supply an example to demonstrate this fact.

2.4 Relabeling, quotient automata, Mealy and Moore automata

2.4.1 Relabeling

To compute our observer, we shall modify the given reporter map iteratively. This modification is carried out by relabeling the transitions of an appropriate automaton obtained from the given map. To define relabeling, let $\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_m)$ be a nondeterministic automaton, where Q is the state set, Σ is the event alphabet, $\delta : Q \times \Sigma \longrightarrow \mathcal{P}(Q)$, q_0 is the initial state and Q_m is the set of marker states. The set of transitions of $\mathbf{G}$ is given by

$$\mathcal{T}_\delta := \{(q, \sigma, q') \in Q \times \Sigma \times Q \mid q' \in \delta(q, \sigma)\} \tag{7}$$

An automaton $\mathbf{G}' = (Q, \Sigma', \delta', q_0, Q_m)$, possibly nondeterministic, is a *relabeling* of $\mathbf{G}$ if there is a surjective mapping $r : \mathcal{T}_\delta \longrightarrow \mathcal{T}_{\delta'} : (q, \sigma, q') \longmapsto (q, \sigma', q')$, where $\sigma' \in \Sigma'$. The function r is called the *relabeling function* from $\mathbf{G}$ to $\mathbf{G}'$. If $\mathbf{G}'$ is deterministic, then we say that $\mathbf{G}'$ is a *deterministic relabeling* of $\mathbf{G}$.

2.4.2 Quotient automaton

In the observer computation, one of the steps, as we shall see, involves the computation of the "quotient" of an automaton, which we now introduce. Let $\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_m)$ be a nondeterministic automaton. For $\Sigma' \subseteq \Sigma$ and $Q' \subseteq Q$, define

$$\delta_{\Sigma'}(Q') := \bigcup\{\delta(q, \sigma) \mid q \in Q',\ \sigma \in \Sigma'\} \tag{8}$$

If $\Sigma' = \{\sigma\}$, we write δ_σ instead of $\delta_{\Sigma'}$. For an equivalence relation ρ on Q, let $Y := Q/\rho$ be the quotient set and $g : Q \longrightarrow Y : q \longmapsto [q]_\rho$ the corresponding canonical projection. Let $y_0 := g(q_0)$ and $Y_m := g(Q_m)$ be respectively the initial state and the marker states in the quotient. Let $\sigma_{unobs} \notin \Sigma$ and $\Sigma_o \subseteq \Sigma$. Intuitively, it might be helpful to think of the labels in Σ_o as *observable* labels and those in $\Sigma - \Sigma_o$ as *unobservable* labels. Define the (induced) transition function on the quotient as follows.

$$\begin{array}{rcll} \eta & : & Y \times (\Sigma_o \cup \{\sigma_{unobs}\}) & \longrightarrow \mathcal{P}(Y) \\ & & (y, \sigma) & \longmapsto \left\{ \begin{array}{ll} g(\delta_\sigma(g^{-1}(y))) & \text{if } \sigma \in \Sigma_o \\ g(\delta_{(\Sigma-\Sigma_o)}(g^{-1}(y))) - \{y\} & \text{otherwise, i.e., } \sigma = \sigma_{unobs} \end{array} \right. \end{array} \tag{9}$$

Here σ_{unobs} labels an 'unobservable' transition. Define the *quotient* of $\mathbf{G}$ for Σ_o and ρ as

$$\overline{\mathbf{G}} = \mathbf{G}/(\Sigma_o, \rho) := (Y, \Sigma_o \cup \{\sigma_{unobs}\}, \eta, y_0, Y_m) \tag{10}$$

In general the quotient $\overline{\mathbf{G}}$ is nondeterministic. One can verify that if $\mathbf{G}$ is trim (for the definition of *trim*, see for example [RW87] or [Wnhm02]), so is $\overline{\mathbf{G}}$.

The following definitions are used in formulating the relabeling procedure (see Section 5). Define a map from the transitions of $\mathbf{G}$ to those of $\overline{\mathbf{G}}$:

$$\begin{array}{rcll} g_t & : & \mathcal{T}_\delta & \longrightarrow \mathcal{T}_\eta \\ & & (q, \sigma, q') & \longmapsto \left\{ \begin{array}{ll} (g(q), \sigma, g(q')) & \text{if } \sigma \in \Sigma_o \\ (g(q), \sigma_{unobs}, g(q')) & \text{if } \sigma \notin \Sigma_o, \text{ and } g(q) \neq g(q') \\ \text{undefined} & \text{otherwise} \end{array} \right. \end{array} \tag{11}$$

We call g_t the *transition mapping* from $\mathbf{G}$ to $\mathbf{G}/(\Sigma_o, \rho)$. Thus, g_t maps a transition of $\mathbf{G}$ to a transition of $\overline{\mathbf{G}}$ by mapping states to their equivalence classes, and either retaining the event label of a transition if it is in Σ_o, or replacing it with σ_{unobs} if it is unobservable (i.e. not in Σ_o) and the transition moves from one equivalence class to another distinct class.

Let $D_{g_t}(\mathcal{T}_\delta)$ denote the set of transitions on which g_t is defined, namely

$$D_{g_t}(\mathcal{T}_\delta) := \{\zeta \in \mathcal{T}_\delta \mid g_t(\zeta) \text{ is defined } \} \tag{12}$$

Thus, $(q, \sigma, q') \in \mathcal{T}_\delta - D_{g_t}(\mathcal{T}_\delta)$ if and only if $\sigma \notin \Sigma_o$ and $g(q) = g(q')$. In words, g_t is undefined on a transition precisely when the label of the transition is unobservable and the exit and entrance states are in the same equivalence class.

2.4.3 Mealy and Moore automata

The observer computation is carried out on an automaton derived from a Mealy automaton representing the reporter map. We recall that a *Mealy automaton* is a 7-tuple

$$\mathbf{A} = (X, \Sigma, T, \xi, \phi, x_0, X_m)$$

where the entries of $\mathbf{A}$ are respectively the state set, event alphabet, output event set, (partial) transition function $\xi : X \times \Sigma \longrightarrow X$, output function $\phi : X \times \Sigma \longrightarrow T$, the initial state and the set of marker states. Sometimes it is convenient to represent a reporter map by a Moore automaton. We recall that a *Moore* automaton is a 7-tuple

$$\mathbf{A} = (X, \Sigma, T, \xi, \psi, x_0, X_m)$$

where the entries of $\mathbf{A}$ are respectively the state set, event alphabet, output event set, (partial) transition function $\xi : X \times \Sigma \longrightarrow X$, output function $\psi : X \longrightarrow T$, the initial state and the set of marker states.

Let

$$\mathbf{G} = (Q, \Sigma, T \cup \{\tau_0\}, \delta, \phi, q_0, Q_m) \tag{13}$$

be a trim, deterministic Mealy automaton representing (L, θ). Here $\tau_0 \notin \Sigma \cup T$. Let $\mathbf{G}_{ts}$ be the *transition structure* of $\mathbf{G}$, i.e.,

$$\mathbf{G}_{ts} := (Q, \Sigma, \delta, q_0, Q_m)$$

For our computation, we first incorporate the "behavior" of the output function ϕ into the transition structure of $\mathbf{G}$. Let

$$\widehat{\mathbf{G}} := (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m) \tag{14}$$

be a relabeling of $\mathbf{G}_{ts}$ with the relabeling function $r_{\mathbf{G}} : \mathcal{T}_\delta \longrightarrow \mathcal{T}_{\widehat{\delta}}$ satisfying the following properties: for $(q, \sigma, q') \in \mathcal{T}_\delta$,

$$r_{\mathbf{G}}((q,\sigma,q')) = \begin{cases} (q, \tau, q') & \text{if } \phi(q,\sigma) = \tau \in T \\ (q, \sigma, q') & \text{if } \phi(q,\sigma) = \tau_0 \end{cases} \tag{15}$$

In general, $\widehat{\mathbf{G}}$ is nondeterministic.

Next we define the dynamic system corresponding to $\widehat{\mathbf{G}}$ as follows. For $\tau \in T$,

$$\Delta_\tau : Q \longrightarrow \mathcal{P}(Q) : q \longmapsto \bigcup \{\widehat{\delta}(q, u\tau u') \mid uu' \in (\Sigma - T)^*\}$$

and

$$\Delta_m : Q \longrightarrow \mathcal{P}(Q_m) : q \longmapsto \bigcup \{\widehat{\delta}(q, u) \cap Q_m \mid u \in (\Sigma - T)^*\}$$

The corresponding dynamic system is

$$\widetilde{\mathbf{G}} := (Q, \{\Delta_\tau \mid \tau \in T\} \cup \{\Delta_m\}, q_0, Q_m) \tag{16}$$

We assume that m is not a symbol in T.

We extend the function Δ_τ to strings as follows. For $q \in Q$ and $t\tau \in T^*$, $\Delta_\epsilon(q) = \{q\}$, and $\Delta_{t\tau}(q) = \bigcup \{\Delta_\tau(q') \mid q' \in \Delta_t(q)\}$.

The following notational convention will be used throughout. For a given Mealy automaton, say $\mathbf{A}$, $\widehat{\mathbf{A}}$ will denote a relabeling of $\mathbf{A}_{ts}$ as in (14) for incorporating the information in the output function into the transition structure $\mathbf{A}_{ts}$, and $\widetilde{\mathbf{A}}$ will denote the corresponding dynamic system of $\widehat{\mathbf{A}}$ as defined in (16).

One of the objectives of the paper is to define a computation on a (finite) Mealy automaton $\mathbf{G}$ producing a causal reporter map θ^* which has π^* as its equivalence kernel. To do this, we need to relate the computation on $\mathbf{G}$ to π^*, which is an equivalence relation on $\overline{L}$. To this end, we represent (L, θ) by a single transition structure as follows. Let

$$\mathbf{T}_{ts} = (X, \Sigma, \xi, x_0, X_m) \tag{17}$$

be a (deterministic) automaton such that there is a bijection $f : \overline{L} \longrightarrow X$ with the properties that $f(\epsilon) = x_0$ and $f(L) = X_m$, and that for $s \in \overline{L}$ and $\sigma \in \Sigma$, $\xi(f(s), \sigma)$ is defined iff $s\sigma \in \overline{L}$, in which case $\xi(f(s), \sigma) = f(s\sigma)$. Extend the domain of ξ to $X \times \Sigma^*$ in the standard way. It is clear that the transition graph of $\mathbf{T}_{ts}$ is a tree (generally infinite).

Relations on $\overline{L}$ are induced on $\mathbf{T}_{ts}$. Given a binary relation ρ on $\overline{L}$, let ρ_X be the induced binary relation on X as follows: for $x, x' \in X$, $(x, x') \in \rho_X$ iff $(f^{-1}(x), f^{-1}(x')) \in \rho$, where f^{-1} is the inverse of f. In this way equivalence relations $ker\,\theta$, π^* and the prefix partial order $\leq$ are induced on X. We shall call the induced relations again $ker\,\theta$, π^* and $\leq$ respectively; the context will determine the meaning. So, $ker\,\theta, \pi^* \in \mathcal{E}(X)$ and, for $x, x' \in X$, $x \leq x'$ iff there is $s \in \Sigma^*$ such that $\xi(x, s) = x'$.

To represent the reporter map θ, define $\lambda : \overline{L} \times \Sigma \longrightarrow T \cup \{\tau_0\}$ according to:

$$\lambda(s, \sigma) = \begin{cases} \tau_0 & \text{if } \theta(s\sigma) = \theta(s) \text{ and } s\sigma \in \overline{L} \\ \tau & \text{if } \theta(s\sigma) = \theta(s)\tau \text{ and } s\sigma \in \overline{L} \\ \text{undefined} & \text{otherwise} \end{cases}$$

Let $\lambda^\circ : X \times \Sigma \longrightarrow T \cup \{\tau_0\}$ be the induced function such that $\lambda^\circ(x, \sigma) = \lambda(f^{-1}(x), \sigma)$, where f^{-1} is the inverse of f. We will drop the superscript $\circ$ if there is no risk of confusion.

The automaton

$$\mathbf{T} := (X, \Sigma, T \cup \{\tau_0\}, \xi, \lambda, x_0, X_m)$$

is a Mealy automaton representing the function θ. Let

$$\widehat{\mathbf{T}} = (X, \Sigma \cup T, \widehat{\xi}, x_0, X_m) \text{ and } \widetilde{\mathbf{T}} = (X, \{\Xi_\tau \mid \tau \in T\} \cup \{\Xi_m\}, x_0, X_m) \tag{18}$$

be the corresponding relabeled system and dynamic system of $\mathbf{T}$ respectively, as in (14) and (16).

It can be verified that

$$\theta \text{ is an } L\text{-observer iff } ker\,\theta \text{ is a quasi-congruence on } \widetilde{\mathbf{T}} \tag{19}$$

and that the desired equivalence relation π^*, when induced on the state set X of $\widehat{\mathbf{T}}$, can be expressed in terms of Ξ_τ and Ξ_m as follows:

$$\pi^* = sup\{\pi \in \mathcal{E}(X) \mid \pi \leq (ker\,\theta) \wedge (\pi \circ pre) \wedge (\bigwedge_{\tau \in T} (\pi \circ \Xi_\tau)) \wedge (\pi \circ \Xi_m)\} \tag{20}$$

where $ker\,\theta$ and pre are respectively the induced equivalence relation and the prefix closure function on the state set X of $\widetilde{\mathbf{T}}$.

Let

$$h : X \longrightarrow Q : x \longmapsto \delta(q_0, f^{-1}(x))$$

It can be verified that $q_0 = h(x_0)$, $Q_m = h(X_m)$ and

$$\begin{gathered} (\forall \tau \in T)\, h \circ \Xi_\tau = \Delta_\tau \circ h \\ h \circ \Xi_m = \Delta_m \circ h \end{gathered} \tag{21}$$

That is, h is a homomorphism from $\widetilde{\mathbf{T}}$ to $\widetilde{\mathbf{G}}$. Let $\pi_{Mealy} \in \mathcal{E}(\overline{L})$ be the right congruence corresponding to the Mealy automaton (cf. (13)): for $s, s' \in \overline{L}$,

$$s \equiv s' (mod\, \pi_{Mealy}) \text{ iff } \delta(q_0, s) = \delta(q_0, s') \tag{22}$$

Let π_{Mealy} be induced on X, and the induced equivalence relation denoted by π_{Mealy}. It can be verified that $ker\,h = \pi_{Mealy}$; this fact will be used in the proof of Proposition 3.1.

3 Computation on finite automata

3.1 Computation of quasi-congruence

In this section we present the computation of the coarsest quasi-congruence. Let

$$\mathbf{G} = (Q, \Sigma, T \cup \{\tau_0\}, \delta, \phi, q_0, Q_m) \tag{23}$$

be a trim, deterministic Mealy automaton representing (L, θ). For computation, we assume for this paper that

$$\mathbf{G} \text{ is a finite transition structure.}$$

From $\mathbf{G}$, we obtain $\widehat{\mathbf{G}}$ and $\widetilde{\mathbf{G}}$ (as in (14) and (16) respectively). The coarsest quasi-congruence on $\widetilde{\mathbf{G}}$ is:

$$\rho_{\mathbf{G}} := sup\{\rho \in \mathcal{E}(Q) \mid \rho \leq (\bigwedge_{\tau \in T} (\rho \circ \Delta_\tau)) \wedge (\rho \circ \Delta_m)\} \tag{24}$$

To compute $\rho_{\mathbf{G}}$, define $\varphi : \mathcal{E}(Q) \longrightarrow \mathcal{E}(Q)$:

$$\varphi(\rho) = (\bigwedge_{\tau \in T} (\rho \circ \Delta_\tau)) \wedge (\rho \circ \Delta_m)$$

It can be verified that φ is monotone and that $\rho_{\mathbf{G}}$ is the greatest fixpoint of φ. Let

$$\rho_0 := \varphi^0(\top) = \top \text{ and } \rho_i := \varphi(\rho_{i-1})$$

for $i > 0$. Since φ is monotone and $\varphi(\top) \leq \top$, $\{\rho_i \mid i \geq 0\}$ is a nonincreasing sequence. Since $\mathbf{G}$ is a finite transition structure,

$$\rho_{\mathbf{G}} = inf\{\rho_i \mid i \geq 0\} = \rho_k$$

for some $k \geq 0$, i.e., $\rho_{\mathbf{G}}$ is reached in a finite number of steps.

This computation of $\rho_{\mathbf{G}}$ can be carried out by using the partition refinement algorithm of [Fer90] modified for the marker states in our models (Appendix D). Assume that $k = |Q|$ and $n = |T|$. From [Fer90], we know that the partition $\rho_{\mathbf{G}}$ can be computed in $O(k^3 n)$ time.

As an illustration consider the following.

Example 3.1. Let a causal reporter map be given by the Mealy automaton in Figure 1. We

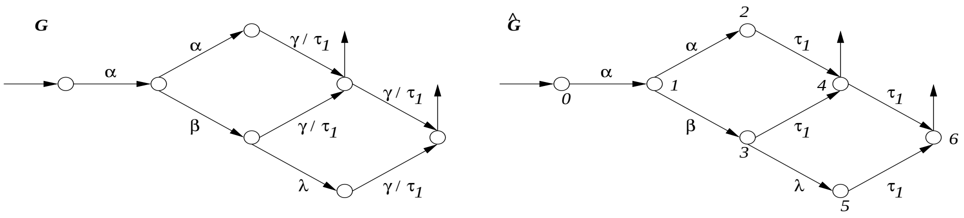

Figure 1: A Mealy automaton.

compute $\rho_{\mathbf{G}}$: $\rho_1 = \varphi(\top) = (\top \circ \Delta_{\tau_1}) \wedge (\top \circ \Delta_m)$ and $\rho_2 := \varphi(\rho_1)$. They are displayed in Figure 2. In this case, $\rho_{\mathbf{G}} = \rho_2$. The details of this computation are provided in Appendix D.

◇

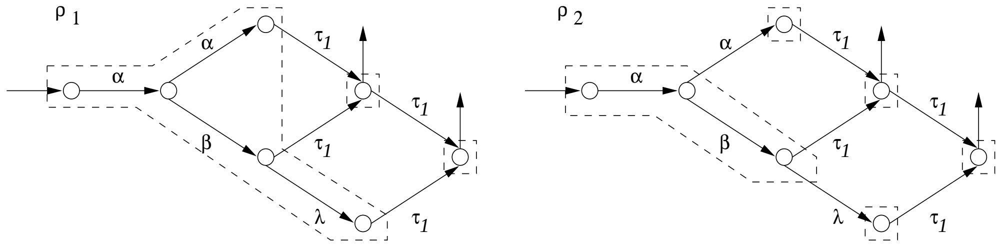

Figure 2: The partitions ρ_1 and ρ_2 on $\mathbf{G}$.

3.2 A property of coarsest quasi-congruences

In this subsection we present a property of $\rho_{\mathbf{G}}$ to be used later; a proof is given in Appendix A.

Proposition 3.1. *Suppose that $\theta : \overline{L} \longrightarrow T^*$ is an L-observer. Then for all $q, q' \in Q$, $q \equiv q' \, (mod \, \rho_{\mathbf{G}})$ iff*

$$\begin{aligned} &[(\forall t \in T^+) \, \Delta_t(q) \cap Q_m \neq \emptyset \Longleftrightarrow \Delta_t(q') \cap Q_m \neq \emptyset] \\ &\qquad\qquad\qquad \& \;\; \Delta_m(q) \neq \emptyset \Longleftrightarrow \Delta_m(q') \neq \emptyset \end{aligned} \tag{25}$$

Thus, given that the reporter map represented by $\mathbf{G}$ is an observer with respect to the marked behaviour, any two states q and q' are related by $\rho_{\mathbf{G}}$ if and only if, whenever a marker state is reachable from q, there will be a sequence of transitions with the same sequence of high-level output labels as that from q, leading from q' to a marker state; and vice versa.

4 A characterization of observers

After the coarsest quasi-congruence $\rho_{\mathbf{G}}$ on $\widetilde{\mathbf{G}}$ has been computed (as in Section 3), θ will be modified by relabeling the transitions of $\widehat{\mathbf{G}}$ in accordance with $\rho_{\mathbf{G}}$. This process is then repeated. In this section, we present a characterization of the observer property which provides a termination criterion for the iteration and, as we shall see in the next section, also suggests a transition relabeling scheme. Roughly, a causal reporter map θ is an L-observer if and only if a certain 'quotient' of a Mealy automaton for (L, θ) is deterministic and contains no 'unobservable' transitions.

Let

$$\widehat{\mathbf{G}} = (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m)$$

be an automaton obtained from a relabeling of $\mathbf{G}$ for (L, θ) as in (14) and (15). Let $\rho_{\mathbf{G}} \in \mathcal{E}(Q)$ be the coarsest quasi-congruence on $\widetilde{\mathbf{G}}$ that is finer than $\top$ as defined in (24). Let

$$\overline{\mathbf{G}} := \widehat{\mathbf{G}}/(T, \rho_{\mathbf{G}}) = (Y, T \cup \{\tau_0\}, \eta, y_0, Y_m)$$

where τ_0 is an event label that is not in $\Sigma \cup T$. A *τ_0-transition* is a transition labelled by τ_0, i.e., $\eta(y, \tau_0)$ is defined, for some $y \in Y$.

The following theorem provides a characterization of the observer property of θ.

Theorem 4.1. *θ is an L-observer iff $\overline{\mathbf{G}}$ is deterministic and contains no τ_0-transitions.*

The following is an informal argument for the characterization; a formal proof is given in Appendix B. If $\overline{\mathbf{G}}$ is nondeterministic, then there are two distinct states, say q_1 and q_2, in $\widehat{\mathbf{G}}$

which are the entry states of some nondeterministic transition, and q_1 and q_2 must belong to distinct $\rho_{\mathbf{G}}$ equivalence classes. It follows from the fact that $\rho_{\mathbf{G}}$ is the coarsest quasi-congruence that the behaviors for reaching the marker states from q_1 and q_2 are different. Since $\widehat{\mathbf{G}}$ is derived from a Mealy representation of (L, θ), there are two strings in $\overline{L}$, say s_1 and s_2, that are related by $ker\,\theta$ but $\theta(pos_L(s_1))$ and $\theta(pos_L(s_2))$ are different. It follows that $ker\,\theta$ is not a quasi-congruence for pos_L; in other words θ is not an L-observer. A similar line of argument can be followed if there is a τ_0-transition in $\overline{\mathbf{G}}$. Conversely, if θ is not an L-observer, then π^* is strictly finer than $ker\,\theta$, and this results in either the transitions with the same label now going from one π^* equivalence class to distinct equivalence classes, or an "unobservable" transition going from one equivalence class to another. From the construction of $\widehat{\mathbf{G}}$ and $\overline{\mathbf{G}}$, it follows that either $\overline{\mathbf{G}}$ is nondeterministic, or it contains a τ_0-transition.

Furthermore, if θ is an L-observer then the minimal recognizer for the language $\theta(L)$ is always simpler than a Mealy automaton representing (L, θ). A proof is also in Appendix B.

Theorem 4.2. *(i) if θ is an L-observer, then $\overline{\mathbf{G}}$ (deterministic and free of τ_0-transitions) is a minimal-state recognizer of the language $\theta(L)$. (ii) $\overline{\mathbf{G}}$ can be computed in polynomial time.*

Thus, if θ is an L-observer, then $\theta(L)$ can be recognized by an automaton with fewer states than that of the Mealy automaton for (L, θ), and this automaton can be constructed in polynomial time. If our causal map is a natural projection and $\mathbf{G}$ is the minimal generator of the language L, we recover the main results of [Won98].

5 Relabeling algorithm

In this section we present a relabeling scheme which carries out the modification of the map θ. Roughly, the scheme is to relabel $\widehat{\mathbf{G}}$ in such a way that the quotient of the relabeled automaton over $\rho_{\mathbf{G}}$ is deterministic and contains no τ_0-transitions. We see from Theorem 4.1 that, if the quotient is deterministic and free of τ_0-transitions, then θ is an L-observer. Intuitively, through relabeling we are modifying the reporter map in the direction of an observer.

Let

$$\widehat{\mathbf{G}} = (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m)$$

be an automaton obtained from a relabeling of $\mathbf{G}$ for (L, θ) as in (14), (15) and define $\widetilde{\mathbf{G}}$ as in (16). Let $\rho_{\mathbf{G}} \in \mathcal{E}(Q)$ be the coarsest quasi-congruence on $\widetilde{\mathbf{G}}$ that is finer than $\top$ as defined in (24), and bring in the quotient

$$\overline{\mathbf{G}} := \widehat{\mathbf{G}}/(T, \rho_{\mathbf{G}}) = (Y, T \cup \{\tau_0\}, \eta, y_0, Y_m)$$

where τ_0 is an event label that is not in $\Sigma \cup T$.

As discussed above, the goal of this procedure is to relabel $\widehat{\mathbf{G}}$ so that $\overline{\mathbf{G}}$ is deterministic and free of τ_0-transitions. To achieve this we first form the quotient $\overline{\mathbf{G}}$, then relabel $\overline{\mathbf{G}}$ so that it is deterministic and τ_0-transition free, and finally relabel $\widehat{\mathbf{G}}$ accordingly. These are the three basic steps in the relabeling procedure.

Let g_t be the transition mapping function from $\widehat{\mathbf{G}}$ to $\overline{\mathbf{G}}$ as defined in (11), and let $D_{g_t}(\mathcal{T}_{\widehat{\delta}})$ be defined as in (12).

Algorithm 5.1 (Relabeling).

Input: $\widehat{\mathbf{G}} = (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m)$ *is a (possibly nondeterministic) automaton and* ρ *is an equivalence relation on* Q.

1. *Compute* $\overline{\mathbf{G}} := \widehat{\mathbf{G}}/(T, \rho) = (Y, T \cup \{\tau_0\}, \eta, y_0, Y_m)$.

2. *Let* $\overline{\mathbf{G}}' = (Y, T', \eta', y_0, Y_m)$ *be a deterministic relabeling of* $\overline{\mathbf{G}}$ *such that* $\tau_0 \notin T'$ *and the relabeling function* $\overline{r} : \mathcal{T}_\eta \longrightarrow \mathcal{T}_{\eta'}$ *satisfies the following.*

$$(i)(\forall \zeta = (y, \tau, y') \in \mathcal{T}_\eta)\, \overline{r}(\zeta) \neq \zeta \Longrightarrow (\exists \tau' \in T' - T \cup \Sigma)\, \overline{r}(\zeta) = (y, \tau', y')$$

(All relabelings are done with new labels.)

$$(ii)(\forall (y_1, \tau, y_2), (y_1', \tau', y_2') \in \mathcal{T}_\eta)(\forall \tau'' \in T')\, \overline{r}((y_1, \tau, y_2)) = (y_1, \tau'', y_2),$$
$$\overline{r}((y_1', \tau', y_2')) = (y_1', \tau'', y_2') \Longrightarrow \tau = \tau'$$

(Two transitions are relabeled with the same label only if the two original labels are the same.)

3. *Let* $\widehat{\mathbf{G}}' = (Q, \Sigma \cup T', \widehat{\delta}', q_0, Q_m)$ *be a relabeling of* $\widehat{\mathbf{G}}$ *such that the relabeling function* $r : \mathcal{T}_{\widehat{\delta}} \longrightarrow \mathcal{T}_{\widehat{\delta}'}$ *satisfies the following.*

$$(i)(\forall \zeta \in \mathcal{T}_{\widehat{\delta}} - D_{g_t}(\mathcal{T}_{\widehat{\delta}}))\, r(\zeta) = \zeta,$$

(Transitions with unobservable labels and having both states in the same coset are not relabeled.)

$$(ii)(\forall \tau \in \Sigma \cup T)(\forall \zeta = (q, \tau, q') \in D_{g_t}(\mathcal{T}_{\widehat{\delta}}))(\exists \tau' \in T')\, r((q, \tau, q')) = (q, \tau', q')$$
and $(\overline{r} \circ g_t)(\zeta) = (g_t' \circ r)(\zeta)$

where g_t' *is the transition mapping from* $\widehat{\mathbf{G}}'$ *to* $\widehat{\mathbf{G}}'/(T', \rho)$.

(Other transitions are relabeled in accordance with $\overline{r}$.*)*

Output: $\widehat{\mathbf{G}}' = (Q, \Sigma \cup T', \widehat{\delta}', q_0, Q_m)$, *possibly nondeterministic.* ◇

For a finite automaton it is clear that the algorithm terminates.
We write

$$(\mathbf{G}', T') = relabel_\rho(\mathbf{G}, T)$$

to indicate an application of the relabeling algorithm.

In Step 2 of the relabeling algorithm, the set of event labels in $\overline{\mathbf{G}}'$ is T', which does not contain τ_0. We interpret this set T' as the set of observable event labels in the relabeled system. Thus the relabeling operation which produces $\overline{\mathbf{G}}'$ from $\overline{\mathbf{G}}$ eliminates τ_0-transitions. Since $\overline{\mathbf{G}}'$ is a deterministic relabeling of $\overline{\mathbf{G}}$, the relabeling operation also eliminates nondeterminism.

The property in $2(i)$ requires that any relabeling be carried out with new event labels (i.e. labels not in $T \cup \Sigma$). With this, assumption (2) remains satisfied in the relabeled system.

Note that the observable events that have not been relabeled remain observable in the relabeled system.

Property 2(ii) states that if, after relabeling, two transitions have the same event label, then the original event labels (i.e. those before relabeling) are also the same. If $(y_1, \sigma, y_2), (y_1', \sigma', y_2') \in \mathcal{T}_\eta$ and $\sigma \neq \sigma'$ then, after relabeling, these transitions will again have different labels. This property of the relabeling algorithm is important: as we shall see in the next section, the iterative computation produces a sequence of quasi-congruences $\rho_{i,Q}$ on the state set Q of $\mathbf{G}$, corresponding to a sequence of reporter maps θ_i, successively approaching the observer θ^* from θ (cf. Figure 3). Property 2(ii) ensures the monotonicity of this sequence, and therefore the convergence of the observer computation algorithm (cf. Lemma 6.1 below).

It can be checked that a relabeling as defined in Step 2 indeed exists. To see this, consider an implementation: (1) Number the states in $\widehat{\mathbf{G}}$ as $\{y_0, y_1, \cdots, y_{n-1}\}$, where y_0 is the initial state and $n = |Y|$. (2) For y_i, where $0 \leq i \leq n-1$, check all the transitions exiting y_i. If there is a τ_0 transition, relabel it with a new label, i.e., a label not in $T \cup \{\tau_0\}$. If there are nondeterministic transitions exiting y_i, relabel them with new labels chosen to satisfy (ii) of Step 2 so that the transitions exiting y_i are deterministic.

In Step 3, $\widehat{\mathbf{G}}$ is relabeled to produce a $\widehat{\mathbf{G}}'$ with the property that the quotient of $\widehat{\mathbf{G}}'$ mod (T', ρ) is $\overline{\mathbf{G}}'$.

Property 3(i) states that any transition with an unobservable event label and with its exit and entrance states in the same equivalence class of ρ will not be relabeled. Other transitions, according to 3(ii), are relabeled in accordance with the relabeling of their g_t-images.

Let $\zeta = (y, \sigma, y') \in D_{g_t}(T_\delta)$. Then by 3$(ii)$ $r((q, \sigma, q')) = (q, \sigma', q')$, for some $\sigma' \in T'$. Suppose that $\overline{r}$ relabels $g_t(\zeta) = (g(q), \sigma, g(q'))$ to $(g(q), \sigma'', g(q'))$, for some $\sigma'' \in T'$. That is, $(\overline{r} \circ g_t)(\zeta) = (g(q), \sigma'', g(q'))$. Thus,

$$\begin{aligned}
(g_t' \circ r)(\zeta) &= (g_t' \circ r)((q, \sigma, q')) \\
&= g_t'((q, \sigma', q')) && r \text{ relabels } \sigma \text{ with } \sigma' \\
&= (g(q), \sigma', g(q')) \\
&= (\overline{r} \circ g_t)(\zeta) \\
&= (g(q), \sigma'', g(q'))
\end{aligned}$$

Thus $\sigma' = \sigma''$, i.e., ζ is relabeled in accordance with the relabeling in $\overline{\mathbf{G}}$.

The lemma below follows immediately from the previous discussion, and it is used in Section 6.

Lemma 5.1. *Under the foregoing assumptions,*

(1) $\widehat{\mathbf{G}}'/(T', \rho)$ *is isomorphic to* $\overline{\mathbf{G}}'$ *and thus is deterministic and contains no* τ_0*-transitions.*

(2) *for all* $(q_1, \sigma, q_2), (q_1', \sigma', q_2') \in \mathcal{T}_\delta$ *and for* $\tau \in T'$ *such that* $r((q_1, \sigma, q_2)) = (q_1, \tau, q_2)$ *and* $r((q_1', \sigma', q_2')) = (q_1', \tau, q_2')$, *then either* $\sigma, \sigma' \in \Sigma - T$ *or* $(\sigma, \sigma' \in T$ *and* $\sigma = \sigma')$

□

6 Observer computation algorithm

With the preparation in previous sections, we now present the algorithm for computing an observer with the coarsest equivalence kernel that is finer than that of a given reporter map.

Algorithm 6.1 (Observer computation).

Input: *$\widehat{\mathbf{G}} = (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m)$ is an automaton obtained from a Mealy automaton $\mathbf{G}$ for (L, θ) by a relabeling as in (14) and (15). Let $i = 0$, $\widehat{\mathbf{G}}_i = \widehat{\mathbf{G}}$ and $T_i = T$. Let $\widetilde{\mathbf{G}}_i$ be the corresponding dynamic system as in (16).*

1. *Compute ρ_{i+1}, the coarsest quasi-congruence on $\widetilde{\mathbf{G}}_i$.*
2. *Compute $\overline{\mathbf{G}}_i = \widehat{\mathbf{G}}_i/(T_i, \rho_{i+1})$.*
3. *If $\overline{\mathbf{G}}_i$ is deterministic and contains no τ_0-transitions then*
   $\widehat{\mathbf{G}}' = \widehat{\mathbf{G}}_i$
   $T' = T_i$
   Stop
   else
   $(\widehat{\mathbf{G}}_{i+1}, T_{i+1}) = relabel_{\rho_{i+1}}(\widehat{\mathbf{G}}_i, T_i)$
   $i = i + 1$
   Return to Step 1
   end if.

Output: $\widehat{\mathbf{G}}' = (Q, \Sigma \cup T', \widehat{\delta}', q_0, Q_m)$, *possibly nondeterministic.* $\diamond$

For the computation, we start with $\widehat{\mathbf{G}} = (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m)$. Let $T_0 := T$ and $\widehat{\mathbf{G}}_0 := \widehat{\mathbf{G}}$. Next we compute the coarsest quasi-congruence $\rho_{1,Q}$ on $\widetilde{\mathbf{G}}_0$. If the quotient of $\widehat{\mathbf{G}}_0$ mod $(T_0, \rho_{1,Q})$ is not deterministic and τ_0-transition free, a relabeling is carried out on $\widehat{\mathbf{G}}_0$, i.e.,

$$(\widehat{\mathbf{G}}_1, T_1) = relabel_{\rho_{1,Q}}(\widehat{\mathbf{G}}_0, T_0)$$

where $\widehat{\mathbf{G}}_1 = (Q, \Sigma \cup T_1, \widehat{\delta}_1, q_0, Q_m)$. As this process is repeated, we generate a sequence $\{(\widehat{\mathbf{G}}_i, T_i) \mid i \geq 0\}$, where $\widehat{\mathbf{G}}_i = (Q, \Sigma \cup T_i, \widehat{\delta}_i, q_0, Q_m)$ and

$$(\widehat{\mathbf{G}}_{i+1}, T_{i+1}) = relabel_{\rho_{i+1,Q}}(\widehat{\mathbf{G}}_i, T_i)$$

with $\rho_{i+1,Q}$ being the coarsest quasi-congruence for $(\widetilde{\mathbf{G}}_i, T_i)$, and where $\widetilde{\mathbf{G}}_i (= (Q, \{\Delta^i_\tau \mid \tau \in T_i\} \cup \{\Delta^i_m\}, q_0, Q_m))$ is the corresponding dynamic system of $\widehat{\mathbf{G}}_i$.

From $\widehat{\mathbf{G}}_i$, we can construct a Mealy automaton, representing the reporter map θ_i in the i'th iteration of the computation. Let

$$\mathbf{G}_i = (Q, \Sigma, T_i \cup \{\tau_0\}, \delta, \phi_i, q_0, Q_m)$$

where δ is the same transition function as in (23), and $\phi_i : Q \times \Sigma \longrightarrow T_i \cup \{\tau_0\}$ is defined according to the following. Let $(r^i_{\mathbf{G}} \circ \cdots \circ r^1_{\mathbf{G}} \circ r_{\mathbf{G}})(q, \sigma, \delta(q, \sigma)) = (q, \tau, \delta(q, \sigma))$; then

$$\phi_i(q, \sigma) = \begin{cases} \tau & \text{if } \tau \in T_i \\ \tau_0 & \text{otherwise} \end{cases}$$

where $r_{\mathbf{G}}$ is given in (15), and $r^j_{\mathbf{G}}$ is the relabeling function from $\widehat{\mathbf{G}}_{j-1}$ to $\widehat{\mathbf{G}}_j$ for $j \geq 1$. For $i \geq 0$, let $\theta_i : \overline{L} \longrightarrow T_i^*$ be the causal reporter map represented by the Mealy automaton $\mathbf{G}_i$.

The following theorem states that, under the assumption that $\mathbf{G}$ is finite, Algorithm 6.1 will terminate after a finite number of steps and, when the computation stops, the equivalence kernel of the causal reporter map equals π^*.

Theorem 6.1. *Let $\widehat{\mathbf{G}} = (Q, \Sigma \cup T, \widehat{\delta}, q_0, Q_m)$ be an automaton obtained from a Mealy automaton $\mathbf{G}$ for (L, θ) by a relabeling as in (14) and (15). Then Algorithm 6.1 terminates in at most $|Q|$ number of steps. If the algorithm stops after n iterations, where $n \geq 1$, then $ker\, \theta_{n-1} = \pi^*$.*

To prove the theorem, we need to establish that (1) the iterative computation of Algorithm 6.1 terminates, and (2) after termination the equivalence kernel of the reporter map θ_{n-1} is π^*, the coarsest quasi-congruence finer than $ker\, \theta_0$ (cf. (20)).

To establish convergence, we shall show that the sequence $\{\rho_{i,Q} \,|\, i \geq 1\}$ is monotone (see Lemma 6.1). From this, Lemma 5.1, and the finiteness of the state set Q, convergence will follow. For a nondeterministic automaton $\mathbf{G}$, let $\mathcal{QC}(\mathbf{G})$ denote the set of quasi-congruences on $\mathbf{G}$.

Lemma 6.1.
(1) For $i \geq 1$, $\mathcal{QC}(\widetilde{\mathbf{G}}_i) \subseteq \mathcal{QC}(\widetilde{\mathbf{G}}_{i-1})$.
(2) The sequence $\{\rho_{i,Q} \,|\, i \geq 1\}$ is nonincreasing.

Proof: (1) Let ρ be a quasi-congruence on $\widetilde{\mathbf{G}}_i$ and let $q_1, q_2 \in Q$ such that $q_1 \equiv q_2\, (mod\, \rho)$. For $\tau \in T_{i-1}$, let $q_1' \in \Delta_\tau^{i-1}(q_1)$. There is then a path from q_1 to q_1' with the string $u\tau u'$, where $\tau \in T_{i-1}$ and $uu' \in (\Sigma - T_{i-1})^*$.

After the relabeling of $\widehat{\mathbf{G}}_{i-1}$ to $\widehat{\mathbf{G}}_i$, the string $u\tau u'$ becomes

$$u_1\tau_1u_2\tau_2 \cdots u_k\tau_ku_{k+1}\tau'u_1'\tau_1'u_2'\tau_2' \cdots u_l'\tau_l'u_{l+1}'$$

where $k, l \geq 0$, $\tau', \tau_j, \tau_j' \in T_i$ and $u_j, u_j' \in (\Sigma - T_i)^*$, where j ranges over the appropriate indices. Thus τ is relabeled to τ', and possibly some unobservable events along u and u' are relabeled to become observable events in $\mathbf{G}_i$. Since ρ is a quasi-congruence on $\widetilde{\mathbf{G}}_i$, there is $q_2' \in Q$ such that $q_2' \equiv q_1'\, (mod\, \rho)$ and there is a path from q_2 to q_2' with the string

$$v_1\tau_1v_2\tau_2 \cdots v_k\tau_kv_{k+1}\tau'v_1'\tau_1'v_2'\tau_2' \cdots v_l'\tau_l'v_{l+1}'$$

where $v_j, v_j' \in (\Sigma - T_i)^*$ and j ranges over the appropriate indices.

By the property of the relabeling algorithm that, if two transitions have the same label after relabeling, then the original labels of these two transitions are either both unobservable, or both observable and coincide (Lemma 5.1 (2)), this path from q_2 to q_2' in $\widehat{\mathbf{G}}_{i-1}$ has the string $v\tau v'$, where $vv' \in (\Sigma - T_{i-1})$. Thus, we have $q_2' \in \Delta_\tau^{i-1}(q_2)$.

This establishes that ρ is a quasi-congruence for (Q, Δ_τ^{i-1}).

It is similar to establish the quasi-congruence property of ρ with respect to (Q, Δ_m^{i-1}), and this concludes the proof.

(2) Since $\rho_{i+1,Q}$ is a quasi-congruence on $\widetilde{\mathbf{G}}_i$, by (1) $\rho_{i+1,Q}$ is a quasi-congruence on $\widetilde{\mathbf{G}}_{i-1}$.
Since $\rho_{i,Q}$ is the coarsest quasi-congruence for $\widetilde{\mathbf{G}}_{i-1}$, $\rho_{i+1,Q} \leq \rho_{i,Q}$. □

We now consider the second problem of showing that $ker\, \theta_{n-1} = \pi^*$. We recall that in one iteration of Algorithm 6.1 there are two basic steps: one is the computation of the coarsest quasi-congruence; the other is the relabeling in accordance with this quasi-congruence. Let us consider the first iteration of the observer computation algorithm. In this iteration, we first compute $\rho_{1,Q}$ on $\widetilde{\mathbf{G}}_0$, and then, assuming that the algorithm does not terminate at the first step, relabel $\widehat{\mathbf{G}}_0$, i.e., $(\widehat{\mathbf{G}}_1, T_1) = relabel_{\rho_{1,Q}}(\widehat{\mathbf{G}}_0, T_0)$.

Let $\widehat{\mathbf{T}}_0 := \widehat{\mathbf{T}}$ (cf. (18)). Let $\widetilde{\mathbf{T}}_0 = (X, \{\Xi_\tau^0 \,|\, \tau \in T_0\} \cup \{\Xi_m^0\}, x_0, X_m)$ be the corresponding dynamic system of $\widehat{\mathbf{T}}_0$. We note that the computation of $\rho_{1,Q}$ on $\widetilde{\mathbf{G}}_0$ corresponds to the computation of

$$\pi_{1,X} := sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq (\bigwedge_{\tau \in T_0} (\pi \circ \Xi_\tau^0)) \wedge (\pi \circ \Xi_m^0)\} \tag{26}$$

on $\widetilde{\mathbf{T}}_0$, in the following sense. Since h is a homomorphism from $\widetilde{\mathbf{T}}_0$ to $\widetilde{\mathbf{G}}_0$, by Fact 2.1

$$\pi_{1,X} = h^\times(\rho_{1,Q}) \text{ and } \rho_{1,Q} = h_\times(\pi_{1,X})$$

It is straightforward to check that π_{Mealy} is a quasi-congruence on $\widetilde{\mathbf{T}}_0$. Thus

$$\pi_{Mealy} \leq \pi_{1,X} \tag{27}$$

Recall that our goal is to compute a reporter map with π^* as its equivalence kernel from the given map θ_0, or in terms of equivalence kernels, to compute π^* from $\pi_0 := ker\,\theta_0$. Let

$$\pi_1' := sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq \pi_0 \wedge (\bigwedge_{\tau \in T_0} (\pi \circ \Xi_\tau^0)) \wedge (\pi \circ \Xi_m^0)\} \tag{28}$$

We have the following equality, to be proved in Lemma 6.4 below.

$$\pi_{1,X} \wedge \pi_0 = \pi_1'$$

Since π^* is a quasi-congruence on $\widetilde{\mathbf{T}}_0$ (cf. (20)),

$$\pi^* \leq \pi_1' \leq \pi_0 \tag{29}$$

Thus the step of computing $\rho_{1,Q}$ on $\widetilde{\mathbf{G}}_0$ may be thought of, in a sense, as computing π_1' from π_0 on $\widetilde{\mathbf{T}}_0$.

Next we investigate the relabeling of $\widehat{\mathbf{G}}_0$. As $\widehat{\mathbf{G}}_0$ is relabeled, the transitions of $\widehat{\mathbf{T}}_0$ are relabeled correspondingly. To make the last statement formal, let us recall that h is a homomorphism from $\widetilde{\mathbf{T}}_0$ to $\widetilde{\mathbf{G}}_0$ (cf. (21)). Let T_{uni} be a sufficiently large set that contains $T_0 (= T)$ and all the new labels in the relabeling process; for example, we could take T_{uni} to be $\bigcup_{i=1}^\infty T_i$. Now define a function

$$h_t = h \times id \times h : X \times (\Sigma \cup T_{uni}) \times X \longrightarrow Q \times (\Sigma \cup T_{uni}) \times Q$$

where id is the identity function on $\Sigma \cup T_{uni}$. Let $\widehat{\mathbf{T}}_1$ be a relabeling of $\widehat{\mathbf{T}}_0$ such that

$$r_{\mathbf{G}}^1 \circ h_t|_{\mathcal{T}_{\widehat{\xi}_0}} = h_t|_{\mathcal{T}_{\widehat{\xi}_1}} \circ r_{\mathbf{T}}^1 \tag{30}$$

where $r_{\mathbf{T}}^1 : \mathcal{T}_{\widehat{\xi}_0} \longrightarrow \mathcal{T}_{\widehat{\xi}_1}$ is the relabeling function from $\widehat{\mathbf{T}}_0$ to $\widehat{\mathbf{T}}_1$. It is easy to see that $r_{\mathbf{T}}^1$ above is unique. Let $\widetilde{\mathbf{T}}_1 = (X, \{\Xi_\tau^1 \,|\, \tau \in T_1\} \cup \{\Xi_m^1\}, x_0, X_m)$ be the corresponding dynamic system of $\widehat{\mathbf{T}}_1$. It can be verified that h is a homomorphism from $\widetilde{\mathbf{T}}_1$ to $\widetilde{\mathbf{G}}_1$, i.e.

$$\begin{aligned} (\forall \tau \in T_1)\, h \circ \Xi_\tau^1 &= \Delta_\tau^1 \circ h \\ h \circ \Xi_m^1 &= \Delta_m^1 \circ h \end{aligned} \tag{31}$$

It can be shown (see Lemma 6.3) that the relabeling of $\widehat{\mathbf{G}}_0$ corresponds to the computation of

$$\pi_1 := sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq \pi_1' \wedge (\pi \circ pre)\} \tag{32}$$

on $\widetilde{\mathbf{T}}_0$.

Since π^* is a quasi-congruence for pre,

$$\pi^* \leq \pi_1 \leq \pi_1' \leq \pi_0 \tag{33}$$

For the other iterations the situation is the same, except that the computation is now carried out with respect to $\widetilde{\mathbf{T}}_{i-1}$ and $\widetilde{\mathbf{G}}_{i-1}$, where $i \geq 1$. The relabeling of $\widetilde{\mathbf{T}}_{i-1}$ is done in accordance with the relabeling of $\widehat{\mathbf{G}}_{i-1}$ as in (30). Explicitly, let $\widehat{\mathbf{T}}_i$ be a relabeling of $\widehat{\mathbf{T}}_{i-1}$ such that

$$r^i_{\mathbf{G}} \circ h_t|_{\mathcal{T}_{\widehat{\xi}_{i-1}}} = h_t|_{\mathcal{T}_{\widehat{\xi}_i}} \circ r^i_{\mathbf{T}} \tag{34}$$

where $r^i_{\mathbf{T}} : \mathcal{T}_{\widehat{\xi}_{i-1}} \longrightarrow \mathcal{T}_{\widehat{\xi}_i}$ is the relabeling function from $\widehat{\mathbf{T}}_{i-1}$ to $\widehat{\mathbf{T}}_i$. Thus, as the computation proceeds a sequence $\{\widehat{\mathbf{T}}_i \,|\, i \geq 0\}$ is generated, corresponding to $\{\widehat{\mathbf{G}}_i \,|\, i \geq 0\}$. It can also be established that

$$h \text{ is a homomorphism from } \widetilde{\mathbf{T}}_i \text{ to } \widetilde{\mathbf{G}}_i \tag{35}$$

Similar to Lemma 6.1 (1), we have the following.

Lemma 6.2. *For $i \geq 1$, $\mathcal{QC}(\widetilde{\mathbf{T}}_i) \subseteq \mathcal{QC}(\widetilde{\mathbf{T}}_{i-1})$.* □

As in the first iteration, one can define

$$\pi_{i,X} = sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq (\bigwedge_{\tau \in T_{i-1}} (\pi \circ \Xi^{i-1}_\tau)) \wedge (\pi \circ \Xi^{i-1}_m)\} \tag{36}$$

where Ξ^{i-1}_τ and Ξ^{i-1}_m are the transition functions on $\widetilde{\mathbf{T}}_{i-1}$. Also, let

$$\pi_i' := sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq \pi_{i-1} \wedge (\bigwedge_{\tau \in T_{i-1}} (\pi \circ \Xi^{i-1}_\tau)) \wedge (\pi \circ \Xi^{i-1}_m)\} \tag{37}$$

and

$$\pi_i := sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq \pi_i' \wedge (\pi \circ pre)\} \tag{38}$$

The Hasse diagram in Figure 3 shows how $\rho_{i,Q}$, $\pi_{i,X}$, π_i', π_i, π_0, π^* and π_{Mealy} are related to each other.

Proposition 6.1. *Under the foregoing assumptions, for $i \geq 0$,*

(1) $ker\,\theta_i = \pi_i$,
(2) $\pi_{i+1,X} \wedge \pi_i = \pi_{i+1}'$,
(3) π_{Mealy} *and* π^* *are quasi-congruences on* $\widetilde{\mathbf{T}}_i$,
(4) $\pi_{Mealy} \leq \cdots \leq \pi_{i+1,X} \leq \cdots \leq \pi_{2,X} \leq \pi_{1,X} \leq \top$,
(5) $\pi^* \leq \cdots \leq \pi_{i+1} \leq \pi_{i+1}' \leq \cdots \leq \pi_1 \leq \pi_1' \leq \pi_0$

Proof: By induction on i.

Base step ($i = 0$): (1) We have the equality $ker\,\theta_0 = \pi_0$ by definition. (2) The equality $\pi_{1,X} \wedge \pi_0 = \pi_1'$ is established in Lemma 6.4 below. (3) Since π_{Mealy} is the right congruence associated with $\mathbf{G}_0$, it can be shown that it is a quasi-congruence on $\widetilde{\mathbf{T}}_0$. The fact that π^* is a quasi-congruence on $\widetilde{\mathbf{T}}_0$ follows from (20). (4) The inequality $\pi_{Mealy} \leq \pi_{1,X} \leq \top$ is proved in (27). (5) The inequality $\pi^* \leq \pi_1 \leq \pi_1' \leq \pi_0$ is proved in (33).

Induction step: Suppose that the proposition holds up to i. That is,

$$\begin{array}{ll} (1) & ker\,\theta_i = \pi_i, \\ (2) & \pi_{i+1,X} \wedge \pi_i = \pi'_{i+1}, \\ (3) & \pi_{Mealy} \text{ and } \pi^* \text{ are quasi-congruences on } \widetilde{\mathbf{T}}_i, \\ (4) & \pi_{Mealy} \leq \cdots \leq \pi_{i+1,X} \leq \cdots \leq \pi_{2,X} \leq \pi_{1,X} \leq \top, \\ (5) & \pi^* \leq \cdots \leq \pi_{i+1} \leq \pi'_{i+1} \leq \cdots \leq \pi_1 \leq \pi'_1 \leq \pi_0 \end{array}$$

We need to show that

$$\begin{array}{ll} (1) & ker\,\theta_{i+1} = \pi_{i+1}, \\ (2) & \pi_{i+2,X} \wedge \pi_{i+1} = \pi'_{i+2}, \\ (3) & \pi_{Mealy} \text{ and } \pi^* \text{ are quasi-congruences on } \widetilde{\mathbf{T}}_{i+1}, \\ (4) & \pi_{Mealy} \leq \cdots \leq \pi_{i+2,X} \leq \cdots \leq \pi_{2,X} \leq \pi_{1,X} \leq \top, \\ (5) & \pi^* \leq \cdots \leq \pi_{i+2} \leq \pi'_{i+2} \leq \cdots \leq \pi_1 \leq \pi'_1 \leq \pi_0 \end{array}$$

We now establish these in turn. For (1), we have the following lemma, proved in Appendix C.

Lemma 6.3. *Under the foregoing assumptions,* $ker\,\theta_{i+1} = \pi_{i+1}$.

The following lemma, proved in Appendix C, supplies the induction step for (2).

Lemma 6.4. *With* $ker\,\theta_{i+1} = \pi_{i+1}$,

$$\pi_{i+2,X} \wedge \pi_{i+1} = \pi'_{i+2}$$

For (3), we have the following two lemmas. Lemma 6.5 follows from the fact that $\mathbf{G}_{i+1}$ is a Mealy automaton for (L, θ_{i+1}) with π_{Mealy} as its right congruence, and Lemma 6.6 is proved in Appendix C.

Lemma 6.5. π_{Mealy} *is a quasi-congruence on* $\widetilde{\mathbf{T}}_{i+1}$.

Lemma 6.6. π^* *is a quasi-congruence on* $\widetilde{\mathbf{T}}_{i+1}$.

For (4), since π_{Mealy} is a quasi-congruence on $\widetilde{\mathbf{T}}_{i+1}$ (Lemma 6.5) and $\pi_{i+2,X}$ is the coarsest quasi-congruence on $\widetilde{\mathbf{T}}_{i+1}$ (cf. 36), it follows that

$$\pi_{Mealy} \leq \pi_{i+2,X}$$

By Lemma 6.2, $\pi_{i+2,X} \in \mathcal{QC}(\widetilde{\mathbf{T}}_i)$. Since $\pi_{i+1,X}$ is the coarsest quasi-congruence on $\widetilde{\mathbf{T}}_i$, it follows from the induction hypothesis that

$$\pi_{Mealy} \leq \pi_{i+2,X} \leq \pi_{i+1,X} \leq \cdots \leq \pi_{1,X} \leq \top$$

For (5), Since π^* is a quasi-congruence on $\widetilde{\mathbf{T}}_{i+1}$ (Lemma 6.6), it follows from the induction hypothesis that

$$\pi^* \leq \pi_{i+2} \leq \pi'_{i+2} \leq \pi_{i+1} \cdots \pi_1 \leq \pi'_1 \leq \pi_0$$

The proposition is proved. □

Proof of Theorem 6.1: By Lemma 6.1 (2) and the fact that $|Q|$ is finite, there is $n > 1$ such that $\rho_{n,Q} = \rho_{n-1,Q}$. Since $(\widehat{\mathbf{G}}_{n-1}, T_{n-1}) = relabel_{\rho_{n-1,Q}}(\widehat{\mathbf{G}}_{n-2}, T_{n-2})$, by Lemma 5.1

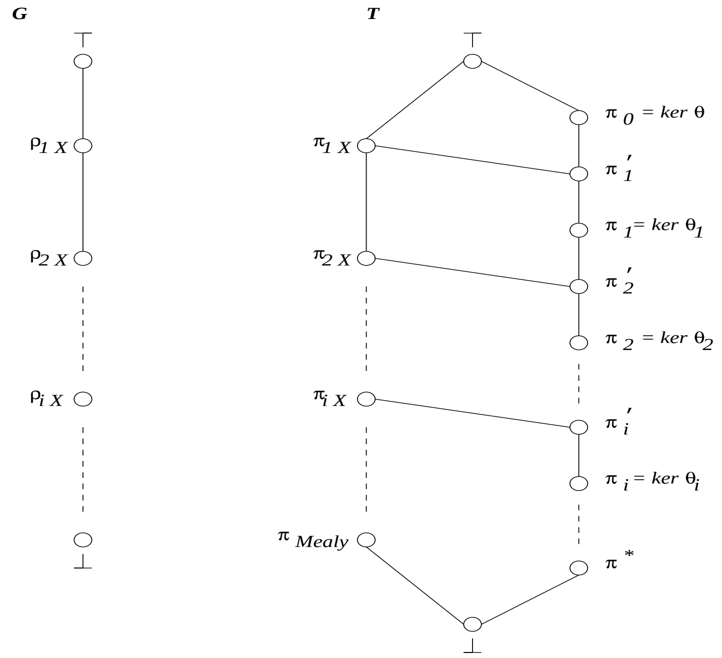

Figure 3: Hasse diagram of equivalence relations.

$\widehat{\mathbf{G}}_{n-1}/(T_{n-1}, \rho_{n-1,Q})$ is deterministic and free of τ_0-transitions. Since $\rho_{n,Q} = \rho_{n-1,Q}$, $\widehat{\mathbf{G}}_{n-1}/(T_{n-1}, \rho_{n,Q})$ is deterministic and free of τ_0-transitions. Thus Algorithm 6.1 stops.

It is easy to see that the algorithm will stop in at most $|Q|$ number of steps.

When Algorithm 6.1 stops after, say, $n \geq 1$ iterations, $\widehat{\mathbf{G}}_{n-1}/(T_{n-1}, \rho_{n,Q})$ is deterministic and free of τ_0-transitions. By Theorem 4.1, θ_{n-1} is an L-observer, i.e., $ker\, \theta_{n-1}$ is a quasi-congruence on $\widetilde{\mathbf{T}}_{n-1}$.

By repeatedly appealing to Lemma 6.2, one concludes that $ker\, \theta_{n-1}$ is a quasi-congruence on $\widetilde{\mathbf{T}}$. By Proposition 6.1, $ker\, \theta_{n-1} = \pi_{n-1} \leq \pi_0 = \ker \theta$. Clearly $ker\, \theta_{n-1}$ is a quasi-congruence for (X, pre). Thus, $ker\, \theta_{n-1} \leq \pi^*$.

By Proposition 6.1, $\pi^* \leq \pi_{n-1} = ker\, \theta_{n-1}$. Therefore $ker\, \theta_{n-1} = \pi^*$. □

Let $k = |Q|$ be the number of states. Recall that δ is the deterministic transition function of the Mealy automaton $\mathbf{G}$. The set of transitions of $\mathbf{G}$ is given by $T_\delta = \{(q, \sigma, q') \mid \delta(q, \sigma) = q'\}$. Let $n = |T_\delta|$. It is clear that $|T_i| \leq n$. Thus $\rho_{i,Q}$ can be computed in $O(k^3(n+1))$ time. Since the algorithm converges in at most k steps, the overall complexity is $O(k^4(n+1))$.

7 Examples

In this section, we illustrate the algorithms with three examples.

Example 7.1. Let $\widehat{\mathbf{G}}$ be the automaton obtained from the Mealy automaton in Figure 1 in Section 3. Let $\widehat{\mathbf{G}}_0 = \widehat{\mathbf{G}}$ and $T_0 = T$. In the first iteration of Algorithm 6.1, the coarsest quasi-congruence $\rho_{1,Q}$ is computed on $\widetilde{\mathbf{G}}_0$ in Step 1. This has already been done in Example 3.1; $\rho_{1,Q}$ equals $\rho_{\mathbf{G}}(= \rho_2)$ in Figure 2.

In Step 2 of Algorithm 6.1, $\overline{\mathbf{G}}_0 := \widehat{\mathbf{G}}_0/(T_0, \rho_{1,Q})$ is computed as displayed in Figure 4.

In Step 3, since $\overline{\mathbf{G}}_0$ contains τ_0-transitions, the relabeling algorithm (Algorithm 5.1) is applied to $(\widehat{\mathbf{G}}_0, T_0, \rho_{1,Q})$ to obtain $\widehat{\mathbf{G}}_1$ and T_1. Here $T_1 = \{\tau_1, \tau_2, \tau_3\}$. The automaton $\widehat{\mathbf{G}}_1$ is also displayed in Figure 4.

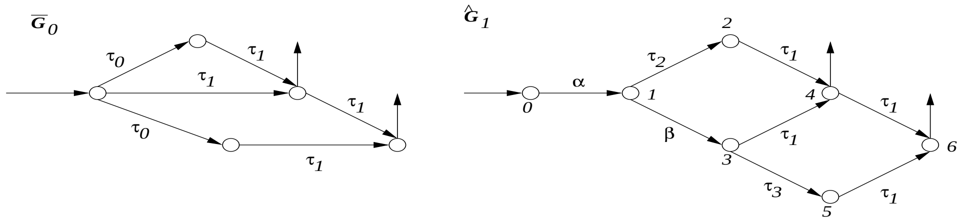

Figure 4: $\overline{\mathbf{G}}_0$ and $\widehat{\mathbf{G}}_1$.

Note that the quotient $\widehat{\mathbf{G}}_1/(T_1, \rho_{1,Q})$ is the same as $\overline{\mathbf{G}}_0$ except that the τ_0-transitions have been relabeled with τ_2 and τ_3.

For the second iteration of the algorithm, the coarsest quasi-congruence $\rho_{2,Q}$ on $\widetilde{\mathbf{G}}_1$ is computed and is displayed in Figure 5. It can be verified that $\overline{\mathbf{G}}_1 (:= \widehat{\mathbf{G}}_1/(T_1, \rho_{2,Q}))$ contains a τ_0-transition. So the relabeling algorithm is used to relabel $\widehat{\mathbf{G}}_1$ to produce $(\widehat{\mathbf{G}}_2, T_2)$. Here $T_2 = \{\tau_1, \tau_2, \tau_3, \tau_4\}$. The automaton $\widehat{\mathbf{G}}_2$ is displayed in Figure 5.

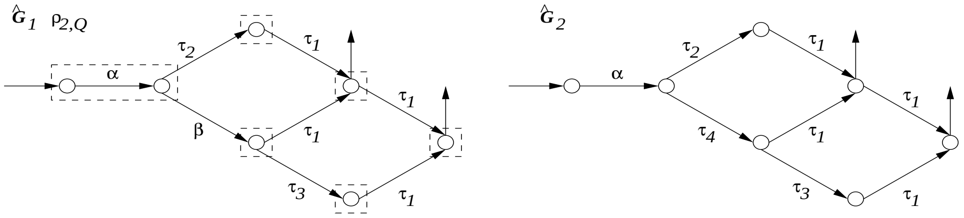

Figure 5: $(\widehat{\mathbf{G}}_1, \rho_{2,Q})$ and $\widehat{\mathbf{G}}_2$.

For the third iteration of the algorithm, we have that $\overline{\mathbf{G}}_2 := \widehat{\mathbf{G}}_2/(T_2, \rho_{3,Q})$ is deterministic and contains no τ_0-transitions, where $\rho_{3,Q}$ is the coarsest quasi-congruence on $\widetilde{\mathbf{G}}_2$. With this the computation terminates. The output of the algorithm is $\widehat{\mathbf{G}}_2$.

The Mealy automaton for the new causal map θ', and the automaton recognizing $\theta'(L_m)$ where L_m is the marked behavior of $\mathbf{G}$ in Figure 1, are displayed in Figure 6. One can verify that the equivalence kernel of the new causal map is indeed π^*. $\diamond$

For the second example, we consider the Moore automata in Figures 5.6.5 and Figure 5.6.7 in [Wnhm02]. As it turns out, the reporter maps represented by these Moore automata are in fact observers.

Example 7.2. Figure 7 displays the automaton obtained from the Moore automaton in Figure 5.6.5 in [Wnhm02]. (One could think of first converting the Moore automaton to a Mealy automaton and then relabeling as in Section 3.) To avoid clutter, the low-level event labels are not displayed as they do not enter into the computation. The automaton $\widehat{\mathbf{G}}$ in Figure 7 is obtained

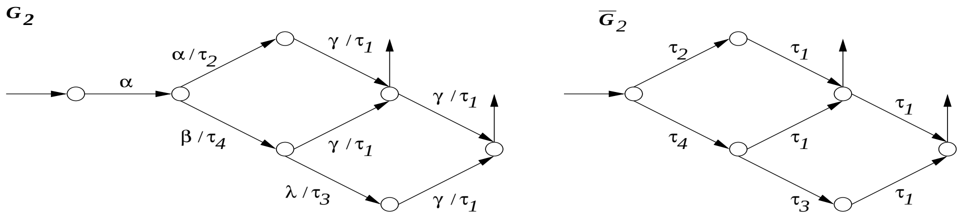

Figure 6: The Mealy automaton representing θ' and the automaton for $\theta'(L_m)$.

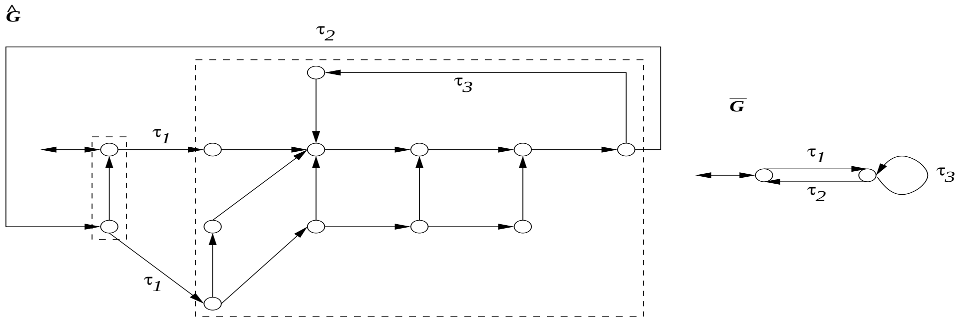

Figure 7: The results of computation.

from the Moore automaton in Figure 5.6.5 of [Wnhm02] by incorporating the "behaviour" of the output function into the transition structure, along the lines of (14). Here $T = \{\tau_1, \tau_2, \tau_3\}$.

The coarsest quasi-congruence $\rho_{1,Q}$ on $\widetilde{\mathbf{G}}$ is shown on $\widehat{\mathbf{G}}$ in Figure 7. The quotient automaton $\overline{\mathbf{G}} := \widehat{\mathbf{G}}/(T, \rho_{1,Q})$ is constructed, with two cells of $\rho_{1,Q}$ on $\widehat{\mathbf{G}}$ as the two states of the quotient. The transition τ_1 goes from the left cell to the right, τ_2 from the right cell to the left, and τ_3 selfloops on the right cell. The quotient is displayed in Figure 7.

Since $\overline{\mathbf{G}}$ is deterministic and contains no τ_0-transitions, the computation terminates and the given causal map is an L_m-observer, where L_m is the marked behavior of the original Moore automaton. According to Theorem 4.2, $\overline{\mathbf{G}}$ recognizes the language $\theta(L_m)$.

The same can be applied to the Moore automaton in Figure 5.6.7 in [Wnhm02] and it turns out that the given causal map is also an L_m-observer. The results of the computation are displayed in Figure 8. The automaton $\overline{\mathbf{G}}$ recognizes the language $\theta(L_m)$. $\diamond$

For the third example, we apply our algorithms to the example in Section 5 of [Gua97].

Example 7.3. In Section 5 of [Gua97], the Small Factory example of [Wnhm02] (Example 3.3.4) is used as the base model to set up a hierarchy. The Small Factory consists of two machines arranged in a line with a buffer between the two machines, accepting parts from the first machine (designated $\mathbf{M}_1$) and delivering parts to the second ($\mathbf{M}_2$). The models of the two machines are the same as that in [Wnhm02], with events *start*, *finish*, *break-down* and *repair*. The specification in this case is only the no-overflow-or-underflow safety requirement for the buffer, which is assumed to have a capacity of one. (There is no specification on the priority of repair as in the original example of [Wnhm02].) An optimal or least restrictive supervisor is designed. To construct the hierarchy which focuses on the breakdown aspect of the system, the following high-level events are

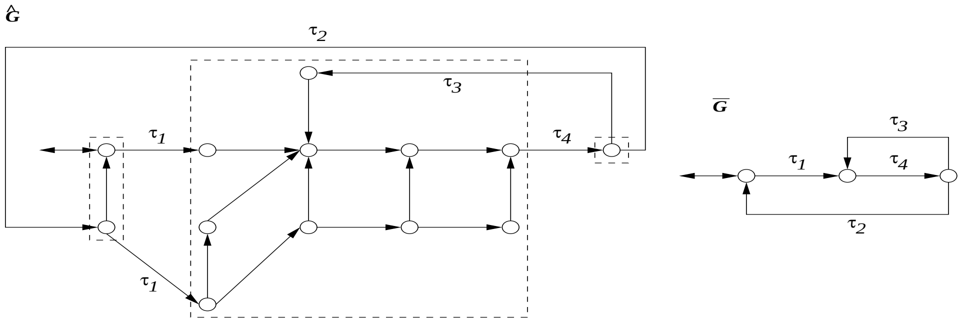

Figure 8: The results of computation for the second Moore automaton.

introduced: τ_1 indicates that only $\mathbf{M}_1$ has broken down, τ_2 that only $\mathbf{M}_2$ has broken down, τ_3 that both machines are down. The closed-loop behavior of the system with the above high-level events is given by the Moore automaton in Figure 9. To avoid clutter, the low-level event labels

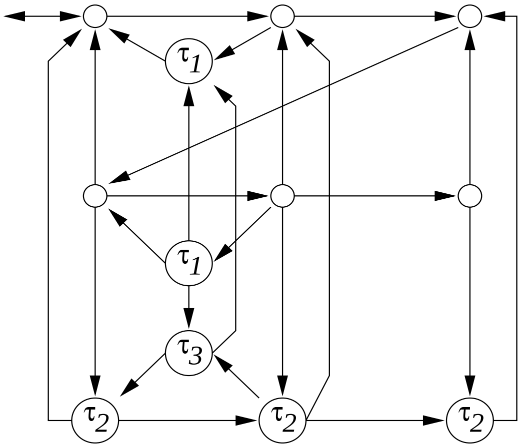

Figure 9: A Moore automaton.

are again not displayed.

We relabel the Moore automaton in Figure 9 as in Section 3. The result is displayed in Figure 10. Let $\widehat{\mathbf{G}}_0$ and T_0 be respectively the automaton in Figure 10 and the high-level event set. In the first iteration of the observer computation algorithm, the coarsest quasi-congruence $\rho_{1,Q}$ is computed and is shown in Figure 10. The quotient $\overline{\mathbf{G}}_0 := \widehat{\mathbf{G}}_0/(T,\rho_{1,Q})$ is also displayed in Figure 10.

To eliminate nondeterminism and τ_0-transitions, $\widehat{\mathbf{G}}_0$ is relabeled to produce $\widehat{\mathbf{G}}_1$ as shown in Figure 11. Here the high-level event set $T_1 = \{\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6\}$. The quotient $\widehat{\mathbf{G}}_1/(T_1, \rho_{1,Q})$ is also displayed in Figure 11, which shows that the relabeling indeed removes nondeterminism and the τ_0-transition.

In the second iteration, the coarsest quasi-congruence $\rho_{2,Q}$ is computed and is displayed in Figure 11. The quotient $\overline{\mathbf{G}}_1 := \widehat{\mathbf{G}}_1/(T_1, \rho_{2,Q})$ is presented in Figure 12. By inspection it is deterministic and contains no τ_0-transitions. Thus the computation terminates and the result is

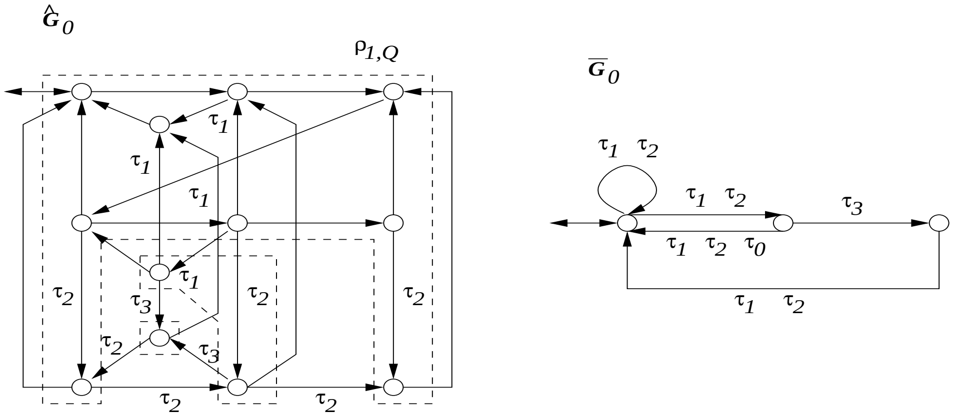

Figure 10: The automaton $\widehat{\mathbf{G}}_0$, the partition $\rho_{1,Q}$ and the quotient $\overline{\mathbf{G}}_0$.

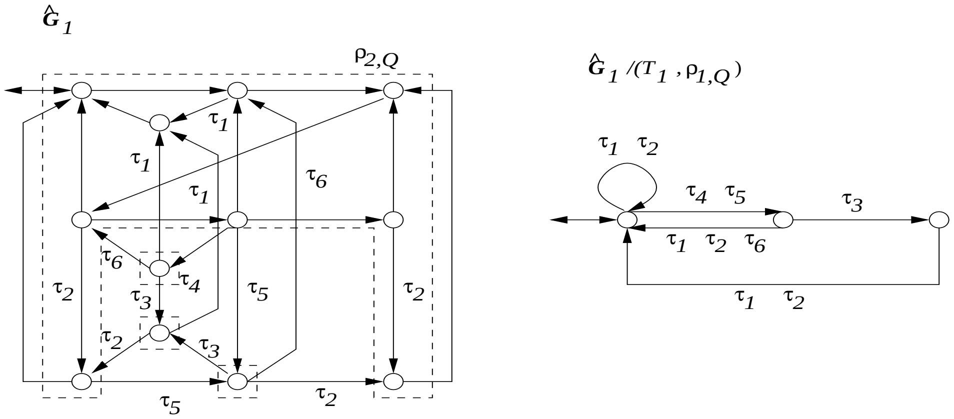

Figure 11: The automata $\widehat{\mathbf{G}}_1$ and $\widehat{\mathbf{G}}_1/(T_1, \rho_{1,Q})$.

$\widehat{\mathbf{G}}_1$ in Figure 11.

From $\widehat{\mathbf{G}}_1$, we obtain the Moore automaton given in Figure 13.

According to our theory, the new causal map represented by the Moore automaton in Figure 13 is an L_m-observer, where L_m is the marked behavior of the low-level system. By Theorem 4.2, $\overline{\mathbf{G}}_1$ recognizes the language $\theta_1(L_m)$. $\diamond$

8 Conclusions

In the case where a given causal reporter map can be represented by a finite Mealy automaton, an effective computational algorithm has been developed, for computing an observer with the coarsest equivalence kernel that is finer than that of a given reporter map. Issues of efficient implementation of the algorithm, as well as the problem of minimal extension of the alphabet in the relabeling process, remain to be investigated. Another topic of research is to study how this algorithm can be adapted to the computation of observers for natural projections in the decentralized control of

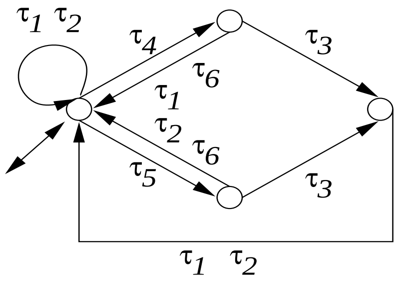

Figure 12: The automaton $\widehat{\mathbf{G}}_1/(T_1, \rho_{2,Q})$.

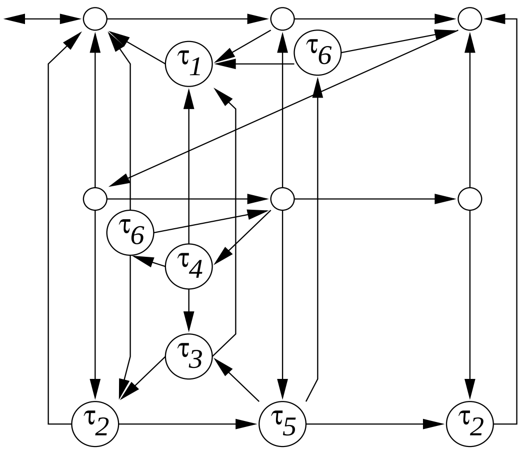

Figure 13: The Moore automaton for the new causal map.

discrete-event systems.

Acknowledgement: Financial support from the Natural Sciences and Engineering Research Council (NSERC) of Canada, RG7399, is gratefully acknowledged.

A Proof of Proposition 3.1

In this appendix, we prove the proposition in Section 3.2. For this we need the following three lemmas. For the first lemma, let $\equiv_{\theta(L)}$ be the Nerode equivalence for $\theta(L)$ and $\equiv_{\theta(L)} |_{\overline{\theta(L)}}$ its restriction to $\overline{\theta(L)}$. Let $\pi_{Mealy} \in \mathcal{E}(\overline{L})$ be as defined in (22).

Lemma A.1. *Suppose that θ is an L-observer. Then*

$$\pi_{Mealy} \leq (\equiv_{\theta(L)} |_{\overline{\theta(L)}}) \circ \theta$$

Proof: Let $s, s' \in \overline{L}$ such that $s \equiv s' \,(mod\, \pi_{Mealy})$. It is equivalent to show that $\theta(s) \equiv_{\theta(L)} \theta(s')$. Let $t \in T^*$ such that $\theta(s)t \in \theta(L)$. Since θ is an L-observer, by Lemma 2.1 there is $u \in \Sigma^*$ such that $su \in L$ and $\theta(su) = \theta(s)t$. Since $s \equiv s' \,(mod\, \pi_{Mealy})$, $s'u \in L$ and $\theta(s'u) = \theta(s')t$. Thus $\theta(s')t \in \theta(L)$. By symmetry we have the other implication. Hence, $\theta(s) \equiv_{\theta(L)} \theta(s')$. $\square$

We induce π_{Mealy} and $(\equiv_{\theta(L)} |_{\overline{\theta(L)}} \circ \theta) \in \mathcal{E}(\overline{L})$ on X, and denote the induced relations by the same symbol, as the context will determine which meaning is intended. It is straightforward to verify that

$$ker\, h = \pi_{Mealy} \leq \{X_m, X - X_m\}$$

Next we let

$$\rho := (\equiv_{\theta(L)} |_{\overline{\theta(L)}} \circ \theta)/\pi_{Mealy}$$

(cf. (5)). So $\rho \in \mathcal{E}(Q)$.

Lemma A.2. *Suppose that θ is an L-observer. Then for $s, s' \in \overline{L}$, $s \equiv s' \,(mod\, (\equiv_{\theta(L)} |_{\overline{\theta(L)}}) \circ \theta)$ iff*

$$\begin{aligned}[(\forall t \in T^+)\, \Xi_t(x_s) \cap X_m \neq \emptyset \iff \Xi_t(x_{s'}) \cap X_m \neq \emptyset] \\ \&\ \ \Xi_m(x_s) \neq \emptyset \iff \Xi_m(x_{s'}) \neq \emptyset\end{aligned}$$

where x_s and $x_{s'}$ are the corresponding states in $\widetilde{\mathbf{T}}$ (cf. (18)) respectively, and Ξ_t is the extension of Ξ_τ to strings.

Proof: (If) Let $s, s' \in \overline{L}$, and $t \in T^+$ such that $\theta(s)t \in \theta(L)$. Since θ is an L-observer, by Lemma 2.1 there is $u \in \Sigma^*$ such that $su \in L$ and $\theta(su) = \theta(s)t$. Thus it follows that $\Xi_t(x_s) \cap X_m \neq \emptyset$. By hypothesis, $\Xi_t(x_{s'}) \cap X_m \neq \emptyset$. Hence there is $u' \in \Sigma^*$ such that $s'u' \in L$ and $\theta(s'u') = \theta(s')t$, so $\theta(s')t \in \theta(L)$. By symmetry we have the other implication. It is similar to establish that $\theta(s) \in \theta(L)$ iff $\theta(s') \in \theta(L)$. Thus, $\theta(s) \equiv_{\theta(L)} \theta(s')$, i.e., $s \equiv s' \,(mod\, ((\equiv_{\theta(L)} |_{\overline{\theta(L)}}) \circ \theta))$.

(Only if) Let $t \in T^+$ such that $\Xi_t(x_s) \cap X_m \neq \emptyset$. So there is $u \in \Sigma^*$ such that $su \in L$ and $\theta(su) = \theta(s)t$. Thus $\theta(s)t \in \theta(L)$. By hypothesis $\theta(s')t \in \theta(L)$. Since θ is an L-observer, by Lemma 2.1 there is $u' \in \Sigma^*$ such that $s'u' \in L$ and $\theta(s'u') = \theta(s')t$, so $\Xi_t(x_{s'}) \cap X_m \neq \emptyset$. The other implication follows by symmetry. The second logical statement can be established similarly. □

The third and last lemma follows from Lemmas A.1 and A.2.

Lemma A.3. *Suppose that θ is an L-observer. Then for $q, q' \in Q$, $q \equiv q' \,(mod\, \rho)$ iff*

$$\begin{aligned}[(\forall t \in T^+)\, \Delta_t(q) \cap Q_m \neq \emptyset \iff \Delta_t(q') \cap Q_m \neq \emptyset] \\ \&\ \ \Delta_m(q) \neq \emptyset \iff \Delta_m(q') \neq \emptyset\end{aligned}$$

Proof: (If) Let $q, q' \in Q$ such that the hypothesis holds. Let $x, x' \in X$ such that $h(x) = q$ and $h(x') = q'$. Since h is a homomorphism from $\widetilde{\mathbf{T}}$ to $\widetilde{\mathbf{G}}$, it can be established that

$$\begin{aligned} h \circ \Xi_t &= \Delta_t \circ h \\ h \circ \Xi_m &= \Delta_m \circ h \end{aligned} \tag{39}$$

for $t \in T^*$. It then follows from this, $ker\, h \leq \{X_m, X - X_m\}$, and the hypothesis, that

$$\begin{aligned}[(\forall t \in T^+)\, \Xi_t(x) \cap X_m \neq \emptyset \iff \Xi_t(x') \cap X_m \neq \emptyset] \\ \&\ \ \Xi_m(x) \neq \emptyset \iff Xi_m(x') \neq \emptyset\end{aligned}$$

By Lemma A.2, $x \equiv x' \,(mod\, ((\equiv_{\theta(L)} |_{\overline{\theta(L)}}) \circ \theta))$. It follows that $q \equiv q' \,(mod\, \rho)$.

(Only if) Let $q, q' \in Q$ such that $q \equiv q' \,(mod\, \rho)$. Let $x, x' \in X$ such that $h(x) = q$ and $h(x') = q'$. It follows that $x \equiv x' \,(mod\, ((\equiv_{\theta(L)} |_{\overline{\theta(L)}}) \circ \theta))$. By Lemma A.2,

$$\begin{aligned}[(\forall t \in T^+)\, \Xi_t(x) \cap X_m \neq \emptyset \iff \Xi_t(x') \cap X_m \neq \emptyset] \\ \&\ \ \Xi_m(x) \neq \emptyset \iff \Xi_m(x') \neq \emptyset\end{aligned}$$

From this, (39) and $ker\, h \leq \{X_m, X - X_m\}$, the result is established. $\square$

Proof of Proposition 3.1: It is sufficient to show that $\rho = \rho_{\mathbf{G}}$.

First we show that ρ is a quasi-congruence on $\widetilde{\mathbf{G}}$. Let $q_1, q_2 \in Q$ such that $q_1 \equiv q_2\, (mod\, \rho)$. Let $\tau \in T$ and $q_1' \in \Delta_\tau(q_1)$. Since $\mathbf{G}$ is trim, there is $t \in T^*$ such that $\Delta_{\tau t}(q_1) \cap Q_m \neq \emptyset$. By Lemma A.3, $\Delta_{\tau t}(q_2) \cap Q_m \neq \emptyset$. Let $q_2' \in \Delta_\tau(q_2)$.

Claim: $q_1' \equiv q_2'\, (mod\, \rho)$.

Let $t \in T^*$ such that $\Delta_t(q_1') \cap Q_m \neq \emptyset$. So $\Delta_{\tau t}(q_1) \cap Q_m \neq \emptyset$. By Lemma A.3, $\Delta_{\tau t}(q_2) \cap Q_m \neq \emptyset$. Since θ is an L-observer, $\Delta_t(q_2') \cap Q_m \neq \emptyset$. The other implication follows by symmetry. So by Lemma A.3 $q_1' \equiv q_2'\, (mod\, \rho)$. This establishes the claim.

Next we show that, for any quasi-congruence $\rho' \in \mathcal{E}(Q)$ on $\widetilde{\mathbf{G}}$, $\rho' \leq \rho$. Let $q_1, q_2 \in Q$ such that $q_1 \equiv q_2\, (mod\, \rho')$. Since ρ' is a quasi-congruence,

$$\begin{aligned} &[(\forall t \in T^+)\, \Delta_t(q_1) \cap Q_m \neq \emptyset \Longleftrightarrow \Delta_t(q_2) \cap Q_m \neq \emptyset] \\ &\qquad \& \;\; \Delta_m(q_1) \neq \emptyset \Longleftrightarrow \Delta_m(q_2) \neq \emptyset \end{aligned}$$

By Lemma A.3, $q_1 \equiv q_2\, (mod\, \rho)$.

Thus, $\rho = \rho_{\mathbf{G}}$. $\square$

B Proofs in Section 4

In this appendix, we provide proofs for the two theorems in Section 4.

B.1 Theorem 4.1

For this theorem, we establish sufficiency and necessity in the following two lemmas.

The following definition is needed: Let $\mathbf{A} = (A, \Gamma, \mu, a_0, A_m)$ be a nondeterministic automaton. A *path from* $a \in A$ in $\mathbf{A}$ is a sequence of the form (a, ϵ, a) or of the form $(a, \gamma_1, a_1)(a_1, \gamma_2, a_2) \cdots (a_{n-1}, \gamma_n, a_n)$, where $n \geq 1$, $a_1 \in \mu(a, \gamma_1)$ and, for i with $2 \leq i \leq n$, $a_i \in \mu(a_{i-1}, \gamma_i)$.

Lemma B.1. *If* $\overline{\mathbf{G}}$ *is deterministic and contains no* τ_0*-transitions, then* θ *is an* L*-observer.*

Proof: We show that $ker\, \theta$ is a quasi-congruence for $(X, (\{\Xi_\tau \mid \tau \in T\}, \Xi_m))$. With this established, it follows from (19) that θ is an L-observer.

Let $x_1, x_2 \in X$ such that $x_1 \equiv x_2\, (mod\, ker\, \theta)$.

Note that there are two paths from x_0 to x_1 and x_2 in $\mathbf{T}$, and that there are two corresponding paths from q_0 to $q_1 := h(x_1)$ and $q_2 := h(x_2)$ in $\widehat{\mathbf{G}}$. Since $x_1 = x_2\, (mod\, ker\, \theta)$ and $\overline{\mathbf{G}}$ is deterministic and contains no τ_0-transitions, $q_1 \equiv q_2\, (mod\, \rho_{\mathbf{G}})$.

We show that $ker\, \theta$ is a quasi-congruence for (X, Ξ_τ), for $\tau \in T$. Let $x_1' \in \Xi_\tau(x_1)$. There is a path from x_1 to x_1' along which there is a transition with label τ and the remaining transitions have labels in $\Sigma - T$. There is a corresponding path from q_1 to $q_1' := h(x_1')$ in $\widehat{\mathbf{G}}$. Thus $q_1' \in \Delta_\tau(q_1)$.

Since $\rho_{\mathbf{G}}$ is a quasi-congruence on $\widetilde{\mathbf{G}}$, there is $q_2' \in \Delta_\tau(q_2)$ such that $q_1' \equiv q_2'\, (mod\, \rho_{\mathbf{G}})$. So there is a path from q_2 to q_2' with only one observable transition with the label τ and the rest with labels in $\Sigma - T$.

By the construction of $\widehat{\mathbf{G}}$ from a Mealy automaton of (L,θ), it follows that there is a path from x_2 to, say, x_2' such that $h(x_2') = q_2'$ and that along the path there is only one observable transition with label τ. Thus, $x_2' \in \Xi_\tau(x_2)$. Clearly $x_1' \equiv x_2' \,(mod\,(ker\,\theta))$.

This establishes that $ker\,\theta$ is a quasi-congruence for (X, Ξ_τ), where $\tau \in T$.

It is similar to show that $ker\,\theta$ is a quasi-congruence for (X, Ξ_m). □

Let $p : (\Sigma \cup T)^* \longrightarrow T^*$ be the natural projection and let $g : Q \longrightarrow Y$ be the canonical projection associated with $\rho_{\mathbf{G}}$.

Lemma B.2. *If θ is an L-observer, then $\overline{\mathbf{G}}$ is deterministic and contains no τ_0-transition.*

Proof: We prove the lemma by contradiction. Suppose that θ is an L-observer and $\overline{\mathbf{G}}$ is nondeterministic. Then there exist $y, y_1, y_2 \in Y$ and $\tau \in T$ such that $y_1 \neq y_2$ and $y_1, y_2 \in \eta(y,\tau)$. By the definition of η, there are $q, q' \in g^{-1}(y)$ such that

$$\widehat{\delta}(q,\tau) \cap g^{-1}(y_1) \neq \emptyset \text{ and } \widehat{\delta}(q',\tau) \cap g^{-1}(y_2) \neq \emptyset$$

Since $q \equiv q' \,(mod\,\rho_{\mathbf{G}})$ and $\rho_{\mathbf{G}}$ is a quasi-congruence, there exists $uw \in (\Sigma - T)^*$ such that $\widehat{\delta}(q, u\tau w) \cap g^{-1}(y_2) \neq \emptyset$. Let $q_1 \in \widehat{\delta}(q,\tau) \cap g^{-1}(y_1)$ and $q_2 \in \widehat{\delta}(q, u\tau w) \cap g^{-1}(y_2)$. Since $y_1 \neq y_2$, $q_1 \not\equiv q_2 \,mod\,\rho_{\mathbf{G}}$. Since θ is an L-observer, by Proposition 3.1 there is $v \in (\Sigma \cup T)^*$ such that

$$\begin{array}{c}[\widehat{\delta}(q_1,v) \cap Q_m \neq \emptyset \text{ and } (\forall r \in (\Sigma \cup T)^*)\, p(r) = p(v) \Longrightarrow \widehat{\delta}(q_2,r) \cap Q_m = \emptyset] \\ \text{or } [\widehat{\delta}(q_2,v) \cap Q_m \neq \emptyset \text{ and } (\forall r \in (\Sigma \cup T)^*)\, p(r) = p(v) \Longrightarrow \widehat{\delta}(q_1,r) \cap Q_m = \emptyset]\end{array} \tag{40}$$

Consider the first disjunct. Since $\widehat{\mathbf{G}}$ is trim, there is $s \in (\Sigma \cup T)^*$ such that $q \in \widehat{\delta}(q_0, s)$. So there is a path in $\widetilde{\mathbf{T}}$ with the string s leading to a state, say, $x \in X$ such that $h(x) = q$. Corresponding to the paths from q to q_1 with the string τ and from q to q_2 with the string $u\tau w$, there are two paths from x to, say x_1, with the string τ and from x to, say x_2, with the string $u\tau w$ such that $h(x_1) = q_1$ and $h(x_2) = q_2$. Clearly $x_1 \equiv x_2 \,(mod\,(ker\,\theta))$.

Since $\widehat{\delta}(q_1, v) \cap Q_m \neq \emptyset$, there is a path from x_1 to, say, $x_1' \in X_m$ with the string v. From (40), it follows that there is no path with any string $r \in (\Sigma \cup T)^*$ such that $p(r) = p(v)$ that leads to a state in X_m. Thus $ker\,\theta$ is not a quasi-congruence on $\widetilde{\mathbf{T}}$, i.e., θ is not an L-observer. This contradicts the hypothesis of the proposition.

Suppose that θ is an L-observer and $\overline{\mathbf{G}}$ contains a τ_0-transition. Then there exist $y, y' \in Y$ such that $y' \in \eta(y, \tau_0)$. From the definition of η, $y \neq y'$ and there exist $q \in g^{-1}(y)$ and $q' \in g^{-1}(y')$ such that $q' \in \widehat{\delta}(q,\sigma)$ for some $\sigma \in \Sigma - T$. Since $y \neq y'$, $q \not\equiv q' \,(mod\,\rho_{\mathbf{G}})$. As before, by Proposition 3.1 there is $v \in (\Sigma \cup T)^*$ such that

$$\widehat{\delta}(q,v) \cap Q_m \neq \emptyset \text{ and } (\forall r \in (\Sigma \cup T)^*)\, p(r) = p(v) \Longrightarrow \widehat{\delta}(q',r) \cap Q_m = \emptyset$$

It then follows as above that θ is not an L-observer, again a contradiction. □

Proof of Theorem 4.1: This follows from Lemmas B.1 and B.2. □

B.2 Theorem 4.2

For this theorem, we first establish the following lemma.

We recall that p is the natural projection from $(\Sigma \cup T)^*$ to T^*. Let $p_0 : (T \cup \{\tau_0\})^* \longrightarrow T^*$ be the natural projection. Recall that for a nondeterministic automaton $\mathbf{A} = (A, \Gamma, \mu, a_0, A_m)$, the *marked behavior* of $\mathbf{A}$ is $L_m(\mathbf{A}) := \{s \in \Gamma^* \mid \mu(a_0, s) \cap A_m \neq \emptyset\}$.

Lemma B.3. $\theta(L) = p(L_m(\widehat{\mathbf{G}})) = p_0(L_m(\overline{\mathbf{G}}))$.

Proof: We show the first equality, $\theta(L) = p(L_m(\widehat{\mathbf{G}}))$.

Let $t \in \theta(L)$, so there exists $s \in L$ such that $t = \theta(s)$. Thus, $\delta(q_0, s) \in Q_m$. By the fact that $\widehat{\mathbf{G}}$ is a relabeling of $\mathbf{G}$ with the property in (15) and the assumption (2), there is a string $u \in (\Sigma \cup T)^*$ such that $\widehat{\delta}(q_0, u) \cap Q_m \neq \emptyset$ and $p(u) = \theta(s)$. Thus $u \in L_m(\mathbf{G})$ and $t = \theta(s) = p(u) \in p(L_m(\widehat{\mathbf{G}}))$.

For the reverse inclusion, let $t \in p(L_m(\widehat{\mathbf{G}}))$, so there exists $s \in L_m(\widehat{\mathbf{G}})$ such that $t = p(s)$. Thus, $\widehat{\delta}(q_0, s) \cap Q_m \neq \emptyset$. Again by the fact that $\widehat{\mathbf{G}}$ is a relabeling of $\mathbf{G}$ with the property in (15) and the assumption (2), there is a string $u \in \Sigma^*$ such that $\delta(q_0, u) \in Q_m$ and $\theta(u) = t$. Hence, $u \in L_m(\mathbf{G}) = L$ and $t = \theta(u) \in \theta(L)$.

We now show the second equality, $p_0(L_m(\overline{\mathbf{G}})) = p(L_m(\widehat{\mathbf{G}}))$.

For the inclusion $(\subseteq)$, first we note the following.

Claim: Let $y, y' \in Y$ and $\tau \in T \cup \{\tau_0\}$ such that $y' \in \eta(y, \tau)$. Let $q' \in Q$ and $u' \in (\Sigma \cup T)^*$ such that $g(q') = y'$ and $\widehat{\delta}(q', u') \cap Q_m \neq \emptyset$. Then there exist $q \in Q$ and $u \in (\Sigma \cup T)^*$ such that $g(q) = y$, $\widehat{\delta}(q, u) \cap Q_m \neq \emptyset$, $(\tau \in T \Longrightarrow p(u) = \tau p(u'))$ and $(\tau = \tau_0 \Longrightarrow p(u) = p(u'))$.

Proof of the claim: From the definition of η and the fact that $y' \in \eta(y, \tau)$, there exist $q \in g^{-1}(y)$ and $q'' \in g^{-1}(y')$ such that $q'' \in \widehat{\delta}(q, \tau)$. Thus, $q' \equiv q'' \,(mod\, \rho_{\mathbf{G}})$. The claim then follows from Proposition 3.1.

Let $s \in p_0(L_m(\overline{\mathbf{G}}))$. Then there exists $u \in L_m(\overline{\mathbf{G}})$ such that $s = p_0(u)$. So $u = \tau_0^k$, for some $k \geq 0$, or $u = \tau_0^{k_1} \tau_1 \tau_0^{k_2} \tau_2 \cdots \tau_0^{k_n} \tau_n \tau_0^{k_n+1}$, for some $n \geq 1$, $\tau_i \in T$ and $k_i \geq 0$. In either case, by using the claim and Proposition 3.1 we conclude that there is a string $v \in (\Sigma \cup T)^*$ such that $\widehat{\delta}(q_0, v) \cap Q_m \neq \emptyset$ and $s = p(v)$. Thus $s = p(v) \in p(L_m(\widehat{\mathbf{G}}))$.

$(\supseteq)$ Let $s \in p(L_m(\widehat{\mathbf{G}}))$. So there exists $u \in L_m(\widehat{\mathbf{G}})$ such that $s = p(u)$. Thus $\widehat{\delta}(q_0, u) \cap Q_m \neq \emptyset$. It follows from the construction of $\overline{\mathbf{G}}$ that there is a string $v \in (T \cup \{\tau_0\})^*$ such that $\eta(y_0, v) \cap Y_m \neq \emptyset$ and $p_0(v) = s$. Thus $s = p_0(v) \in p_0(L_m(\overline{\mathbf{G}}))$. □

Proof of Theorem 4.2: (i) From Theorem 4.1 and Lemma B.3, we have that $\overline{\mathbf{G}}$ is deterministic, free of τ_0-transitions and recognizes $\theta(L)$.

To prove that $\overline{\mathbf{G}}$ is minimal, it sufficient to show that, for $s, s' \in \overline{\theta(L)}$,

$$s \equiv_{\theta(L)} s' \text{ iff } \eta(y_0, s) = \eta(y_0, s')$$

Let $s, s' \in \overline{\theta(L)}$. Since $s, s' \in \overline{\theta(L)} = \theta(\overline{L})$, there are $u, u' \in \overline{L}$ such that $s = \theta(u)$ and $s' = \theta(u')$. Let $q := \delta(q_0, u)$ and $q' := \delta(q_0, u')$, where δ is the transition function of the Mealy automaton $\mathbf{G}$ (see (23)). Since $\overline{\mathbf{G}}$ is deterministic and free of τ_0-transitions, it can be shown that $\eta(y_0, s) = g(q)$ and $\eta(y_0, s') = g(q')$, where g is the canonical projection associated with $\rho_{\mathbf{G}}$.

Suppose that $s \equiv_{\theta(L)} s'$. To prove $\eta(y_0, s) = \eta(y_0, s')$, it is sufficient to show that

$$q \equiv q' \,(mod\, \rho_{\mathbf{G}})$$

Since θ is an L-observer, by Proposition 3.1 it is sufficient to show that

$$(\forall t \in T^*)\, \Delta_t(q) \cap Q_m \neq \emptyset \Longleftrightarrow \Delta_t(q') \cap Q_m \neq \emptyset$$

Let $t \in T^*$ such that $\Delta_t(q) \cap Q_m \neq \emptyset$. So there is a string $v \in \Sigma^*$ such that $uv \in L$ and $\theta(uv) = \theta(u)t$. Thus $st \in \theta(L)$. Since $s \equiv_{\theta(L)} s'$, $s't \in \theta(L)$. Thus $\theta(u')t \in \theta(L)$. Since θ is an L-observer, there is $v' \in \Sigma^*$ such that $u'v' \in L$ and $\theta(u'v') = \theta(u')t$. It follows that $\Delta_t(q') \cap Q_m \neq \emptyset$. By symmetry we have the reverse implication. So by Proposition 3.1, $q \equiv q' \, (mod\, \rho_{\mathbf{G}})$.

Now suppose that $\eta(y_0, s) = \eta(y_0, s')$. Thus, $g(q) = g(q')$, i.e. $q \equiv q' \, (mod\, \rho_{\mathbf{G}})$. Let $t \in T^*$ such that $st \in \theta(L)$. Since θ is an L-observer, there is $v \in \Sigma^*$ such that $uv \in L$ and $\theta(uv) = \theta(u)t$. Thus $\Delta_t(q) \cap Q_m \neq \emptyset$. Since $q \equiv q' \, (mod\, \rho_{\mathbf{G}})$, by Proposition 3.1 $\Delta_t(q') \cap Q_m \neq \emptyset$. Thus, there is $v' \in \Sigma^*$ such that $u'v' \in L$ and $\theta(u'v') = \theta(u')t$. Hence $s't \in \theta(L)$. The other implication follows by symmetry. Therefore $s \equiv_{\theta(L)} s'$.

(ii) This follows from the fact that $\rho_{\mathbf{G}}$ can be computed in polynomial time (see Appendix D).
□

C Proofs in Section 6

C.1 Lemma 6.3

Proof: We first show that

$$ker\, \theta_{i+1} \leq ker\, \theta_i = \pi_i \tag{41}$$

The equality follows from the induction hypothesis. Let $x, x' \in X$ such that $x \equiv x' \, (mod\, (ker\, \theta_{i+1}))$; we show $x \equiv x' \, mod\, ker\, \theta_i$.

Let $s, s' \in (\Sigma \cup T_{i+1})^*$ be the (unique) strings of events from x_0 to x and to x' respectively. Let $p_{i+1} : (\Sigma \cup T_{i+1})^* \longrightarrow (T_{i+1})^*$ be the natural projection. Since $x \equiv x' \, (mod\, ker\, \theta_{i+1})$, $p_{i+1}(s) = p_{i+1}(s')$.

Consider the following two cases.

(1) $s, s' \in (\Sigma - T_{i+1})^*$. Since any transition in $\widehat{\mathbf{T}}_i$ with an observable event label in T_i remains observable in $\widehat{\mathbf{T}}_{i+1}$, with either the same event label or a new observable event label (cf. Algorithm 5.1), it follows that the transitions along the paths from x_0 to x and from x_0 to x' are all unobservable in $\widehat{\mathbf{T}}_i$, i.e., $s, s' \in (\Sigma - T_i)^*$. Thus, $x \equiv x' \, (mod\, (ker\, \theta_i))$.

(2) $s = s_1\tau_1 s_2\tau_2 \cdots s_n\tau_n s_{n+1}$, $s' = s'_1\tau_1 s'_2\tau_2 \cdots s'_n\tau_n s'_{n+1}$, where $n \geq 1$, $\tau_j \in T_1$ for $1 \leq j \leq n$ and $s_j, s'_j \in (\Sigma - T_{i+1})^*$ for $1 \leq j \leq n+1$. By the argument (1) above, $s_j, s'_j \in (\Sigma - T_i)^*$. By Lemma 5.1 (2) the original labels (in $\Sigma \cup T_i$) of two transitions that have the same label (in T_{i+1}) after relabeling are either the same observable label (in T_i) or both unobservable (i.e. in $\Sigma - T_i$). It follows that $x \equiv x' \, (mod\, (ker\, \theta_i))$.

We are now in position to establish the lemma. Recall that $\pi_{i+1} = sup\{\pi \in \mathcal{E}(X) \,|\, \pi \leq \pi'_{i+1} \wedge (\pi \circ pre)\}$.

To show that $ker\, \theta_{i+1} \leq \pi_{i+1}$, let $x, x' \in X$ such that $x \equiv x' \, (mod\, ker\, \theta_{i+1})$.

Let $s, s' \in (\Sigma \cup T_{i+1})^*$ be the strings of events from x_0 to x and to x' respectively. It follows from (34) that the strings s and s' lead from q_0 to $h(x)$ and to $h(x')$ respectively.

Let $p_{i+1} : (\Sigma \cup T_{i+1})^* \longrightarrow T^*_{i+1}$ be the natural projection. Since $x \equiv x' \, (mod\, ker\, \theta_{i+1})$, $p_{i+1}(s) = p_{i+1}(s')$.

By Lemma 5.1 (1), $\mathbf{G}_{i+1}/(T_{i+1}, \rho_{i+1,Q})$ is deterministic and free of τ_0-transitions. It follows from this and the fact $p_{i+1}(s) = p_{i+1}(s')$ that $h(x) \equiv h(x') \, (mod\, \rho_{i+1,Q})$. Since h is a homomorphism from $\widetilde{\mathbf{T}}_i$ to $\widetilde{\mathbf{G}}_i$, $h_\times(\pi_{i+1,X}) = \rho_{i+1,Q}$ and $h^\times(\rho_{i+1,Q}) = \pi_{i+1,X}$. Thus, $x \equiv x' \, (mod\, \pi_{i+1,X})$.

By (41), $x \equiv x' \, (mod\, (\pi_{i+1,X} \wedge ker\, \theta_i))$. By the induction hypothesis $(\pi_{i+1,X} \wedge \pi_i = \pi'_{i+1})$, $x \equiv x' \, (mod\, \pi'_{i+1})$, so $ker\, \theta_{i+1} \leq \pi'_{i+1}$.

It is clear that $ker\, \theta_{i+1}$ is a quasi-congruence for (X, pre).

Thus, $ker\,\theta_{i+1} \leq \pi_{i+1}$.

To show the reverse implication, let $x, x' \in X$ such that $x \equiv x'\,(mod\,\pi_{i+1})$.

Since π_{i+1} is a quasi-congruence for (X, pre) (cf. 38), there are $n \geq 0$ and $x_1, \cdots, x_n \in X$ such that

$$x_0 < x_1 < x_2 < \cdots < x_n < x$$

where x_0, x_1, x_2, $\cdots$, x_n are in distinct equivalence classes of π_{i+1}, and for all $0 \leq j < n$ and $x'' \in X$ with $x_j \leq x'' < x_{j+1}$, $x_j \equiv x''\,(mod\,\pi_{i+1})$, and for $x'' \in X$ with $x_n \leq x'' \leq x'$, $x_n \equiv x''\,(mod\,\pi_{i+1})$.

Let $x'_1, x'_2, \cdots, x'_m \in X$ be the similar states for x', where $m \geq 0$. Thus

$$x_0 < x'_1 < x'_2 < \cdots < x'_m < x'$$

Again since π_{i+1} is a quasi-congruence for (X, pre), $m = n$ and $x_j \equiv x'_j\,(mod\,\pi_{i+1})$ for $j = 0, 1, 2, \cdots, n$.

Let $\sigma_j, \sigma'_j \in (\Sigma \cup T_i)$ be the labels of the transitions in $\widehat{\mathbf{T}}_i$ entering x_j and x'_j respectively, for $0 \leq j \leq n$.

Since $\pi_{i+1} \leq ker\,\theta_i$, the transitions, that are along the paths from x_0 to x and from x_0 to x' and have their entrance and exit states in the same equivalence class of π_{i+1}, have unobservable labels, and σ_j, σ'_j are either both unobservable or $\sigma_j = \sigma'_j \in T_i$ for $j = 0, 1, \cdots, n$.

Let (y, σ, y') be a transition along the paths from x_0 to x or from x_0 to x' with y and y' in the same equivalence class of π_{i+1}. Thus $\sigma \in (\Sigma - T_i)$, i.e. σ is unobservable.

Since $\pi_{i+1} \leq \pi'_{i+1}$ and $\pi'_{i+1} = \pi_{i+1,X} \wedge ker\,\theta_i$ (induction hypothesis), y and y' are equivalent modulo $\pi_{i+1,X}$. By the fact that $h_\times(\pi_{i+1,X}) = \rho_{i+1,Q}$, $h(y)$ and $h(y')$ are equivalent modulo $\rho_{i+1,Q}$. Since $\sigma \in \Sigma - T_i$, by Property 3(i) of the relabeling algorithm the transition $(h(y), \sigma, h(y'))$ is not relabeled. So, neither is (y, σ, y'). Thus, all the transitions that are along the paths from x_0 to x and from x_0 to x' and have their entrance and exit states equivalent modulo π_1 will not be relabeled, and hence their labels remain unobservable.

Let $(\widehat{x_j}, \sigma_j, x_j)$ and $(\widehat{x_j}', \sigma'_j, x'_j)$ be the transitions entering x_j and x'_j respectively, for some $j \leq n$. Since $\widehat{x_j} \equiv \widehat{x_j}'\,(mod\,\pi_{i+1})$ and $x_j \equiv x'_j\,(mod\,\pi_{i+1})$, as above $\widehat{x_j} \equiv \widehat{x_j}'\,(mod\,\pi_{i+1,X})$ and $x_j \equiv x'_j\,(mod\,\pi_{i+1,X})$. By $h_\times(\pi_{i+1,X}) = \rho_{i+1,Q}$, $h(\widehat{x_j}) \equiv h(\widehat{x_j}')\,(mod\,\rho_{i+1,Q})$ and $h(x_j) \equiv h(x'_j)\,(mod\,\rho_{i+1,Q})$. By Property 3($ii$) of the relabeling algorithm, if the transitions $(h(\widehat{x_j}), \sigma_j, h(x_j))$ and $(h(\widehat{x_j}'), \sigma'_j, h(x'_j))$ are relabeled, they are relabeled with the same event label.

It follows that $x \equiv x'\,(mod\,ker\,\theta_{i+1})$. □

C.2 Lemma 6.4

Proof: We note that $\pi_{i+1} = ker\,\theta_{i+1}$. It is clear that $\pi'_{i+2} \leq \pi_{i+2,X}$ and $\pi'_{i+2} \leq ker\,\theta_{i+1}$. So $\pi'_{i+2} \leq \pi_{i+2,X} \wedge (ker\,\theta_{i+1}) = \pi_{i+2,X} \wedge \pi_{i+1}$.

For the reverse refinement, it suffices to show that $\pi_{i+2,X} \wedge \pi_{i+1}$ is a quasi-congruence on $\widetilde{\mathbf{T}}_{i+1}$. Let $x_1, x_2 \in X$ such that $x_1 \equiv x_2\,(mod\,(\pi_{1,X} \wedge (ker\,\theta)))$. Then $x_1 \equiv x_2\,(mod\,\pi_{i+2,X})$ and $x_1 \equiv x_2\,(mod\,\pi_{i+1})$. Let $\tau \in T_{i+1}$ and $x'_1 \in \Xi_\tau(x_1)$. Since $\pi_{i+2,X}$ is a quasi-congruence on $\widetilde{\mathbf{T}}_{i+1}$, there is $x'_2 \in \Xi_\tau(x_2)$ such that $x'_1 \equiv x'_2\,(mod\,\pi_{i+2,X})$. Since $x_1 \equiv x_2\,(mod\,\pi_{i+1})$, $x'_1 \equiv x'_2\,(mod\,\pi_{i+1})$. Thus $x'_1 \equiv x'_2\,(mod\,(\pi_{i+2,X} \wedge \pi_{i+1}))$.

By symmetry we have that $(\forall x'_2 \in \Xi_\tau(x_2))(\exists x'_1 \in \Xi_\tau(x_1))\, x'_1 \equiv x'_2\,(mod\,(\pi_{i+2,X} \wedge \pi_{i+1}))$. It is similar to establish the quasi-congruence property for Ξ_m. Thus, we have shown that $\pi_{i+2,X} \wedge \pi_{i+1}$ is a quasi-congruence on $\tilde{\mathbf{T}}_{i+1}$. The lemma is established. □

C.3 Lemma 6.6

Proof: We first note the following. Since both π_{Mealy} and π^* are quasi-congruences on $\widetilde{\mathbf{T}}_i$ (induction hypothesis), so is their join $\pi_{Mealy} \vee \pi^*$. Since h is a homomorphism from $\widetilde{\mathbf{T}}_i$ to $\widetilde{\mathbf{G}}_i$, by Fact 2.1, $\rho_Q^* := h_\times(\pi_{Mealy} \vee \pi^*)$ is a quasi-congruence on $\widetilde{\mathbf{G}}_i$. Since $\rho_{i+1,Q}$ is the coarsest quasi-congruence on $\widetilde{\mathbf{G}}_i$,

$$\rho_Q^* \leq \rho_{i+1,Q}$$

Let $x_1, x_2 \in X$ such that $x_1 \equiv x_2 \,(mod\, \pi^*)$. Let $\tau \in T_{i+1}$ and let $x_1^1 \in \Xi_\tau^{i+1}(x_1)$. Thus there is $uu^1 \in (\Sigma - T_{i+1})^*$ such that $x_1^1 \in \widehat{\xi}_{i+1}(x_1, u\tau u^1)$, so there is a path from x_1 to x_1^1 in $\widehat{\mathbf{T}}_{i+1}$ with the string $u\tau u^1$.

Since an observable transition in $\widehat{\mathbf{T}}_i$ remains observable in $\widehat{\mathbf{T}}_{i+1}$, $uu^1 \in (\Sigma - T_i)^*$ and there is a path from x_1 to x_1^1 in $\widehat{\mathbf{T}}_i$ with the string $u\tau^1 u^1$, for some $\tau^1 \in \Sigma \cup T_i$.

Consider the following two cases:

Case (1) $\tau^1 \in T_i$:

Thus $x_1^1 \in \Xi_{\tau^1}^i(x_1)$. Since π^* is a quasi-congruence on $\widetilde{\mathbf{T}}_i$, there is $x_2^1 \in \Xi_{\tau^1}^i(x_2)$ such that $x_1^1 \equiv x_2^1 \,(mod\, \pi^*)$. Thus there is a path from x_2 to x_2^1 in $\widehat{\mathbf{T}}_i$ with the string $v\tau^1 v^1$, where $vv^1 \in (\Sigma - T_i)^*$. The strings can be displayed as in Figure 14. Let x_1^2,x_1^3,x_2^2 and x_2^3 be as defined in Figure 14.

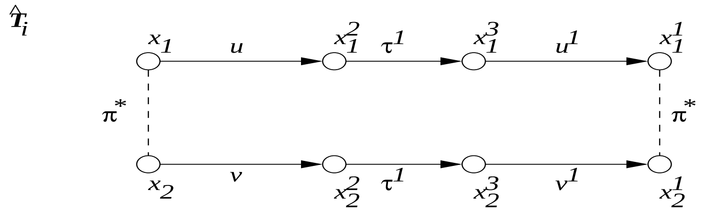

Figure 14: A display of relations among various states in $\widehat{\mathbf{T}}_i$.

Since $\pi^* \leq \pi_i = ker\, \theta_i$ (induction hypothesis), x_1^2 cannot be equivalent modulo π^* to any state along $x_2^3 \xrightarrow{v^1} x_2^1$ (and to any state along $x_1^3 \xrightarrow{u^1} x_1^1$). Since π^* is a quasi-congruence for (X, pre), x_1^2 is equivalent to a state along $x_2 \xrightarrow{v\tau^1 v^1} x_2^1$. By the previous argument it is equivalent modulo π^* to a state along $x_2 \xrightarrow{v} x_2^2$. Again, using the fact that π^* is a quasi-congruence for (X, pre), $x_1^2 \equiv x_2^2 \,(mod\, \pi^*)$.

Similarly one can establish that $x_1^3 \equiv x_2^3 \,(mod\, \pi^*)$.

Since π^* is a quasi-congruence for (X, pre), states along $x_1 \xrightarrow{u} x_1^2$ have their corresponding equivalent $(mod\, \pi^*)$ states along $x_2 \xrightarrow{v} x_2^2$ and vice versa. A similar statement can be made for $x_1^3 \xrightarrow{u^1} x_1^1$ and $x_2^3 \xrightarrow{v^1} x_2^1$.

Let $q_i := h(x_i)$, $q_i^1 := h(x_i^1)$, $q_i^2 := h(x_i^2)$ and $q_i^3 := h(x_i^3)$, where $i = 1, 2$. Since $x_1 \equiv x_2 \,(mod\, \pi^*)$, $x_1 \equiv x_2 \,(mod\, (\pi_{Mealy} \vee \pi^*))$. So $q_1 \equiv q_2 \,(mod\, \rho_Q^*)$. Similar conclusions can be drawn for the other pairs of states. The relationships of these states are displayed in Figure 15.

Since $\rho_Q^* \leq \rho_{i+1,Q}$, the relations displayed in Figure 16 hold.

Since $uu^1 \in (\Sigma - T_{i+1})^*$, by the property of the relabeling algorithm (i.e., there is no τ_0-transition in $\widehat{\mathbf{G}}_{i+1}/(T_{i+1}, \rho_{i+1,Q})$ (cf. Lemma 5.1)), all the states along $q_1 \xrightarrow{u} q_1^2$ are equivalent modulo $\rho_{i+1,Q}$ and so are the states along $q_1^3 \xrightarrow{u^1} q_1^1$. Thus we have the situation in Figure 17.

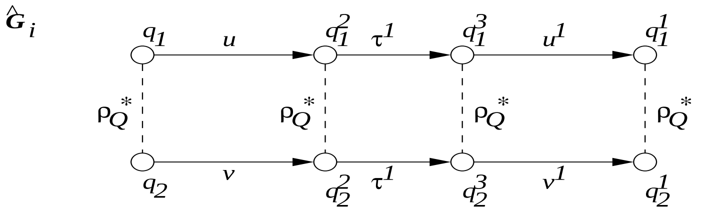

Figure 15: A display of relations among $q_i, \cdots, q_i^3$ in $\widehat{\mathbf{G}}_i$.

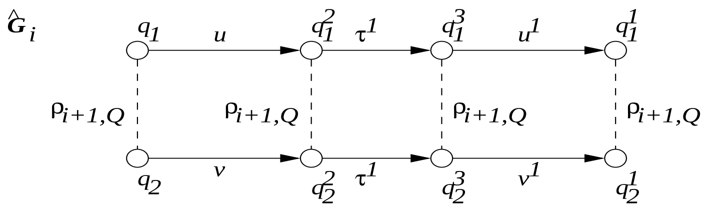

Figure 16: Another display of relations among $q_i, \cdots, q_i^3$ in $\widehat{\mathbf{G}}_i$.

By the properties of the relabeling algorithm $3(i)$ and $3(ii)$, all the transitions along $q_2 \xrightarrow{v} q_2^2$ and along $q_2^3 \xrightarrow{v^1} q_2^1$ have the unobservable event labels in $\Sigma - T_{i+1}$, and (q_2^2, τ^1, q_2^3) is also relabeled with the label τ.

Thus, by relabeling done correspondingly on $\widehat{\mathbf{T}}_i$ (cf. (30)) $x_2^1 \in \Xi_\tau^{i+1}(x_2)$ and $x_1^1 \equiv x_2^1 \,(mod\, \pi^*)$.
Case (2) $\tau^1 \in \Sigma - T_i$:
Since $\mathbf{T}_i$ is trim, there is a path from x_1^1 to a state, say $\widehat{x}_1$, in X_m. Since π^* is a quasi-congruence on $\widetilde{\mathbf{T}}_i$, there is a path from x_2 to a state, say $\widehat{x}_2$, in X_m such that $\widehat{x}_1 \equiv \widehat{x}_2 \,(mod\, \pi^*)$. Since π^* is a quasi-congruence for (X, pre), there is a state $x_2^1 \in X$ such that $x_2 \leq x_2^1 \leq \widehat{x}_2$ and $x_1^1 \equiv x_2^1 \,(mod\, \pi^*)$. Since $\pi^* \leq ker\, \theta_i$ and the string $u\tau^1 u^1 \in (\Sigma - T_i)^*$, the string from x_2 to x_2^1 consists of event labels in $\Sigma - T_I$. Call this string $\widehat{v}$. So $\widehat{v} \in (\Sigma - T_i)^*$. The states are as displayed in Figure 18. Let x_1^2 and x_1^3 be as in Figure 18.

We note that $x_1^2 \not\equiv x_1^3 \,(mod\, \pi^*)$. Suppose that $x_1^2 \equiv x_1^3 \,(mod\, \pi^*)$. Then $h(x_1^2) = h(x_1^3) \,(mod\, \rho_Q^*)$. Since $\rho_Q^* \leq \rho_{i+1,Q}$, $h(x_1^2) \equiv h(x_1^3) \,(mod\, \rho_{i+1,Q})$. Since $\tau^1 \in \Sigma - T_i$, according to the property $3(i)$ of Algorithm 5.1, $(h(x_1^2), \tau', h(x_1^3))$ will not be relabeled. Thus, (x_1^2, τ^1, x_1^3) will not be relabeled.

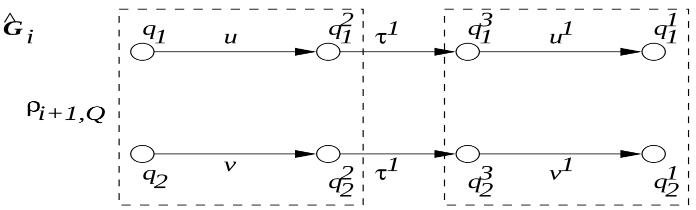

Figure 17: A display of the states $q_i, \cdots, q_i^3$ and the partition $\rho_{i+1,Q}$.

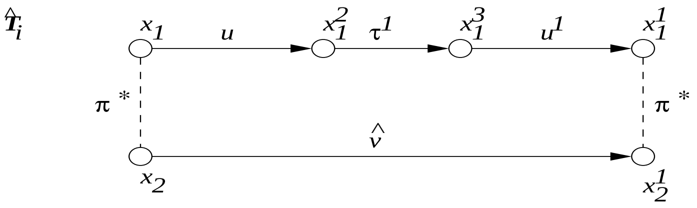

Figure 18: A display of states in $\widehat{\mathbf{T}}_i$.

This contradicts to the fact that (x_1^2, τ^1, x_1^3) is relabeled to (x_1^2, τ, x_1^3). So $x_1^2 \not\equiv x_1^3\,(mod\,\pi^*)$, after all.

Since π^* is a quasi-congruence for (X, pre), it follows that $\widehat{v} > \epsilon$ and we have the situation in Figure 19.

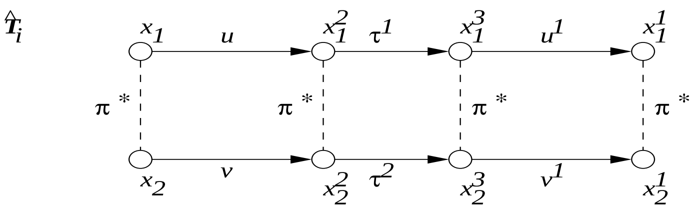

Figure 19: A display of states in $\widehat{\mathbf{T}}_i$.

By the same argument as in case (1) above, we have the situation in Figure 20, where $q_i := h(x_i)$,

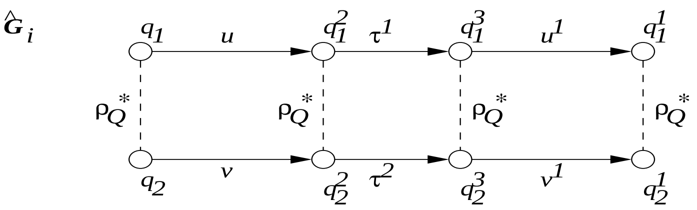

Figure 20: A display of states in $\widehat{\mathbf{G}}_i$.

$q_i^1 := h(x_i^1)$, $q_i^2 := h(x_i^2)$ and $q_i^3 := h(x_i^3)$, for $i = 1, 2$. Since $\rho_Q^* \le \rho_{i+1,Q}$, the situation is as displayed in Figure 21.

Note that since $\tau^1, \tau^2 \in \Sigma - T_i$ and relabeling is done for (q_1^2, τ^1, q_1^3), $q_1^2 \not\equiv q_1^3\,(mod\,\rho_{i+1,Q})$ and $q_2^2 \not\equiv q_2^3\,(mod\,\rho_{i+1,Q})$. By the relabeling algorithm, $vv^1 \in (\Sigma - T_{i+1})^*$ and (q_2^2, τ^2, q_2^3) is relabeled to (q_2^2, τ, q_2^3). The corresponding relabeling is done on $\widehat{\mathbf{T}}_i$ (cf. (30)). It follows that $x_2^1 \in \Xi_\tau^{i+1}(x_2)$ and $x_1^1 \equiv x_2^1\,(mod\,\pi^*)$.

Thus we establish that π^* is a quasi-congruence for $(X, \{\Xi_\tau^{i+1} \mid \tau \in T_{i+1}\})$.

To show that π^* is a quasi-congruence for (X, Ξ_m^{i+1}), let $x_1^1 \in \Xi_m^{i+1}(x_1)$. Thus $x_1^1 \in X_m$ and there is $u \in (\Sigma - T_{i+1})^*$ such that $x_1^1 \in \xi_{i+1}(x_1, u)$. So there is a path from x_1 to x_1^1 in $\widehat{\mathbf{T}}_{i+1}$ with

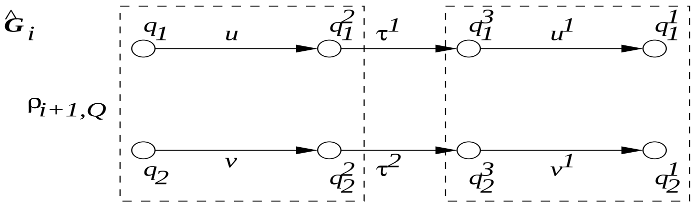

Figure 21: Another display of states in $\widehat{\mathbf{G}}$.

the string u. Since relabeling is done with new observable events, $u \in (\Sigma - T_i)^*$.

Since π^* is a quasi-congruence for (X, Ξ_m^i), there is a path from x_2 to x_2^1, a state in X_m, with a string v in $(\Sigma - T_i)^*$, and $x_1^1 \equiv x_2^1 \, (mod\, \pi^*)$.

By a previous argument, we have that $h(x_1) \equiv h(x_2)$ and $h(x_1^1) \equiv h(x_2^1)$ modulo ρ_Q^* and that states along $h(x_1) \xrightarrow{u} h(x_1^1)$ are equivalent modulo ρ_Q^* to states along $h(x_2) \xrightarrow{v} h(x_2^1)$. Since $\widehat{\mathbf{G}}_{i+1}/(T_{i+1}, \rho_{i+1,Q})$ has no τ_0-transition, all the states along $h(x_1) \xrightarrow{u} h(x_1^1)$ are equivalent modulo $\rho_{i+1,Q}$. Since $\rho_Q^* \leq \rho_{i+1,Q}$, all the states along $h(x_2) \xrightarrow{v} h(x_2^1)$ are equivalent modulo $\rho_{i+1,Q}$.

By the property $3(i)$ of the relabeling algorithm (Algorithm 5.1), $v \in (\Sigma - T_{i+1})^*$. So $x_2^1 \in \Xi_m^{i+1}(x_2)$.

This establishes the lemma. $\square$

D A partition refinement algorithm

In this appendix we present some of the basics of the partition refinement algorithm of [Fer90], with the required modification for the marker states in our models.

We continue with the development in Section 3. Following [Fer90], we represent an equivalence relation ρ on Q as a partition $\rho = \{B_1, \cdots, B_n\}$ for some $n \geq 1$, where each B_i represents one of its equivalence classes. Given a partition $\rho = \{B_i \mid i \in I\}$ on Q, ρ is *compatible* with $(\{\Delta_\tau \mid \tau \in T\}, \Delta_m)$ [Fer90] if

$$(\forall i, j \in I)(\forall q, q' \in B_i)\{[(\forall \tau \in T)\,(\Delta_\tau(q) \cap B_j \neq \emptyset \iff \Delta_\tau(q') \cap B_j \neq \emptyset)] \\ \text{and } [\Delta_m(q) \cap B_j \neq \emptyset \iff \Delta_m(q') \cap B_j \neq \emptyset)]\}$$

A partition is compatible with $(\{\Delta_\tau \mid \tau \in T\}, \Delta_m)$ if and only if it is a quasi-congruence.

We make the following definition: For $\tau \in T_m := T \cup \{m\}$, define $\Phi_\tau : Q \longrightarrow \mathcal{P}(Q)$ as follows. For $q \in Q$,

$$\Phi_\tau(q) = \begin{cases} \{q' \in Q \mid q \in \Delta_\tau(q')\} & \text{if } \tau \neq m, \\ \{q' \in Q \mid q \in \Delta_m(q')\} & \text{if } \tau = m. \end{cases} \tag{42}$$

Thus, $\Phi_\tau(q)$ is the set of states from which it is possible to reach q with Δ_τ or Δ_m.

Extend Φ_τ to $\Phi_\tau : \mathcal{P}(Q) \longrightarrow \mathcal{P}(Q)$ as follows. For $B \subseteq Q$,

$$\Phi_\tau(B) := \bigcup\{\Phi_\tau(q) \mid q \in B\}$$

We note that ρ is compatible with $(\{\Delta_\tau \mid \tau \in T\}, \Delta_m)$ if and only if

$$(\forall \tau \in T_m)(\forall B, B' \in \rho)\, B' \cap \Phi_\tau(B) = \emptyset \vee B' \subseteq \Phi_\tau(B)$$

where $\vee$ represents logical *or*.

We introduce the following properties Π, Π_B and $\Pi_{\tau,B}$ for $\tau \in T_m$ and $B \subseteq Q$:

$$\begin{aligned}
&\Pi_{\tau,B}(\rho) = (\forall X \in \rho)(X \cap \Phi_\tau(B) = \emptyset \vee X \subseteq \Phi_\tau(B)),\\
&\Pi_B(\rho) = (\forall \tau \in T_m)\, \Pi_{\tau,B}(\rho),\\
&\Pi(\rho_1, \rho_2) = (\forall B \in \rho_1)\, \Pi_B(\rho_2).
\end{aligned}$$

For $\tau \in T_m$ and $B \subseteq Q$, define the operator $\Phi_{\tau,B}$ as follows. For $\rho \in \mathcal{E}(Q)$,

$$\Phi_{\tau,B}(\rho) := \{X \cap \Phi_\tau(B) \mid X \in \rho\} \cup \{X - \Phi_\tau(B) \mid X \in \rho\} \tag{43}$$

The following properties can be established.

Proposition D.1 ([Fer90]). *Let $\tau, \tau_1, \tau_2 \in T_m$, $B, B_1, B_2 \subseteq Q$ and $\rho, \rho_1, \rho_2 \in \mathcal{E}(Q)$.*

(i) $\Phi_{\tau,B}(\rho) \in \mathcal{E}(Q)$,
(ii) *monotonicity:* $\rho_1 \le \rho_2 \Longrightarrow \Phi_{\tau,B}(\rho_1) \le \Phi_{\tau,B}(\rho_2)$,
(iii) $\Phi_{\tau,B}(\rho) \le \rho$,
(iv) $\Phi_{\tau_1,B_1} \circ \Phi_{\tau_2,B_2} = \Phi_{\tau_2,B_2} \circ \Phi_{\tau_1,B_1}$,
(v) $\Pi_{\tau,B}(\Phi_{\tau,B}(\rho))$,
(vi) $\Pi_{\tau,B}(\rho) \Longleftrightarrow \Phi_{\tau,B}(\rho) = \rho$.

$\square$

By Proposition D.1, the sequence of refinements with respect to $B \subseteq Q$ and $\tau \in T_m$ may be computed in any order. Let $T_m = \{\tau_1, \tau_2, \cdots, \tau_n\}$ for some $n \ge 1$ and $B \subseteq Q$. Define the operator Φ_B as follows:

$$\Phi_B = \Phi_{\tau_1,B} \circ \cdots \circ \Phi_{\tau_n,B} \tag{44}$$

Proposition D.2 ([Fer90]). *Let $B, B_1, B_2 \subseteq Q$ and $\rho, \rho_1, \rho_2 \in \mathcal{E}(Q)$.*

(i) $\Phi_B(\rho) \in \mathcal{E}(Q)$,
(ii) *monotonicity:* $\rho_1 \le \rho_2 \Longrightarrow \Phi_B(\rho_1) \le \Phi_B(\rho_2)$,
(iii) $\Phi_B(\rho) \le \rho$,
(iv) $\Phi_{B_1} \circ \Phi_{B_2} = \Phi_{B_2} \circ \Phi_{B_1}$,
(v) $\Pi_B(\Phi_B(\rho))$,
(vi) $\Pi_B(\rho) \Longleftrightarrow \Phi_B(\rho) = \rho$,
(vii) $\Phi_{B_1} \circ \Phi_{B_2}(\rho) \le \Phi_{B_1 \cup B_2}(\rho)$,
(viii) $\Phi_{B_1} \circ \Phi_{B_2} \circ \Phi_{B_1 \cup B_2} = \Phi_{B_1} \circ \Phi_{B_2}$.

$\square$

Let $\rho = \{B_i \mid 1 \le i \le n\}$ be a family of subsets of Q for some $n \ge 1$, and ρ' be a partition of Q. Define an operator Φ as follows:

$$\Phi(\rho, \rho') = (\Phi_{B_1} \circ \cdots \circ \Phi_{B_n})(\rho') \tag{45}$$

Proposition D.3 ([Fer90]). *Let $B \subseteq Q$ and $\rho, \rho_1, \rho_2 \in \mathcal{E}(Q)$.*

(i) $\Phi(\rho_1, \rho_2) \in \mathcal{E}(Q)$,
(ii) *left-monotonicity:* $\rho_1 \le \rho_2 \Longrightarrow \Phi(\rho_1, \rho) \le \Phi(\rho_1, \rho)$,
(iii) *right-monotonicity:* $\rho_1 \le \rho_2 \Longrightarrow \Phi(\rho, \rho_1) \le \Phi(\rho, \rho_2)$,
(iv) $\Phi(\rho_1, \rho_2) \le \rho_2$,
(v) $\Pi(\rho_1, \Phi(\rho_1, \rho_2))$,
(vi) $\Pi(\rho_1, \rho_2) \Longleftrightarrow \Phi(\rho_1, \rho_2) = \rho_2$,
(vii) *if* $B \in \rho_1$ *and* $\Pi_B(\rho_2)$, *then* $\Phi(\rho_1, \rho_2) = \Phi(\rho_1 - \{B\}, \rho_2)$.

$\square$

Define an operator $\tilde{\Phi}$ on the set of partitions on Q as follows: For a partition ρ on Q,

$$\tilde{\Phi}(\rho) := \Phi(\rho, \rho).$$

Proposition D.4 ([Fer90]). *Let $\rho_1, \rho_2 \in \mathcal{E}(Q)$.*

(i)	*monotonicity:* $\rho_1 \leq \rho_2 \Longrightarrow \tilde{\Phi}(\rho_1) \leq \tilde{\Phi}(\rho_2)$,
(ii)	*if $\Phi_\tau, \tau \in T_m$, is image-finite, then $\tilde{\Phi}$ is downward continuous,*
(iii)	*given an initial partition ρ, the maximal fixpoint of $\tilde{\Phi}$, i.e., the coarsest partition compatible with $(\{\Delta_\tau \mid \tau \in T\}, \Delta_m)$ and ρ, is the limit of the sequence:* $\rho_0 = \rho, \rho_{r+1} = \tilde{\Phi}(\rho_r)$
(vi)	*the maximal fixpoint of $\tilde{\Phi}$ with $\rho_0 = \{Q\}$ is the same as $\rho_{\mathbf{G}}$.*

where $\rho_{\mathbf{G}}$ is defined in (24). □

We now develop an algorithm for computing the maximal fixpoint of $\tilde{\Phi}$, as the limit of the following sequence.

$$\begin{aligned} &\rho_0 = \rho_I \\ &\rho_{r+1} = \Phi(\rho_r, \rho_r) \end{aligned} \tag{46}$$

where ρ_I is the initial partition.

By Proposition D.3, $\Pi(\rho_r, \rho_{r+1})$. Thus, the blocks in ρ_r will not 'split' ρ_{r+1}. Hence, one can remove those blocks of ρ_r that remain in ρ_{r+1}.

Let $W_{r+1} := \rho_{r+1} - \rho_r$. By Proposition D.3, the following sequence has the same limit as that of (46):

$$\begin{aligned} &\rho_0 = \rho_I \\ &W_0 = \rho_I \\ &\rho_{r+1} = \Phi(W_r, \rho_r) \\ &W_{r+1} = \rho_{r+1} - \rho_r \end{aligned}$$

We can compute this sequence with the following algorithm.

$$\begin{aligned} &W, \rho = \rho_I, \rho_I \\ &\text{repeat} \\ &\quad W, \rho = \Phi(W, \rho) - \rho, \Phi(W, \rho) \\ &\text{until } W = \emptyset \end{aligned} \tag{47}$$

Note that in '$W, \rho = \Phi(W, \rho) - \rho, \Phi(W, \rho)$' compute $\Phi(W, \rho)$ and $\Phi(W, \rho) - \rho$ and then assign to W, ρ on the left.

The elements of W are called the *splitters*.

Assume that $W = \{B_1, B_2, \cdots, B_n\}$ for some $n \geq 1$. Then

$$\Phi(W, \rho) = (\Phi_{B_1} \circ \cdots \circ \Phi_{B_n})(\rho)$$

In the application of Φ_{B_n} to ρ, if B_i, for some $i < n$, is split into B_i^1 and B_i^2, then by Proposition D.2 $(viii)$ we have that

$$\Phi_{B_i^1} \circ \Phi_{B_i^2} \circ \Phi_{B_i^1 \cup B_i^2} = \Phi_{B_i^1} \circ \Phi_{B_i^2}$$

Thus, we can skip the step Φ_{B_i}.

By Proposition D.2 (*iv*), refinement steps with respect to W may be performed in any order. Recalling the definition of $\Phi_{\tau,B}$, we have the following algorithm.

$$
\begin{array}{l}
W, \rho = \rho_I, \rho_I \\
\text{repeat} \\
\quad \text{Choose and remove any } B \text{ in } W \\
\quad \text{for each } \tau \in T_m \\
\quad\quad I_{\tau,B} = \{X \in \rho \,|\, X \cap \Phi_\tau(B) \neq \emptyset \wedge X \not\subseteq \Phi_\tau(B)\} \\
\quad\quad I^{1,2}_{\tau,B} = \{X \cap \Phi_\tau(B) \,|\, X \in I_{\tau,B}\} \cup \{X - \Phi_\tau(B) \,|\, X \in I_{\tau,B}\} \\
\quad\quad \rho = (\rho - I_{\tau,B}) \cup I^{1,2}_{\tau,B} \\
\quad\quad W = (W - I_{\tau,B}) \cup I^{1,2}_{\tau,B} \\
\text{until } W = \emptyset
\end{array}
\tag{48}
$$

The time complexity of this algorithm can be improved by using special data structures [Fer90]. This will not be reviewed here. To estimate the complexity of the algorithm given in [Fer90], define

$$
R := \{(q, \tau, q') \in Q \times T_\epsilon \times Q \,|\, q' \in \Delta_m(q) \text{ or } (\exists \tau \in T)\, q' \in \Delta_\tau(q)\}
\tag{49}
$$

If $|Q| = k$ and $|R| = r$, then the greatest fixpoint of $\tilde{\Phi}$ (with $\{Q\}$ as the initial partition) can be computed in $O(kr)$ time.

To conclude this appendix, we present the computations for some of the examples in this paper. *Computation for Example 3.1:* Let the transition structure be that of $\widehat{\mathbf{G}}$ in Figure 1. The following table displays Δ_{τ_1} and Δ_m. The vertical and horizontal axes list all the states in $\widehat{\mathbf{G}}$. For a given state, say, 0, and under a given transition, say, Δ_{τ_1}, the table lists the states in $\Delta_{\tau_1}(0)$. In this case, $\Delta_{\tau_1}(0) = \{4, 6\}$. The table represents this by putting τ_1 in row 0 and column 4, and in row 0 and column 6.

	0	1	2	3	4	5	6
0					τ_1		τ_1
1					τ_1		τ_1
2					τ_1		
3					τ_1		τ_1
4					m		τ_1
5							τ_1
6							m

The relations of Φ_{τ_1} and Φ_ϵ are the inverse relations of Δ_{τ_1} and Δ_m.

Let $\rho_I = \{B_1\}$, where $B_1 = \{0, 1, 2, 3, 4, 5, 6\}$, and $W, \rho = \rho_I, \rho_I$.

The first iteration: Choose and remove B_1 in W.

For τ_1,

$$
\begin{array}{l}
\Phi_{\tau_1}(B_1) = \{0, 1, 2, 3, 4, 5\}, \\
I_{\tau_1,B_1} = \{B_1\}, \\
I^{1,2}_{\tau_1,B_1} = \{B_2, B_3\}, \\
\rho = W = \{B_2, B_3\}
\end{array}
$$

where $B_2 := \{0, 1, 2, 3, 4, 5\}$ and $B_3 := \{6\}$.

For m,

$$\begin{aligned}
&\Phi_m(B_1) = \{4, 6\},\\
&I_{m,B_1} = \{B_2\},\\
&I^{1,2}_{m,B_1} = \{B_4, B_5\},\\
&\rho = W = \{B_4, B_5, B_3\}
\end{aligned}$$

where $B_4 := \{0, 1, 2, 3, 5\}$ and $B_5 := \{4\}$. Note that ρ here is ρ_1 in Example 3.1.

The second iteration: Choose and remove B_4 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_4) = \emptyset,\\
&I_{\tau_1,B_4} = \emptyset,\\
&I^{1,2}_{\tau_1,B_4} = \emptyset,\\
&\rho = \{B_4, B_5, B_3\},\\
&W = \{B_5, B_3\}
\end{aligned}$$

For m,

$$\begin{aligned}
&\Phi_m(B_4) = \emptyset,\\
&I_{m,B_4} = \emptyset,\\
&I^{1,2}_{m,B_4} = \emptyset,\\
&\rho = \{B_4, B_5, B_3\},\\
&W = \{B_5, B_3\}
\end{aligned}$$

The third iteration: Choose and remove B_5 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_5) = \{0, 1, 2, 3\},\\
&I_{\tau_1,B_5} = \{B_4\},\\
&I^{1,2}_{\tau_1,B_5} = \{B_6, B_7\},\\
&\rho = \{B_6, B_7, B_5, B_3\},\\
&W = \{B_6, B_7, B_3\}
\end{aligned}$$

where $B_6 := \{0, 1, 2, 3\}$ and $B_7 := \{5\}$.

For m,

$$\begin{aligned}
&\Phi_m(B_5) = B_5,\\
&I_{m,B_5} = \emptyset,\\
&I^{1,2}_{m,B_5} = \emptyset,\\
&\rho = \{B_6, B_7, B_5, B_3\},\\
&W = \{B_6, B_7, B_3\}
\end{aligned}$$

The fourth iteration: Choose and remove B_3 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_3) = \{0, 1, 3, 4, 5\},\\
&I_{\tau_1,B_3} = \{B_6\},\\
&I^{1,2}_{\tau_1,B_3} = \{B_8, B_9\},\\
&\rho = \{B_8, B_9, B_7, B_5, B_3\},\\
&W = \{B_8, B_9, B_7\}
\end{aligned}$$

where $B_8 := \{0, 1, 3\}$ and $B_9 := \{2\}$.

For m,

$$\begin{aligned}
&\Phi_m(B_3) = B_3, \\
&I_{m,B_6} = \emptyset, \\
&I^{1,2}_{m,B_6} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \{B_8, B_9, B_7\}
\end{aligned}$$

The fifth iteration: Choose and remove B_8 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_8) = \emptyset, \\
&I_{\tau_1,B_8} = \emptyset, \\
&I^{1,2}_{\tau_1,B_8} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \{B_9, B_7\}
\end{aligned}$$

For m,

$$\begin{aligned}
&\Phi_m(B_8) = \emptyset, \\
&I_{m,B_8} = \emptyset, \\
&I^{1,2}_{m,B_8} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \{B_9, B_7\}
\end{aligned}$$

The sixth iteration: Choose and remove B_9 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_9) = \emptyset, \\
&I_{\tau_1,B_9} = \emptyset, \\
&I^{1,2}_{\tau_1,B_9} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \{B_7\}
\end{aligned}$$

For m,

$$\begin{aligned}
&\Phi_m(B_9) = \emptyset, \\
&I_{m,B_9} = \emptyset, \\
&I^{1,2}_{m,B_9} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \{B_7\}
\end{aligned}$$

The seventh iteration: Choose and remove B_7 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_7) = \emptyset, \\
&I_{\tau_1,B_7} = \emptyset, \\
&I^{1,2}_{\tau_1,B_7} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \emptyset
\end{aligned}$$

For m,

$$\begin{aligned}
&\Phi_m(B_7) = \emptyset, \\
&I_{m,B_7} = \emptyset, \\
&I^{1,2}_{m,B_7} = \emptyset, \\
&\rho = \{B_8, B_9, B_7, B_5, B_3\}, \\
&W = \emptyset
\end{aligned}$$

Here the computation terminates and

$$\rho = \{\{0,1,2\},\{2\},\{4\},\{5\},\{6\}\}$$

is $\rho_{\mathbf{G}}$ in Example 3.1.

Computation for Example 7.1: Let the transition structure be as $\widehat{\mathbf{G}}_1$ in Figure 4. The following table displays Δ_{τ_1}, Δ_{τ_2}, Δ_{τ_3} and Δ_m.

	0	1	2	3	4	5	6
0			τ_2		τ_1	τ_3	
1			τ_2		τ_1	τ_3	
2					τ_1		
3					τ_1	τ_3	
4					m		τ_1
5							τ_1
6							m

Since $\rho_{2,Q} \leq \rho_{1,Q} = \rho_{\mathbf{G}}$, we can take $\rho_I = \rho_{\mathbf{G}} = \{B_8, B_9, B_7, B_5, B_3\}$, where the B_i's are defined in the computation for Example 3.1 above. Let $W, \rho = \rho_I, \rho_I$.

We shall present one iteration of this computation.

Choose and remove B_9 in W.

For τ_1,

$$\begin{aligned}
&\Phi_{\tau_1}(B_9) = \emptyset,\\
&I_{\tau_1,B_9} = \emptyset,\\
&I^{1,2}_{\tau_1,B_9} = \emptyset,\\
&\rho = \{B_8, B_9, B_7, B_5, B_3\},\\
&W = \{B_8, B_7, B_5, B_3\}
\end{aligned}$$

For τ_2,

$$\begin{aligned}
&\Phi_{\tau_2}(B_9) = \{0,1\},\\
&I_{\tau_2,B_9} = \{B_8\},\\
&I^{1,2}_{\tau_2,B_9} = \{B_{10}, B_{11}\},\\
&\rho = \{B_{10}, B_{11}, B_9, B_7, B_5, B_3\},\\
&W = \{B_8, B_7, B_5, B_3\}
\end{aligned}$$

where $B_{10} := \{0,1\}$ and $B_{11} := \{3\}$.

For τ_3,

$$\begin{aligned}
&\Phi_{\tau_3}(B_9) = \emptyset,\\
&I_{\tau_3,B_9} = \emptyset,\\
&I^{1,2}_{\tau_3,B_9} = \emptyset,\\
&\rho = \{B_{10}, B_{11}, B_9, B_7, B_5, B_3\},\\
&W = \{B_8, B_7, B_5, B_3\}
\end{aligned}$$

For m,

$$\begin{aligned}
&\Phi_m(B_9) = \emptyset,\\
&I_{m,B_9} = \emptyset,\\
&I^{1,2}_{m,B_9} = \emptyset,\\
&\rho = \{B_{10}, B_{11}, B_9, B_7, B_5, B_3\},\\
&W = \{B_8, B_7, B_5, B_3\}
\end{aligned}$$

The other iterations are similar. It can be verified that ρ will not be refined further. So $\rho_{2,Q} = \{\{0,1\},\{2\},\{3\},\{4\},\{5\},\{6\}\}$.

The partitions in the computation of Example 3.1 and 7.1 are summarized in the following table. The vertical axis indicates the iteration, with 0 indicating the initial partition, and the horizontal axis lists the states. For example, at row 2 column 3 the entry B_4 indicates that after the second iteration state 3 is in equivalence class B_4, which is defined earlier in the computation.

Iteration	0	1	2	3	4	5	6
0	B_1	B_1	B_1	B_1	B_1	B_1	B_1
1	B_4	B_4	B_4	B_4	B_5	B_4	B_3
2	B_4	B_4	B_4	B_4	B_5	B_4	B_3
3	B_6	B_6	B_6	B_6	B_5	B_7	B_3
4	B_8	B_8	B_9	B_8	B_5	B_7	B_3
5	B_8	B_8	B_9	B_8	B_5	B_7	B_3
6	B_8	B_8	B_9	B_8	B_5	B_7	B_3
7	B_8	B_8	B_9	B_8	B_5	B_7	B_3
8	B_{10}	B_{10}	B_9	B_{11}	B_5	B_7	B_3

The B_i's are defined in the computation above.

References

[Arn94] A. Arnold. *Finite Transition Systems.* Prentice Hall, 1994.

[BS81] S. Burris and H. P. Sankappanavar. *A Course in Universal Algebra.* Springer-Verlag, New York, 1981.

[Fer90] J.-C. Fernandez. An implementation of an efficient algorithm for bisimulation equivalence. *Science of Computer Programming*, 13:219–236, 1990.

[Gua97] Y. Guan. *Implementation of Hierarchical Observer Theory.* M.A.Sc thesis, Department of Electrical and Computer Engineering, University of Toronto, 1997.

[HC02] P. Hubbard and P.E. Caines. Dynamical consistency in hierarchical supervisory control. *IEEE Transactions on Automatic Control*, 47(1):37–52, January 2002.

[LW88] F. Lin and W. M. Wonham. Decentralized supervisory control of discrete-event systems. *Information Sciences*, 44:199–224, 1988.

[Mil89] R. Milner. *Communication and Concurrency.* Prentice Hall, New York, 1989.

[Pu00] K. Q. Pu. *Modeling and Control of Discrete-Event Systems with Hierarchical Abstractions.* M.A.Sc thesis, Department of Electrical and Computer Engineering, University of Toronto, 2000.

[RW87] P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete-event processes. *SIAM J. Control and Optimization*, 25(1):206–230, 1987.

[Sif82] J. Sifakis. A unified approach for studying the properties of transition systems. *Theoretical Computer Science*, 18:227–258, 1982.

[Wnhm02] W. M. Wonham. Notes on Control of Discrete-Event Systems: ECE 1636F/1637S. Department of Electrical and Computer Engineering, University of Toronto, 2002. Can be downloaded from www.control.utoronto.ca/people/profs/wonham/wonham.html.

[Wnhm76] W. M. Wonham. Towards an abstract internal model principle. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(11):735–740, November 1976.

[Won98] K. C. Wong. On the complexity of projections of discrete-event systems. In *IEE Workshop on Discrete Event Systems*, pages 201–206, Cagliari (Sardinia) Italy, August 1998.

[WTMH00] K. C. Wong, J. G. Thistle, R. P. Malhamé, and H.-H. Hoang. Supervisory control of distributed systems: conflict resolution. *Discrete Event Dynamical Systems: Theory and Applications*, 10(1):131–186, January 2000.

[WW96] K. C. Wong and W. M. Wonham. Hierarchical control of discrete-event systems. *Discrete Event Dynamic Systems: Theory and Applications*, 6(3):241–273, July 1996.

[WW98] K. C. Wong and W. M. Wonham. Modular control and coordination of discrete-event systems. *Discrete Event Dynamic Systems: Theory and Applications*, 8(3):247–297, October 1998.

[WW00] K. C. Wong and W. M. Wonham. On the computation of observers in discrete-event systems. In *2000 Conference on Information Sciences and Systems*, pages TP1–7 to TP1–12, Princeton, NJ, March 2000.

[ZW90] 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.