15th Triennial World Congress, Barcelona, Spain

HIERARCHICALLY CONSISTENT CONTROLLED DISCRETE EVENT SYSTEMS

Antonio E. C. da Cunha [*,1] **José E. R. Cury** [**,2]

[*] *Departamento de Engenharia Elétrica*
Instituto Militar de Engeharia
Pr. Gen Tibúrcio 80 – 22.290-270 – Rio de Janeiro – RJ – Brasil
[**] *Departamento de Automação e Sistemas*
Universidade Federal de Santa Catarina
Caixa Postal 470 – 88040-001 – Florianópolis – SC – Brasil

Abstract: In the hierarchical control of discrete event systems (DES), the hierarchical consistency expresses the requirement that a control *task* is solvable within the model at a given level if it is in fact executable by the infrastructure one level down. In this paper we present a method for construction of a two level hierarchy of DES with hierarchical consistency. By application of a generalized model for controlled DES for the high level system, hierarchical consistency is obtained directly with no refinement of the hierarchy. This approach makes itself distinct from other approaches for hierarchical control for that they all add complexity by refining the the hierarchy to ensure hierarchical consistency.

Keywords: Discrete-Event Dynamic Systems, Hierarchical Control, Consistency

1. INTRODUCTION

In the Ramadge & Wonham (RW) framework for supervisory control of discrete event systems (DES), although the supervisor synthesis algorithms have polynomial complexity with the number of states of the system, the number of states grows exponentially with the number of system components (Ramadge and Wonham, 1989). The general idea of vertical decomposition of the system to reduce the overall complexity is considered in the hierarchical control of DES, first introduced in Zhong and Wonham (1990) and also subject of Wong and Wonham (1996), Pu (2000) and Hubbard and Caines (2002).

In the hierarchical control of DES, the *hierarchical consistency* expresses the requirement that a control *task* is solvable within the model at a given level if it is in fact executable by the infrastructure one level down. As it will be shown later, to achieve hierarchical consistency in a two level hierarchy, the above approaches impose some conditions to the hierarchy. If the conditions are not valid, the approaches propose refinements for the hierarchy by insertion of new high level events and modification of the low level system until the conditions are valid. Therefore this refinement adds complexity to build the hierarchy with hierarchical consistency.

In this work, hierarchical consistency is achieved for a two level hierarchy with no refinement for the hierar-

[1] da Cunha (aecc@das.ufsc.br) is supported (in part) by CNPq under grant 200463/01-2.
[2] Dr. Cury (cury@das.ufsc.br) is supported (in part) by CNPq under grant number 300953/93-3 and PRONEX 015/98 (Control of Dynamical Systems).

chy. This is done by modelling the high level system using a generalized framework for controlled DES introduced in Cury *et al.* (2001). It is also provided a constructive procedure to obtain the complete hierarchy, given the low level system and the high level set of events.

The paper has the following outline: section 2 presents the problem formulation; section 3 reviews the generalized framework for supervisory control of DES of Cury *et al.* (2001); section 4 introduces the proposed model for the high level system; section 5 contains the main results of the paper; section 6 presents the method to build the hierarchy illustrated by an example; section 7 makes some comments on related work, summarizes the contributions of the paper and give directions of future research.

2. PROBLEM FORMULATION

This section presents the basic problem of hierarchical control for DES and the concept of hierarchical consistency.

We introduce some preliminary language formalism to describe the models from Wonham (1999). Let Σ be a finite set of symbols, Σ^+ be the set of all finite length strings formed by concatenation of symbols in Σ, and Σ^* be the set Σ^+ plus the empty string ϵ. Given two strings $s, t \in \Sigma^*$, s is a *prefix* of t, $s \leq t$, if there exists $u \in \Sigma^*$ such that $s \cdot u = t$; also, s is a *strict prefix* of t, $s < t$, if s is a prefix of t and $s \neq t$. Any subset L of Σ^* is a *language* on Σ. Given a language L on Σ, the *prefix closure* of L is a language on Σ, denoted by $\overline{L}$, that contains every prefix of strings in L. A language is said to be *prefix closed* if it is equal to its prefix closure.

Consider the two level hierarchical control scheme in figure 1, where the low level DES is D_{lo} and the high level DES is D_{hi} (Zhong and Wonham, 1990). The behavior of D_{hi} is an abstraction of the behavior of D_{lo}, generated by an information channel inf_{lohi}. We consider that the control action of supervisor f_{hi} is virtual, that is, it is in fact implemented by a supervisor f_{lo} which controls D_{lo} following the directives of supervisor f_{hi}, transmitted through by the command channel com_{hilo}.

The model for D_{lo} is the standard RW model for controlled DES (Ramadge and Wonham, 1989). Let Σ_{lo} be the set of events of D_{lo}. D_{lo} is defined by the pair $(L_{lo}, L_{m,lo})$, where $L_{lo} \subseteq \Sigma_{lo}^*$ is a prefix closed language representing every string that can be generated by D_{lo}; $L_{m,lo} \subseteq L_{lo}$ is a language of marked strings, that is, defining *completed tasks* for D_{lo}. The event control mechanism for D_{lo} is defined by

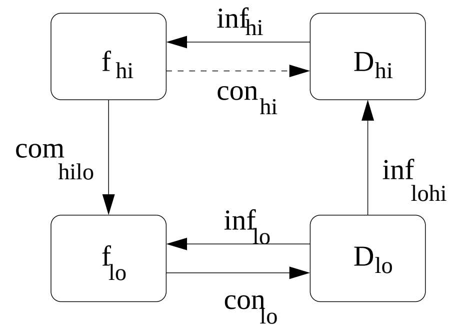

Fig. 1. Scheme for hierarchical control of DES.

partitioning the alphabet Σ_{lo} into a set of *controllable events*, $\Sigma_{lo,c}$, that can be inhibited, and a set of *uncontrollable events*, $\Sigma_{lo,u}$.

The following presents a summary of the supervisory control results of Ramadge and Wonham (1989). A supervisor for D_{lo} is a map $f_{lo} : L_{lo} \rightarrow 2^{\Sigma_{lo}}$ for that given $s \in L_{lo}$, $f_{lo}(s)$ is the set of enabled events after s. The closed loop f_{lo}/D_{lo} is characterized by the language $L(f_{lo}/D_{lo})$, the set of all strings in L_{lo} that *survive* under supervision, and $L_m(f_{lo}/D_{lo}) = L(f_{lo}/D_{lo}) \cap L_{m,lo}$, the set of closed loop marked strings. A non-blocking supervisor f_{lo} is one that $\overline{L_m(f_{lo}/D_{lo})} = L(f_{lo}/D_{lo})$, that is, all closed loop strings are prefixes of closed loop marked strings. For a language $E_{lo} \subseteq \Sigma_{lo}^*$, there is a non-blocking supervisor f_{lo} for D_{lo} such that $L_m(f_{lo}/D_{lo}) = E_{lo}$ if and only if E_{lo} is $L_{m,lo}$-closed and controllable with respect to D_{lo}.[3] Given D_{lo} and $E_{lo} \subseteq \Sigma_{lo}^*$ there is a unique maximal controllable and $L_{m,lo}$-closed language contained in E_{lo}, denoted by $\sup CF(E_{lo})$. Given $E_{lo} \subseteq \Sigma_{lo}^*$, if $\sup CF(E_{lo})$ is non-empty, the the supervisor that corresponds to the less restrictive behavior of D_{lo} that follows E_{lo} can be implemented by using a finite state generator for $\sup CF(E_{lo})$.

Let Σ_{hi} be the event set for D_{hi}. The information channel Inf_{lohi} is modelled by a *reporter map* $\theta : L_{lo} \rightarrow \Sigma_{hi}^*$, formally defined by the recursion $\theta(\epsilon) = \epsilon$ and $\theta(s\sigma) = \theta(s)$ or $\theta(s)\tau$, where ϵ represents the empty string on both alphabets, $s \in L_{lo}$, $\sigma \in \Sigma_{lo}$ and $\tau \in \Sigma_{hi}$ (Zhong and Wonham, 1990). In words, a reporter map notifies the occurrence of events in D_{hi} by the observation of the sequences generated by D_{lo}. At this point, let D_{hi} be a controlled DES generating the language $\theta(L_{lo}) = \{t \in \Sigma_{hi}^* \mid (\exists s \in L_{lo})\ \theta(s) = t\}$.

There is *hierarchical consistency* between D_{lo} and D_{hi} if and only if, for any $E_{hi} \subseteq \Sigma_{hi}^*$, *if* there exists a non-blocking supervisor f_{hi} for D_{hi} such that $L_m(f_{hi}/D_{hi}) = E_{hi}$, *then* there exists a non-blocking supervisor f_{lo} for D_{lo} such that $\theta(L_m(f_{lo}/D_{lo})) =$

[3] $E \subseteq \Sigma_{lo}^*$ is said to be controllable with respect to D_{lo} if $\overline{K} \cdot \Sigma_{lo,u} \cap L_{lo} \subseteq \overline{K}$. $K \subseteq L$ is said to be L-closed if $\overline{K} \cap L = K$.

E_{hi} (Zhong and Wonham, 1990). There is *strong hierarchical consistency* between D_{lo} and D_{hi} if and only if for any $E_{hi} \subseteq \Sigma_{hi}^*$ there exists a non-blocking supervisor f_{hi} for D_{hi} such that $L_m(f_{hi}/D_{hi}) = E_{hi}$ *if and only if* there exists a non-blocking supervisor f_{lo} for D_{lo} such that $\theta(L_m(f_{lo}/D_{lo})) = E_{hi}$.[4] The hierarchical control problem for DES is: given D_{lo} and Σ_{hi}, find D_{hi} with (strong) hierarchical consistency.

3. A GENERALIZED MODEL FOR CONTROLLED DES

This section reviews the generalized model for controlled DES of Cury *et al.* (2001).

Given a set of events Σ, a controlled DES D on Σ is a tuple $(L, \Gamma) \subseteq (\Sigma^*, \Sigma^* \times 2^{2^{\Sigma} \times \{M,N\}})$, where $L \subseteq \Sigma^*$ is a prefix closed language and Γ is a map of $s \in L$ into control sets $\Gamma(s) \subseteq 2^{\Sigma} \times \{M, N\}$. For the system D, the language L represents the set of all strings in Σ that can be generated by the system, and for a string $s \in L$, $\Gamma(s)$ is a set of controls $\gamma_{\#} \in 2^{\Sigma} \times \{M, N\}$, where γ is the set of enabled events after s, and if $\# = M$, s is considered marked, and if $\# = N$, s is considered not-marked. The generalized model controlled DES above defined differs from the standard RW model in that the control set depends on the string generated by the system, and the marked strings are determined by the control rather than being a subset of all strings.

The following defines some operations for controls and control sets. For the set $\{M, N\}$, define the partial order $\geq$ and the operations *or* ($\vee$) and *and* ($\wedge$) as for the binary set $\{1, 0\}$, with M playing the role of 1. For controls $\gamma^1_{\#^1}$ and $\gamma^2_{\#^2}$ in $2^{\Sigma} \times \{M, N\}$, $\gamma^2_{\#^2} \supseteq \gamma^1_{\#^1}$ if and only if $\gamma^2 \supseteq \gamma^1$ and $\#^2 \geq \#^1$; and the *union* is $\gamma^1_{\#^1} \cup \gamma^2_{\#^2} = (\gamma^1 \cup \gamma^2)_{(\#^1 \vee \#^2)}$. For control sets Γ_1 and Γ_2 in $2^{2^{\Sigma} \times \{M,N\}}$, $\Gamma_2 \supseteq \Gamma_1$ if and only if for all $\gamma^1_{\#^1} \in \Gamma_1$ there is $\gamma^2_{\#^2} \in \Gamma_2$ such that $\gamma^2_{\#^2} \supseteq \gamma^1_{\#^1}$; and the *union* $\Gamma_1 \cup \Gamma_2$ is a set of controls $\gamma_{\#} \in 2^{\Sigma} \times \{M, N\}$ for which there is $\gamma^1_{\#^1} \in \Gamma_1$ and $\gamma^2_{\#^2} \in \Gamma_2$ such that $\gamma_{\#} = \gamma^1_{\#^1} \cup \gamma^2_{\#^2}$.

The results for supervisory control in Cury *et al.* (2001) are the following. Given the CDES $D = (L, \Gamma)$, a supervisor for D is a map $f : L \to 2^{\Sigma \times \{M,N\}}$ which maps $s \in L$ to the control $f(s) \in \Gamma(s)$. For $s \in L$, if $f(s) = \gamma_{\#}$, the active event set in L after s is restricted to $\gamma \cap \Sigma_L(s)$ and s is considered as marked if $\# = M$, otherwise, not-marked.[5] The closed loop behavior f/D is characterized by a closed language $L(f/D) \subseteq L$, the strings in L allowed by f under supervision, and a marked language $L_m(f/D) \subseteq L(f/D)$, the strings in $L(f/D)$ where the supervisor chooses a control with the M attribute. A non-blocking supervisor is one for which $L(f/D) = \overline{L_m(f/D)}$. Given $D = (L, \Gamma)$ and $K \subseteq L$, K is said to be Γ-*compatible* if and only if $K = \emptyset$ or for all $s \in \overline{K}$, there is $\gamma_{\#} \in \Gamma(s)$ such that $\gamma \cap \Sigma_L(s) = \Sigma_K(s)$, and $\# = M$ if and only if $s \in K$, otherwise $\# = N$. Given $D = (L, \Gamma)$ and $K \subseteq L$, there is a nonblocking supervisor f such that $L_m(f/D) = K$ if and only if K is Γ-compatible. For the case that the control set $\Gamma(s)$ is closed for arbitrary unions of controls for all $s \in L$, there is a unique maximal Γ-compatible language contained in K, denoted by $\sup G(K)$. The maximal Γ-compatible language is used to implement the less restrictive supervisor for a desired behavior.

4. MODEL FOR THE HIGH LEVEL SYSTEM

This section introduces the model for D_{hi} for the two level hierarchy in figure 1.

Let $w : L_{lo} \to \Sigma_{hi} \cup \{\tau_0\}$ be the *tail map*, defined recursively as $w(\epsilon) = \tau_0$, $w(s\sigma) = \tau_0$, if $\theta(s\sigma) = \theta(s)$, and $w(s\sigma) = \tau$, if $\theta(s\sigma) = \theta(s)\tau$, where τ_0 is the *silent event*, a new symbol not in Σ_{hi}, representing that the reporter map has not notified the occurrence of a new event, $s \in L_{lo}$, $\sigma \in \Sigma_{lo}$ and $\tau \in \Sigma_{hi}$ (Zhong and Wonham, 1990). Define set $L_{voc} \subseteq L_{lo}$ as the set of *vocal strings* of D_{lo}, the strings $s \in L_{lo}$ for that $w(s) \neq \tau_0$ and the empty string ϵ. A string in L_{lo} that is not vocal is a *silent string*.

For the string $s \in L_{lo}$, define the following set $L(s) = \{u \in L_{lo} \mid (\forall u' \in \Sigma_{lo}^+)\ u' < u \ \Rightarrow\ w(s \cdot u') = \tau_0\}$. $L(s)$ is a prefix closed language on Σ_{lo} that contains, besides the empty string, all the non-empty strings in Σ_{lo} that, when concatenated with s, form either a silent string or a vocal string that occur in D_{lo} *after* s and *until* the the reporter map notifies the occurrence a new high level event. Define also the sets $L_{voc}(s) = \{v \in L(s) - \{\epsilon\} \mid w(s \cdot v) \neq \tau_0\}$ and $L_m(s) = L_{voc}(s) \cup \{u \in L(s) \mid s \cdot u \in L_{m,lo}\}$. $L_{voc}(s)$ contains the non-empty strings in $L(s)$ that correspond to vocal strings in D_{lo}, and $L_m(s)$ contains $L_{voc}(s)$ plus the strings in $L(s)$ that correspond to marked strings in D_{lo}. Finally, define the subsystem $D(s)$ as the controlled DES in the RW framework $(L(s), L_m(s))$, with the same control structure than D_{lo}, that is, defined by the partition of Σ_{lo}. The subsystem $D(s)$ corresponds to the behavior of D_{lo} after the occurrence of s and until the notification of a new high level event by the reporter map, and a *task* for the subsystem is to reach either a vocal or a marked string of D_{lo}.

[4] The strong hierarchical consistency appears in Wong and Wonham (1996) with the name *control consistency*.

[5] For $L \subseteq \Sigma^*$ and $s \in L$, the active event set in L after s is $\Sigma_L(s) = \{\sigma \in \Sigma \mid s\sigma \in \overline{L}\}$.

Also, for the string $s \in L_{lo}$ define the set $\Sigma_{voc}(s) = \{\tau \in \Sigma_{hi} \mid (\exists v \in L_{voc}(s))\ \tau = w(s \cdot v)\}$, that contains the *next* events to be reported to the high level after the occurrence of s. Therefore, let $\Gamma_{voc}(s)$ be the set of high level controls $\gamma_{\#} \in 2^{\Sigma hi} \times \{M, N\}$ such that (i) there is a non-blocking supervisor f for the subsystem $D(s)$ such that $w(s \cdot [L_m(f/D(s)) \cap L_{voc}(s)]) = \gamma \cap \Sigma_{voc}(s)$, and (ii) if $\# = M$ then $L_m(f/D(s)) - L_{voc}(s) \neq \emptyset$, else $L_m(f/D(s)) - L_{voc}(s) = \emptyset$. The set $\Gamma_{voc}(s)$ is the control set that can be implemented in the high level by means of supervisory control *after* the occurrence of s. Notice that a control with attribute M corresponds to a supervisor that allows marked strings in D_{lo}, and a control with attribute N corresponds to a supervisor that don't enable marked strings in D_{lo}. Call $\Gamma_{voc}(s)$ the vocal control set for s.

The computation of the vocal control set for $s \in L_{lo}$ is done by solving a supervisory control problem for the subsystem $D(s)$. Define the following specification languages: for the control γ_M, $E_s(\gamma_M) = L_m(s) - \{u \in L_m(s) \mid w(s \cdot u) \notin \gamma\}$, and for γ_N, $E_s(\gamma_N) = L_{voc}(s) - \{u \in L_{voc}(s) \mid w(s \cdot u) \notin \gamma\}$. The specification $E_s(\gamma_M)$ inhibits every vocal string after s whose output is not an event in γ, and $E_s(\gamma_N)$, besides inhibiting every vocal string after s whose output is not an event in γ, also inhibits the silent marked strings after s.

Proposition 1. For $s \in L_{lo}$ and $\gamma_{\#} \in 2^{\Sigma hi} \times \{M, N\}$, if (i) $K = \sup CF(E_s(\gamma_{\#})) \neq \emptyset$, (ii) $w(s \cdot [K \cap L_{voc}(s)]) = \gamma \cap \Sigma_{voc}(s)$, and (iii) if $\# = M$, then $K - L_{voc}(s) \neq \emptyset$, else $K - L_{voc}(s) = \emptyset$, then $\gamma_{\#} \in \Gamma_{voc}(s)$.

By application of proposition 1, the vocal control set for $s \in L_{lo}$ is built by testing if each control $\gamma_{\#}$ in the set $2^{\Sigma hi} \times \{M, N\}$ is an element of the vocal control set.

Finally, define D_{hi} as the controlled DES in the generalized framework of Cury *et al.* (2001) (L_{hi}, Γ_{hi}) on Σ_{hi}. The language $L_{hi} \subseteq \Sigma_{hi}^*$ is given by $L_{hi} = \theta(L_{lo})$, the image of L_{lo} by the reporter map. For $t \in L_{hi}$ define the inverse image map as $\theta^{-1}(t) = \{s \in L_{lo} \mid \theta(s) = t\}$. For $t \in L_{hi}$ the control set $\Gamma_{hi}(t) = \bigcup \Gamma_{voc}(v)$, for $v \in \theta^{-1}(t) \cap L_{voc}$. Therefore, each control set for $t \in L_{hi}$ is the union of the vocal control sets for the vocal strings corresponding to t. It can be proved that $\Gamma_{hi}(t)$ is closed for the union of controls.

5. MAIN RESULTS

This section contains the proof that there is hierarchical consistency for the proposed two level hierarchy.

Given a Γ_{hi}-compatible specification $E_{hi} \subseteq \Sigma_{hi}^*$ for D_{hi}, refer figure 1, there is a non-blocking supervisor f_{hi} for D_{hi} such that $L_m(f_{hi}/D_{hi}) = E_{hi}$. Define f_{hi} for $t \in \overline{E_{hi}}$ as $f_{hi}(t) = \gamma_M$ for $t \in E_{hi}$ and $f_{hi}(t) = \gamma_N$ for $t \in \overline{E_{hi}} - E_{hi}$, where $\gamma \cap \Sigma_{L_{hi}}(t) = \Sigma_{E_{hi}}(t)$. The control input $\gamma_{\#}$ is not actually applied to D_{hi} by f_{hi}, it is in fact sent through com_{hilo} as a control directive for a supervisor f_{lo} for D_{lo}. To follow the directive $\gamma_{\#}$ of f_{hi} at t, the supervisor f_{lo} for D_{lo} decomposes the current string $s \in L_{lo}$ into two strings $s = v \cdot u$, where $v = \sup\{v' \in L_{voc} \mid v' \leq s\}$, the greatest prefix of s which is a vocal string, and $u \in L(v)$. Notice that $t = \theta(v) = \theta(s)$. The control policy of f_{lo} at s is the same that a supervisor for the subsystem $D(v)$ applies at u to follow the directive of f_{hi} at t, given by the optimal control $\gamma_{\#,opt} = \sup\{\gamma'_{\#'} \in \Gamma_{voc}(v) \mid \gamma'_{\#'} \subseteq \gamma_{\#}\}$, and the language $E_v(\gamma_{\#,opt}) = \sup CF(E_v(\gamma_{\#,opt}))$. Therefore, define the control action of f_{lo} for $s \in L_{lo}$ as $f_{lo}(s) = \{\sigma \in \Sigma_{lo} \mid u\sigma \in \overline{E_v(\gamma_{\#,opt})}\}$. It can be proved that, for the above supervisory scheme, if E_{hi} is Γ_{hi}-compatible, then f_{lo} is nonblocking and $\theta(L_m(f_{lo}/D_{lo})) = E_{hi}$. This is exactly the definition of hierarchical consistency (section 2), therefore:

Theorem 1. For the proposed two level hierarchy, there is hierarchical consistency between D_{lo} and D_{hi}.

The next step is to determine a hierarchy with strong hierarchical consistency. When two or more vocal strings have the same image through the reporter map, the union of their vocal control sets may lead to the loss of some particular controls of some of the control sets. Therefore, some behaviors implementable by supervisor in D_{lo} may be lost in the process of abstraction that builds D_{hi}. If any pair of vocal strings of D_{lo} corresponds to different strings in D_{hi}, there is no such loss. Therefore, define the reporter map θ to be *deterministic* when for any $v_1, v_2 \in L_{voc}$, if $v_1 \neq v_2$ then $\theta(v_1) \neq \theta(v_2)$.

Theorem 2. If the reporter map θ is deterministic, then there is strong hierarchical consistency between D_{lo} and D_{hi}.

6. EXAMPLE

This section presents the method for (strong) hierarchical consistency, illustrated by an example.

Let the state representation for D_{lo} and the inf_{lohi} be the Moore automaton G_{lo} (Wonham, 1999), for that the recognized languages are the languages of D_{lo} and the state output function defines inf_{lohi}. Consider, the DES in figure 2: the transition diagram follows the conventional notation of (Wonham, 1999), where transitions with a tick correspond to controllable events and the output τ_0 is not represented. Notice that the vocal strings of D_{lo} correspond to the states in G_{lo} with output and the initial state, called *vocal states*.

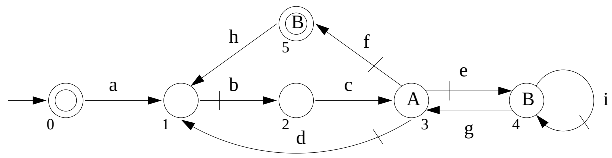

Fig. 2. Moore automaton G_{lo}: D_{lo} and inf_{lohi}

Given a string $s \in L_{lo}$, the state representation for the subsystem $D(s)$ is the Moore automaton $G(x)$, where $x = [s]$, the state of G_{lo} equivalent to s. $G(x)$ is built by taking the reachable component of G_{lo}, starting from x, and stopping when a state with an output or a visited state is found. The vocal control sets for the state x are computed by application of proposition 1. Figure 3 displays the subsystems and the vocal control sets for the vocal states.

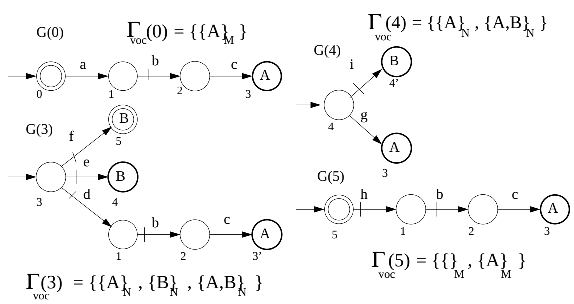

Fig. 3. Subsystems for the vocal states of G_{lo}

The state representation for D_{hi} is the pair (G_{hi}, Γ_{hi}). G_{hi} is obtained by first substituting the transition labels of G_{lo} for the state output labels, and then taking the deterministic (considering τ_0 null event) equivalent automaton (Wonham, 1999). Γ_{hi} is a table relating each state of G_{hi} to its control set. By section 4, each entry in Γ_{hi} for a state x is the union of the vocal control sets for the vocal states of G_{lo} corresponding to x. The automaton G_{hi} and Γ_{hi} for the running example are shown in figure 4.

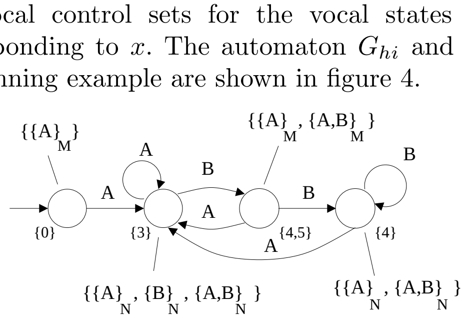

Fig. 4. D_{hi}: automaton G_{hi} and table Γ_{hi}

Analyze the high level specification E_{hi}, recognized by the automaton in figure 5. E_{hi} is Γ_{hi}-compatible, and the controls that implement a corresponding high level supervisor are also represented in figure 5. The

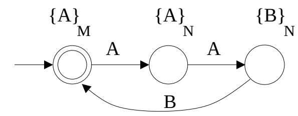

Fig. 5. E_{hi} and controls

low level implementation for E_{hi} is given by the language E_{lo} recognized by the automaton in figure 6. When implementing a supervisor for E_{lo} it is not necessary to implement the automaton in figure 6 directly. E_{lo} is in fact implemented by loading the corresponding subsystem and implementing for each subsystem the control policy that follows the high level control directive. This process is shown by the indications of subsystem and high level controls in figure 6.

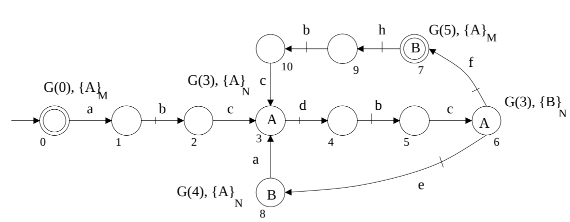

Fig. 6. Language implemented in the low level

Consider the second specification as $E'_{hi} = \{\epsilon, AB\}$. Although E'_{hi} is not Γ_{hi}-compatible, there is $E'_{lo} = \{\epsilon, abcf\}$ controllable with respect to D_{lo}, $L_{m,lo}$-closed, and such that $\theta(E'_{lo}) = E'_{hi}$. From this example, there is not strong hierarchical consistency between D_{lo} and D_{hi}.

To achieve strong hierarchical consistency, examine the outputs of G_{lo} and modify the event labels to eliminate the non-determinism in the reporter map. By inspection of figure 2, the vocal states 4 and 5 correspond to the same state in G_{hi}, figure 4. Therefore, to make the information channel deterministic, a new instance of event B, B', is created for state 4. The resulting high level system is shown in figure 7, with its corresponding control structure. Finally, it can be checked that the specification $E'_{hi} = \{\epsilon, AB\}$ is Γ_{hi}-compatible for the new D_{hi}.

7. CONCLUSION

This section summarizes the contributions of this paper and makes some comments regarding related work found in the literature.

In the works of Zhong and Wonham (1990), Wong and Wonham (1996), Pu (2000) and Hubbard and Caines

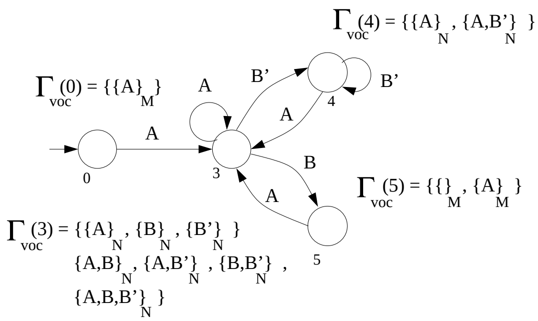

Fig. 7. Refined G'_{hi} and Γ'_{hi}

(2002), the common approach to the hierarchical control problem for a two level hierarchy (recall figure 1) is to fix a model to D_{hi} and refine D_{lo} and inf_{lohi} until there is hierarchical consistency. This refinement is done by creating new event labels for Σ_{hi} and modifying the transition structure of D_{lo}, therefore adding complexity both in state size of hierarchy and high level event semantics. For example, in (Zhong and Wonham, 1990) the model for D_{lo} and D_{hi} is the standard RW model with prefix closed languages, and to create the partition of the alphabet Σ_{hi} into controllable and uncontrollable events, the condition of *strong output control consistency* is imposed to D_{lo} and inf_{lohi}. When the marking behavior is considered in the hierarchy, the hierarchical consistency also involves the problem of matching non-blocking behaviors for D_{hi} and D_{lo}. This is treated with additional conditions for D_{lo} and inf_{lohi}: *observer reporter map* and *marking consistency* in (Wong and Wonham, 1996), *weak observer reporter map* in (Pu, 2000), and *non-blocking trace-dynamical consistency* in (Hubbard and Caines, 2002). For some of the above conditions no constructive method was proposed since then. Also, most of the above approaches don't give directions for implementation of hierarchical control, that is, the construction of f_{lo} given f_{hi}.

This paper provides a solution for the hierarchical control problem considering the general case of marking behavior for the systems. By application of the generalized model for controlled DES from Cury *et al.* (2001), hierarchical consistency is achieved with no additional condition or refinement for D_{lo} and inf_{lohi}. Moreover, this work also provides a method to build the hierarchy with hierarchical consistency, considers the refinement necessary for the strong hierarchical consistency, and finally gives directions of construction of a low level supervisor, given a high level designed supervisor.

With an extension of the results presented in this paper, by considering a two level hierarchy with the same generalized model for controlled DES of Cury *et al.* (2001), the following extensions for the hierarchical control problem are possible: multilevel hierarchies, hierarchical coordination and the consideration of systems with more elaborated control structure, as discrete state abstractions of hybrid systems as in González *et al.* (2001).

Similar approach for the hierarchical control of DES is found in Torrico and Cury (2001), but for the state aggregation viewpoint.

ACKNOWLEDGMENTS

The authors thank Dr. Bruce Krogh for the valuable comments when preparing the final version for this paper.

8. REFERENCES

Cury, J. E. R., C. R. C. Torrico and A. E. C. da Cunha (2001). A new approach for supervisory control of discrete event systems. In: *Proceedings of the European Control Conference*. Porto – Portugal.

González, J. M. E., A. E. C. da Cunha, J. E. R. Cury and B. H. Krogh (2001). Supervision of event driven hybrid systems: Modeling and synthesis. In: *In Proceedings of Hybrid Systems: Computation and Control*. LNCS. Italy.

Hubbard, Paul and Peter E. Caines (2002). Dynamical consistency in hierarchical superviosry control. *IEEE Transactions on Automatic Control* **47**(1), 37–52.

Pu, Ken Quian (2000). Modeling and Control of Discrete-Event Systems with Hierarchical Abstraction. Master's thesis. Systems Control Group, Department of Electrical & Computer Engineering, University of Toronto. Toronto, Canada.

Ramadge, Peter J. G. and W. Murray Wonham (1989). The control of discrete event systems. *Proceedings of the IEEE* **77**(1), 81–98.

Torrico, C. C. and J. E. R. Cury (2001). Hierarchical supervisory control of discrete event systems based on state aggregation. To appear in the 2002 IFAC World Congress.

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

Wonham, W. M. (1999). *Notes on Control of Discrete-Event Systems*. Systems Control Group, Department of Electrical & Computer Engineering, University of Toronto. Toronto, Canada.

Zhong, Hao and W. Murray Wonham (1990). On the consistency of hierarchical supervision in discrete-event systems. *IEEE Transactions on Automatic Control* **35**(10), 1125–1134.