

# CONTROLE SUPERVISÓRIO DE SISTEMAS A EVENTOS DISCRETOS COM AUTO-APROXIMAÇÕES

MARCELO ROSA\*, MARCELO TEIXEIRA\*, ROBI MALIK†

\**Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Tecnológica Federal do Paraná  
Pato Branco, Brasil*

†*Depto. de Ciência da Computação  
Universidade de Waikato  
Hamilton, Nova Zelândia*

Emails: marcelorosa@alunos.utfpr.edu.br, marceloteixeira@utfpr.edu.br,  
robi@waikato.ac.nz

**Abstract**— *Distinguishers and Approximations* have been used in *Supervisory Control Theory* (SCT) of *Discrete Event Systems* (DESs), as a way to simplify modeling tasks and to reduce synthesis cost. Despite the advantages, supervisors obtained from approximated DES plants can be suboptimal, so that the modeling structure has to be reconstructed empirically, until the proper supervisor is found. In this paper we present an approach to construct *suitable-approximations* that always lead to least restrictive controllable synthesis. Examples are provided to illustrate the approach.

**Keywords**— Discrete Event Systems; Supervisory Control; Automation.

**Resumo**— *Distinguidores e Aproximações* têm sido usados na *Teoria de Controle Supervisório* (TCS) de *Sistemas a Eventos Discretos* (SEDs), como forma de simplificar tarefas de modelagem e reduzir o esforço de síntese. Apesar das vantagens, supervisores obtidos a partir de modelos aproximados da planta de um SED podem ser subótimos, implicando na reconstrução da estrutura de modelagem até que o supervisor apropriado seja obtido. Este trabalho mostra como construir *auto-aproximações* que, por construção, garantem a obtenção de supervisores controláveis e minimamente restritivos. Exemplos ilustram a abordagem.

**Palavras-chave**— Sistemas a Eventos Discretos, Controle Supervisório, Automação.

## 1 Introdução

A *Teoria de Controle Supervisório* (TCS) (Ramadge and Wonham, 1989) formaliza a síntese de controladores para *Sistemas a Eventos Discretos* (SEDs), usando *Autômatos e Linguagens* (Cassandras and Lafortune, 2008). Apesar das várias extensões, a TCS ainda enfrenta limitações quando aplicada sobre problemas reais de controle industrial. Uma delas é o nível de dificuldade enfrentado para expressar especificações de controle.

Recentemente, o conceito de *distinguidor* (Bouzon et al., 2008; Cury et al., 2015) tem sido explorado como forma de simplificar tarefas de modelagem. Essa abordagem fornece mais detalhes sobre a ocorrência de certos eventos no modelo da planta de um SED, sob o preço de incorporar à síntese de controle um modelo adicional, o *distinguidor*, que refina e distingue eventos.

No sentido de tornar essa abordagem computacionalmente mais atrativa, a literatura (Bouzon et al., 2009; Cury et al., 2015) tem sugerido o uso de uma *aproximação*, cujo mecanismo de construção remove parte, ou todo, o modelo do *distinguidor* da planta, simplificando consideravelmente a síntese. É mostrado que, em alguns casos, essa abordagem ainda leva a um controlador ótimo. Entretanto, em outros casos, pode acontecer que a aproximação leve a um controlador subótimo, i.e., mais restritivo, justamente pela carência do meca-

nismo de distinção que induz a síntese a ser mais conservativa para preservar o conjunto de especificações. Nesses casos, melhorar a qualidade do controlador implica em selecionar partes do *distinguidor* para serem incluídas à síntese.

Todavia, a tarefa de antecipar qual parte do *distinguidor* é realmente necessária à síntese é empírica. Em (Aguiar et al., 2013) são propostas duas meta-heurísticas para identificar os chamados *distinguidores-essenciais* baseando-se na evolução randômica de um conjunto de configurações iniciais. Estima-se que essa abordagem seja computacionalmente dispendiosa, uma vez que partes do *distinguidor* são incluídas e testadas incrementalmente na síntese para que se possa determinar se elas realmente são ou não necessárias. Além disso, isso não garante minimidade, ou seja, que somente as partes essenciais serão selecionadas.

Neste artigo propõe-se um método para construir *auto-aproximações* que, quando usadas na síntese, levam diretamente ao resultado menos restritivo e controlável, sem a necessidade de verificação pós-síntese. A abordagem é estruturada sobre os conceitos de *evento-alvo* e *estado-alvo*, os quais fundamentam a identificação de partes específicas do modelo *distinguidor* que devem compor a planta para síntese, materializando a ideia de *auto-aproximação*. O resultado da abordagem é um controlador que se comporta equivalentemente

a um obtido usando todo o modelo do distinguidor, embora sua computação seja mais simples. Exemplos são fornecidos ao longo do artigo para ilustrar a abordagem.

O restante do documento está estruturado da seguinte forma: a Seção 2 aborda e exemplifica a fundamentação teórica do artigo; a Seção 3 introduz os principais resultados; e a Seção 4 discute algumas conclusões e perspectivas futuras.

## 2 Conceitos Preliminares

Na *Teoria de Controle Supervisório* (TCS) (Ramadge and Wonham, 1989), a planta de um *Sistema a Eventos Discretos* (SED) é modelada por um *Autômato Finito* (AF) (Cassandras and Lafortune, 2008), formalmente representado por uma tupla  $G = \langle \Sigma, Q, q^\circ, \rightarrow \rangle$ , em que  $\Sigma$  é o alfabeto de eventos,  $Q$  é um conjunto finito de estados,  $q^\circ \in Q$  é o estado inicial, e  $\rightarrow \subseteq Q \times \Sigma \times Q$  é a relação de transição.

Eventos são tomados de  $\Sigma$  e compõem cadeias, tal que  $\Sigma^*$  denota o conjunto de todas as cadeias, incluindo a cadeia vazia  $\varepsilon$ . Um subconjunto  $L \subseteq \Sigma^*$  é chamado de linguagem e  $L(G) \subseteq \Sigma^*$  denota a linguagem gerada por  $G$ , i.e., o conjunto de todas as cadeias  $s \in \Sigma^*$  possíveis em  $G$  (ou  $G \xrightarrow{s}$ ).

Para um estado  $q \in Q$  e um alfabeto  $\Sigma$ , denota-se por  $\Sigma(q)$  o conjunto dos eventos habilitados em  $q$ . A operação de *composição síncrona* de dois AFs,  $G^a$  e  $G^b$ , é denotada por  $G^a \parallel G^b$  e a mesma notação é utilizada para a composição entre linguagens. Duas cadeias  $s, t \in \Sigma^*$  podem ser concatenadas como  $st$  e o *prefixo-fechamento* de uma linguagem  $L$  é tal que  $\bar{L} = \{s \in \Sigma^* \mid st \in L \text{ para algum } t \in \Sigma^*\}$ . Neste artigo, somente linguagens prefixo-fechada são consideradas.

Para o propósito de controle, o conjunto de eventos é particionado nos conjuntos  $\Sigma_c$  e  $\Sigma_u$ , incluindo os eventos *controláveis* e *não-controláveis*, respectivamente. Os eventos em  $\Sigma_c$  pode ser desabilitados pelo controlador, enquanto os eventos em  $\Sigma_u$  ocorrem espontaneamente. Uma linguagem de *especificação* prefixo-fechada  $K \subseteq \Sigma^*$  é *controlável* (Ramadge and Wonham, 1989) e.r.a. uma planta prefixo-fechada  $L(G) \subseteq \Sigma^*$  se  $K\Sigma_u \cap L(G) \subseteq K$ , i.e., se cada evento não-controlável, possível na planta, também é possível em  $K$ . Se  $K$  não é controlável, ela pode ser reduzida à sua *máxima sub-linguagem controlável*, denotada por  $\sup\mathcal{C}(K, G) = \cup\{K' \subseteq K \mid K' \text{ é controlável e.r.a. } L(G)\}$  (Ramadge and Wonham, 1989). O processo de cálculo de  $\sup\mathcal{C}(K, G)$  é chamado de *síntese*.

### 2.1 TCS com distinção de eventos

Para um SED modelado por um autômato  $G$ , com eventos em  $\Sigma$ , assume-se que cada evento  $\sigma \in \Sigma$  esteja associado a um conjunto de eventos  $\Delta^\sigma \neq \emptyset$ ,

tal que para todo  $\sigma_1, \sigma_2 \in \Sigma$ , com  $\sigma_1 \neq \sigma_2$ ,  $\Delta^{\sigma_1} \cap \Delta^{\sigma_2} = \emptyset$ . Eventos em  $\Delta^\sigma$  refinam a ocorrência de  $\sigma$  no sistema (Cury et al., 2015), tornando  $\Sigma$  um conjunto de máscaras para os eventos em  $\Delta = \Delta_c \cup \Delta_u$ , para  $\Delta_u = \bigcup_{\sigma \in \Sigma_u} \Delta^\sigma$  e  $\Delta_c = \bigcup_{\sigma \in \Sigma_c} \Delta^\sigma$ .

A relação entre  $\Sigma$  e  $\Delta$  pode ser definida pelos mapas: (i) *mascarador*  $\Pi : \Delta^* \rightarrow \Sigma^*$ , definido por  $\Pi(\varepsilon) = \varepsilon$  e  $\Pi(t\delta) = \Pi(t)\sigma$ , para  $t \in \Delta^*$ ,  $\delta \in \Delta^\sigma$  e  $\sigma \in \Sigma$ ; (ii) *mascarador inverso*  $\Pi^{-1} : \Sigma^* \rightarrow 2^{\Delta^*}$ , definido para qualquer cadeia  $s \in \Sigma^*$  como  $\Pi^{-1}(s) = \{t \in \Delta^* \mid \Pi(t) = s\}$ . O mapa  $\Pi$  pode ser estendido para qualquer linguagem  $L^d \subseteq \Delta^*$ , como  $\Pi(L^d) = \{s \in \Sigma^* \mid \exists t \in L^d, \Pi(t) = s\}$ . O mapa  $\Pi^{-1}$  também pode ser estendido para qualquer linguagem  $L \subseteq \Sigma^*$  por  $\Pi^{-1}(L) = \{t \in \Delta^* \mid \Pi(t) \in L\}$ .

Dado um alfabeto  $\Sigma$ , e sua versão refinada  $\Delta$ , um *distinguidor* é um mapa  $D : \Sigma^* \rightarrow 2^{\Delta^*}$  definido, para todo  $s \in \Sigma^*$ ,  $\sigma \in \Sigma$ ,  $r \in \Delta^*$  e  $\delta \in \Delta$ , por: (i)  $D(\varepsilon) = \{\varepsilon\}$ ; (ii) se  $r\delta \in D(s\sigma)$  então  $r \in D(s)$  e  $\delta \in \Delta^\sigma$ ; (iii) se  $r \in D(s)$  então  $\exists \delta' \in \Delta^\sigma$  tal que  $r\delta' \in D(s\sigma)$  (Cury et al., 2015). Ou seja,  $D$  é um mapa prefixo-preservante que sempre distingue e nunca desabilita totalmente uma máscara.  $D$  simplesmente substitui cada cadeia em  $\Sigma^*$  por um conjunto não vazio de cadeias em  $\Delta^*$ .  $D$  pode ser estendido para qualquer linguagem  $L \subseteq \Sigma^*$  por  $D(L) = \{r \in \Delta^* \mid \exists s \in L, r \in D(s)\}$ . Pode ser mostrado (Cury et al., 2015) que  $\Pi(D(L)) = L$  e, para uma linguagem  $L_d = D(\Sigma^*)$ ,  $\Pi(L_d) = \Sigma^*$ . Observe que  $D$  aproxima cadeias em  $\Sigma^*$ , e o mapa  $\Pi^{-1}$  torna-se um caso particular do distinguidor (quando  $L_d = \Delta^*$ ). Outro caso particular é quando, para todo  $s \in \Sigma^*$ ,  $|D(s)| = 1$ . Nesse caso  $D$  é dito ser *preditível* e qualquer  $s \in \Sigma^*$  é mapeada em exatamente um  $r \in \Delta^*$  ou, ainda, após qualquer cadeia  $r \in L_d$ , existe exatamente um evento  $\delta \in \Delta^\sigma$  elegível para cada  $\sigma \in \Sigma$ .

Dado um SED modelado por  $G$ , o efeito de  $D$  sobre  $G$  pode ser expressado por  $D(L(G)) = \Pi^{-1}(L(G)) \cap L_d$ . A mesma notação é assumida para AFs e  $\Pi^{-1}(G)$  é um modelo que substitui os eventos de cada transição em  $G$  por seu respectivo conjunto de refinamentos, e  $H_d$  é o autômato para  $L_d$ , de modo que  $G_d = D(G) = \Pi^{-1}(G) \parallel H_d$  modela a planta distinguida.

Para as plantas  $G$  e  $G_d$ , sejam  $E$  e  $E_d$  especificações tais que  $K = E \parallel G$  e  $K_d = G_d \parallel E_d$ . Pode ser mostrado (Cury et al., 2015) que, se  $D$  é preditível e se  $K = \Pi(K_d)$ , então  $\Pi(\sup\mathcal{C}(K_d, G_d)) = \sup\mathcal{C}(K, G)$  e  $\sup\mathcal{C}(K_d, G_d) = D(\sup\mathcal{C}(K, G))$ .

Tornar  $D$  preditível é uma tarefa de modelagem que, em geral, pode ser feita de forma modular. Nesse artigo assume-se que, para qualquer  $\sigma \in \Sigma$ , cada evento  $\delta \in \Delta^\sigma$  está implicitamente associado a um submodelo distinguidor  $H_d^\delta$ , de modo que um distinguidor preditível  $D$  pode ser modelado por um composição  $H_d = \parallel H_d^i$ ,  $i = 1, \dots, n$ , para  $L(H_d) = L_d \subseteq \Delta^*$ .

## 2.2 TCS com Aproximações

Distinguidores preditíveis são normalmente associados a modelos extensos, com muitos estados, condição necessária para projetar eventos de  $\Sigma$  em instâncias únicas em  $\Delta$ . De certa forma, isso anula parte das vantagens de modelagem trazidas pelos refinamentos. Aproximações são alternativas que tornam distinguidores eficazes também na síntese.

Formalmente, seja  $D : \Sigma^* \rightarrow 2^{L_d}$  um distinguidor preditível para uma planta  $G$  e seja  $D_a : \Sigma^* \rightarrow 2^{L_{da}}$  um distinguidor tal que  $L_d \subseteq L_{da} \subseteq \Delta^*$ . Então, qualquer AF  $G_a = D_a(G)$  é uma *Aproximação* para  $G_d = D(G)$ .

Usar  $L_{da}$  como modelo de aproximação para  $L_d$  implica que  $G_a$  tende a ter menos estados que  $G_d$ . Por exemplo, a aproximação  $G_a$ , para  $L_{da} = \Delta^*$ , tem o mesmo número de estado de que  $G$  e, portanto, seria uma escolha natural para a síntese  $\text{sup}\mathcal{C}(K_a, G_a)$ . No entanto, pode ser mostrado (Cury et al., 2015) que, em certos casos,  $\text{sup}\mathcal{C}(K_a, G_a) \cap L_d \subseteq \text{sup}\mathcal{C}(K_d, G_d)$  e, portanto, o resultado de síntese pode ser subótimo. O exemplo apresentado a seguir ilustra essa propriedade.

## 2.3 Um exemplo

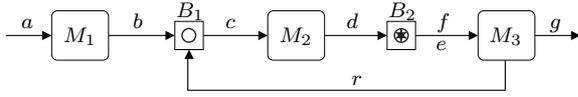


Figura 1: Sistema de manufatura com retrabalho

Considere o sistema de manufatura com retrabalho mostrado na Fig. 1, adaptado de (Teixeira et al., 2014), o qual é composto por três máquinas,  $M_1$ ,  $M_2$  e  $M_3$ , interconectadas por dois *buffers* unitários,  $B_1$  e  $B_2$ . O *buffer*  $B_1$  recebe novas peças (evento  $b$ ) ou peças a serem retrabalhadas (evento  $r$ ). A máquina  $M_2$  remove peças de  $B_1$  (evento  $c$ ), manufatura e as deposita em  $B_2$  (evento  $d$ ), onde um teste de qualidade determina qual operação será executada por  $M_3$  (eventos  $e$  ou  $f$ ), que resulta na liberação da peça (evento  $g$ ) ou no retorno para retrabalho (evento  $r$ ).  $G^1$ ,  $G^2$  e  $G^3$  na Fig. 2 modelam  $M_1$ ,  $M_2$  e  $M_3$ , respectivamente.

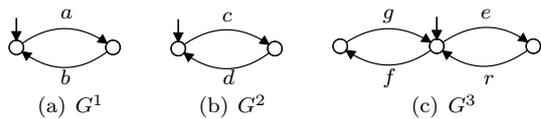


Figura 2: Modelos para  $M_1$ ,  $M_2$  e  $M_3$

A planta do sistema é dada pela composição  $G = G^1 \| G^2 \| G^3$ , com  $\Sigma = \{a, b, c, d, e, f, g, r\}$ , para  $\Sigma_c = \{a, c, e, f\}$  e  $\Sigma_u = \{b, d, g, r\}$ .

Os objetivos de controle para este exemplo são: (i) evitar o *underflow* e *overflow* em  $B_j$ ,  $j =$

1, 2; (ii) limitar a um o número de ciclo de retrabalho de cada peça. O requisito (i) é implementado pelos modelos  $E^1$  e  $E^2$  da Fig. 3.

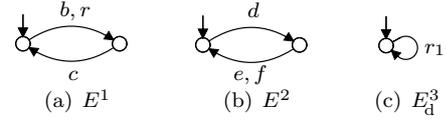


Figura 3: Modelos para  $E^1$ ,  $E^2$  e  $E_d^3$

Já o requisito (ii) (modelo  $E^3$ ) requer memorizar a quantidade de retrabalhos para cada peça inserida em  $B_1$ , o que pode levar a um modelo extenso, visto que o sistema pode manufaturar até três peças em paralelo. Esse modelo pode ser simplificado por refinamentos que identifiquem quantas vezes cada peça foi retrabalhada.

Assim, seja  $\Delta^\sigma = \{\sigma_0, \sigma_1\}$ , para  $\sigma \in \{c, d, e\}$ , e  $\Delta^r = \{r_1, r_2\}$ , tal que 0, 1 e 2 indicam a quantidade de retrabalhos efetuados. Então, a versão refinada de  $\Sigma$  é dada por  $\Delta = \bigcup_{\sigma \in \Sigma} \Delta^\sigma$  e, utilizando-se das informações em  $\Delta$ ,  $E_d^3$  pode ser modelado por  $E_d^3$  da Fig. 3, para  $\Delta_{E_d^3} = \Delta^r$ . O modelo  $E_d^3$  desabilita o evento  $r_2$  de modo que o segundo ciclo de retrabalho é evitado. As versões  $E^1$  e  $E^2$  passam a ser modeladas em  $\Delta$  por  $E_d^1 = \Pi^{-1}(E^1)$  e  $E_d^2 = \Pi^{-1}(E^2)$ , e a especificação global é dada pela composição  $E_d = E_d^1 \| E_d^2 \| E_d^3$ .

Resta mostrar como os eventos em  $\Delta$  podem ser distinguidos. Essa tarefa é atribuída aos distinguidores  $H_d^c$ ,  $H_d^d$ ,  $H_d^e$  e  $H_d^r$  da Fig. 4. O distinguidor preditível, para este exemplo, é dado pela composição  $H_d = H_d^c \| H_d^d \| H_d^e \| H_d^r$ .

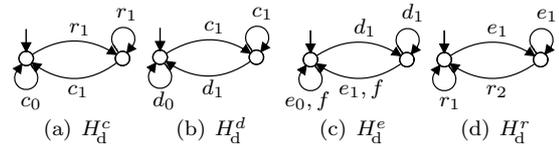


Figura 4: Versão modular do distinguidor  $H_d$

Agora, para  $G_a = \Pi^{-1}(G)$  e  $K_a = E_d \| G_a$ , sintetiza-se um supervisor  $\text{sup}\mathcal{C}(K_a, G_a)$ , modelado por  $V_a$  na segunda linha da Tabela 1. Para fins comparativos com o caso preditível, pode ser também calculado um supervisor  $\text{sup}\mathcal{C}(K_d, G_d)$ , para  $G_d = G_a \| H_d$  e  $K_d = E_d \| G_d$ , o qual é modelado por  $V_d$  na primeira linha da Tabela 1.

Tabela 1: Síntese - estados (transições)

$G_d$	$E_d$	$K_d$	$V_d$
84 (276)	4 (18)	192 (364)	26 (45)
$G_a$	$E_d$	$K_a$	$V_a$
12 (60)	4 (18)	48 (132)	18 (40)
			$V_a \  H_d$
			18 (32)

Note que a síntese de  $V_a$ , a partir de um modelo com 48 estados, é mais simples do que o caso

preditível  $V_d$ , com 192 estados. No entanto, a versão implementável do controlador,  $V_a \parallel H_d$ , é mais restritiva do que  $V_d$ , pois possui apenas 18 estados enquanto  $V_d$  possui 26 estados. Com o intuito de obter um supervisor  $V_a'$ , tal que  $V_a' \parallel H_d = V_d$ , partes do distinguidor podem ser usadas para enriquecer a aproximação mantendo-a, ao mesmo tempo, mais simples do que o caso preditível. Resta descobrir quais partes de  $H_d$  podem ser úteis a este propósito, o que é investigado a seguir.

### 3 TCS com Auto-Aproximações

A combinação dos conceitos de *distinguidor* e *aproximação* leva a um mecanismo de síntese que mantém os benefícios de modelagem trazidos por refinamentos de eventos e, ao mesmo tempo, tende a simplificar o cálculo do controlador. No entanto, essa abordagem pode resultar em soluções mais restritivas, devido ao baixo nível de distinção, usual de uma planta aproximada (exemplo).

Considerando que um distinguidor possa ser exposto de forma modular, uma possível alternativa para enriquecer uma aproximação, i.e., aumentar o seu poder de distinção, é selecionar partes específicas para serem compostas à planta.

Resta descobrir quais módulos poderiam ser compostos à planta, o que pode não ser trivial e, em geral, é associado a um processo de tentativa e erro. Na sequência, será descrita uma sistemática que leva ao conceito de *auto-aproximações*, ou seja, plantas que por construção garantem a síntese controlável e menos restritiva. As noções de *estados alvo* e *eventos alvo* são a chave para se definir auto-aproximações.

#### 3.1 Estados alvo

Quando uma aproximação é usada em síntese, ao menos três classes de estados podem ser identificadas, de acordo com a maneira como as transições (rotuladas com eventos refinados) são habilitadas.

**Definição 1 (Estados alvo)** Para uma planta  $G$ , com eventos em  $\Sigma$ , e um distinguidor preditível  $D$ , seja  $G_a$  uma aproximação para  $G_d = D(G)$ , e seja  $E_d$  um modelo de especificação. Defina os seguintes conjuntos de estados:

$S^u$  (Estados Certamente Não-Controláveis):

$$S^u = \{(x, y) \in Q_{G_a} \times Q_{E_d} \mid \text{existe } s \in \Delta^* \text{ e } \mu \in \Sigma_u \text{ tal que para todo } \mu_i \in \Delta^\mu, \text{ se } G_a \xrightarrow{s} x \xrightarrow{\mu_i} x' \text{ então } E_d \xrightarrow{s} y \xrightarrow{\mu_i} y'\}.$$

$S^e$  (Estados Eventualmente-Controláveis):

$$S^e = \{(x, y) \in Q_{G_a} \times Q_{E_d} \mid (x, y) \notin S^u \text{ e existe } \mu \in \Sigma_u, s \in \Delta^* \text{ e } \mu_i, \mu_j \in \Delta^\mu, \text{ para } \mu_i \neq \mu_j, \text{ tal que } G_a \xrightarrow{s} x \xrightarrow{\mu_i, \mu_j} x', E_d \xrightarrow{s} y \xrightarrow{\mu_i} y', \text{ e } E_d \xrightarrow{s} y \xrightarrow{\mu_j} y'\}.$$

$S^c$  (Estados Certamente-Controláveis):

$$S^c = \{(x, y) \in Q_{G_a} \times Q_{E_d} \mid (x, y) \notin \{S^u \cup S^e\}\}.$$

Os estados de uma composição  $G_a \parallel E_d$  são separados de acordo com o nível de restrições imposto sobre as transições que partem de cada estado. Quando uma restrição é não-ambígua e r.a. eventos de um mesmo conjunto de refinamentos (que é o caso de  $S^u$  e  $S^c$ ), a interpretação do conceito de mau-estado no procedimento de síntese segue como usual. Do contrário (caso  $S^e$ ), a síntese pode levar à diferentes desfechos, pois um estado pode ser classificado como proibido por efeito de um evento que pode nem mesmo ser habilitado pela planta. Assim, ao elevar o grau de distinção da planta, eleva-se a expectativa de que mais estados possam ser mantidos no controlador final.

**Exemplo 1 (Estados alvo)** Seja  $G_a = \Pi^{-1}(G)$  e  $E_d$  como na Fig. 5, com  $\Delta^\beta = \{\beta\}$ ,  $\Delta^\mu = \{\mu_a, \mu_b, \mu_c\}$  e  $\Delta^\alpha = \{\alpha\}$ , para  $\beta \in \Sigma_c$  e  $\mu, \alpha \in \Sigma_u$ .

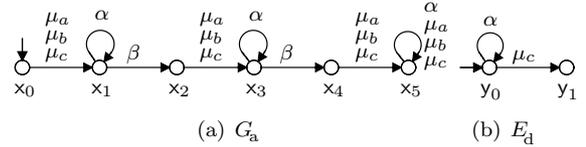


Figura 5: Modelos para  $G_a$  e  $E_d$

Denota-se por  $(x_i, y_j)$ , para  $i = 0, \dots, 5$  e  $j = 0, 1$ , os estados em  $Q_{G_a} \times Q_{E_d}$ . Note que  $E_d$  desabilita os eventos  $\mu_c$  e  $\alpha$  no estado  $y_1$ . Portanto, os estados  $(x_i, y_1)$  são candidatos a maus estados. Como  $(x_m, y_1)$ , para  $m \in \{1, 3, 5\}$ , desabilitam todo o conjunto  $\Delta^\alpha$ , então eles são certamente maus estados, i.e.,  $(x_m, y_1) \in S^u$ .

Diferentemente, os estados  $(x_n, y_1)$ , com  $n \in \{2, 4\}$ , desabilitam o evento  $\mu_c$  após sua primeira ocorrência, enquanto que  $\Delta^\mu \setminus \{\mu_c\}$  estão livres para ocorrer. Como a síntese é incapaz de reconhecer qual evento em  $\Delta^\mu$  é realmente elegível, então  $(x_n, y_1)$  podem ser maus estados para  $\mu_c$ , mas são certamente seguros para os demais refinamentos em  $\Delta^\mu$ . Assim,  $(x_n, y_1) \in S^e$ .  $\square$

#### 3.2 Eventos Alvo

A noção de *eventos alvo* pode ser apresentada como uma função dos estados alvo.

**Definição 2 (Eventos Alvo)** Para uma planta  $G$ , com eventos em  $\Sigma$ , e um distinguidor preditível  $D$ , seja  $G_a$  uma aproximação para  $G_d = D(G)$  e seja  $E_d$  um modelo de especificação para  $G_a$ , tal que  $K_a = G_a \parallel E_d$ . Um evento  $\mu_j \in \Delta^\mu \subseteq \Delta_u$ , para  $\mu \in \Sigma_u$ , é dito ser alvo em  $G_a$  se ele leva um estado de  $K_a$  a pertencer a  $S^e$ , i.e., se  $G_a \xrightarrow{s} x \xrightarrow{\mu_j} x'$  e  $E_d \xrightarrow{s} y \xrightarrow{\mu_j} y'$ , e  $(x, y) \in S^e$ .

Denota-se por  $\Delta_{\mathcal{V}}(G_a)$  o conjunto de todos os eventos alvos em  $G_a$ .

**Exemplo 2 (Eventos alvo)** No exemplo da Fig. 5, o evento  $\mu_c$  é alvo em  $G_a$ , uma vez que esse evento leva os estados  $(x_n, y_1)$ , com  $n \in \{2, 4\}$ , a serem incluídos em  $\mathcal{S}^e$ . De fato,  $(x_n, y_1)$  são estados seguros para as transições rotuladas com  $\mu_a$  e  $\mu_b$  que partem desses estados, mas inseguros para o rótulo  $\mu_c$ .

Tal ambiguidade não aconteceria no caso de  $G_a$  estar associada ao distinguidor preditível. Note que, ao utilizar uma planta  $G_d$  na síntese, o supervisor resultante é diferente de quando  $G_a$  é utilizado, como ilustrado na Fig. 6.

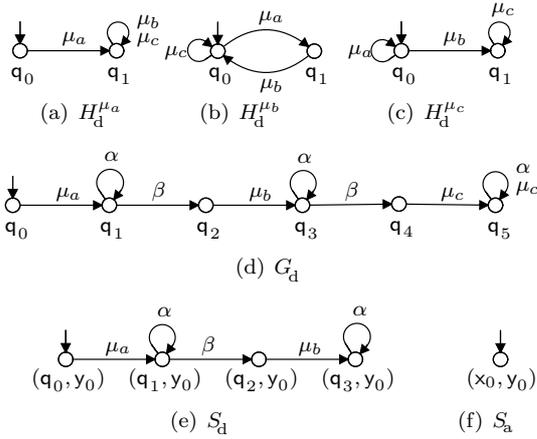


Figura 6: Distinguidores  $H_d^{\mu_a}$ ,  $H_d^{\mu_b}$  e  $H_d^{\mu_c}$ , planta  $G_d$ , e supervisores  $S_d$  e  $S_a$ , com  $\Delta_{S_a} = \Delta$

Como  $G_d$  “reconhece” que  $\mu_c$  é possível somente após  $\mu_a$  e  $\mu_b$ , então  $S_d$  evita  $\mu_c$  (desabilitando  $\beta$ ) somente quando ele seria realmente elegível. Diferentemente,  $G_a$  habilita  $\mu_c$  já no estado inicial e um sistema sob o controle de  $S_a$  não inicia.  $\square$

### 3.3 Auto-aproximação

Quando  $\Delta_{\mathcal{V}}(G_a) = \emptyset$ , para uma dada aproximação  $G_a$ , é sempre possível determinar univocamente se um dado estado sobrevive ou não sob controle e nenhuma parte do distinguidor é necessária na síntese. Porém, quando  $\Delta_{\mathcal{V}}(G_a) \neq \emptyset$ ,  $G_a$  tem que ser melhorada com alguma parte do distinguidor.

Neste artigo, mostra-se que  $G_a$  pode ser melhorada de tal forma que possa distinguir univocamente somente os eventos alvo. Ou seja,  $G_a$  é construída de modo a se tornar preditível e.r.a. um evento em particular, ou a um conjunto de eventos, mesmo que ainda possa permanecer não preditível para outros. Essa ideia relaxa a propriedade de preditibilidade (total) da subseção 2.1, e introduz a *preditibilidade parcial* como se segue.

**Definição 3 (preditibilidade-parcial)** Seja  $D_a$  um distinguidor (subseção 2.2) modelado por

$L(H_{da}^i = \|H_{da}^i) = L_{da}$ , para  $L_{da} \subseteq \Delta^*$ . Para  $\sigma \in \Sigma$ ,  $\delta^i \in \Delta^\sigma$ , uma planta  $G$  e  $G_a = D_a(G)$ ,  $D_a$  é dito ser  $\delta^i$ -preditível se, para todo  $t \in \Delta^*$ ,

$$t\delta^i \in L(G_a) \Rightarrow |t\Delta^\sigma \cap L(G_a)| = 1.$$

Ou seja,  $D_a$  é  $\delta^i$ -preditível quando, após qualquer cadeia em  $\Delta^*$ , se  $\delta^i$  é elegível sob a distinção de  $D_a$ , ele é único entre os eventos em  $\Delta^\sigma$ . Para qualquer conjunto  $\Delta' \subseteq \Delta$ ,  $D_a$  é dito ser  $\Delta'$ -preditível se é  $\delta$ -preditível para todo  $\delta \in \Delta'$ . A mesma notação é generalizada para autômatos.

**Lema 1** Dados dois autômatos  $G^1$  e  $G^2$  modelados com eventos em  $\Delta$ , seja  $G^1$   $\delta$ -preditível, para  $\delta \in \Delta$ . Então,  $G^1 \| G^2$  é  $\delta$ -preditível.

### Exemplo 3 (Preditibilidade-parcial)

Considere uma planta  $G$  e um distinguidor  $D_a$ , modelado por  $L(H_{da}) = L_{da} = \Delta^*$ , tal que  $G_a = D_a(G)$  é como mostrado na Fig. 5 (a).

Note, por exemplo, que  $G_a$  não é  $\mu_c$ -preditível, pois após qualquer cadeia  $t \in L(G_a)$ , sempre que uma transição com eventos em  $\Delta^\mu$  é possível,  $|t\Delta^\mu \cap L(G_a)| > 1$  e, portanto, essa versão de  $G_a$  não pode distinguir  $\mu_c$  em  $\Delta^\mu$ . Diferentemente,  $G_a$  poderia ser melhorada por um distinguidor para  $\mu_c$  a fim de torná-la  $\mu_c$ -preditível. Considere que  $D_a'$  é modelado por  $L_{da}' = L_{da} \| H_d^{\mu_c}$ , para  $H_d^{\mu_c}$  como na Fig. 6 (c), tal que  $G_a' = G_a \| H_d^{\mu_c}$  é mostrada na Fig. 7.

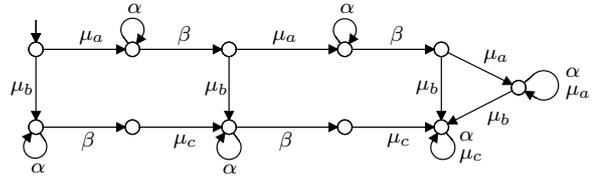


Figura 7: Modelo de  $G_a' = G_a \| H_d^{\mu_c}$

Observe que, agora após qualquer cadeia  $t \in L(G_a')$ , se  $\mu_c$  é elegível (estados alcançados pelas cadeias  $\mu_b\alpha^*\beta$  e  $\mu_a\alpha^*\beta[\mu_b\alpha^*\beta + \mu_a\alpha^*\beta\mu_b]$ ),  $|t\Delta^\mu \cap L(G_a')| = 1$ .  $\square$

### 3.4 Construindo Auto-aproximações

Intuitivamente, uma aproximação apropriada em síntese, aqui chamada de *aproximação-candidata*, é uma construída pela adição à planta de distinguidores que tornam os eventos alvos preditíveis, como formalizado a seguir.

**Definição 4 (Aproximação-candidata)** Para uma planta preditível  $G_d$  e uma aproximação  $G_a = \Pi^{-1}(G)$ , seja  $\Delta_{\mathcal{V}}(G_a)$  o conjunto de eventos alvo em  $G_a$  e seja  $H_d^\delta$ , para todo  $\delta \in \Delta$ , um conjunto de distinguidores  $\delta$ -preditíveis para  $G_a$ . Defina:

$$\mathcal{G}_c = G_a \| \left[ \begin{array}{c} \parallel \\ \delta \in \Delta_{\mathcal{V}}(G_a) \end{array} H_d^\delta \right]$$

com uma aproximação-candidata para  $G_d$ .

Como  $\mathcal{G}_c$  é  $\Delta_{\mathcal{V}}(G_a)$ -preditível, é intuitivo pensar em utilizá-la como auto-aproximação. No entanto, ao usar  $\mathcal{G}_c$  na síntese, pode ocorrer que eventos anteriormente não alvos tornem-se alvos indiretos. É o caso, por exemplo, de eventos não-controláveis que precederem os eventos tidos até então como alvos. Tais eventos devem ser igualmente distinguidos. Os conceitos de estado *local*, *global*, e *sub-estado* (Åkesson et al., 2002), são usados para identificar alvos indiretos.

### Definição 5 (Global, local e sub-estado)

Para um conjunto de autômatos  $G = \{G^1, \dots, G^n\}$ , seja  $G' = \{G^{1'}, \dots, G^{n'}\}$  um subconjunto de  $G$ . Defina:

- estado-local como um estado  $q_l \in Q_{G^i}$ , para um  $G^i \in G$ ;
- estado-global como um estado  $q_g \in Q_{G^t}$ , para  $G^t = G^1 \parallel \dots \parallel G^n$ ;
- sub-estado com um estado  $q_s \in Q_{G^p}$ , para  $G^p = G^{1'} \parallel \dots \parallel G^{n'}$ .

Denote por  $sS(q_g, G^p)$  a função que retorna um sub-estado  $q_s \in Q_{G^p}$  associado a um estado global  $q_g \in Q_{G^t}$ .

### Definição 6 (Eventos Indiretamente Alvos)

Dada uma planta preditível  $G_d$  e uma especificação  $E_d$ , seja  $G_a = \Pi^{-1}(G)$  uma aproximação para  $G_d$ . Seja  $\Delta_{\mathcal{V}}(G_a)$  o conjunto de eventos alvo em  $G_a$ , e  $\mathcal{G}_c$  uma aproximação  $\Delta_{\mathcal{V}}(G_a)$ -preditível como na Def. 4. Para  $\delta \in \Delta_{\mathcal{V}}(G_a)$  e  $t = t'c \in L(\mathcal{G}_c)$ , com  $t' \in \Delta^*$  e  $c \in \Delta_c$ , os Eventos Indiretamente Alvos de  $\mathcal{G}_c$  e.r.a.  $\Delta_{\mathcal{V}}(G_a)$  são todos os eventos que formam cadeias  $\gamma \in \Delta_u^*$ , tal que  $\mathcal{G}_c \xrightarrow{t} q \xrightarrow{\gamma} q' \xrightarrow{\delta} q''$  e  $E_d \xrightarrow{t} y \xrightarrow{\gamma} y' \xrightarrow{\delta} y''$  e  $sS((q', y'), G_a \parallel E_d) \in \mathcal{S}^e$ .

Denote por  $\mathcal{I}_{\mathcal{V}}(\mathcal{G}_c, \Delta_{\mathcal{V}}(G_a))$  a função que retorna o conjunto de eventos indiretamente alvos de  $\mathcal{G}_c$  e.r.a.  $\Delta_{\mathcal{V}}(G_a)$ , denotado por  $\Delta_{\mathcal{I}\mathcal{V}}(\mathcal{G}_c)$ . Assim o seguinte resultado pode ser apresentado.

**Lema 2** Para uma dada aproximação  $G_a$  e uma especificação  $E_d$ , sejam  $\Delta_{\mathcal{V}}(G_a)$  e  $\Delta_{\mathcal{I}\mathcal{V}}(G_a)$  respectivamente os conjuntos de eventos alvo e indiretamente alvo de  $G_a$ . Pode ser mostrado que, se  $\Delta_{\mathcal{V}}(G_a) = \emptyset$ , então  $\Delta_{\mathcal{I}\mathcal{V}}(G_a) = \emptyset$ .

**Prova:** Como  $\Delta_{\mathcal{V}}(G_a) = \emptyset$ , então  $\mathcal{S}^e = \emptyset$ , pela Def. 2, o que implica que, para qualquer estado global  $q_g$  em  $G_a \parallel E_d$ ,  $sS(q_g, G_a \parallel E_d) \notin \mathcal{S}^e$ .  $\square$

Agora, utiliza-se esse raciocínio para mostrar que, se  $\Delta_{\mathcal{V}}(G_a)$  é vazio, então  $G_a$  e  $G_d$  levam ao supervisor controlável menos restritivo.

**Teorema 3** Dada uma planta preditível  $G_d$  e uma especificação  $E_d \subseteq \Delta^*$ , com  $K_d = E_d \cap L(G_d)$ , seja  $G_a$  uma aproximação para  $G_d$ , tal que  $K_a = E_d \cap L(G_a)$ . Para os conjuntos  $\Delta_{\mathcal{V}}(G_a)$  e  $\Delta_{\mathcal{I}\mathcal{V}}(G_a)$

de eventos alvos e indiretamente alvos em  $G_a$  (definições 2 e 6), se  $\Delta_{\mathcal{V}}(G_a) \cup \Delta_{\mathcal{I}\mathcal{V}}(G_a) = \emptyset$ , então  $\sup\mathcal{C}(K_a, G_a) \cap L_d = \sup\mathcal{C}(K_d, G_d)$ .

**Prova:** Para mostrar  $\sup\mathcal{C}(K_a, G_a) \cap L_d \subseteq \sup\mathcal{C}(K_d, G_d)$  (Cury et al., 2015), note que  $L(G_d) \subseteq L(G_a)$  por definição. Portanto, a controlabilidade de  $\sup\mathcal{C}(K_a, G_a)$  e.r.a.  $L(G_a)$  implica na controlabilidade de  $\sup\mathcal{C}(K_a, G_a)$  e.r.a.  $L(G_d)$ . Além disso, a controlabilidade de  $L_d$  e.r.a.  $L(G_d)$  segue de  $L(G_d) \subseteq L_d$ , e então  $\sup\mathcal{C}(K_a, G_a) \cap L_d$  é controlável e.r.a.  $L(G_d)$ . Assim,  $\sup\mathcal{C}(K_a, G_a) \cap L_d \subseteq K_a \cap L_d = K_d$  implica

$$\sup\mathcal{C}(K_a, G_a) \cap L_d \subseteq \sup\mathcal{C}(K_d, G_d). \quad (1)$$

Para mostrar  $\sup\mathcal{C}(K_d, G_d) \subseteq \sup\mathcal{C}(K_a, G_a) \cap L_d$ , seja  $s\mu_i \in \sup\mathcal{C}(K_d, G_d)$  para  $\mu_i \in \Delta^\mu$  e  $\mu \in \Sigma_u$ . Como  $\sup\mathcal{C}(K_d, G_d)$  é controlável e.r.a.  $L(G_d)$ , então  $G_d \xrightarrow{s} q \xrightarrow{\mu_i} q'$  e  $E_d \xrightarrow{s} y \xrightarrow{\mu_i} y'$ .

Pelo pressuposto  $\Delta_{\mathcal{V}}(G_a) = \emptyset$ , tem-se que para qualquer  $\mu_j \in \Delta^\mu$ , tal que  $G_a \xrightarrow{s} x \xrightarrow{\mu_i, \mu_j} x'$ , é verdade que  $E_d \xrightarrow{s} y \xrightarrow{\mu_i, \mu_j} y'$  pois, do contrário, a combinação de estados  $(x, y) \in \mathcal{S}^e$  e  $\Delta_{\mathcal{V}}(G_a) \neq \emptyset$ . Agora, para  $s = t\gamma$ , tal que  $t \in \Delta^*$  e  $\gamma \in \Delta_u^*$ ,  $t\gamma \in L(G_a)$  e  $G_a \xrightarrow{t} x \xrightarrow{\gamma} x' \xrightarrow{\mu_i, \mu_j} x''$ , implica que  $E_d \xrightarrow{t} y \xrightarrow{\gamma} y' \xrightarrow{\mu_i, \mu_j} y''$ , visto que  $\Delta_{\mathcal{I}\mathcal{V}}(G_a) = \emptyset$ , por suposição.

Logo,  $s\mu_i \in \sup\mathcal{C}(K_d, G_d)$  e  $\Delta_{\mathcal{V}}(G_a) = \emptyset$  implicam  $s\mu_i \in \sup\mathcal{C}(K_a, G_a)$ . Como ainda  $s\mu_i \in L(G_d) \subseteq L_d$ , então  $s\mu_i \in \sup\mathcal{C}(K_a, G_a) \cap L_d$ .  $\square$

Do Teorema 3, resta mostrar como se pode construir uma auto-aproximação  $\mathcal{G}_c$  que sempre leve aos conjuntos vazio de eventos alvo e indiretamente alvo, i.e.,  $\Delta_{\mathcal{V}}(\mathcal{G}_c) = \Delta_{\mathcal{I}\mathcal{V}}(\mathcal{G}_c) = \emptyset$ . O Alg. 1 sistematiza essa construção.

### Algoritmo 1 Computação de $\mathcal{G}_c$

**Entrada:** planta  $G_a = \Pi^{-1}(G)$ ; especificação  $E_d$ ; eventos  $\Sigma$ , eventos  $\Delta$ ;  $\mu \in \Sigma_u$ ; conjunto de distinguidores  $\delta$ -preditível  $H_d^\delta$ , para todo  $\delta \in \Delta$ , tal que  $H_d = \parallel H_d^\delta$  é preditível;

- 1:  $\mathcal{G}_c \leftarrow G_a$ ;  $\Delta_{\mathcal{V}}(\mathcal{G}_c) \leftarrow \emptyset$ ;  $\Delta_{\mathcal{I}\mathcal{V}}(\mathcal{G}_c) \leftarrow \emptyset$ ;
- 2: **para todo**  $(x, y) \in Q_{G_a} \times Q_{E_d}$  **faça**
- 3:   **se** (para  $\mu_i, \mu_j \in \Delta^\mu$  tal que  $\mu_i \neq \mu_j$ ,  $x \xrightarrow{\mu_i, \mu_j} x'$  e  $y \xrightarrow{\mu_i} y'$  e  $y \xrightarrow{\mu_j} y''$  e  $(x, y) \notin \mathcal{S}^u$ ) **então**
- 4:      $\Delta_{\mathcal{V}}(G_a) \leftarrow \Delta_{\mathcal{V}}(G_a) \cup \{\mu_j \in \Delta_u \mid \mu_j \text{ é alvo}\}$ ;
- 5: **fim para**
- 6: **para todo**  $\delta \in \Delta_{\mathcal{V}}(G_a)$  **faça**
- 7:    $\mathcal{G}_c \leftarrow \mathcal{G}_c \parallel H_d^\delta$ ;
- 8: **fim para**
- 9:  $\Delta_{\mathcal{I}\mathcal{V}}(\mathcal{G}_c) \leftarrow \mathcal{I}_{\mathcal{V}}(\mathcal{G}_c, \Delta_{\mathcal{V}}(G_a))$ ;
- 10: **para todo**  $\alpha \in \Delta_{\mathcal{I}\mathcal{V}}(\mathcal{G}_c)$  **faça**
- 11:    $\mathcal{G}_c \leftarrow \mathcal{G}_c \parallel H_d^\alpha$ ;
- 12: **fim para**
- 13: **retornar**  $\mathcal{G}_c$ ;

Inicialmente,  $\mathcal{G}_c$  é considerado como  $G_a = \Pi^{-1}(G)$  (linha 1). Computacionalmente, esse é o melhor caso, pois nenhum distinguidor é incluído na síntese ( $L_{da} = \Delta^*$ ) e verifica-se se o produto é o supervisor controlável e menos restritivo.

Para isso, visita-se cada estado de  $\mathcal{G}_c$  verificando se é *eventualmente controlável* ( $\mathcal{S}^e$  na linha 3). Se não há estados em  $\mathcal{S}^e$ , conseqüentemente não existem eventos alvos em  $G_a$  (pela Def. 2). Então, pelo Teorema 3, a síntese pode ser conduzida utilizando a planta  $\mathcal{G}_c = G_a = \Pi^{-1}(G)$ .

Diferentemente, a existência de um estado em  $\mathcal{S}^e$  sugere a reconstrução de  $\mathcal{G}_c$ , de modo a melhorar sistematicamente seu grau de distinção por meio da composição de distinguidores específicos (linhas 7 e 11). Tais modelos são selecionados de acordo com: (i) eventos alvo (linha 4); (ii) eventos indiretamente alvo (linha 9). Então, o seguinte resultado pode ser anunciado.

**Lema 4** *Seja uma planta  $G$ , uma aproximação  $G_a = \Pi^{-1}(G)$  e um conjunto de eventos alvo em  $G_a$  ( $\Delta_{\mathcal{V}}(G_a)$ ). Ao término do Alg. 1,  $\mathcal{G}_c$  é ( $\Delta_{\mathcal{V}}(G_a) \cup \Delta_{\text{ind}}(\mathcal{G}_c)$ )-preditível.*

**Prova:** Se segue do Lema 1, associado aos distinguidores  $\delta$ -preditíveis parametrizados como entrada do Alg. 1.  $\square$

**Exemplo 4 (Aproximação-alvo)** *No exemplo da Fig. 5, o evento  $\mu_c$  é alvo em  $G_a = \Pi^{-1}(G)$ , pela Def. 2. Assim,  $G_a$  é composta à  $H_d^{\mu_c}$  para formar a auto-aproximação  $\mathcal{G}_c$  da Fig. 7. O conjunto de eventos indiretamente alvo para  $\mathcal{G}_c$  é  $\Delta_{\text{ind}}(\mathcal{G}_c) = \emptyset$ , visto que o único estado  $q_i \in Q_{\mathcal{G}_c \| E_d}$ , tal que  $s\mathcal{S}(q_i, G_a \| E_d) \in \mathcal{S}^e$ , é alcançado pela cadeia  $\mu_b \alpha^* \beta \mu_c \beta$ , de modo que o único evento que será desabilitado para evitar esse mau estado é  $\beta \in \Delta_c$ .*

*Diferentemente de  $G_a$  (Fig. 5 (a)),  $\mathcal{G}_c$  (Fig. 7) somente permite  $\mu_c$  após  $\mu_b$ , exatamente do mesmo modo que em  $G_d$  da Fig. 6 (d). Como  $E_d$  implementa uma ação de controle sob o evento  $\mu_c$ , então a planta  $\mathcal{G}_c$  sabe exatamente quando  $\mu_c$  pode realmente ocorrer, independente do caminho que o levou a ocorrer. Dessa forma, se um evento  $\sigma_i \in \Delta_u$  é proibido por  $E_d$  em um dado estado  $q \in Q_{\mathcal{G}_c \| E_d}$ ,  $q$  é classificado como  $\mathcal{S}^u$  pela Def. 1, como ilustra a Fig. 8.*  $\square$

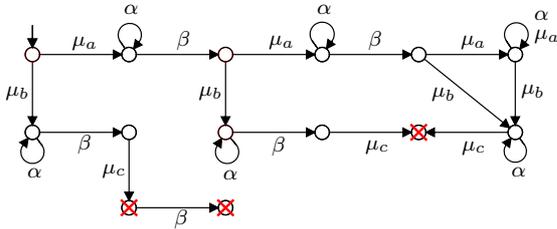


Figura 8: Modelo para  $\mathcal{G}_c \| E_d$

Observe que, do Alg. 1, a inclusão

$$L(G_d) \subseteq L(\mathcal{G}_c) \subseteq L(G_a) \subseteq \Delta^* \quad (2)$$

é verdadeira, i.e.,  $\mathcal{G}_c$  é simplesmente uma aproximação particular que trata dos eventos alvo e indiretamente alvo. Então, a seguinte propriedade pode ser apresentada para  $\mathcal{G}_c$ .

**Lema 5** *Para uma planta  $G$  e uma aproximação  $G_a = \Pi^{-1}(G)$ , seja  $\mathcal{G}_c$  uma aproximação computada como no Alg. 1. Se  $\mathcal{G}_c$  é  $\Delta_{\mathcal{V}}(G_a)$ -preditível, então  $\Delta_{\mathcal{V}}(\mathcal{G}_c) = \emptyset$ .*

**Prova:** Pelo Lema 4,  $\mathcal{G}_c$  é  $\Delta_{\mathcal{V}}(G_a)$ -preditível. Então para qualquer  $\mu_i \in \Delta_{\mathcal{V}}(G_a)$  e  $s \in \Delta^*$ , tal que  $\mathcal{G}_c \xrightarrow{s} q \xrightarrow{\mu_i} q'$  e  $E_d \xrightarrow{s} y \xrightarrow{\mu_i}$ , se segue que  $|\Delta(q) \cap \Delta^\mu| = 1$ , pela Def. 3. Então, pela Def. 1, a combinação de estados  $(q, y) \in \mathcal{S}^u$  e, generalizando,  $\Delta_{\mathcal{V}}(\mathcal{G}_c) = \emptyset$ , pela Def. 2.  $\square$

Finalmente, pode-se mostrar que uma *auto-aproximação* construída como no Algoritmo 1 sempre leva a síntese controlável menos restritiva.

**Teorema 6** *Para uma planta  $G$ , uma especificação  $E$ , e um distinguidor predictível  $D$ , seja  $G_d = D(G)$  e  $E_d \subseteq \Delta^*$  a versão de  $E$  modelada em  $\Delta$ , com  $K_d = E_d \cap L(G_d)$ . Para  $\mathcal{G}_c$  calculado como no Algoritmo 1, tal que  $K_c = E_d \cap L(\mathcal{G}_c)$ , é verdade que  $\sup\mathcal{C}(K_c, \mathcal{G}_c) \cap L_d = \sup\mathcal{C}(K_d, G_d)$ .*

**Prova:** Dos lemas 4 e 5, tem-se que  $\Delta_{\mathcal{V}}(\mathcal{G}_c) = \emptyset$ . Então, pelo Lema 2,  $\Delta_{\text{ind}}(\mathcal{G}_c) = \emptyset$  e a prova para o Teorema 6 segue diretamente do Teorema 3.  $\square$

### 3.5 Exemplo revisitado

Considere agora resolver o exemplo da subseção 2.3 usando uma auto-aproximação  $\mathcal{G}_c$  para  $G_d$ . Para isso, utiliza-se o Alg. 1 para classificar os estados de  $G_a \| E_d$ , com  $G_a = \Pi^{-1}(G)$ , e então construir a auto-aproximação.

Inicialmente, note que  $E_d^1 = \Pi^1(E^1)$ ,  $E_d^2 = \Pi^1(E^2)$  e  $E_d^3$  desabilitam eventos não-controláveis ( $\Delta^b$ ,  $\Delta^d$  e  $\Delta^r$ ). Então é possível que existam maus estados em  $G_a \| E_d$ . Porém, as especificações  $E_d^1$  e  $E_d^2$ , ao desabilitar um refinamento não-controlável, desabilitam todo o conjunto de eventos com a mesma máscara. Logo, os estados a partir dos quais esses eventos partem são incluídos em  $\mathcal{S}^u$  pela Def. 1.

Diferentemente, as combinações de estados para  $E_d^3$  são classificadas como eventualmente controlável ( $\mathcal{S}^e$  na linha 3), uma vez que,  $E_d^3$  desabilita o evento  $r_2$  enquanto  $r_1$  está habilitado. Então, pela Def. 2, o Alg. 1 identifica  $r_2$  como alvo em  $G_a$  (linha 4).

Para fins de ilustração, considere a composição  $G_a^3 \| E_d^3$  na Fig. 9 (a). Após  $\Delta^e$ , o estado alcançado é identificado como não-controlável e.r.a.  $r_2$ . No entanto, o mesmo estado pode ser visto como controlável e.r.a.  $r_1$ . Portanto, como a síntese com  $G_a$  não distingue  $r_2$  em  $\Delta^r$ , esse estado é removido, o que significa que o supervisor resultante irá restringir qualquer retrabalho de peças.

A fim de melhorar o supervisor, reconstrói-se  $G_a$  (linha 7) por meio da composição com  $H_d^r$  da Fig. 4 (d), resultando em uma aproximação  $\mathcal{G}_c$  que é  $\Delta_{\mathcal{V}}(G_a)$ -preditível (pelo Lema 4). A escolha

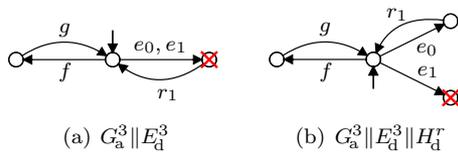


Figura 9: Modelos para  $G_a^3 || E_d^3$  e  $G_a^3 || E_d^3 || H_d^r$

por  $H_d^r$  é baseada na identificação do conjunto de eventos alvo  $\Delta_{\mathcal{V}}(G_a) = \{r_2\}$  do Alg. 1.

Uma comparação entre  $G_a$  e sua versão  $\Delta_{\mathcal{V}}(G_a)$ -preditível pode ser vista respectivamente nas figuras 9 (a) e 9 (b). Observe que, em 9 (b), o estado alcançado por  $e_0$  é seguro, diferentemente do caso da Fig. 9 (a). Isso significa que cada nova peça poderá ser retrabalhada no máximo uma vez.

A função  $\mathcal{I}_{\mathcal{V}}(\mathcal{G}_c, \Delta_{\mathcal{V}}(G_a))$  retorna  $\Delta_{\text{in}}(\mathcal{G}_c) = \{b\}$  (linha 9). Como o evento  $b$  não possui refinamento em  $\Delta$  então, por construção,  $G_a$  é  $\{b\}$ -preditível. Assim, a auto-aproximação para  $G_d$  é simplesmente  $\mathcal{G}_c = G_a || H_d^r$ .

Agora, para  $\mathcal{G}_c = G_a || H_d^r$  e  $\mathcal{K}_c = \mathcal{G}_c || E_d$ , sintetiza-se um supervisor  $\text{sup}\mathcal{C}(\mathcal{K}_c, \mathcal{G}_c)$ , modelado por  $V_a'$  na terceira linha da Tabela 2. As duas primeiras linhas reproduzem a Tabela 1.

Tabela 2: Síntese - estados (transições)

$G_d$	$E_d$	$K_d$	$V_d$	
84 (276)	4 (18)	192 (364)	26 (45)	
$G_a$	$E_d$	$K_a$	$V_a$	$V_a    H_d$
12 (60)	4 (18)	48 (132)	18 (40)	18 (32)
$\mathcal{G}_c$	$E_d$	$\mathcal{K}_c$	$V_a'$	$V_a'    H_d$
16 (72)	4 (18)	64 (160)	19 (42)	26 (45)

O supervisor  $V_a'$ , obtido de  $\mathcal{K}_c$  com 64 estados, é a solução controlável e menos restritiva, i.e.,  $V_a' || H_d = V_d$ . Além disso,  $\mathcal{K}_c$  é mais simples que  $K_d$ , com 192 estados, e essa diferença pode ser substancialmente mais acentuada para casos com maior quantidade de retrabalhos.

#### 4 Conclusões

Este artigo explora a síntese de controladores para SEDs usando o refinamento de eventos. Nessa abordagem, distinguidores facilitam tarefas de modelagem, mas eles carregam a síntese com modelos adicionais para identificar o contexto dos refinamentos. Já aproximações simplificam a síntese ao desconsiderar parte, ou todo, o modelo distinguidor, mas elas podem levar a resultados subótimos, além de serem construídas empiricamente.

Esse artigo mostra como construir um tipo particular de (auto)aproximação, que detém a propriedade de sempre levar a um supervisor controlável e maximamente permissivo, ao mesmo tempo em que preserva importantes ganhos computacionais na síntese. Essa propriedade é, em geral, associada ao nível de distinção de uma planta

refinada em relação a certos eventos. Então, é mostrado como construir uma versão dessa planta que incorpora tão somente as distinções necessárias à síntese.

Perspectivas de trabalhos futuros apontam para a possibilidade de modularizar a abordagem, considerar as marcações nos estados, além de prover o ferramental necessário para a prática de controle que incorpora essa proposta.

#### Agradecimentos

Este trabalho foi apoiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (projeto 402145/2016-0), pela Fundação Araucária (FA) e Financiadora de Estudos e Projetos (FINEP).

#### Referências

- Aguiar, R. S. S., Cunha, A. E. C., Cury, J. E. R. and Queiroz, M. H. (2013). Heuristic search of supervisors by approximated distinguishers, *Dependable Control of Discrete Systems (DCDS'13)*, York, UK, pp. 121–126.
- Bouzon, G., de Queiroz, M. H. and Cury, J. E. R. (2008). Supervisory control of DES with distinguishing sensors, *International Workshop on Discrete Event Systems, WODES'08*, Gothenburg, Sweden, pp. 22–27.
- Bouzon, G., de Queiroz, M. H. and Cury, J. E. R. (2009). Exploiting distinguishing sensors in supervisory control of DES, *Int. Conf. on Control and Automation, ICCA'09*, Christchurch, NZ, pp. 442–447.
- Cassandras, C. G. and Lafortune, S. (2008). *Introduction to Discrete Event Systems*, 2 edn, Springer Science, NY.
- Cury, J. E. R., de Queiroz, M. H., Bouzon, G. and Teixeira, M. (2015). Supervisory control of discrete event systems with distinguishers, *Automatica* **56**: 93 – 104.
- Åkesson, K., Flordal, H. and Fabian, M. (2002). Exploiting modularity for synthesis and verification of supervisors, *IFAC Proceedings Volumes* **35**(1): 175 – 180.
- Ramadge, P. J. and Wonham, W. M. (1989). The control of discrete event systems, *Discrete Event Dynamic Systems* **77**: 81–98.
- Teixeira, M., Malik, R., Cury, J. and de Queiroz, M. (2014). Supervisory control of DES with extended finite-state machines and variable abstraction, *Automatic Control, IEEE Transactions on* **60**(1): 118–129.