

# SÍNTESE MODULAR DO CONTROLE SUPERVISÓRIO EM DIAGRAMA ESCADA PARA UMA CÉLULA DE MANUFATURA

MAX H. DE QUEIROZ<sup>1</sup>, EDUARDO A. P. SANTOS<sup>2</sup>, JOSÉ E. R. CURY<sup>1</sup>

<sup>1</sup> *LCMI, Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina  
Florianópolis SC 88040-900 Brasil*

<sup>2</sup> *Laboratório de Automação e Sistemas, Pontifícia Universidade Católica do Paraná  
Curitiba PR 80215-901 Brasil*

*E-mails: max, eaps, cury@lcmi.ufsc.br*

**Resumo**— A aplicação da Teoria de Controle Supervisório numa célula de manufatura comandada por um controlador lógico programável (CLP) é apresentada. Do processo de ligação entre a teoria e a prática surgem diversos novos desafios e soluções que abrangem desde aspectos de modelagem do sistema em malha aberta até a estrutura final do programa em diagrama escada. Utiliza-se uma metodologia que explora a modularidade dos modelos da planta e das especificações para a síntese automática de supervisores minimamente restritivos com tamanhos reduzidos. Para implementar os supervisores em diagrama escada no CLP, desenvolve-se uma estrutura em três níveis que, além de executar a ação dos supervisores modulares, funciona como interface entre o modelo teórico e o sistema real. O funcionamento flexível e produtivo da célula de manufatura, após a execução prática do sistema de controle, e a clareza e flexibilidade do programa final do CLP são indicadores positivos da qualidade da metodologia usada.

**Abstract**— The Supervisory Control Theory is applied to a manufacturing cell commanded by a programmable logical controller (PLC). Several new challenges and solutions arise from the process of linking the theory to reality, concerning aspects from the open-loop system modeling to the final structure of the ladder diagram program. A methodology exploiting the modularity of the plant and of the behavioral specifications models is used to automatically synthesize minimally restrictive supervisors with reduced number of states. For the purpose of simplifying the ladder diagram implementation of supervisors in the PLC, the paper presents a three level structure that, beyond executing the modular supervisors action, works as an interface between the theoretical model and the real system. The flexible and productive manufacturing cell behavior, after the control system practical implementation, and the final PLC code readability and flexibility are positive quality indicators to the applied methodology.

**Keywords**— Supervisory control; manufacturing systems; control system synthesis; programmable logic controllers.

## 1 Introdução

A Teoria de Controle Supervisório (Ramadge e Wonham, 1989) tem sido desenvolvida nas últimas duas décadas como uma proposta de metodologia formal para a síntese automática de controladores ótimos para sistemas a eventos discretos (SEDs), entre os quais se inclui grande parte dos sistemas de manufatura. Essa teoria se fundamenta na modelagem da planta e das especificações de controle por linguagens controláveis, de modo que o supervisor resultante gere a máxima linguagem controlável contida nessas linguagens.

Entretanto, a aplicação da Teoria de Controle Supervisório em sistemas reais (Balemi et al., 1993) (Brandin, 1996) tem esbarrado na complexidade computacional dos cálculos envolvidos e no grande número de estados dos modelos resultantes. A razão para isso é que o número de estados do modelo que representa o sistema cresce exponencialmente em função do número de subsistemas que o compõem. Assim, a complexidade do cálculo de supervisores ótimos, embora polinomial no número de estados do modelo da planta e da especificação, é um fator limitante em aplicações reais, as quais são geralmente modeladas pela composição de múltiplos sistemas locais.

Em trabalhos anteriores (Queiroz e Cury, 2000a, 2000b e 2000c), os autores propõem tratar

este fator limitante explorando, além da modularidade das especificações (Wonham e Ramadge, 1988), a própria modularidade natural da planta em sistemas de maior porte. Ao invés de se construir um único controlador monolítico para toda a planta, na abordagem modular proposta procura-se construir, sempre que possível, um controlador local para cada especificação, modelando-o apenas em termos dos subsistemas afetados por sua ação. Neste caso, deseja-se que os supervisores resultantes sejam localmente modulares, isto é, que a ação conjunta dos supervisores tenha o mesmo desempenho que a do supervisor monolítico. Quando essa propriedade é verificada, a abordagem de controle modular é bastante vantajosa no sentido de promover maior flexibilidade, maior eficiência computacional e segurança na aplicação do controle.

Este artigo apresenta os resultados obtidos pela implementação prática da metodologia proposta em Queiroz e Cury (2000c) na síntese do programa de controle para uma célula de manufatura real coordenada por um controlador lógico programável (CLP). Do processo de ligação entre a teoria e a prática surgem diversos novos desafios e soluções que abrangem desde aspectos de modelagem do sistema em malha aberta até a estrutura final do programa em diagrama escada.

A metodologia usada para a implementação do programa de controle é inovadora em relação aos trabalhos de Lauzon et al. (1996), de Fabian e Hellgren (1998), entre outros, por explorar a estrutu-

ra modularizada da planta e dos supervisores sintetizados, de forma a contornar os problemas com razoável simplicidade. A qualidade dos resultados obtidos é comprovada pelo bom funcionamento do sistema após a implementação do controle.

A seqüência do artigo é a seguinte: na Seção 2, são revistos alguns conceitos fundamentais sobre a abordagem usada de Controle Supervisório; na Seção 3, apresenta-se o problema da célula de manufatura, bem como o processo de síntese da lógica de controle, cuja implementação em diagrama escada é descrita na Seção 4; e na última seção são discutidos os resultados.

## 2 Controle Supervisório Modular

Na abordagem de linguagens controláveis proposta por Ramadge e Wonham (1989), a planta a controlar é modelada por um gerador  $G$ . O gerador é um autômato  $G = (\Sigma, Q, \delta, q_0, Q_m)$ , onde  $\Sigma$  é um alfabeto de eventos,  $Q$  é um conjunto de estados,  $q_0 \in Q$  é o estado inicial,  $Q_m \subseteq Q$  é o conjunto de estados marcados e  $\delta: \Sigma \times Q \rightarrow Q$ , a função de transição, é uma função parcial definida em cada estado de  $Q$  para um subconjunto de  $\Sigma$ . Seja  $\Sigma^*$  o conjunto de todas as cadeias finitas de  $\Sigma$ , incluindo a cadeia nula  $\epsilon$ . Então,  $G$  é caracterizado por dois subconjuntos de  $\Sigma^*$  chamados de *comportamento fechado* de  $G$  (todas as seqüências de eventos que a planta pode gerar), denotado por  $L(G)$ , e de *comportamento marcado* de  $G$  (seqüências representando tarefas completas), denotado por  $L_m(G)$ .

Eventos controláveis  $\Sigma_c \subseteq \Sigma$  são aqueles que podem ser ativados ou desativados por agentes externos. Eventos incontroláveis  $\Sigma_u \subseteq \Sigma$  são aqueles que não podem ser evitados de ocorrer e por isso são considerados permanentemente habilitados. Uma linguagem  $K \subseteq L(G)$  é *controlável e.r.a*  $G$  se  $\overline{K} \Sigma_u \cap L(G) \subseteq \overline{K}$  (a barra superior representa o prefixo fechamento). Isso quer dizer que a ocorrência de um evento incontrolável e fisicamente possível, após uma cadeia de  $\overline{K}$ , mantém a seqüência no conjunto  $\overline{K}$ . A classe de linguagens controláveis contidas em  $G$  é denotada por  $C(M, G) = \{ K | K \subseteq M \text{ e } K \text{ é controlável e.r.a } G \}$  e, por ser fechada sobre união, contém um (único) elemento supremo, chamado  $\text{Sup}C(M, G)$ .

Os autômatos podem ser ilustrados por diagramas de transição de estado, que são grafos direcionados onde os nós representam os estados e os ramos representam os eventos. Nesses diagramas, os estados marcados são caracterizados por nós desenhados com linhas duplas e o estado inicial é identificado por uma seta. Os eventos controláveis são representados por ramos interceptados.

Para fins de desenvolvimento do sistema de controle supervisório, o comportamento da planta e

das correspondentes especificações é modelado diretamente na forma de diagramas de transição de estado. A metodologia de síntese de supervisores proposta por Queiroz e Cury (2000c) explora o fato de que grande parte dos sistemas discretos de manufatura pode ser modelada como um conjunto de sistemas a eventos discretos concorrentes, assíncronos, interagindo entre si, de forma a garantir uma economia computacional. Esse tipo de modelagem é chamado de Sistema Produto (Ramadge e Wonham, 1989), o qual consiste de um conjunto de subsistemas completamente assíncronos entre si. Para obter uma representação por sistema produto (RSP) a partir de uma modelagem inicial de todos os subsistemas, faz-se a composição dos subsistemas síncronos originais (que têm eventos em comum).

Seja a RSP de um sistema de manufatura formada por subsistemas  $G_i = (\Sigma_i, Q_i, \delta_i, q_{0i}, Q_{mi})$ ,  $i = 1, \dots, n$ . Para  $j = 1, \dots, m$ , sejam agora as especificações genéricas locais  $E_{xj}$  definidas respectivamente em  $\Sigma_{xj} \subseteq \Sigma$ . Para  $j = 1, \dots, m$ , a planta local  $G_{xj} = (\Sigma_{xj}, Q_{xj}, \delta_{xj}, q_{0xj}, Q_{mxj})$  associada à especificação  $E_{xj}$  é definida por  $G_{xj} = \parallel_{i \in I_{xj}} G_i$ , com  $I_{xj} = \{k \in I | \Sigma_k \cap \Sigma_{xj} \neq \emptyset\}$ ,

onde o operador  $\parallel$  representa a composição síncrona (Ramadge e Wonham, 1989). Assim, a planta local  $G_{xj}$  é composta apenas pelos subsistemas da modelagem original que estão diretamente (e indiretamente) restringidos por  $E_{xj}$ .

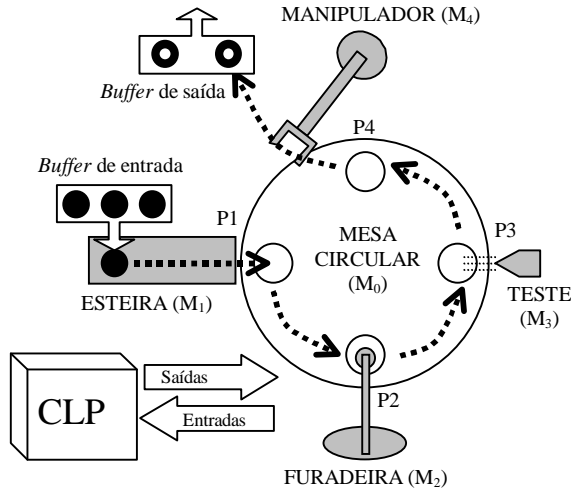
Para a síntese de supervisores modulares ótimos para as especificações locais  $E_{xj}$ ,  $j = 1, \dots, m$ , é necessário expressá-las apenas em termos das respectivas plantas locais  $G_{xj}$ ,  $j = 1, \dots, m$ , fazendo-se  $E_{xj} = E_{xj} \parallel L_m(G_{xj})$ . Calculam-se, então, as máximas linguagens controláveis  $\text{Sup}C(E_{xj}, G_{xj})$ ,  $j = 1, \dots, m$ , contidas nas especificações locais e controladores modulares que satisfaçam essas linguagens. A modularidade local dos supervisores pode ser verificada se, ao fazer a composição síncrona de todos os geradores para as linguagens resultantes, for obtido um autômato TRIM (não bloqueante). Essa condição, conforme os resultados de Queiroz e Cury (2000b), garante que o procedimento modular não resulte em qualquer perda de performance em relação ao controle centralizado.

## 3 A Célula de Manufatura

### 3.1 Descrição da Célula de Manufatura

A célula de manufatura estudada (Figura 1) é composta por uma mesa circular de quatro posições ( $M_0$ ), onde são efetuadas operações de furo e teste de peças metálicas, e de mais quatro dispositivos operacionais: a esteira de entrada ( $M_1$ ), a furadeira ( $M_2$ ), o aparelho de teste ( $M_3$ ) e o manipulador robótico ( $M_4$ ). O funcionamento da mesa circular é comandado por um controlador lógico programável conforme a seguinte seqüência:

1. a esteira gira até que uma peça seja posicionada em P1 (figura 1);
2. a mesa gira 90°;
3. a peça é furada;
4. a mesa gira 90°;
5. a peça é testada;
6. a mesa gira 90°;
7. manipulador robótico retira a peça da mesa.



**Fig. 1:** Célula de Manufatura

O programa de controle original da mesa, fornecido pelo fabricante, permite operar em seqüência apenas uma peça por vez, ou seja, a esteira só pode ser acionada novamente depois que o manipulador retirar a peça da mesa. Esta restrição da lógica de controle evita os problemas que podem ocorrer na operação de múltiplas peças em paralelo, entre os quais se destacam:

- operar a esteira, a furadeira, o teste ou o manipulador enquanto a mesa girar;
- sobrepor peças na posição P1;
- girar a mesa sem que as peças em P2, P3 e P4 tenham sido furadas, testadas ou retiradas, respectivamente;
- furar, testar ou acionar o robô sem peças nas posições P2, P3 e P4, respectivamente;
- furar ou testar duas vezes a mesma peça;
- girar a mesa sem nenhuma peça.

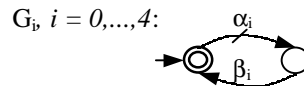
Entretanto, esse modo de funcionamento é muito pouco eficiente, visto que a esteira, a furadeira, o teste e o manipulador passam a maior parte do tempo parados enquanto poderiam estar operando em paralelo. O objetivo deste trabalho é, então, sintetizar um novo programa de controle para o CLP que evite os problemas anteriormente descritos de forma minimamente restritiva.

### 3.2 Modelagem da Planta

De acordo com a metodologia proposta por Queiroz e Cury (2000c), o primeiro passo para a síntese de controladores locais é a modelagem dos subsistemas

envolvidos. O funcionamento de cada dispositivo corresponde a uma seqüência de operações específicas, como acionamento de motores e atuadores pneumáticos e leitura de sensores. No entanto, os problemas operacionais descritos não ocorrem nas seqüências operacionais particulares de cada subsistema, mas decorrem da descoordenação entre o início e o final das diversas seqüências. Com isso, para a síntese da lógica de controle, os modelos dos dispositivos podem ser abstraídos, bastando representá-los em termos dos eventos de início e final de operação. O funcionamento particular de cada subsistema pode ser implementado diretamente no programa do CLP, como mostrado na Seção 4. Essa simplificação ajuda a reduzir o tamanho dos modelos e, por consequência, a complexidade do processo de síntese e o número de estados dos supervisores resultantes.

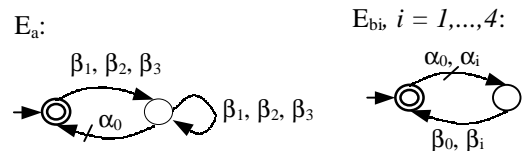
Assim, considera-se que os subsistemas  $M_i$ ,  $i=0, \dots, 4$ , têm suas operações iniciadas respectivamente pelos eventos controláveis  $\alpha_i$ ,  $i=0, \dots, 4$ , e terminam de operar com o evento incontrolável  $\beta_i$ ,  $i=0, \dots, 4$ . As máquinas  $M_i$ ,  $i=0, \dots, 4$ , podem então ser modeladas respectivamente pelos autômatos  $G_i$ ,  $i=0, \dots, 4$ , apresentados na figura 2.



**Fig. 2:** Autômato para  $M_i$ ,  $i = 0, \dots, 4$

### 3.3 Modelagem das Especificações

O autômato  $E_a$  descrito na figura 3 modela uma especificação que garante que a mesa não vai girar à toa, isto é, sem ao menos uma peça bruta em P1 ou uma peça furada em P2 ou uma peça testada em P3. As quatro especificações de segurança que impedem a mesa de girar enquanto a esteira ( $E_{b1}$ ), a furadeira ( $E_{b2}$ ), o teste ( $E_{b3}$ ) ou/e o manipulador ( $E_{b4}$ ) estiverem operando podem ser descritas pelo autômato  $E_{bi}$  de índice  $i$  da figura 3.

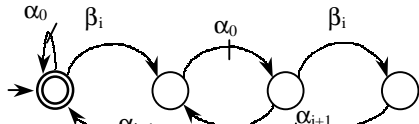


**Fig. 3:** Autômatos para  $E_a$  e para  $E_{bi}$ ,  $i = 1, \dots, 4$

Os possíveis problemas decorrentes do fluxo de múltiplas peças na mesa podem ser evitados pelas especificações:  $E_{c1}$ , relativa à movimentação de peças brutas entre as posições P1 e P2;  $E_{c2}$ , para a manipulação de peças furadas entre P2 e P3; e  $E_{c3}$ , correspondente ao fluxo de peças testadas entre P3 e P4. Para isso, a especificação  $E_{c1}$  evita sobrepor peças em P1, furar sem peça bruta em P2 e girar a mesa com peça bruta em P2. Já a especificação  $E_{c2}$  proíbe furar duas vezes a mesma peça, testar sem peça furada em P3 e girar a mesa com peça furada e

não testada em P3, enquanto  $E_{c3}$  impede testar duas vezes a mesma peça, acionar o manipulador sem peça em P4 e girar a mesa com peça em P4. Como as especificações têm a mesma estrutura, pode-se ilustrá-las pelo modelo indexado  $E_{ci}$  da figura 4.

$E_{ci}, i = 1, 2, 3$ :



**Fig. 4:** Autômato para  $E_{ci}, i = 1, 2, 3$

Como os subsistemas estão modelados de forma assíncrona, pode-se observar que o sistema encontra-se na mais refinada RSP. Assim, podem-se obter as plantas locais  $G_A, G_{B1}, G_{B2}, G_{B3}, G_{B4}, G_{C1}, G_{C2}$  e  $G_{C3}$  respectivas às especificações  $E_a, E_{b1}, E_{b2}, E_{b3}, E_{b4}, E_{c1}, E_{c2}$  e  $E_{c3}$  fazendo-se a composição dos modelos que tenham eventos comuns a elas. Geram-se desta forma os seguintes autômatos:

- $G_A = G_0 \parallel G_1 \parallel G_2 \parallel G_3$ ;
- $(G_{Bi} = G_0 \parallel G_i), i = 1, \dots, 4$ ;
- $(G_{Ci} = G_0 \parallel G_i \parallel G_{i+1}), i = 1, 2, 3$ .

Calculam-se, então, as especificações locais  $E_A, E_{B1}, E_{B2}, E_{B3}, E_{B4}, E_{C1}, E_{C2}$  e  $E_{C3}$  pela composição síncrona das especificações  $E_a, E_{b1}, E_{b2}, E_{b3}, E_{b4}, E_{c1}, E_{c2}$  e  $E_{c3}$  com suas respectivas plantas locais.

### 3.4 Síntese dos Supervisores

O passo seguinte é o cálculo das máximas linguagens controláveis contidas nas especificações locais. Fazendo a composição síncrona de todas as linguagens resultantes, obtém-se o autômato  $S$ , que é TRIM. Isso indica que a condição de modularidade local é verificada e, por conseguinte, a abordagem modular é ótima. A tabela 1 apresenta os resultados desse passo.

**Tabela 1:** Máximas Linguagens Controláveis

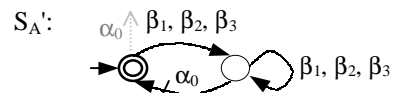
	Cálculo	estados	trans.
$S_A$	$\text{SupC}(E_A, G_A) = E_A$	32	120
$S_{B1}$	$\text{SupC}(E_{B1}, G_{B1}) = E_{B1}$	3	4
$S_{B2}$	$\text{SupC}(E_{B2}, G_{B2}) = E_{B2}$	3	4
$S_{B3}$	$\text{SupC}(E_{B3}, G_{B3}) = E_{B3}$	3	4
$S_{B4}$	$\text{SupC}(E_{B3}, G_{B3}) = E_{B3}$	3	4
$S_{C1}$	$\text{SupC}(E_{C1}, G_{C1}) \subset E_{C1}$	24	52
$S_{C2}$	$\text{SupC}(E_{C2}, G_{C2}) \subset E_{C2}$	24	52
$S_{C3}$	$\text{SupC}(E_{C3}, G_{C3}) \subset E_{C3}$	24	52
$S$	$S_A \parallel S_{B1} \parallel \dots \parallel S_{B4} \parallel S_{C1} \parallel \dots \parallel S_{C3}$	151	350

Para cada uma dessas linguagens resultantes, poderia ser elaborado um supervisor que atuasse na respectiva planta local, ativando ou desativando eventos controláveis, de forma a atender localmente à especificação resultante. Entretanto, busca-se reduzir ainda mais o tamanho dos supervisores finais, uma vez que a memória do CLP é limitada. Observando-se que a composição síncrona do autômato  $S_A$  com a planta  $G$  é equivalente à composição do au-

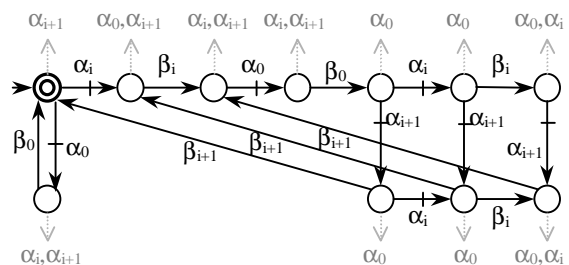
tômato  $E_a$  (especificação controlável) com a mesma planta, pode-se concluir que o próprio autômato  $S_A' = E_a$  (de apenas dois estados) é um supervisor reduzido que pode substituir  $S_A$ . Já o número de estados e transições dos supervisores  $S_{C1}, S_{C2}$  e  $S_{C3}$ , por uma particularidade de suas estruturas, é reduzido para 11 estados e 16 transições pelas composições:

- $S_{C1}' = S_{C1} \parallel S_{B1} \parallel S_{B2}$ ;
- $S_{C2}' = S_{C2} \parallel S_{B2} \parallel S_{B3}$ ;
- $S_{C3}' = S_{C3} \parallel S_{B3} \parallel S_{B4}$ .

Os supervisores resultantes desse processo são representados pelos autômatos, em cujos estados os eventos desabilitados são ilustrados por setas tracejadas, apresentados na figura 5.



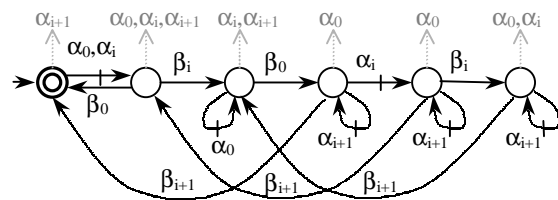
$S_{Ci}', i = 1, 2, 3$ :



**Fig. 5:** Supervisores  $S_A'$  e  $S_{Ci}', i = 1, 2, 3$

Pode ser observado nos supervisores  $S_{Ci}', i = 1, 2, 3$ , que algumas transições provocam mudanças de estados mas não alteram as ações de controle. Então, agregam-se estados dos autômatos  $S_{Ci}', i = 1, 2, 3$ , verificando, para cada conjunto de estados agregados, se as ações de controle não entram em conflito com as saídas habilitadas pelos estados do conjunto e se a agregação não gera indeterminismos no autômato final. Os supervisores reduzidos  $S_{Ci}', i = 1, 2, 3$ , obtidos por esse processo de agregação são ilustrados pelo autômato da figura 6. Um algoritmo formal para redução de supervisores é apresentado por Vaz e Wonham (1986).

$S_{Ci}'', i = 1, 2, 3$ :



**Fig. 6:** Supervisores  $S_{Ci}'', i = 1, 2, 3$

Como resultado, são obtidos 4 supervisores localmente modulares ( $S_A', S_{C1}'', S_{C2}'$  e  $S_{C3}''$ ) com um número bastante reduzido de estados e transições. Esses supervisores, atuando em conjunto sobre a planta, garantem que a célula de manufatura funcione da forma mais permissível para que não ocorram os problemas anteriormente especificados.

#### 4 Implementação em Diagrama Escada

Os supervisores resultantes do processo de síntese apresentado na Seção 3 são descritos como máquinas de estados finitos em que, para cada estado ativo, um conjunto de eventos controláveis deve ser desabilitado. Assim, a implementação prática do programa de controle consiste basicamente em fazer o CLP se comportar como um jogador de autômatos.

Porém, algumas suposições subjacentes feitas na modelagem do sistema físico precisam ser resolvidas para que a implementação prática funcione. Por exemplo, as seqüências de operação de cada dispositivo não são executadas automaticamente pela planta física, como a modelagem pressupõe, e precisam ser comandadas pelo CLP de acordo com os sinais dos sensores. Também os eventos controláveis e incontrolláveis do modelo não correspondem diretamente a sinais de saída ou de entrada do CLP. Desta forma, a ação do programa de controle não é simplesmente desabilitar eventos controláveis como a teoria de controle supervisionário pressupõe. Portanto, o CLP deve também funcionar como interface entre o modelo teórico e o sistema real.

Para explorar a estrutura modular da planta e dos supervisores resultantes do processo de síntese, o programa de controle é desenvolvido em três níveis estruturais: o nível dos Supervisores Modulares que desabilitam eventos da planta de acordo com as mudanças de estado da mesma; o nível do Sistema Produto que, seguindo os modelos supervisionados das plantas, é responsável por comandar o início das seqüências de operação; e o nível das Seqüências Operacionais que, observando os sinais de entrada do CLP e ajustando os sinais de saída, executa os ciclos de funcionamento de cada dispositivo. Essa estrutura de controle é esquematizada na figura 7.

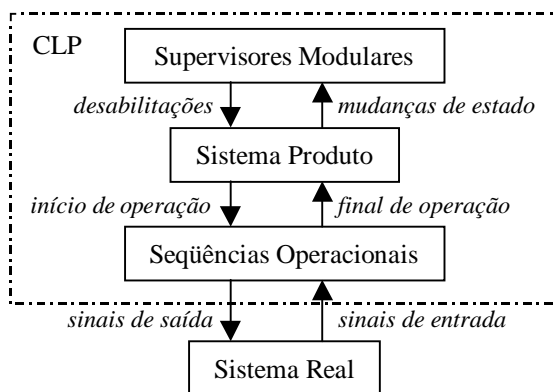


Fig. 7: Estrutura básica do sistema de controle

Para programação de máquinas de estado em diagrama escada, assim como em Brandin (1996) e em Fabian e Hellgren (1998), representa-se cada estado por uma variável booleana interna ("flag"). Para cada transição do autômato é implementado um degrau do diagrama escada segundo o qual, sempre que o relé representando o estado atual e o relé rela-

tivo a um determinado evento interno ou externo ao CLP estiverem ativos, atribuem-se os valores 1 ao estado seguinte e 0 ao estado atual através de bobinas sustentáveis (*latched coil* - norma IEC1131-3).

Para o controle da célula de manufatura, são implementadas no nível dos Supervisores Modulares as 4 máquinas de estado referentes aos autômatos de  $S_A$ ,  $S_{C1}$ ,  $S_{C2}$  e  $S_{C3}$ . As transições dos supervisores ocorrem em função de eventos indicando mudança de estado no Sistema Produto. É fácil notar que não há necessidade de se representar transições que saem e entram no mesmo estado (*self-loops*). A ação dos supervisores é ativar por bobinas normais as desabilitações de eventos controláveis do Sistema Produto (atribuídas a variáveis booleanas internas) quando um dos estados correspondentes à ação estiver ativo. Como exemplo, o diagrama escada da figura 8 ilustra parte do programa para o supervisor  $S_{C1}$ .

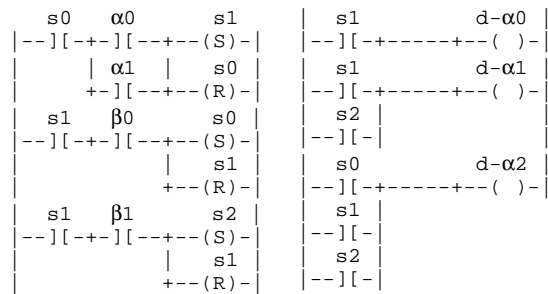


Fig. 8: Parte do Supervisor  $S_{C1}$ : máquina de estado (esq.) e desabilitações (dir.).

No nível do Sistema Produto são programadas as máquinas de estado para os subsistemas  $G_0$ ,  $G_1$ ,  $G_2$ ,  $G_3$  e  $G_4$ . As transições incontrolláveis são disparadas pela sinalização de final de operação das Seqüências Operacionais. Já as transições controláveis são executadas automaticamente sempre que não forem desabilitadas pelos supervisores, ativando as variáveis que iniciam os ciclos de operação. Cada transição ativa também os sinais que representam as mudanças de estado para os supervisores. Para a coerência do sistema de controle, é importante cuidar para que não ocorram duas transições seguidas no Sistema Produto sem que os supervisores tenham sido atualizados. No caso particular do CLP em questão, isso pode ser realizado fazendo-se um salto do final de cada transição para o primeiro degrau dos supervisores. A figura 9 mostra a implementação em diagrama escada da planta da esteira ( $G_1$ ).

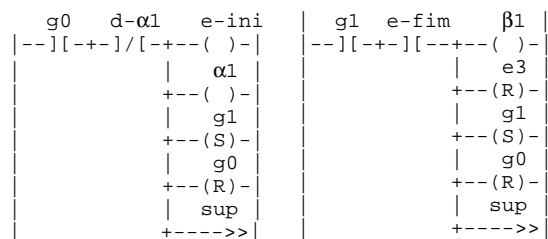
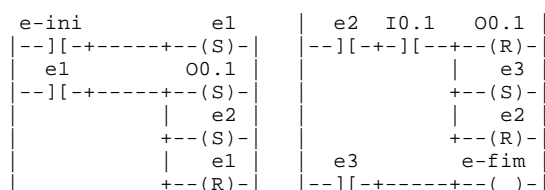


Fig. 9: Planta  $G_1$ : transição controlável (esq.) e transição incontrollável (dir.).

As Seqüências Operacionais para a mesa circular, a esteira, a furadeira, o teste e o manipulador robótico também podem ser programadas como máquinas de estados, cujas transições iniciais são disparadas pelos sinais do Sistema Produto e as seguintes pelos sinais de entrada do CLP. Assim, os sinais de saída do CLP são ativados seqüencialmente conforme a lógica de funcionamento interna de cada dispositivo. O final de cada ciclo de operação deve ser sinalizado ao Sistema Produto ativando-se a respectiva variável booleana. O programa para a seqüência operacional da esteira, mostrado na figura 10, serve de exemplo.



**Fig. 10:** Seqüência Operacional da Esteira.

As outras estruturas periféricas do programa, como inicialização do sistema físico, ativação dos estados iniciais dos autômatos e procedimentos de parada, podem ser implementadas sem grandes dificuldades. O diagrama escada final para o controle do CLP tem 92 degraus (5233 Bytes) e utiliza 76 dos 256 “flags” disponíveis no CLP. Portanto, a memória do CLP não é um obstáculo para a implementação prática do controlador.

Após a aplicação do novo sistema de controle, a célula de manufatura passou a funcionar com bastante produtividade e flexibilidade, pois a lógica de controle permite operar de 0 a 4 peças em paralelo, conforme a mesa for sendo alimentada. Além disso, a estrutura do programa se mostrou bastante flexível e compreensível, possibilitando uma depuração rápida e mudanças no programa com bastante facilidade.

## 5 Conclusão

A abordagem de controle supervísório modular local se mostrou uma ferramenta bastante favorável para a síntese de supervisores ótimos e reduzidos para a célula de manufatura. A modularidade e a localidade das soluções permitiram o desenvolvimento de um modelo estruturado para a programação do sistema de controle em diagrama escada. Este modelo sugere a sistematização da abordagem para uma proposta de metodologia de geração automática de controladores para CLPs. O funcionamento flexível e produtivo da célula de manufatura e a clareza e flexibilidade do programa final do CLP são indicadores positivos da qualidade da metodologia usada.

## Agradecimentos

O primeiro autor foi financiado pelo CNPq (Processo 140.825/2000-2) e o terceiro autor também recebeu apoio financeiro do CNPq (Processo 300.953/93-3 e PRONEX 015/98). Os autores agradecem também à Universidade PUC-PR, especialmente ao professor Agnelo Denis Vieira, por disponibilizar por três dias a célula de manufatura para os ensaios do presente trabalho.

## Referência Bibliográfica

- Balemi, S., Hoffmann, G. J., Gyugyi, P., Wong-Toi, H. e Franklin, G. F. (1993). Supervisory control of a rapid thermal multiprocessor. *IEEE Transactions on Automatic Control*, 38(7):1040-1059.
- Brandin, B. A. (1996). The Real-Time Supervisory control of an experimental manufacturing cell. *IEEE Transactions on Robotics and Automation*, 12(1):1-14.
- Fabian, M. e Hellgren, A. (1998). PLC-based implementation of supervisory control for discrete event systems. *Proc. of the 37<sup>th</sup> Conference on Decision and Control*. Tampa, USA.
- ISO/IEC (1993). *International Standard IEC 1131-3, Programmable Controllers – Part 3*.
- Lauzon, S.C., Ma, A. K. L., Mills, J. K. e Benhabib, B. (1996). Application of discrete-event-system theory to flexible manufacturing. *IEEE Control Systems Magazine*. pp41-48, Feb. 1996.
- Queiroz, M. H. e Cury, J. E. R.. (2000a). Modular control of composed systems. *Proc. of the American Control Conference*. Chicago, USA.
- Queiroz, M. H. e Cury, J. E. R.. (2000b). Modular supervisory control of large scale discrete-event systems. *In: Discrete Event Systems: Analysis and Control*. Kluwer Academic Publishers, pp. 103-110. (*Proceedings of the WODES 2000 - Ghent, Belgium*)
- Queiroz, M. H. e Cury, J. E. R. (2000c). Controle modular de sistemas de manufatura. *Revista Controle & Automação, SBA* (aprovado para publicação). Versão preliminar: *Anais do XIII CBA*, UFSC, Florianópolis.
- Ramadge, P. J. e Wonham, W. M. (1989). The control of discrete event systems. *Proceedings IEEE, Special Issue on Discrete Event Dynamic Systems*, 77(1):81-98.
- Vaz, A. F. e Wonham, W. M. (1986). On supervisor reduction in discrete-event systems. *International Journal of Control*, 44(2):475-491.
- Wonham, W. M. e Ramadge, P. J. (1988). Modular supervisory control of discrete event systems. *Mathematics of control of discrete event systems*, 1(1):13-30.