

# APLICAÇÕES E LIMITAÇÕES DO CONTROLE DIRIGIDO ÓTIMO NÃO-BLOQUEANTE DE SISTEMAS A EVENTOS DISCRETOS

GUSTAVO L. VIEIRA<sup>1</sup>, PATRÍCIA N. PENNA<sup>2</sup>.

1. *Engenharia de Controle e Automação, Universidade Federal de Minas Gerais  
LACSED – DELT (Laboratório de Análise e Controle de Sistemas a Eventos Discretos)  
Av. Antônio Carlos, 6627, Pampulha - Belo Horizonte - MG, 31270-901  
E-mail: gustavolvieira@ufmg.com.br*

2. *LACSED - DELT, Universidade Federal de Minas Gerais  
Av. Antônio Carlos, 6627, Pampulha - Belo Horizonte - MG, 31270-901  
E-mail: ppenna@ufmg.br*

**Abstract**— For discrete event systems, supervisory control guarantees a safe nonblocking minimally restrictive control. However, it only determines the actions allowed in each state, but does not determine which one should be taken. The method of direct control refines the results obtained, obtaining a the best sequence of control commands among all possible. The optimal director is the controller that defines the control actions leading to the accomplishment of the system's goals with minimum cost. In this work, an algorithm for synthesis of the optimal director in any automata is computationally implemented. The applicability of such method is verified in several kinds of problems of the literature.

**Keywords**— discrete event systems, directed control, supervisory control, optimization.

**Resumo**— Para sistemas a eventos discretos, o controle supervisão garante um controle seguro, não bloqueante e minimamente restritivo. No entanto, ele apenas determina as ações permitidas em cada estado, mas não define qual deve ser seguida. O método de controle dirigido refina os resultados obtidos, encontrando a melhor sequência de comandos de controle dentre todas as possíveis. É chamado diretor ótimo o controlador que define as ações de controle que levam à realização dos objetivos do sistema com custo mínimo. Nesse trabalho, é implementado computacionalmente um algoritmo para síntese do diretor ótimo em autômatos quaisquer. A aplicabilidade desse método é verificada então em diferentes tipos de problemas recorrentes na literatura.

**Palavras-chave**— sistemas a eventos discretos, controle dirigido, controle supervisão, otimização

## 1 Introdução

A Teoria de Controle Supervisório busca um controlador, chamado supervisor, que implementa especificações (de segurança, justiça, etc.) em uma planta, de maneira minimamente restritiva e não bloqueante. O supervisor desabilita apenas eventos controláveis que possam levar o autômato a estados bloqueantes ou que violam especificações. Dessa forma, em uma planta sob controle supervisão monolítico (Ramadge e Wonham, 1989) sobrevivem todos os caminhos legais possíveis para o sistema (Cassandras e Lafortune, 1999).

No entanto, para determinados tipos de problemas, não basta determinar um conjunto de caminhos legais e possíveis. Para exercer controle na planta, pode ser necessário selecionar, para cada estado, um único comando dentre todos os habilitados. É possível solucionar o problema de determinar um caminho dentre todos os possíveis com métodos como a escolha aleatória dentre todos os eventos habilitados em um estado.

Há contextos, porém, onde o objetivo não é apenas que o sistema cumpra suas tarefas, mas também que o faça com gasto mínimo de algum recurso utilizado durante o processo. Para otimização do uso desse recurso, o método de escolha aleatória de ação de controle não atinge resultados satisfatórios. Um método comumente empregado (escolha do evento de menor custo em cada estado) também não garante um resultado ótimo, devido à lógica gulosa utilizada

e à natureza dinâmica do problema, apesar de reduzir em média o gasto do recurso considerado.

O método formal para solução do problema de sequenciamento de tarefas com gasto mínimo de recursos em uma planta é a observação de todos os caminhos possíveis e comparação dos seus custos totais. O custo computacional de tal método, entretanto, cresce exponencialmente com o número de estados da planta, o que o torna inviável para problemas maiores.

Métodos de otimização computacional têm sido aplicados a problemas específicos de controle em sistemas a eventos discretos com o objetivo de minimizar os gastos de recursos em um sistema sob controle supervisão. Os trabalhos de (Oliveira et al., 2013), (Costa et al., 2012) e (Silva et al., 2011) utilizam heurísticas e meta-heurísticas para realizar o sequenciamento de tarefas em sistemas flexíveis de manufatura industriais, com minimização do tempo de produção de uma batelada de produtos. Os resultados alcançados mostram redução significativa nos tempos obtidos, e a complexidade é menor que de testes exaustivos. Entretanto, devido à própria natureza dos métodos utilizados, não se pode garantir a obtenção do resultado ótimo.

A noção de controle dirigido surge para refinar a do controle supervisão (Huang e Kumar, 2005). Um controlador dirigido, chamado diretor, habilita no máximo um evento controlável para cada estado de uma planta (mantendo habilitados os eventos não-controláveis). Desse modo, um sistema sob controle

de um diretor já tem determinada para cada estado, no máximo uma ação de controle.

Uma mesma planta possui diversos possíveis diretores. O diretor que leva a planta ao resultado desejado com gasto mínimo de um recurso é chamado Diretor Ótimo. Em (Huang e Kumar, 2008), foi desenvolvido um método para obtenção do diretor ótimo para um autômato qualquer, com complexidade polinomial com o número de estados da planta. Enxerga-se neste método uma possível alternativa para otimização aplicada ao controle supervísório para fins de sequenciamento de tarefas em sistemas de manufatura industriais, por exemplo. Essa abordagem destaca-se de métodos baseados em otimizações meta-heurísticas por utilizar puramente conceitos relacionados à teoria de sistemas a eventos discretos.

Para verificar a aplicabilidade do método de controle dirigido nos problemas testados em (Oliveira et al., 2013) e (Costa et al., 2012) foram realizados testes com os métodos de obtenção do diretor ótimo apresentados. O algoritmo proposto foi então implementado computacionalmente, de modo a permitir a execução dos testes desejados com segurança e em menor tempo.

Os conceitos básicos de sistemas a eventos discretos, controle supervísório e controle dirigido são apresentados na seção 2. Na seção 3, é discutido o método para obtenção do diretor ótimo e sua implementação computacional. A seção 4 apresenta testes realizados sobre a aplicabilidade do controle dirigido para obtenção do caminho ótimo em problemas de interesse. Os resultados obtidos são discutidos na seção 5 e a última seção apresenta as conclusões.

## 2 Noções Preliminares

Esse trabalho se baseia no formalismo estabelecido por Ramadge e Wonham (1987). Um Sistema a Eventos Discretos (SED) é modelado como um autômato  $G := (X, \Sigma, f, x_0, X_m)$ , onde  $X$  é o conjunto de estados,  $\Sigma$  o conjunto de eventos,  $f: X \times \Sigma \rightarrow X$  denota a função de transição e pode ser estendida para cadeias,  $x_0$  é o estado inicial e  $X_m \subseteq X$  o conjunto de estados marcados. O conjunto de eventos factíveis no estado  $x \in X$  é d por  $\Sigma(x) := \{\sigma \in \Sigma \mid f(x_0, s) \text{ é definido}\}$ .  $\Sigma^*$  é o conjunto de todas as sequências finitas de eventos de  $\Sigma$ , chamadas de cadeias, e contém a cadeia vazia  $\epsilon$ .

Um subconjunto de  $\Sigma^*$  é chamado linguagem. A linguagem gerada de  $G$  é o conjunto de cadeias finitas que podem ser obtidas em  $G$ , definida como  $\mathcal{L}(G) := \{s \in \Sigma^* \mid f(x_0, s) \text{ definida}\}$ . A linguagem marcada de  $G$  é o conjunto de cadeias que levam o autômato a um estado marcado a partir do estado inicial,  $\mathcal{L}_m(G) := \{s \in \mathcal{L}(G) \mid f(x_0, s) \in X_m\}$ . Um estado  $x$  é dito terminal se pertence ao conjunto  $X_t := \{x \in X \mid \Sigma(x) = \emptyset\}$ , e  $X_{tm} := X_t \cap X_m$  denota o conjunto de estados marcados terminais.

Um estado  $x \in X$  é chamado acessível se  $\exists s \in \Sigma^*$  tal que  $f(x_0, s) = x$ . Um estado  $x \in X$  é chamado

coacessível se  $\exists s \in \Sigma^*$  tal que  $f(x, s) \in X_m$ . Um autômato é aparado se todos seus estados são acessíveis e coacessíveis.

Na modelagem do sistema, divide-se o conjunto de eventos em dois subconjuntos: eventos controláveis  $\Sigma_c$  e eventos não-controláveis  $\Sigma_u$ . No contexto do controle dirigido (Huang e Kumar, 2005), define-se ainda um subconjunto dos eventos não-controláveis chamados eventos de perturbação, que indicam eventos que não podem ser previstos no sistema, como por exemplo uma falha. Os eventos não-controláveis de perturbação são denotados por  $\Sigma_d \subseteq \Sigma_u$ . Define-se  $\Sigma_c(x) := \Sigma(x) \cap \Sigma_c$ ,  $\Sigma_u(x) := \Sigma(x) \cap \Sigma_u$  e  $\Sigma_d(x) := \Sigma(x) \cap \Sigma_d$ .

O controle supervísório de SEDs baseia-se no modelo da planta e nas especificações desejadas para síntese do supervisor. Suas ações de controle consistem na inibição de determinados eventos controláveis de maneira que a planta atinja o comportamento desejado. Caso o comportamento desejado não possa ser alcançado sem desabilitar eventos não-controláveis, o supervisor implementa a máxima sublinguagem controlável, representada por  $S^{\uparrow c}(K, \mathcal{L}_m(G))$ , onde  $K$  é a linguagem desejada e  $\mathcal{L}_m(G)$  é a linguagem que representa o comportamento da planta em malha aberta. O controle supervísório monolítico é realizado através de uma planta única (que consiste na composição de todas as subplantas) e uma especificação única (composição de todas as especificações), utilizadas no cálculo de um supervisor único  $S^{\uparrow c}(K, G)$ .

No contexto de otimização em questão, é definido para cada evento  $\sigma$  um custo, denotado  $c(\sigma)$ . Esse custo representa a quantidade de um recurso (e.g. dinheiro ou tempo, dentre outros) dispendido para a execução do evento.

Seguindo a base formal estabelecida em (Huang e Kumar, 2008), utiliza-se neste trabalho a definição de Diretor baseada em estados. Nela, um diretor é um mapa  $D: X \rightarrow 2^{\Sigma_c}$  tal que  $\forall x \in X: |D(x)| \leq 1$ ;  $X_e \Rightarrow |D(x)| = 1$ , onde  $X_e := \{x \in X_m - X_t \mid \Sigma_u(x) = \emptyset\} \cup \{x \in X \mid (\Sigma_d(x) \neq \emptyset) \wedge (\emptyset \subsetneq \Sigma_u(x) \subseteq \Sigma_d)\}$ . Ou seja, o diretor é um mapa que habilita, para cada estado, no máximo um evento controlável. O conjunto de estados  $X_e$  reúne os estados nos quais não se deve desabilitar todos os eventos controláveis. São estes:

- Estados marcados não terminais onde não é habilitado nenhum evento não-controlável (de forma que desabilitar todos os eventos controláveis impediria o sistema de realizar tarefas adicionais);
- Estados que possuem como factíveis pelo menos um evento controlável (de modo que nele o controle pode ser exercido) e somente (e pelo menos um) eventos não-controláveis de perturbação. Nestes estados, desabilitar todos os eventos controláveis manteria o sistema parado até que um evento de perturbação ocorresse.

Uma planta  $G$  sob controle de um diretor  $D$  é denotada como  $G^D$ .

### 3 Implementação Computacional do Diretor Ótimo

#### 3.1 Obtenção do Diretor Ótimo

É de interesse do grupo de pesquisa comparar resultados obtidos no campo de otimização aplicada a SEDs com outros já estabelecidos na literatura. Entre os trabalhos existentes, optou-se por estudar o trabalho desenvolvido em (Huang e Kumar, 2008) e tentar aplicar os seus resultados a problemas de sequenciamento de tarefas em sistemas de manufatura. Neste trabalho, o método para síntese do diretor ótimo apresentado no trabalho citado foi implementado computacionalmente, com utilização da plataforma MATLAB. Por meio de um programa que execute o algoritmo, é possível realizar um grande número de testes com maior facilidade, bem como lidar com problemas de maior complexidade com segurança.

Durante a execução do método de síntese do diretor, são atribuídos custos a cada estado  $x \in X$ , denotados por  $\rho(x)$ . Esse conceito pode ser interpretado como o custo de deixar o estado  $x$  e atingir o estado  $x' \in X_m$ , onde  $x'$  é o estado marcado mais próximo de  $x$  (i.e., que pode ser atingido com menor custo a partir de  $x$ ). Esse custo pode ser visto como o custo máximo entre todos os caminhos que permitem atingir  $x'$  a partir de  $x$ .

Neste método, quando o autômato alcança um estado marcado não terminal, o sistema é “reinicializado”. O estado marcado alcançado passa a agir como novo estado inicial, e os demais estados marcados continuam a existir normalmente. O conjunto de estados em uma planta  $G$  a partir dos quais ela pode ser “reinicializada” é definido por  $I(G) := x_0 \cup (X_m - X_t)$ .

O custo de um diretor para uma planta  $G$  é definido como  $P(D) := \max_{x \in I(G^v)} \rho(x)$ . O objetivo do método de otimização é encontrar o diretor  $D'$  tal que:

$$D' \in \arg \{ \min_D P(D) \},$$

ou seja, o diretor com mínimo custo dentre todos os diretores possíveis para o autômato.

Apresenta-se, na seção 3.2, o pseudo-código que descreve simplificada o funcionamento do algoritmo implementado. Na seção 3.3 seguem-se alguns testes realizados sobre a implementação.

#### 3.2 Pseudo-código

*Algoritmo\_Diretor:* Recebe como entrada uma planta  $G := (X, \Sigma, f, x_0, X_m)$ , e retorna seu diretor ótimo  $D$ .

##### Inicialização

- 1) Define-se  $G' = G$
- 2) Aplica-se em  $G'$  uma função para obter um diretor localmente ótimo baseado em uma lógica gulosa. O diretor resultante é denotado  $D'$ .

- 3) Determina-se  $D = D'$ .

##### Iteração

- 1) Define-se o conjunto  $I(G^{D'})$  contendo os estados marcados não terminais e o estado inicial da planta  $G'$  sob controle de  $D'$ .
- 2) Dentro do conjunto  $I(G^{D'})$ , determina-se o conjunto  $V$  como o conjunto de estados com custo máximo.
- 3) Os estados pertencentes ao conjunto  $V$  são removidos de  $G'$  e a planta é aparada.
- 4) Aplica-se a função para obtenção de um diretor localmente ótimo<sup>1</sup> na nova planta  $G'$ . O diretor resultante é denotado  $D'$ .
- 5) Se  $P(D) \geq P(D')$ ,  $D = D'$ .

##### Término

Se  $x_0 \in V$  ou  $G^* = \emptyset$ , o algoritmo para. Se  $P(D) = \infty$ , não existe diretor ótimo para o autômato. Caso contrário,  $D$  é o diretor ótimo.

O algoritmo apresentado utiliza uma função generalizada para qualquer tipo de autômato capaz de obter um diretor baseado em uma lógica gulosa. Essa função obtém um diretor localmente ótimo, ou seja, que direciona a planta de maneira tal que em cada estado é selecionada a transição que leva a um estado marcado com menor custo.

O Algoritmo\_Diretor aplica essa função repetidamente, e a cada iteração faz uma redução do espaço de busca para a lógica gulosa. Essa redução é feita removendo-se da planta fornecida o estado marcado não terminal que gera o maior custo para a planta. Na nova planta reduzida é novamente obtido um diretor, cujo custo é comparado com o resultado anterior.

A aplicação da função baseada em lógica gulosa individualmente não garante que o resultado ótimo seja alcançado. Sua utilização em conjunto com o método de refinamento do espaço de estados, no entanto, pode garantir que todos os caminhos relevantes são observados. Desse modo, a obtenção do diretor ótimo é garantida, segundo (Huang e Kumar, 2008).

A execução detalhada do algoritmo e as provas de sua correteza podem ser obtidas em (Huang e Kumar, 2008).

#### 3.3 Testes do Algoritmo

Após a implementação computacional do algoritmo de síntese do diretor ótimo, foram realizados diversos testes com o mesmo, com objetivo de verificar seu funcionamento. Durante os testes, os resultados obtidos com o algoritmo foram verificados por meio de algum outro método, de maneira a garantir que o resultado obtido estivesse correto. Esses métodos consistiram na aplicação manual do algoritmo de

<sup>1</sup>Este algoritmo foi omitido por falta de espaço. Detalhes podem ser encontrados em (Huang e Kumar, 2008).

síntese do diretor ótimo para plantas gerais, seguindo passo a passo as instruções do algoritmo apresentado em (Huang e Kumar, 2008) e na listagem manual de todos os diretores possíveis do autômato e comparação do custo de cada um deles (em autômatos cujo tamanho permitisse).

Os testes apresentados foram realizados com plantas e figuras utilizadas em (Huang e Kumar, 2008). Desse modo, o objetivo era verificar que o resultado obtido na implementação computacional coincidia com aqueles obtidos no trabalho original.

O primeiro dos testes foi realizado com o autômato da Figura 1. Nas figuras apresentadas, a seta com linha cheia indica evento controlável e a seta tracejada com duas pontas indica evento não-controlável. Na planta da Figura 1, todos os eventos não-controláveis são de perturbação.

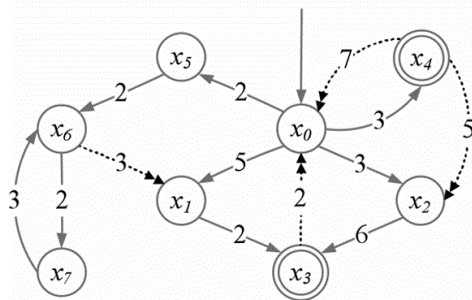


Figura 1. Planta G1 testada com o Algoritmo

A aplicação da ferramenta implementada sobre esta planta alcançou exatamente os mesmos resultados obtidos em (Huang e Kumar, 2008), tanto nas etapas intermediárias como na final, o que indica a correteza da implementação. Ao final da execução do programa, obteve-se um diretor que indica na planta o mesmo resultado apresentado no trabalho original, Figura 2. Como pode ser observado, esse é o caminho ótimo da planta G1.

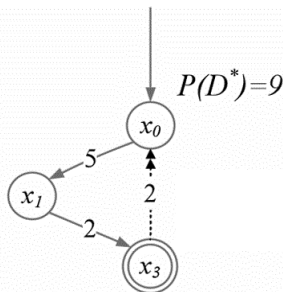


Figura 2. Planta G1 sob controle dirigido ótimo

Para o segundo teste apresentado, tem-se um sistema de controle de tráfego em uma seção de trilhos de trem representado na Figura 3 e modelado pelo autômato da Figura 4. Cada um dos estados representa uma das sessões marcadas na linha de trem. Através de um sistema de semáforos, é possível permitir ou proibir a passagem do trem em alguns trechos, controlando assim seu caminho. O custo de cada uma das transições representadas no autômato indica o

tempo gasto pelo trem para ir da seção de origem da transição até a seção de destino.

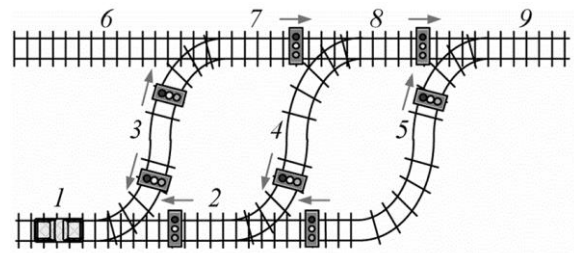


Figura 3. Sistema de linhas de trem

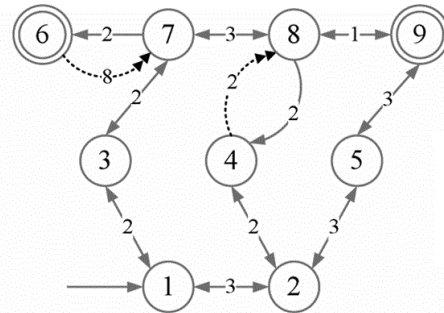


Figura 4. Planta G2 que modela o sistema

Devido à maneira com que os semáforos estão posicionados, os eventos correspondentes às transições da seção 4 para a seção 8 e da seção 6 para a 7 são eventos não-controláveis de perturbação. Todos os outros eventos são controláveis.

O trem inicia sua trajetória no estado 1, e tem-se como objetivo direcionar seu movimento para qualquer uma das sessões 6 ou 9, com mínimo custo. Para alcançar esse resultado, foi utilizado o algoritmo implementado. O diretor obtido indica, na planta, o caminho da figura 5 abaixo.

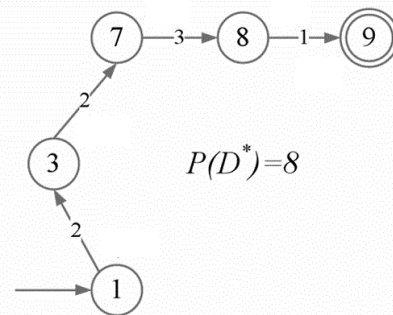


Figura 5. Planta G2 sob controle dirigido ótimo

Esse resultado também é o mesmo daquele obtido em (Huang e Kumar, 2008), indicando mais uma vez que o algoritmo foi programado corretamente.

Estes e diversos outros testes realizados com resultados corretos serviram como forte indício de que a implementação computacional do método para obtenção do diretor ótimo está correta.

A próxima seção tem como objetivo verificar o funcionamento do Diretor em outros contextos, focados na sua aplicação em problemas de sequenciamento de tarefas em sistemas a eventos discretos sob controle supervisorio.

#### 4 Diretor aplicado a Sistemas Concorrentes sob Controle Supervisório

O método de controle dirigido foi desenvolvido para autômatos genéricos para a obtenção de uma sequência que permita a realização de um objetivo com custo mínimo. A partir de sistemas modelados e controlados por meio da teoria de controle supervisório, pretende-se aplicar o controle dirigido e verificar se este é capaz de selecionar o caminho ótimo. Sabendo que o supervisor monolítico é implementado por meio de um autômato que codifica todas as sequências legais do sistema sob controle (autômato maximamente permissivo), optou-se por aplicar o algoritmo para síntese do diretor sobre o supervisor monolítico.

Em algumas das situações testadas, os resultados encontrados não foram aqueles esperados. Sobre estes casos, foram feitos estudos detalhados. Segue abaixo um exemplo de um desses problemas.

##### 4.1) Estudo de caso: Linha de Produção Industrial

O teste realizado consiste na obtenção da sequência ótima para produção de peças no sistema apresentado abaixo na Figura 6. Trata-se de uma simplificação do problema de SED apresentado em (de Queiroz e Cury, 2000) para modelagem de um sistema de produção industrial.

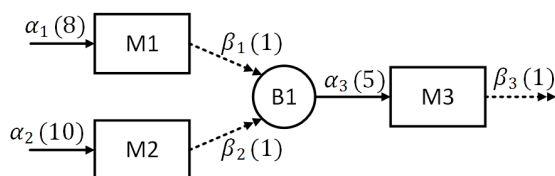


Figura 6. Linha de produção industrial

Neste sistema, M1, M2 e M3 representam 3 máquinas em uma linha de produção, e B1 um buffer com capacidade para duas peças. As máquinas M1 e M2 produzem peças que são depositadas no buffer ao final do processo. Quando duas peças estão posicionadas no buffer, a máquina M3 as retira e com ambas produz um único produto final.

A modelagem do sistema por meio de autômatos é representada na Figura 7. Os autômatos que modelam o comportamento cíclico de cada máquina tem dois estados possíveis: máquina desligada (estado inicial) ou em funcionamento. A especificação Eb1 do buffer B1 indica sua capacidade máxima de duas peças, evitando ocorrências de overflow ou underflow. Para cada evento, o número entre parênteses que o segue indica seu custo associado. Os eventos  $\alpha_i$  são eventos controláveis que indicam o ligamento das máquinas  $M_i$ , e tem como custo o tempo que as máquinas permanecem ligadas, em segundos. Os eventos  $\beta_i$  são eventos não-controláveis que indicam o desligamento das máquinas  $M_i$ , e tem custos iguais a 1 para indicar que a máquina não tem um tempo

mínimo para permanecer no estado desligado (ou seja, após ser desligada ela já pode ser religada imediatamente).

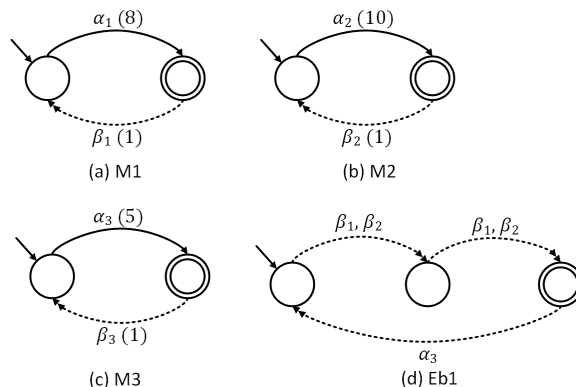


Figura 7. Autômatos para modelagem da planta

A partir dos autômatos acima, é obtido o supervisor monolítico para a planta. Sobre ele, é aplicado o algoritmo para obtenção do diretor ótimo. A planta controlada pelo diretor obtido obedece à seguinte sequência de eventos para produção de um produto:

$$s1: \alpha_1 - \beta_1 - \alpha_1 - \beta_1 - \alpha_3 - \beta_3$$

O custo do diretor encontrado é dado por  $P(D) = X$ . Observando-se passo a passo a sequência de eventos, esse custo pode ser verificado:

1. ( $t = 0$ ) A máquina M1 é ligada;
2. ( $t = 8$ ) A máquina M1 termina seu funcionamento e é desligada;
3. ( $t = 9$ ) A máquina M1 é ligada novamente;
4. ( $t = 17$ ) A máquina M1 novamente termina seu funcionamento e é desligada;
5. ( $t = 18$ ) A máquina M3 é ligada;
6. ( $t = 23$ ) A máquina M3 chega ao fim do seu funcionamento e é desligada.
7. Tempo final:  $T(s1) = 24s$ .

No entanto, considera-se essa segunda sequência de eventos alternativa:

$$s2: \alpha_2 - \alpha_1 - \beta_1 - \beta_2 - \alpha_3 - \beta_3$$

Analogamente, é feita a verificação passo a passo da sequência s2.

1. ( $t = 0$ ) A máquina M2 é ligada;
2. ( $t = 1$ ) Com a máquina M2 ainda ligada, a máquina M1 é ligada, e ambas passam a funcionar simultaneamente;
3. ( $t = 9$ ) A máquina M1 termina seu funcionamento e é desligada;
4. ( $t = 10$ ) A máquina M2 termina seu funcionamento e é desligada;
5. ( $t = 11$ ) A máquina M3 é ligada;
6. ( $t = 16$ ) A máquina M3 chega ao fim do seu funcionamento e é desligada.
7. Tempo final:  $T(s2) = 17s$ .

Como pode ser observado, em ambas as situações o objetivo é alcançado e uma peça é produzida. No entanto, a sequência s2 a produz em um tempo MENOR que o tempo obtido na sequência s1. Nesse

caso, a otimização com utilização do método de controle dirigido proposto não alcança o resultado ótimo desejado.

## 5 Discussão dos Resultados

A otimização através do método de controle dirigido desenvolvido em (Huang e Kumar, 2008) e (Huang e Kumar, 2007) atua no sentido de minimizar o gasto de um recurso qualquer utilizado pelo sistema. A quantidade de recurso dispendida para a ocorrência de um evento atribuída como custo deste evento. A aplicação do algoritmo de síntese do diretor ótimo pode garantir que a quantidade de recursos utilizada por elementos do sistema seja mínima. Em grande parte dos casos, o método realmente permite que o objetivo seja alcançado com gasto mínimo do recurso em questão. Verificou-se, no entanto, que em situações nas quais se busca minimizar o tempo gasto, o objetivo nem sempre é atingido, como no exemplo discutido anteriormente.

Essa divergência se dá devido a diferentes objetivos de otimização. Pela concepção de otimização utilizada na síntese do diretor ótimo, o recurso utilizado no sistema é minimizado, no sentido de que o tempo total gasto individualmente por cada elemento do sistema é totalmente reduzido. Ou seja, a soma dos tempos em que cada elemento do sistema permanece executando uma tarefa é a menor possível. No entanto, há situações onde o conceito desejado de otimização do recurso tempo é outro. O objetivo pode ser minimizar o tempo total para que o sistema realize uma tarefa ou alcance determinado estado. Nesses casos, a aplicação do método de otimização desenvolvido não garante o resultado ótimo no sentido desejado.

Isso ocorre porque o método utilizado não considera a possibilidade de *recursos compartilhados* no sistema, isto é, a utilização simultânea do mesmo recurso para realização de diferentes tarefas. O recurso que mais facilmente exemplifica esse conceito é o tempo. Em um mesmo instante de tempo, duas ou mais tarefas podem estar em execução simultaneamente. Desse modo, o tempo para conclusão de ambas é menor que a soma dos tempos de execução de cada uma separadamente.

Nos resultados observados, não foram obtidos os resultados desejados para problemas nos quais ocorre compartilhamento de recursos. É razoável supor-se então a existência de uma classe de problemas na qual o método de otimização com controle dirigido utilizado não pode garantir resultado ótimo.

## 6 Conclusão

No trabalho, foi realizada a implementação computacional do algoritmo para síntese de um diretor ótimo. Além disso, verificou-se a aplicabilidade do método de controle dirigido para otimização de sistemas de tipos diversos, com ênfase em sistemas

concorrentes de manufatura industriais, como alternativa aos métodos baseados em técnicas de otimização utilizando meta-heurísticas.

Observa-se que a técnica discutida nesse trabalho não se aplica para os tipos de problemas tratados. Em trabalhos futuros, propõe-se adaptar o método de controle dirigido ótimo para sistemas com compartilhamento de recurso, permitindo sua aplicação nos casos de interesse.

## Agradecimentos

Os autores agradecem o apoio da PRO-GRAD/UFGM, Procad-Capes e Fapemig.

## Referências Bibliográficas

- Cassandras C. G. e Lafortune S. (1999). Introduction to Discrete Event Systems. Boston, MA: Kluwer.
- Costa, T. A. ; Oliveira, A. C. ; Pena, P. N. ; Takahashi, R. H. C. (2012). An Ant System Algorithm for Task Scheduling in a Flexible Manufacturing Cell with Supervisory Control. XIX Congresso Brasileiro de Automática, CBA'12, pp. 2515-2522.
- Huang, J. e Kumar, R. (2005). Nonblocking Directed Control of Discrete Event Systems. Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005, pp. 7627- 7632.
- Huang, J. e Kumar, R. (2007). An optimal directed control framework for discrete event systems, IEEE Trans. Syst., Man, Cybern. A, vol. 37, No. 5, pp. 780- 791.
- Huang, J. e Kumar, R. (2008). Optimal Nonblocking Directed Control of Discrete Event Systems. IEEE Transactions on Automatic Control, Vol. 53, No. 7, pp. 1592- 1603.
- Oliveira, A. C. ; Costa, T. A. ; Pena, P. N. e Takahashi, R. H. C. (2013). Supervisory Control of Flexible Manufacturing Cells with an Evolutionary Computation Approach for Schedule Synthesis. Submetido para IEEE Transactions on Automation, Science and Engineering, 2013
- de Queiroz, M. H. e Cury, J. E. R. (2000). Proceedings of the 5th Workshop on Discrete Event Systems, WODES'00, Vol. 1, Ghent, Belgium, pp. 103-110.
- Ramadge, P. e Wonham, W. (1989). The control of discrete event systems. IEEE, Special Issue on Discrete Event Dynamic Systems, pp. 81-98.
- Silva, R. S. ; Oliveira, A. C. ; Pena, P. N. ; Takahashi, R. H. C. (2011). Algoritmo Clonal para Job Shop Scheduling com Controle Supervisório. X Simpósio Brasileiro de Automação Inteligente, pp. 1376-1381.