

SIMULAÇÃO DE SISTEMAS A EVENTOS DISCRETOS SOB O CONTROLE SUPERVISÓRIO MODULAR LOCAL

JÉSSICA K. S. SANTOS¹, PEDRO P. B. BRAGA¹, LEANDRO DE S. D. DE P. VIEIRA¹, PATRÍCIA N. PENNA²

1. *Engenharia de Controle e Automação, LACSED-DELT - Laboratório de Análise e Controle de Sistemas a Eventos Discretos, Departamento de Engenharia Eletrônica, Universidade Federal de Minas Gerais -*

E-mails: jessicakatielle@ufmg.br, pedrobraga@ufmg.br, leandro.vieira1@gmail.com.

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

Abstract— This paper presents a methodology for simulation of discrete event systems under local modular supervisory control. The proposed methodology adapts the simulation of discrete event systems as in Cassandras and Lafortune (1999) to deal with systems composed of multiple subsystems and multiple supervisors. The simulations presented are used in other researches from the group to evaluate sequences obtained through optimization methods in the problem of scheduling in manufacturing cells. The simulations are implemented in MATLAB.

Keywords— supervisory control, local modular control, discrete event systems, simulation.

Resumo— Neste artigo apresenta-se uma metodologia para simulação de sistemas a eventos discretos sob o controle supervisório modular local. A metodologia adapta a proposta de simulação de sistemas a eventos discretos de Cassandras e Lafortune (1999) para tratar de sistemas compostos de múltiplas plantas e múltiplos supervisores. As simulações apresentadas são utilizadas em outras pesquisas do grupo, com objetivo de avaliar seqüências obtidas por meio de métodos de otimização no problema de sequenciamento de tarefas em células de manufatura. As simulações são implementadas em MATLAB.

Palavras-chave— controle supervisório, controle modular local, sistemas a eventos discretos, simulação.

1 Introdução

O avanço da tecnologia e da automação de processos tem dado origem a sistemas de funcionamento complexo e de difícil modelagem segundo as técnicas convencionais. A automatização de processos de grande porte do ambiente industrial tais como a operação de robôs, esteiras entre outros apresentam caráter discreto e não podem ser tratados por meio das técnicas de controle clássica.

Esses sistemas diferem dos convencionais por serem geridos não pela passagem do tempo e sim pela ocorrência de eventos. Esses eventos são caracterizados, por exemplo, pela mudança de estado de um sensor de fim de curso que pode indicar o término de uma tarefa. Devido a essa dependência dos eventos, eles são chamados de Sistemas a Eventos Discretos (SED).

Diversas formas de modelagem são sugeridas para esse tipo de sistema, desde Redes de Petri a Teoria de Linguagens e Autômatos e a partir dessa modelagem técnicas de controle específicas podem ser aplicadas. A Teoria de Controle Supervisório (TCS) (Ramadge e Wonham, 1989) é uma das soluções para o problema do controle de SEDs baseada em sistemas modelados por autômatos.

Uma das principais vantagens de se utilizar a TCS para realizar o controle sequencial está na garantia de obtenção de solução não-bloqueante. Por

outro lado, a TCS provê uma solução minimamente restritiva, ou seja, a TCS implementa o conjunto de todas as trajetórias que não violam as especificações. No caso de uma indústria, é necessário que exista uma seqüência dentre todas que seja a escolhida para executar.

Desta forma, acredita-se que a efetiva aplicação da TCS em sistemas industriais reais passa pela introdução de alguma metodologia que indique qual evento deve ser executado e em qual ordem. O uso dos métodos heurísticos e meta-heurísticos em conjunto com a TCS provê este complemento. Por um lado tem-se a execução segura e livre de bloqueios garantida. Por outro lado, com auxílio dos métodos de otimização, é possível selecionar a execução ótima segundo algum funcional.

Silva et al (2011) e Costa et al (2012) propõem o uso de heurísticas e metaheurísticas para obter uma seqüência de eventos que corresponda à operação otimizada e fazem uso de um simulador que executa as seqüências geradas pra avaliar o custo das soluções. Dessa forma, este artigo apresenta os simuladores que foram construídos com o objetivo de servir à pesquisa de SED junto com Otimização e permitindo validar a metodologia.

Ao todo, até o presente momento, foram implementados simuladores para 5 exemplos dos quais 3 já estão sendo utilizados em outros trabalhos

(Silva et al, 2011, Costa et al, 2012 e Oliveira et al, 2013a).

2 Noções preliminares

Nesta seção, apresentam-se as noções preliminares relativas ao Controle Supervisório de SEDs e também, de forma resumida, os principais aspectos da simulação de SEDs.

2.1 SEDs e a TCS

No trabalho utiliza-se como base o formalismo desenvolvido por Ramadge e Wonham (1987) segundo o qual SEDs são modelados por autômatos. Um autômato G é representado por:

$$G = (Q, \Sigma, \delta, q_0, Q_m)$$

onde Q é o conjunto de estados; Σ é o conjunto de eventos; δ é a função de transição; q_0 é o estado inicial e Q_m é o conjunto de estados marcados. Tem-se ainda que, Σ^* é o conjunto de todas as cadeias finitas de elementos que incluem a cadeia vazia ϵ . Seguindo essas definições, linguagens são subconjuntos de Σ^* .

O conjunto de cadeias geradas pelo autômato G é denotado por:

$$L(G) \subseteq \Sigma^*,$$

e

$$L_m(G) \subseteq L(G),$$

é o conjunto de cadeias que terminam em estados marcados e representam tarefas completas.

Na TCS os autômatos são divididos em grupos chamados “especificações” e “plantas”. Os eventos usados para modelar a planta são divididos em dois subconjuntos correspondentes aos eventos controláveis e eventos não-controláveis. Um evento controlável é aquele sobre o qual o operador pode atuar e um não controlável, aquele cuja execução não pode ser manipulada representando, normalmente, a resposta da planta.

A TCS utiliza a composição das plantas e especificações para obter um terceiro autômato que implementa a linguagem desejada. Quando não é possível implementar a linguagem desejada, a especificação é dita não-controlável. Nestes casos, o supervisor implementa a máxima sublinguagem controlável da linguagem desejada, $\text{SupC}(K, G)$, em que K é a linguagem desejada e é obtida pela composição da especificação dada por E e $L_m(G)$. Esse controlador visa obter o comportamento desejado da planta de uma forma minimamente restritiva, ou seja, ele procura evitar que estados indesejáveis sejam alcançados. Nesse sentido a atuação do supervisor se resume a desabilitar a ocorrência de eventos controláveis visando alcançar um comportamento desejado. Fazendo analogia com sistemas de controle realimentado, o controlador da TCS observa o estado atual da planta, processa quais eventos não podem ocorrer e retorna como ação de

controle – realimentação – somente os eventos possíveis ou factíveis naquele momento.

O supervisor obtido pela TCS está sujeito a crescimento exponencial do número de estados a medida que a planta tratada se torna mais complexa, tornando-se inviável para sistemas de grande porte. Uma extensão do controle monolítico é proposta por Wonham e Ramadge (1988) a fim de reduzir essa complexidade, a abordagem modular. Esta abordagem propõe obter um supervisor para cada especificação, tratando a planta como única. O Controle Supervisório Modular Local (de Queiroz e Cury, 2000) é outra extensão de Wonham e Ramadge (1988) em que cada supervisor é projetado com uma visão parcial da planta. Como cada supervisor obtido não tem informação da atuação dos outros supervisores, é necessário avaliar se os supervisores são não conflitantes. Em caso positivo, pode-se afirmar que o desempenho do conjunto de supervisores é equivalente ao desempenho do supervisor monolítico. Supervisores reduzidos são obtidos para cada supervisor modular local com o objetivo de separar no supervisor a informação redundante da planta da ação de controle.

A simulação proposta neste trabalho consiste da implementação e execução dos autômatos dos componentes da planta sob o controle exercido pelos supervisores reduzidos.

2.2 Simulação de Sistemas a Eventos Discretos

A simulação de sistemas a eventos discretos proposta na seção 10.2 de Cassandras e Lafortune (1999), denominada de “Esquema de Escalonamento de Eventos”, baseia-se na criação de uma lista de eventos (factíveis no estado corrente) e o tempo associado a eles (na proposta original é um tempo aleatório). Essa lista é ordenada pelo tempo associado aos eventos factíveis e o evento com o menor tempo é escolhido para execução.

O esquema abaixo mostra como ocorre a simulação de um autômato seguindo esta metodologia.

- Inicializa-se o estado corrente (como sendo o estado inicial), o tempo (para zero), a lista de eventos (aqueles factíveis no estado inicial) atribuindo tempos aleatórios aos eventos. Ordena-se a lista pelo evento de menor tempo associado.
 1. Inicia-se o loop
 2. O primeiro evento é removido da lista de eventos escalonados;
 3. Atualiza-se o tempo saltando para o momento de execução do evento removido;
 4. O estado atual do autômato é atualizado de acordo com o evento executado anteriormente;
 5. São apagados da lista os eventos que não são mais possíveis no estado atual;
 6. Os novos eventos possíveis são adicionados à lista;

7. A lista é ordenada de acordo com o menor tempo.
8. Fim do “loop”, com condição de parada dependente da aplicação.

É importante ressaltar que essa metodologia prevê a simulação de somente um autômato, podendo ser usada para simular o Supervisor Monolítico obtidos pela TCS.

3 Simulação de Sistemas Concorrentes

A solução de controle obtida pela Teoria de Controle Supervisório Modular Local gera um conjunto de autômatos que, quando executados em conjunto levam à operação segura da planta.

A simulação de inúmeros autômatos não está prevista na abordagem apresentada na seção 2.2. A obtenção de um autômato só que seja equivalente à operação dos supervisores é inviável devido à explosão do número de estados (o exemplo da seção 3.1 excede os 2.000.000 de estados) sendo inviável a implementação de um simulador que percorre os estados.

Nesse sentido, torna-se necessária a adaptação do método de simulação para atender o caso em que se quer simular múltiplos autômatos que executam concorrentemente. Além disto, há que se implementar a estrutura de desabilitações do supervisor, que também não é prevista.

Os simuladores desenvolvidos são baseados no modelo apresentado por Cassandras e Lafortune (1999) o qual segue uma lógica de escalonamento de eventos.

Os autômatos referentes à planta e ao supervisor reduzido, obtidos por meio da Teoria de Controle Supervisório Modular Local, são representados por matrizes tridimensionais que carregam informações sobre o estado atual, a transição ou evento, o próximo estado e o autômato ao qual esses dados pertencem. A matriz do supervisor é uma matriz de desabilitações que informa, a partir dos estados atuais dos supervisores, os eventos desabilitados em cada estado. Estes autômatos (matrizes) são analisados concorrentemente no decorrer do algoritmo.

Os simuladores utilizam a passagem do tempo como guia e, por isso, a cada transição é associado um intervalo de tempo. Este tempo pode ser interpretado como o tempo que o sistema fica no estado alcançado pela execução do evento. Desta forma, os eventos possíveis no estado só ficam habilitados após a passagem do tempo correspondente. Desta forma, monta-se uma lista de eventos factíveis a cada execução do “loop”. Esses eventos são consumidos de acordo com um parâmetro de escolha, inicialmente, o tempo em que cada um pode ser executado.

Para tanto, implementam-se matrizes que guardam o tempo de execução de cada evento. Aos eventos não controláveis, que, como exposto anteriormente, sinalizam o fim de uma operação, é

associada 1 unidade de tempo. É necessária também, uma variável “Tempo” que guarde o tempo atual de simulação. Outros vetores são utilizados para caracterizar o sistema, como um vetor de estados correntes dos subsistemas e dos supervisores.

O algoritmo implementado em MATLAB é regido pelas regras apresentadas a seguir.

- São carregadas as matrizes das plantas, supervisores e tempos dos eventos.
- São inicializados: estado corrente, tempo da simulação e variáveis auxiliares.
 1. Início do “loop”
 2. Baseado no estado corrente, a matriz correspondente às plantas é percorrida e os eventos habilitados no estado formam uma lista de eventos possíveis de serem realizados.
 3. A matriz de desabilitação é percorrida e os eventos desabilitados no estado corrente são retirados da lista de eventos possíveis.
 4. A cada evento que sobrevive aos passos 4 e 5 é associado o tempo em que ele pode acontecer.
 5. A lista de eventos é ordenada de forma crescente de acordo com o tempo que cada evento pode ocorrer.
 6. Um evento é selecionado se o tempo decorrido “Tempo” for maior ou igual ao tempo de pelo menos um evento da lista.
 7. Em caso de empate nas primeiras posições da lista, o evento não-controlável é sempre escolhido. Nas ocasiões em que há dois ou mais eventos controláveis e nenhum não controlável o desempate é feito de maneira aleatória. O mesmo é feito para o caso de dois ou mais eventos não controláveis empatados.
 8. Executa-se o evento escolhido (caso seja escolhido algum) e atualiza-se o estado corrente (estado atual dos subsistemas e supervisores).
 9. Armazena-se o tempo de execução do evento para atualização do tempo de execução dos eventos relacionados.
 10. $Tempo = Tempo + 1$
 11. Fim do “loop”, com condição de parada dependente da aplicação.

As simulações nos moldes apresentados acima foram feitas para os cinco exemplos apresentados a diante.

3.1 Sistema Flexível de Manufatura - SFM

A planta mostrada na Figura 1, originalmente apresentada em de Queiroz et al. (2005) é constituída por 8 equipamentos: 3 esteiras (C1, C2 e C3), 1 robô, 1 fresa, 1 torno, 1 máquina de pintura (MP) e 1 máquina de montagem (MM), além destes há também 8 equipamentos de armazenamento temporário, depósitos de capacidade unitária. São

produzidos 2 tipos de produtos a partir de blocos brutos. Os produtos finais são constituídos de uma base com um pino cônico (Produto A) e uma base com um pino cilíndrico pintado (Produto B).

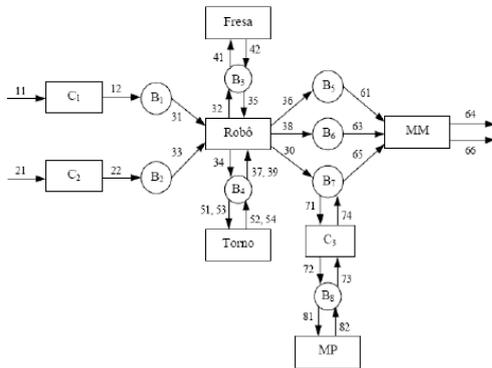


Figura 1. Sistema Flexível de Manufatura

A matéria-prima bruta entra no sistema pelas esteiras C1 e C2 e, a depender do que irá constituir, ela passa pela fresa (para formar as bases) ou pelo torno (para formar os pinos). Nota-se na Figura que os eventos são representados pelas setas entre os blocos de equipamentos e depósitos. O Controle Supervisório Modular Local desta planta, conforme obtido por Pena et. al.(2010), e possui 7 supervisores.

Foram implementados os autômatos dos 8 componentes, bem como os autômatos dos 7 supervisores reduzidos bem como suas estruturas de desabilitações. Este simulador desse sistema vem sendo usado nos trabalhos de Silva et al (2011), Costa et al (2012), Oliveira et al (2013a) e Oliveira et al (2013b) em que o objetivo é obter a sequência ótima para produção de lotes específicos de produtos A e B, sendo a duração da produção do lote a variável a ser otimizada.

3.2 Sistema Integrado de Manufatura

O Sistema Integrado de Manufatura da Figura 2, originalmente apresentado em Pena et. al. (2009) é composto 2 tornos, 2 máquinas de pintura (MP), 2 veículos auto-guiados (AGV), 1 máquina de montagem (MP) e 6 depósitos de capacidade unitária.

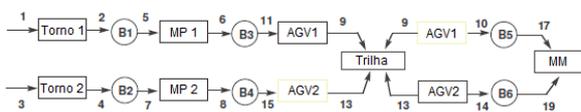


Figura 2. Sistema Integrado de Manufatura

Os veículos auto-guiados retiram as peças do depósito correspondente e as levam até o destino final na máquina de montagem. Observa-se o veículo 1 começa sua trajetória no depósito da esquerda, B3, e o veículo 2, no depósito da direita, B6. Os eventos correspondentes a ir de um depósito a outro e voltar são os mesmos. Além de não *overflow* e não *underflow* dos depósitos, deve-se garantir que somente um AVG ocupa a trilha por vez.

O objetivo da otimização é obter a sequência ótima para produção de lotes de 1 a 6 produtos, sendo a duração da produção do lote a variável a ser otimizada (Oliveira et. al.,2013a).

3.3 Linha de Transferência Industrial

A Linha de Transferência Industrial mostrada na Figura 3 (de Queiroz, 2000) é composta de 6 máquinas (M_i , $i = 1, \dots, 6$) e 4 depósitos (B_j , $j = A, \dots, D$) de capacidade unitária e somente um tipo de produto é manufaturado.

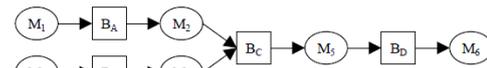


Figura 3. Linha de Transferência Industrial

No simulador esses dois ramos (M_1M_2 e M_3M_4) possuem tempos associados bem diferentes para caracterizar um ramo mais rápido e outro mais lento. Assim a produção de blocos de forma paralela pode ser mais bem analisada. Os resultados dos trabalhos com este simulador podem ser encontrados em (Costa et. al.,2013).

3.4 Célula de Manufatura

A Célula de Manufatura (de Queiroz, 2004) é composta por uma mesa circular com quatro posições que correspondem a estações de operação, como visto na Figura 4.

O funcionamento do sistema é baseado nas rotações da mesa em torno dessas estações. Inicialmente a matéria-prima bruta sai de um depósito e entra numa esteira que a levará até a posição P1. Com a peça na primeira estação, a mesa gira 90° e a peça é levada à segunda estação onde há uma furadeira. Outro giro de 90° a leva para uma estação de teste, após mais um giro a peça é retirada da mesa por um braço robótico.

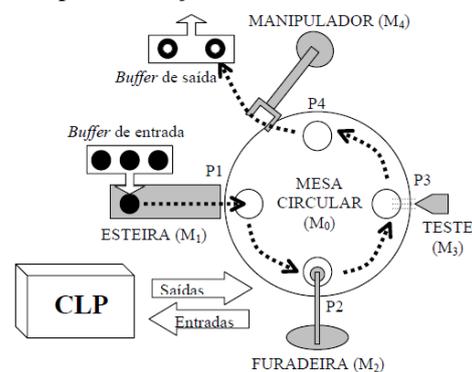


Figura 4. Célula de Manufatura

As especificações deste sistema consistem no giro correto da mesa, ou seja, que esta não gire quando uma estação estiver operando e garantindo, também, que a operação segura das estações. O simulador encontra-se funcional, porém ainda não foi utilizado em conjunto com as técnicas de otimização.

3.5 Sistema de Manufatura

O Sistema de Manufatura da Figura 6 (Ezpeleta, Colom e Martinez, 1995) pode produzir 3 tipos diferentes de produtos, A, B e C. Este sistema é composto por 13 equipamentos entre eles 6 esteiras ($C_i, i = 1, \dots, 6$), 4 máquinas ($M_j, j = 1, \dots, 4$) e 3 robôs ($R_m, m = 1, 2, 3$). Os equipamentos são conectados por 10 depósitos ($B_k, k = 1, \dots, 10$) dos quais B2, B4, B7 e B9 possuem capacidade 2 e os demais capacidade unitária.

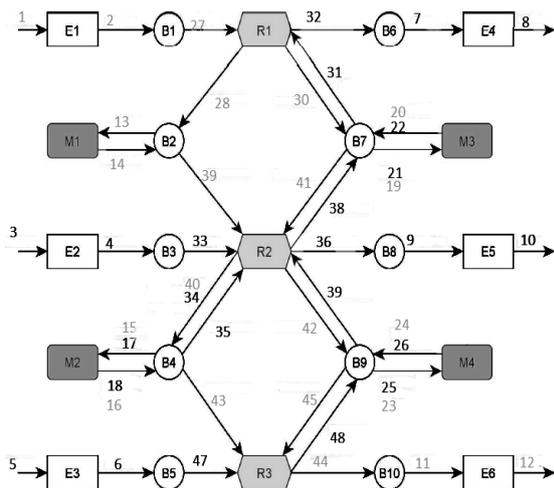


Figura 5. Sistema de Manufatura

O produto A pode ser produzido de duas formas diferentes sendo a primeira correspondente aos eventos [1 27 13 39 15 43 11] e o segundo, [1 27 19 41 23 45 11]. O produto B segue o caminho dado por [3 33 17 35 9] e o C por [5 47 25 37 21 31 7]. As especificações implementam não *overflow* e não *underflow* nos depósitos.

Este sistema é, entre todos apresentados, o de maior complexidade. Deverá ser utilizado para validar a metodologia proposta em (Costa et al., 2012) e (Oliveira et al., 2013).

4 Estudo de Caso: Integração com a Otimização

Os simuladores das soluções da TCS tem grande utilidade na Pesquisa Operacional. Como a TCS fornece soluções minimamente restritivas para os sistemas em questão, métodos de pesquisa operacional podem ser aplicados para se selecionar soluções que atinjam certos objetivos de forma ótima. Dessa forma, as simulações realizadas foram utilizadas com métodos de otimização que procuravam minimizar o tempo de produção.

Até então foram utilizados métodos de otimização como o Algoritmo de Seleção Clonal, Colônia de Formigas, VNS, ILS e variações destas (Costa, et al, 2012). Foram feitos ajustes no modelo de simulação apresentado na seção 3 para que este interagisse da forma desejada com os algoritmos de otimização (também feitos em MATLAB).

O simulador é utilizado como uma função, chamada pelo otimizador, que recebe como parâmetros o número de produtos a serem produzidos e uma sequência de eventos que supostamente leva à produção destes. Como resposta para o algoritmo de otimização é dado o tempo gasto para se realizar tal sequência – caso seja uma sequência válida – ou a ponto da produção (evento) na qual o simulador travou e uma sugestão de eventos factíveis para serem executados – caso a sequência não seja válida. Nota-se que o simulador assume o papel de avaliador do resultado da otimização, sendo assim a função objetivo.

Dentre as modificações necessárias no simulador está a inclusão de “máscaras” na sequência de eventos e a modificação da condição de parada. Foi necessário adequar o simulador com a codificação adotada no algoritmo de otimização. Na otimização procura-se a melhor sequência de eventos que produzam um certo lote de produtos, assim o critério de parada passa a ser o final da produção (esgotamento dos eventos da sequência de entrada).

Um detalhamento maior das alterações pode ser exemplificado pelo Sistema Flexível de Manufatura (seção 3.1). Neste sistema produzem-se dois produtos: produto A (base e pino cônico) e produto B (base e pino cilíndrico pintado). A sequência de eventos controláveis que, se executadas, dão origem a cada parte é dada por:

- Base = [11 31 41 35 61].
- Pino cônico = [21 33 51 37 63].
- Pino cilíndrico pintado = [21 33 53 39 71 81 73 65].

Os eventos das sequências correspondentes ao produto A foram identificadas como a_0 – para a base – e a_1 – para o pino, e para B tem-se b_0 – para a base – e b_1 – para o pino. Dessa forma um produto A passa a ser identificado pelo simulador como Base A + Pino A = [$a_0 a_0 a_0 a_0 a_0 a_1 a_1 a_1 a_1 a_1$], onde o primeiro a_0 é sempre o primeiro evento do vetor Base, o segundo a_0 corresponde ao segundo evento do vetor Base e assim por diante. O mesmo ocorre para o produto B.

Como as heurísticas de otimização utilizadas baseiam-se na geração de novos indivíduos e esta geração pode se dar pela troca de posição de elementos da sequência, o uso das máscaras a_0, a_1, b_0 e b_1 reduz bastante a infactibilidade. Por meio destas, garante-se que o sequenciamento mínimo que deve existir, representados pelos vetores, seja mantido; reduzindo o esforço computacional. Mais detalhes sobre o uso conjunto dos simuladores com os algoritmos de otimização são encontrados em (Oliveira et al., 2013).

5 Conclusões

A metodologia para implementar os simuladores foi adaptada de (Cassandras e Lafortune, 1999) para lidar com múltiplos autômatos de forma a permitir a simulação da solução do Controle Supervisório

Modular Local. Nesta solução, autômatos das plantas e dos supervisores reduzidos são implementados e a simulação ocorre pela execução conjunta de todos estes autômatos. Desta forma, a metodologia torna-se mais geral, podendo ser utilizada em sistemas de maior porte (que não são solúveis pela TCS clássica).

Os simuladores implementados mostraram-se importante ferramenta para realização da Otimização por meio de heurísticas e metaheurísticas. Várias simulações foram desenvolvidas para validar a metodologia proposta em Oliveira et. al.(2013a), Oliveira et. al. (2013b), Costa et.al. (2012) e Silva et. al. (2011).

O simulador do Sistema de Manufatura da Seção 3.5, bem como sua adequação para operar em conjunto com a Otimização, por se tratar de um sistema bem mais complexo que os demais, permitirá, acredita-se, um posicionamento mais definitivo sobre a viabilidade de apresentar a metodologia (otimização + TCS) como opção para a solução de problemas de escalonamento de tarefas de sistemas a eventos discretos.

Agradecimentos

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

Referências Bibliográficas

- CASSANDRAS, C. e LAFORTUNE, S. (1999). Introduction to Discrete Event Systems - First Edition, Springer.
- RAMADGE, P. e WONHAM, W. (1987). Supervisory Control of a Class of Discrete Event Processes, SIAM Journal Control and Optimization pp. 206–230.
- WONHAM, W. e RAMADGE, P. (1988). Modular Supervisory Control of Discrete Event Systems, Math. Control, Signals Systems 1: 13–30.
- DE QUEIROZ, M. H. DE. Controle Supervisório Modular e Multitarefa de Sistemas Compostos. Tese de Doutorado, Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia Elétrica, Florianópolis, maio de 2004.
- DE QUEIROZ, M. H. DE. Controle Supervisório Modular de Sistemas de Grande Porte. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia Elétrica, Florianópolis, maio de 2000.
- DE QUEIROZ, M., CURY, J. E WONHAM, W.(2005). Multitasking Supervisory Control of Discrete-Event Systems, Discrete Event Dynamic Systems: Theory and Applications 15: 375–395.
- EZPELETA, J.; COLOM J.M.; MARTINEZ, J.. "A Petri net based deadlock prevention policy for flexible manufacturing systems," Robotics and Automation, IEEE Transactions on, vol.11, no.2, pp.173-184, Apr 1995.
- COSTA, T.A., OLIVEIRA, A. C., PENA, P. N., TAKAHASHI, R. H. C.. An Ant System Algorithm for Task Scheduling in a Flexible Manufacturing Cell with Supervisory Control In: XIX Congresso Brasileiro de Automática, CBA'12, 2012, Campina Grande. Anais do XIX Congresso Brasileiro de Automática, 2012. p.2515 – 2522.
- OLIVEIRA, A. C., COSTA T.A., PENA, P. N., TAKAHASHI, R. H. C.. Supervisory Control of Flexible Manufacturing Cells with an Evolutionary Computation Approach for Schedule Synthesis. In: Submetido para IEEE Transactions on Automation and Science Engineering., 2013a
- OLIVEIRA, A.C., COSTA, T.A., PENA, P.N., TAKAHASHI, R.H.C., Clonal Selection Algorithms for Task Scheduling in a Flexible Manufacturing Cell with Supervisory Control, Proceedings of the IEEE Congress on Evolutionary Control - Cancun - Mexico, 2013b, pp. 982–988
- PENA, P.N.; CURY, J.E.R.; LAFORTUNE,S.. Verification of Nonconflict of Supervisors Using Abstractions. In: IEEE Transactions on Automatic Control, 2009, 54, 2803-2815
- PENA, P.N.; CUNHA, A.E.C.; CURY, J.E.R.; LAFORTUNE, S.. Metodologia e Ferramenta de Apoio ao Teste de Não-Conflito no Controle Modular de Sistemas a Eventos Discretos. In: Controle e Automação, 2010, 21, 58-68
- SILVA, R. S.; OLIVEIRA, A. C.; PENA, P. N.; TAKAHASHI, R. H. C.. Algoritmo Clonal para Job Shop Scheduling com Controle Supervisório. In: X Simpósio Brasileiro de Automação Inteligente, 2011, São João del Rei. Anais do X Simpósio Brasileiro de Automação Inteligente, 2011. p. 1376-1381.