

ESTRUTURAÇÃO DA MODELAGEM DE PROCESSOS EM SISTEMAS PRODUTIVOS

DIOLINO JOSÉ DOS SANTOS FILHO*#, JOSÉ REINALDO SILVA*#,
NEWTON MARUYAMA*, PAULO EIGI MIYAGI*

* Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Escola Politécnica da USP
Av. Prof. Mello Moraes, 2231
CEP05508-900 – São Paulo - SP

Centro de Pesquisa em Informática – UNIFIEO
Rua Narciso Sturlini, 883
CEP 06018-903 - Bela Vista - Osasco - SP

E-mails: diolinos@usp.br, reinaldo@usp.br, maruyama@usp.br, pemiyagi@usp.br

Resumo— Os avanços recentes em recursos computacionais tem permitido o tratamento de problemas complexos no projeto de sistemas de controle de sistemas de produção. Dentro deste contexto, este trabalho apresenta uma abordagem algorítmica para o projeto de sistemas de controle. O conceito fundamental considerado é a estruturação do processo de modelagem. Uma analogia aos conceitos clássicos de linguagens estruturadas definida, no sentido de estabelecer uma metodologia para o projeto estruturado de sistemas de controle de sistemas de produção que contempla o E-MFG (Enhanced Mark Flow Graph) e o PFS (Production Flow Schema). Neste sentido, o processo de sequenciamento é sintetizado através de uma estrutura hierárquica, o que facilita a interpretação e a manutenção do projeto além de permitir a sua reutilização.

Abstract— Recent improvements in computational resources have been used to address complexity in control system design of production systems. In this context, this work presents an algorithmic approach to handle control system design of these systems. The fundamental concept considered here is the structuring of modelling process. An analogy of the classical concepts of structured languages is defined in order to establish a methodology for a structured control system design of production systems which contemplates the E-MFG (Enhanced Mark Flow Graph) and the PFS (Production Flow Schema). In this way, the process sequencing is expressed through a hierarchical structure, which allows design interpretation and maintenance together with design reuse.

Keywords: Petri Nets, Production systems, Structured Programming, Mark Flow Graph.

1 Introdução

De uma forma genérica, pode-se dizer que o projeto de um sistema de controle de um SP (sistema produtivo) inicia-se a partir da especificação inicial da funcionalidade que se deseja associar ao sistema para que seja possível realizar um determinado conjunto de processos. A partir desta abstração inicial correspondente ao modelo funcional de cada um dos processos, o objetivo é obter uma descrição formal de um algoritmo de controle para ser implementado em uma arquitetura de controladores programáveis capaz de atuar fisicamente sobre a planta que constitui o SP propriamente dito.

Há várias propostas no sentido de sistematizar o procedimento de projeto do sistema de controle de um SP. Por exemplo, em Santos Filho (1998), a proposta de sistematização baseia-se na metodologia Enhanced Mark Flow Graph/ Production Flow Schema and Resources, ou ainda, E-MFG/PFS-R (Santos Filho e Miyagi, 1995; Miyagi *et alii*, 1998).

Ainda que os resultados obtidos tenham sido efetivos, há um outro aspecto que deve ser observado e que não foi considerado em tais trabalhos. Trata-se do problema de estruturação do processo de modelagem.

Mesmo que seja possível aplicar o conceito de refinamentos sucessivos para a modelagem dos

processos utilizando modelos PFS-R e, posteriormente, utilizando modelos E-MFG, o projetista ainda está sujeito ao nível de dificuldade correspondente à representação do compartilhamento de recursos e, dependendo da flexibilidade operacional do sistema, torna-se necessário também projetar compensadores para programar a alocação destes recursos para que não haja “deadlock” no sistema. Tanto esta alocação de recursos quanto a inserção destes compensadores comprometem a estruturação durante o processo de modelagem.

Portanto, ainda que as metodologias existentes auxiliem no processo de modelagem de SPs, pretende-se contribuir de uma forma efetiva, objetivando estruturar o processo de modelagem do controle de SPs utilizando-se os conceitos de programação estruturada.

Uma vez estabelecida a identidade entre o projeto de algoritmos computacionais e o projeto de sistemas de controle de SPs, este trabalho descreve a proposta de uma estruturação da modelagem do sistema de controle e apresenta uma metodologia descrevendo em detalhes os procedimentos que devem ser aplicados para tornar mais eficiente o projeto do algoritmo de controle de processos em SPs.

2 PFS e a Técnica de Refinamento Sucessivos

Estabelecendo um paralelo com a programação estruturada, o PFS é um tipo de modelo baseado em rede

de Petri elementar estendida que permite uma descrição funcional hierárquica do sistema a partir de um modelo conceitual de elevado nível de abstração. O procedimento de modelagem em Production Flow Schema ou PFS (Miyagi, 1996) baseia-se em aplicar a técnica de refinamentos sucessivos, de maneira que seja possível inserir progressivamente o detalhamento do processo dentro do modelo, correspondente ao comportamento desejado para o sistema.

É importante observar que o PFS não representa o comportamento dinâmico de um SP, uma vez que se trata de uma representação gráfica baseada em redes em que não há marcação. Neste contexto, o objetivo do PFS é representar o fluxo de operações (sequenciamento) tendo como referência a evolução de um determinado processo.

O refinamento em níveis de abstração é realizado através da aplicação de uma abordagem “top-down” em que se assume que a relação entre os elementos básicos está inteiramente descrita no nível anterior. Assim, o processo de estruturação (Marca, 1988) pode contribuir enormemente para o projeto e verificação do modelo, desde que as propriedades principais de um nível de abstração não sejam alteradas pelo refinamento.

Considerando estes aspectos envolvendo o conceito de refinamentos sucessivos, há uma perfeita correspondência desta técnica de modelagem com os procedimentos de programação estruturada. Este fato é uma das razões porque o PFS tem efetivamente contribuído para uma série de aplicações de sistemas discretos, em um contexto bastante abrangente, como pode ser observado em vários trabalhos recentes (Gustin *et alii*, 2000; Miyagi *et alii*, 2000; Santos Filho *et alii*, 2000; Villani *et alii*, 2000).

Entretanto, há o aspecto de modularização que deve ser observado com mais critério. Sabe-se que em um modelo PFS os *elementos-atividade* têm como função representar um sub-processo. Também neste caso, a escolha de um sub-processo (caracterizado por uma sequência de atividades pertencente a um processo maior) não é guiada por nenhuma teoria ou esquema, do mesmo modo que a escolha de procedimentos e funções em uma linguagem estruturada. Há portanto um grande número de possibilidades para o refinamento de uma atividade, alguns dos quais podem levar a dependências que não podem ser inseridas em um esquema de refinamentos (no caso dos sistemas discretos).

Observa-se, portanto, que as atividades podem ser interpretadas como módulos e que há uma entrada e saída bem definida que caracteriza a funcionalidade de cada um destes módulos.

Analisando criteriosamente a proposta do PFS em Miyagi (1988), verifica-se que o *elemento-atividade* corresponde a um macro-elemento delimitado por “[“ e “]”, havendo inscrições em linguagem natural (ou formal) para especificá-lo. Os arcos orientados que estiverem diretamente conectados aos símbolos “[“ e “]” representam o fluxo principal, enquanto os arcos conectados à parte interna da atividade representam o fluxo secundário (Figura 1).

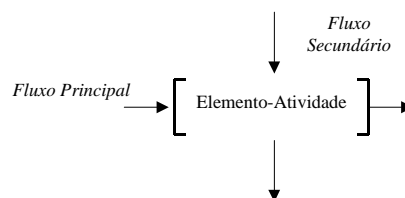


Figura 1: Os fluxos no PFS.

Uma vez que se está investigando o aspecto modular de um *elemento-atividade* no processo de modelagem, observa-se que o fluxo secundário gera um precedente para que possam existir entradas e saídas incondicionais quebrando a estruturação. Portanto, para manter o PFS estruturado a primeira alteração que deve ser efetuada no procedimento original é de não permitir a representação de fluxos secundários em *elementos-atividade*. A presença deste tipo de fluxo descaracteriza a atividade como um elemento próprio (Linger, 1975), o que é a pré-condição para se obter um esquema estruturado.

Assim, o fluxo secundário estaria associado a "falhas" na modularização (equivalente a cortar um procedimento simples no meio de uma iteração), o que obriga - para manter a funcionalidade do processo - a recorrer a desvios incondicionais como foi já apontado em (Silva, 1995).

3 PFS e o uso Racional de GOTOS

O comando GOTO é uma forma de implementar desvios na estrutura de controle de um programa. Basicamente, este comando pode ser aplicado em duas situações distintas:

Que geram um desvio incondicional, tratando-se de um risco para a manutenção da estruturação do programa.

Que geram desvios condicionais no contexto de implementar uma declaração iterativa do tipo padrão (WHILE/REPEAT).

De acordo com Ghezzi e Jazayeri (1987) a controvérsia a respeito do uso de GOTO em programação computacional já é motivo de discussões desde 1970. A proposição do Teorema de Bohm e Jacopini (Teorema da estruturação) estabelece o conjunto mínimo de estruturas de controle que podem ser incluídas em um programa estruturado. Portanto, o GOTO deve ser usado somente como uma técnica de implementação de estruturas de controle legítimas¹ caso a linguagem não as inclua como sinônimos em seus recursos.

A estruturação do processo de modelagem em PFS pode perfeitamente adotar esta abordagem de programação estruturada. A representação de desvios incondicionais no caso de modelar-se a funcionalidade de SPs só teria sentido de ser utilizada caso fosse necessário representar situações anormais de falhas, erros, etc. Em situações normais de realização dos processos, deve-se representar as estruturas baseando-se nas formas padrões de controle, herdadas da programação estruturada, de acordo com a Figura 2.

¹ Entende-se por estruturas legítimas as declarações do tipo sequencial, iterativa e condicional que fazem parte do contexto de aplicação do conceito de programação estruturada no desenvolvimento de algoritmos computacionais.

Um processo de um SP é predominantemente seqüencial envolvendo essencialmente estruturas de modelo baseadas no item (a) da Figura 2.

Em determinadas situações pode ser possível a coexistência de diferentes rotas para a realização do processo, implicando em uma estrutura do tipo condicional (item (b) da Figura 2).

Há casos ainda em que é necessário repetir o processamento de um determinado elemento, podendo utilizar-se as estruturas (c) ou (d) da Figura 2.

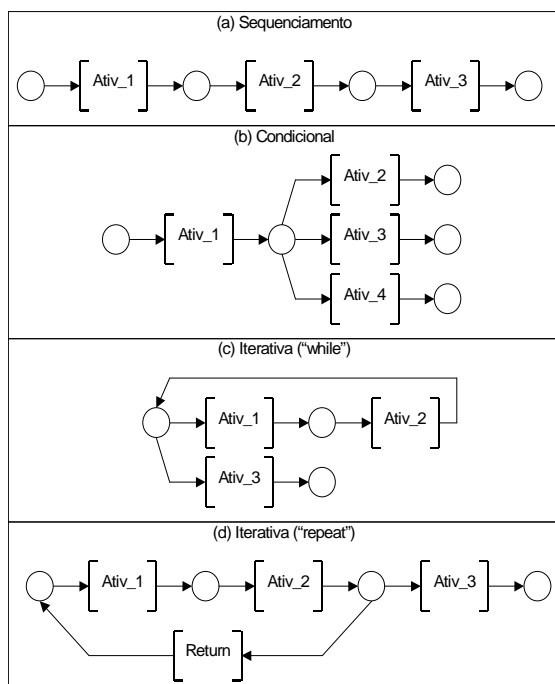


Figura 2: Estruturas padrões para modelagem em PFS estruturado.

Portanto, a essência do processo de modelagem estruturada em PFS está caracterizada. Contudo, para haver consistência nas observações realizadas é necessário considerar algumas hipóteses fundamentais:

Sendo o PFS um modelo funcional, deve-se associar a cada modelo um determinado processo a ser realizado em um SP. Múltiplos processos implicam em uma série de modelos PFS pertinentes a cada deles.

O objetivo do modelo PFS é estritamente representar a lógica do fluxo de operações (work flow) que está associada à realização de um determinado processo, não devendo questionar o mérito de alocação de recursos para a realização de cada uma das operações. Isto significa que o modelo PFS do processo é um modelo que representa como se deseja que as operações que constituem o processo sejam realizadas, representando, portanto, o que se denomina neste trabalho de nível de Controle de Processos.

As questões que envolvem o problema de quem deve processar cada uma das etapas deve ser delegado a outro nível de controle que não faz parte do escopo deste trabalho.

Este aspecto de não se alocar os recursos no modelo PFS é fundamental porque no caso em que o sistema é flexível, operando com vários processos diferentes e simultâneos compartilhando os mesmos recursos, tem sido exaustivamente discutido em Banaszak e Krogh

(1990), Viswanadham *et alii* (1990), Cho *et alii* (1995), Ezpeleta *et alii* (1995), Xing *et alii* (1995), Hasegawa (1996), Hasegawa *et alii* (1996A), Hasegawa *et alii* (1996c) e Xing *et alii* (1996).

Apesar de haver um controle adequado no nível de cada processo que está sendo executado em um sistema produtivo, quando há o compartilhamento dos recursos, provoca-se um 'fluxo dos recursos' entre os processos. Este aspecto pode ser interpretado como o surgimento de um fluxo adicional de controle importante que é mais complexo do que uma simples alocação de recursos conforme é sugerido no modelo PFS da Figura 3.

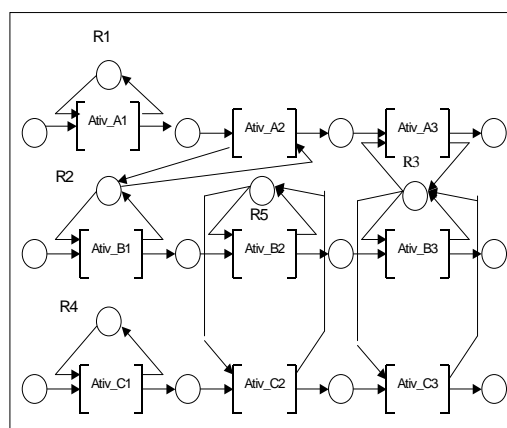


Figura 3: Representação em PFS de recursos compartilhados (R1, R2, R3, R4, R5).

O modelo PFS da Figura 3 exige que técnicas adicionais de controle sejam incorporadas ao modelo para estabelecer uma política de alocação de recursos que evite o travamento do sistema. Em muitos casos práticos, principalmente em sistemas discretos automatizados, usa-se separar os processos sequenciais (nível sequencial direto) da identificação da política de alocação de recursos e resolução de conflitos (nível de controle inteligente).

4 Metodologia Estruturada E-MFG/PFS

O conceito de modelagem estruturada aplicado ao controle de processos de SPs sugere que o problema de alocação de recursos para realizar cada uma das etapas que constituem um determinado processo seja realizado em uma etapa diferente da modelagem do próprio processo.

Outro fator digno de nota é que o nível sequencial e o nível dito inteligente (DiCesare, 1994) têm diferentes níveis de aderência aos modelos físicos. Portanto seria melhor a abordagem estruturada que nos garante a parte sequencial, antes de tratar com o nível de controle inteligente, incluindo aí a política de compartilhamento de recursos.

Todas as observações sobre estruturação evidenciadas para a situação em que se aplica o PFS para a modelagem funcional de um sistema, aplica-se também para o E-MFG. O modelo E-MFG introduz identificação nas marcas sem no entanto fugir do modelo de rede elementar convencional]. A identificação de marcas é um fator à mais na análise de fluxo de controle, o que

justifica a abordagem E-MFG, sem no entanto introduzir maior complexidade de análise com um formalismo mais elaborado.

A abordagem estruturada delineada no PFS pode ser ainda utilizada no E-MFG desde que se introduza um elemento passivo (um box) entre o início e o final de cada atividade do PFS, o que faz com o que o PFS deixe de ser um esquema estruturado de modelagem de MFG para passar a ser uma rede hierárquica (Silva, 1996). Em Miyagi (1996) já foi proposto um processo sistemático de tradução de modelos PFS em MFG.

Portanto, deve-se tomar o devido cuidado para não haver problemas de "vazamento" e "intrusão" de marcas dentro de uma atividade que foi detalhada conforme abordado em Miyagi (1996).

Em síntese, a metodologia PFS/MFG descrita em Miyagi (1996) precisa ser revisada em três aspectos:

Em relação ao contexto de modelagem estruturada basta obedecer aos critérios de construção do PFS já descritos anteriormente.

Em relação à extensão de MFG para E-MFG, basta considerar uma etapa adicional de definição do conjunto de atributos que devem ser utilizados para representar a individualidade de cada marca.

Não faz mais sentido executar a etapa de representação de recursos e seu controle no modelo.

Portanto, a metodologia PFS/E-MFG (ou E-MFG/PFS) engloba os seguintes procedimentos:

Representação de cada processo por um modelo PFS conceitual de alto nível de abstração.

Detalhamento de cada processo em atividades, definindo a lógica de seqüenciamento inerente a cada um deles.

Definição do conjunto de atributos que deve ser associado a cada marca do grafo E-MFG.

Detalhamento as atividades dos processos através de modelos E-MFG.

Mapeamento dos sinais de controle indicando os eventos gerados pelo controlador a partir de arcos de sinal de saída e os eventos ocorridos na planta a partir de portas externas.

5 Exemplo Aplicativo

Considere um SP em estudo composto pelos seguintes elementos:

Quatro estações de trabalho para processamento: EST_1, EST_2, EST_3 e EST_4.

Cada uma destas estações possui algum dispositivo manipulador dedicado que é utilizado para efetuar as operações de carga e descarga de peças a serem processadas, bem como uma máquina-ferramenta que processa uma peça por vez.

Uma estação de entrada de materiais (EST_IN) que armazena o estoque de matéria-prima (peças) que abastece as estações de trabalho.

Uma estação de saída (EST_OUT) que armazena os estoques de peças acabadas.

Para a realização do fluxo de materiais entre as estações, há um conjunto de veículos de transporte (VTs) movimentando-se sobre um circuito de via única, transportando um produto de cada vez. À medida que os

produtos são processados, os VTs são requisitados pela estação correspondente e a retirada dos produtos dos magazines de saída é realizada na ordem FIFO.

No presente sistema executa-se um processo identificado por *PROC_A* que envolve uma seqüência de operações e utiliza os recursos disponíveis de acordo com as relações (5.1) e (5.2) descritas a seguir:

$$E_A = \{s_A(0), s_A(1), s_A(2), s_A(3), s_A(4), s_A(5)\} \quad (5.1)$$

Onde:

- $s_q(i)$ representa a i -ésima etapa da seqüência de produção do elemento q .
- $s_q(0)$ representa a etapa inicial da seqüência de produção.

$$r_A^* = \{B_IN, EST1, EST2, EST3, EST4, B_OUT\} \quad (5.2)$$

Onde:

- $r_q(j)$ representa o recurso utilizado na etapa $s_q(i)$ da seqüência de produção do elemento q

Para construir o modelo estruturado do controle do processo, será aplicada a metodologia estruturada PFS/E-MFG:

A) Identificação dos Processos

Representa-se os processos a serem executados no sistema a partir de modelos PFS, de acordo com a Figura 4. Apesar deste procedimento inicial parecer desnecessário por ser evidente, é importante porque destaca a natureza orientada a produto desta metodologia de modelagem.



Figura 4: Modelo PFS dos Processos.

B) Detalhamento dos Processos em PFS

Refina-se o modelo PFS anterior para descrever a lógica de seqüenciamento das atividades. A Figura 5 ilustra esse modelo detalhado em PFS. Neste modelo é importante observar que há três tipos básicos de atividades que compõem cada processo:

A atividade de requisição de uma peça para ser processado, de acordo com a ordem de serviço prescrita ou o pedido prescrito (atividade "Requis...").

A atividade de despacho que libera a peça pronta ou atendida (atividade "Dispatch...").

A atividade de processamento de cada etapa do processo a ser executada em cada recurso específico (atividade "Proc...")

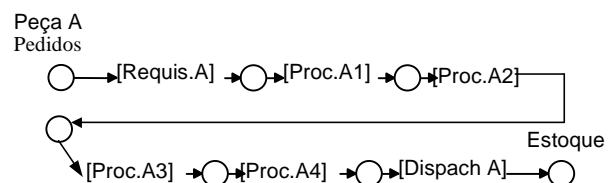


Figura 5: Modelo PFS dos Processos.

C) Definição do Conjunto de Atributos e Detalhamento das Atividades

Uma vez que já se elaborou um modelo do processo detalhando-se as diversas atividades concatenadas logicamente, nesta fase deve ser definido o conjunto de atributos a serem associados às marcas individuais do E-MFG. A Figura 6 mostra como os diversos valores são atribuídos ao conjunto de atributos especificado.

ATRIBUTOS				
V	(a)	(b)	(c)	(d)
	Peça	Pedido	Destino	VT
A	IF PROC_A THEN a=A	IF PED_1 THEN b=1	IF EST_1 THEN c=1	IF VT_1 THEN d=1
L		IF PED_2 THEN b=2	IF EST_2 THEN c=2	IF VT_2 THEN d=2
O		...	IF EST_3 THEN c=3	IF VT_3 THEN d=3
R		...	IF EST_4 THEN c=4	
E			IF EST_In THEN c=5	
S			IF EST_Out THEN c=6	
			IF ESTAC THEN c=7	

Figura 6: Definição dos atributos das marcas individuais.

O modelo detalhado da atividade de requisição pode ser observado na Figura 7.

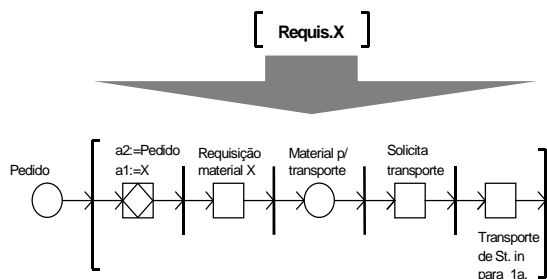


Figura 7: Modelo genérico da atividade "Requis.X".

O modelo detalhado genérico de uma atividade de processamento pode ser obtido a partir da lógica utilizada para modelar a primeira etapa de processamento do processo A que está exemplificado na Figura 8.

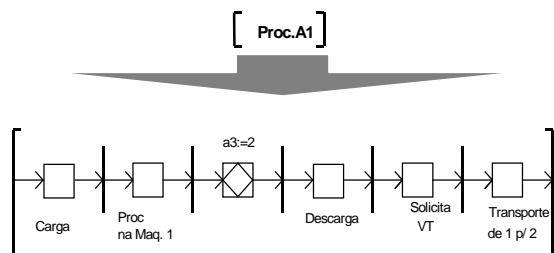


Figura 8: Exemplo de modelo da atividade de processamento.

Finalizando esta etapa há o modelo da atividade de despacho, conforme ilustrado na Figura 9.

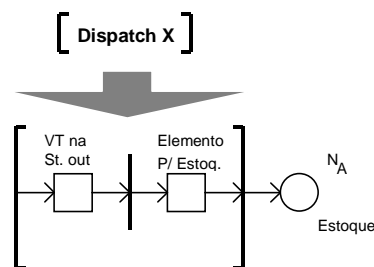


Figura 9: Modelo genérico da atividade "Dispatch.X".

D) Mapeamento dos Sinais Externos de Controle

Os eventos gerados pelo controlador devem ser enviados para o exterior através de arcos de sinal de saída. Por sua vez, os eventos provocados na planta devem ser enviados ao controlador através de portas habilitadoras/inibidoras externas. As Figuras 10, 11 e 12 ilustram este procedimento, devendo estar claro que para mapear-se todos os sinais de controle seria necessário enumerar todas as atividades modeladas em E-MFG.

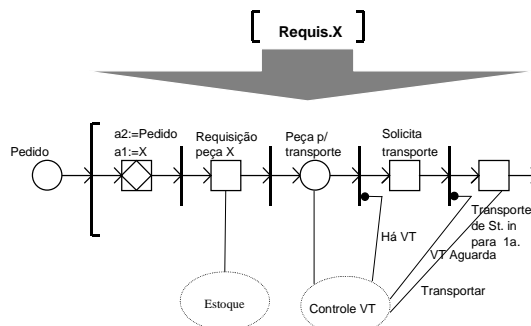


Figura 10: Mapeamento dos sinais externos no controle de "Requis.X".

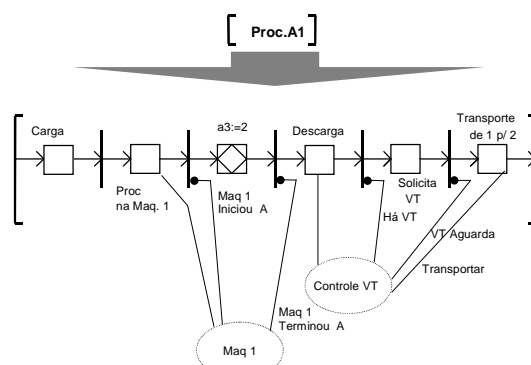


Figura 11: Mapeamento dos sinais externos no controle de "Proc.X".

Conclui-se, portanto, a modelagem do controle de processos do estudo de caso.

6 Observações Finais

Este artigo discute a aplicação do conceito de estruturação no processo de modelagem, mostrando como construir um modelo PFS de forma estruturada, questionando a alocação de recursos durante esta etapa

de modelagem. Em seguida, discute-se as vantagens obtidas com o procedimento de separar o problema de alocação de recursos do problema de garantir o perfeito seqüenciamento das operações que constituem um determinado processo. Finalizando, a metodologia E-MFG/PFS estruturada é apresentada como resultado de um aprimoramento da metodologia PFS/MFG original.

Como evolução deste trabalho, uma série de atividades tem sido realizadas no sentido de estabelecer uma metodologia que aborde também um conjunto de procedimentos para programar a utilização dos recursos disponíveis em situações que envolvem a execução de processos simultâneos e um intenso compartilhamento de recursos.

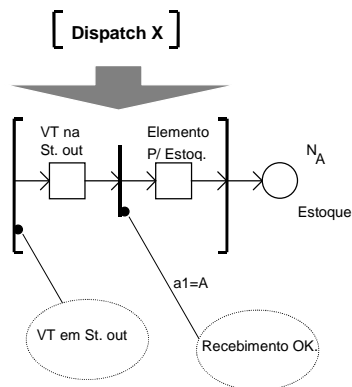


Figura 12: Mapeamento dos sinais externos no controle de "Dispatch.X".

Bibliografia

Banaszak, Z.A.; Krogh, B.H. Deadlock avoidance in Flexible Manufacturing Systems with concurrently competing process flows. IEEE Transaction on Robotics and Automation, v.6, n.6, p.724-34, 1990.

Cho, H. et al. Graph-Theoretic Deadlock Detection and Resolution for Flexible Manufacturing Systems. IEEE Transaction on Robotics and Automation, v.11, n.3, p.413-21, 1995.

DiCesare, T., Mu der Jeng, F.; Synthesis of Manufacturing Systems Integration, in Practice of Petri Nets in Manufacturing, T.

DiCesare, G. Harhalakis, J.M. Proth, M. Silva, F. Vernadat, Chapman & Hall, 1993.

Ezpeleta, J. et al. A Petri net based deadlock prevention policy for Flexible Manufacturing Systems. IEEE Transactions on Robotics and Automation v.11, n.2, p.173-84, 1995.

Ghezzi, C.; Jazayeri, M. Programming Language Concepts. John Wiley & Sons, 1987.

Gustin, G.B.; Miyagi, P.E.; Santos Filho, D.J.; Metodologia para a Modelagem de Sistemas de Elevadores em edifícios Inteligentes. In: CONEM 2000, Anais. Natal, RN, 2000.

Hasegawa, K. Modeling, control and deadlock avoidance of flexible manufacturing systems. In: 11° CBA - CONGRESSO BRASILEIRO de AUTOMÁTICA, Anais. SBA, São Paulo, SP, v.I, p.37-51, 09/1996.

Hasegawa, K. et al. Graphical Analysis and Synthesis of in Flexible Manufacturing Systems. In: 17th Internacional Conference on Application and

Theory of Petri Net, Proceedings. First Internacional Workshop, Osaka, Japan, p.161-176, 06/1996a.

Hasegawa, K. et al. Deadlock avoidance policy based on request/allocation graph. In: '96 Joint Symposium on System and Information, Proceedings. SICE, Matsuyama, Japan, p.219-24, 10/1996c (in Japanese). LINGER, R.C., Mills, H.D., Witt, B.I.; Structured Programming: Theory and Practice, Adison Wesley, 1979.

Marca, D.; SADT: Structured Analysis and Design Technique, McGraw Hill, 1988.

Miyagi, P.E. Control System Design, Programming and Implementation for Discrete Event Production Systems by using Mark Flow Graph, Japan, 1988. Tese de Doutorado - Tokyo Institute of Technology.

Miyagi, P.E. Controle Programável - Fundamentos do controle de sistemas a eventos discretos. S. Paulo, Ed. Edgard Blücher Ltda., 1996.

Miyagi, P.E. et al. Modelagem e Análise de Sistemas Complexos com Recursos Compartilhados Através do PFS (Production Flow Schema), In: 12° CBA - CONGRESSO BRASILEIRO de AUTOMÁTICA, Anais. SBA, Uberlândia, MG, 1998.

Miyagi, M.; Kisil, M.; Santos Fº, D.J.; Maruyama, N.; Miyagi, P.E. Modelagem de Serviço de Saúde Utilizando Rede de Petri. In 4o. SBAI - SIMPÓSIO BRASILEIRO de AUTOMAÇÃO INTELIGENTE, Anais., SBA, São Paulo, SP, 04/2000.

Santos Filho, D.J. Controle de Sistemas Antropocêntricos de Produção Baseado em Redes de Petri Interpretadas. S. Paulo, 1998. Tese (Doutorado) - Escola Politécnica, Universidade de S. Paulo.

Santos Filho, D.J.; Miyagi, P.E. Enhanced Mark Flow Graph to Control Flexible Manufacturing Systems. Jornal of The Brazilian Society of Mechanical Sciences, ABCM, Rio de Janeiro, RJ, v.17, n.2, p.232-248, 1995.

Silva, J.R., Miyagi, P.; PFS/MFG : A High Level Net for the modeling of Discrete Manufacturing Systems, in Balanced Automation Systems, cap. 33, L.Camarinha-Matos, H. Afsarmanesh (eds.), Chapman & Hall, 1995.

Silva, J.R., Miyagi, P.E., Sebastião, J.R., Rodrigues, M.; A High Level Integrated Petri Net Environment to the Design of FMS, IASTED, Int. Conference on Modeling, Simulation and Optimization, Golden Cost, Australia, May, 1996.

Villani, E.; Miyagi, P.E.; Santos Filho, D.J.; Maruyama, N.; Metodologia para Modelagem de Sistemas de Ar condicionado em Edifícios Inteligentes, Controle & Instrumentação, ano 5 no46, SUCESU-SP, São Paulo, SP, 2000.

Xing, K. et al. Deadlock avoidance policiy for Petri-net modeling of Flexible Manufacturing Systems with shared resources. IEEE Transactions on Robotics and Automation, v.41, n.2, p.289-95, 1996.