

**MODELAGEM DO TRATAMENTO DE FALHAS EM  
SISTEMAS DE MANUFATURA ATRAVÉS DE  
REDE DE PETRI**

Dissertação apresentada à Escola Politécnica da  
Universidade de São Paulo para obtenção do título  
de Mestre em Engenharia Mecânica.

São Paulo

1998

LUIS ALBERTO MARTÍNEZ RIASCOS

LUIS ALBERTO MARTINEZ RIASCOS

**MODELAGEM DO TRATAMENTO DE FALHAS EM  
SISTEMAS DE MANUFATURA ATRAVÉS DE  
REDE DE PETRI**

Dissertação apresentada à Escola Politécnica da  
Universidade de São Paulo para obtenção do título  
de Mestre em Engenharia Mecânica.

Área de Concentração:  
Engenharia Mecânica / Mecatrônica.

Orientador:

Prof. Dr. Paulo Eigi Miyagi.

São Paulo

1998

**MODELAGEM DO TRATAMENTO DE FALHAS EM  
SISTEMAS DE MANUFATURA ATRAVÉS DE  
REDE DE PETRI**

LUIS ALBERTO MARTINEZ RIASCOS

DEDALUS - Acervo - EPMN



31600010158

Autor: Luis Alberto Martinez Riascos.  
Orientador: Paulo Eigi Miyagi

Prof. Dr. Paulo Eigi Miyagi, Presidente  
Dep. Eng. Mecânica / EPUSP

Prof. Dr. Oswaldo Horikawa.  
Dep. Eng. Mecânica / EPUSP

Prof. Dr. Márcio Rillo  
Dep. Eng. Eletrônica / EPUSP

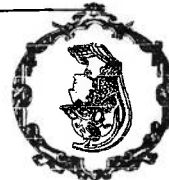
A minha mãe e aos meus irmãos, Carmen Helena e Carlos  
Arturo, que sempre acreditaram em mim.

## **AGRADECIMENTOS**

Inicialmente quero agradecer ao orientador da pesquisa, Prof. Dr. Paulo Figli Miyagi, pela sua constante supervisão e permanente incentivo para o desenvolvimento deste trabalho.

Ao CNPq (CONSELHO NACIONAL DE DESENVOLVIMENTO CIENTÍFICO E TECNOLÓGICO), que financiou com uma bolsa de estudos a realização desta pesquisa.

A todos os colegas da pós-graduação e todos os meus amigos que direta ou indiretamente colaboraram na execução deste trabalho.



PROGRAMA:

NÍVEL: MESTRADO: ( X )  
DOUTORADO: ( )

CANDIDATO: LUIS ALBERTO MARTINEZ RIASCOS

TÍTULO DO TRABALHO: "MODELAGEM DO TRATAMENTO DE FALHAS  
EM SISTEMAS DE MANUFATURA ATRAVÉS DE REDE DE PETRI"

ORIENTADOR: PROF.DR. PAULO EIGI MIYAGI

DATA DA DEFESA: 15.05.1998

## ERRATA

Página 45, linha 3

Onde diz:

... respectivas TRANSIÇÕES (t<sub>n</sub>, t<sub>i</sub>). A figura 3.10-c) ilustra o mesmo esquema utilizando ARCOS auto-modificáveis condicionados a marcação do LUGAR p.

Deve-se ler:

... respectivas TRANSIÇÕES (t<sub>n</sub>, t<sub>i</sub>). A figura 3.10-c) ilustra o mesmo esquema utilizando ARCOS auto-modificáveis condicionados a marcação do LUGAR p. Nesta rede deve-se assumir que TRANSIÇÕES sem ARCOS de entrada não dispararam, para evitar TRANSIÇÕES fonte.

Página 47, linha 3

Onde diz:

... (meditando, pegando o garfo da direita ou jantando) com 3-LUGARES e 3-TRANSIÇÕES cada.

Deve-se ler:

... (meditando, pegando o palito da direita ou jantando) com 3-LUGARES e 3-TRANSIÇÕES cada.

(caso necessário utilize folhas adicionais numeradas)

## SUMÁRIO

1	1. INTRODUÇÃO.....
1	1.1 MOTIVAÇÃO E JUSTIFICATIVA.....
3	1.2 OBJETIVOS.....
4	1.3 ORGANIZAÇÃO DO TEXTO.....
8	2. DETECÇÃO E TRATAMENTO DE FALHAS EM SISTEMAS DE MANUFATURA.8
8	2.1 SISTEMAS FLEXÍVEIS DE MANUFATURA (FMS).....
9	2.1.1 Sistemas Antropocêntricos de Produção (APS).....
12	2.1.2 Sistemas de Automação Balançada (BAS).....
14	2.1.3 Falhas e erros em Sistemas de Manufatura.....
14	2.2 DETECÇÃO DE FALHAS E ERROS EM SISTEMAS DE MANUFATURA.....
20	2.3 TRATAMENTO DE FALHAS E ERROS EM SISTEMAS DE MANUFATURA.....
25	2.4 PROCEDIMENTOS DE MANUTENÇÃO.....
26	2.5 NECESSIDADE DE MODELO.....
29	3 REDE DE PETRI (RP).....
31	3.1 VANTAGENS DA REDE DE PETRI.....
33	3.2 MODELAGEM DO TRATAMENTO DE FALHAS ATRAVÉS DE RP.....
36	3.2.1 Manutenção das propriedades da RP.....
37	3.3 REDE DE PETRI AUTO-MODIFICÁVEL (RPAM) PARA TRATAMENTO DE FALHAS/ ERROS EM SISTEMAS DE MANUFATURA.....
40	3.3.1 Exemplo de Aplicação.....
43	3.3.2 Equivalência entre as Condições Restritivas das RP.....
45	3.3.3 Sistema de Processos Espontâneos com Restrições.....
50	4 ESTUDO DE CASOS.....
50	4.1 CASO 1.....
55	4.1.1 Modos Alternativos de Operação das RPAM.....

4.1.2 Equivalência da RPAM com outros tipos de RP ..... 57

4.2 CASO 2 ..... 58

4.3 IMPLEMENTAÇÃO E TESTE NO SIMULADOR DE RPAM ..... 63

4.4 ANÁLISE DE RESULTADOS ..... 67

5. COMENTÁRIOS FINAIS E CONCLUSÕES ..... 70

ANEXO ..... 76

A.1 RP-PROCEDURAL PARA DETECÇÃO E DIAGNÓSTICO (BPN) ..... 76

A.2 EQUIVALÊNCIA DA RPAM COM RP-COLORIDA ..... 79

REFERÊNCIAS BIBLIOGRÁFICAS ..... 83

APÊNDICE I ..... 11

APÊNDICE II ..... 11

APÊNDICE III ..... 11



## LISTA DE FIGURAS

16	Fig. 2.1 - Estrutura básica de um sistema especialista.
17	Fig. 2.2 - DIF combinando as metodologias analítica e baseada no conhecimento.
19	Fig. 2.3 - Estrutura básica DIF baseada no modelo analítico do sistema.
19	Fig. 2.4 - Conceito básico de um sistema DIF baseado no observador.
21	Fig. 2.5 - Estrutura de um equipamento para auto-correção de falhas.
24	Fig. 2.6 - Algoritmo para execução de reparos num sistema autónomo.
34	Fig. 3.1 - Esquema de recuperação pelo método da entrada condicionada.
34	Fig. 3.2 - Esquema de recuperação pelo método da rota alternativa.
35	Fig. 3.3 - Esquema de recuperação de erros "Backward".
35	Fig. 3.4 - Esquema de recuperação de erros "Forward".
36	Fig. 3.5 - Rede de Petri associada (S*).
38	Fig. 3.6 - RP com ARCO auto-modificável.
41	Fig. 3.7 - Modelagem da solução dos processos de leitura/escrita.
42	Fig. 3.8 - Modelagem da solução dos proc. de leitura/escrita utilizando ARCOS inibidores.
43	Fig. 3.9 - Solução do problema dos processos de leitura/escrita utilizando RPAM.
44	Fig. 3.10 - Equivalência entre várias condições habilitadoras e inibidoras nas RP.
47	Fig. 3.11 - RPAM que modela o problema dos 5 filósofos chineses.
51	Fig. 4.1 - FMS constituído por 2 máquinas, 1 robô e 2 esteiras.
53	Fig. 4.2 - Modelo em RPAM dos processos de fabricação do sistema produtivo.
54	Fig. 4.3 - Modelo em RPAM do estado de operação das máquinas.
55	Fig. 4.4 - Modo alternativo de operação do modelo da fig. 4.2 (MI operando norm).
56	Fig. 4.5 - Modo alternativo de operação. a) MI com falha. b) MI requer manutenção.
58	Fig. 4.6 - Modelo em RP com arcos inibidores do sistema da figura 4.1.
59	Fig. 4.7 - Sistema de manufatura composto por um Robô e uma máquina CNC.
60	Fig. 4.8 - Modelo em RPAM do sistema robô/máquina.
63	Fig. 4.9 - Modelo em RP do estado dos equipamentos.
65	Fig. 4.10 - RPAM modelando os processos de leitura e escrita.

66	Fig. 4.11 - RPAM modelando os processos do sistema da figura 4.1
66	Fig. 4.12 - RPAM do sistema da figura 4.7
77	Fig. A.1.1 - Equivalência da TRANSIÇÃO Tor e T <sub>n</sub>
78	Fig. A.1.2 - Regra de disparo inversa em RP para diagnóstico
80	Fig. A.2.1 - Modelo de RP com ARCOS com inscrições fixas do sistema da figura 4.7
81	Fig. A.2.2 - Modelo em RP colorida do sistema da figura 4.7
I-5	Fig. I.1 - Exemplos de "trajetórias" de estados em DEDS e CVDS
II-1	Fig. II.1 - Classificação de Sistemas Automatizados de Manufatura
II-2	Fig. II.2 - Linha de montagem de veículos
II-3	Fig. II.3 - Célula flexível de manufatura
II-4	Fig. II.4 - Centro de usinagem com sistema de alimentação múltiplo de peças
II-4	Fig. II.5 - Máquina ferramenta CNC para torneamento
III-2	Fig. III.1 - a) ARCOS múltiplos. b) Representação compacta
III-2	Fig. III.2 - a) TRANSIÇÃO habilitada. b) Marcação após o disparo
III-5	Fig. III.3 - Sistema composto por uma máquina-ferramenta e um robô
III-5	Fig. III.4 - a) RP do sistema da fig. III.3 aprese. um bloqueio. b) RP sem bloqueio
III-6	Fig. III.5 - Arvore de alcançabilidade da rede da figura III.8-b)
III-9	Fig. III.6 - a) RP com ARCO inibidor. b) RP com ARCO habilitador
III-10	FIG. III.7 - Sistema de manufatura com 2 máquinas-ferramentas
III-11	Fig. III.8 - a) modelo em RP dos processos. b) modelo em RP baseado nos recursos
III-12	Fig. III.9 - Rede com ARCOS com inscrições fixas
III-13	Fig. III.10 - Rede com ARCOS de inscrições variáveis, quando $x=A$ e $y=B$
III-15	Fig. III.11 - Representação dos elementos de uma C-NET

## LISTA DE TABELAS

Tabela 2.1 - Comparação dos conceitos de produção centrado na tecnologia e centrado no homem.....	10
Tabela 3.1 - Algumas interpretações típicas de lugares e transições.....	32
Tabela 4.1 - Descrição dos lugares da figura 4.8.....	61
Tabela 4.2 - Esquema de recuperação utilizado.....	62
Tabela A.2.1 - Descrição dos lugares da figura A.2.1 e A.2.2.....	80
Tabela A.2.2 - Definição das inscrições (cores) da rede da figura A.2.2.....	81
Tabela I.1 - Classificação de modelos de DEDS.....	1-8
Tabela III.1 - Representação da construção lógica por RP e LLD.....	III-17

## LISTA DE ABREVIATURAS OU SIGLAS

AI	inteligência artificial.
APS	sistemas antropocêntricos de produção.
BAS	sistemas balanceados de produção.
BPN	rede de Petri procedural.
CAD	projeto auxiliado por computador.
CAM	manufatura auxiliada por computador.
CAP	planejamento auxiliado por computador.
CNC	comando numérico computadorizado.
CVDS	sistemas dinâmicos de variáveis contínuas.
DEDS	sistemas dinâmicos a eventos discretos.
DIF	deteção e isolamento de falhas.
FMS	sistemas flexíveis de manufatura.
GSPM	processos generalizados semi-Markovianos.
LLD	diagrama lógico de relés.
PFS/MFG	<i>“Production Flow Schema”/“Mark Flow Graph”</i> .
RCN	rede para controle de recursos.
RP	rede de Petri.
RPAM	rede de Petri auto-modificável.
SPSC	sistema de processos espontâneos com restrições.
TIG	gráfico de iteração de tarefas.

## LISTA DE SÍMBOLOS

$S^*$  é a rede formada pela sub-rede  $S$ , o lugar  $p$  e as transições  $t'$  e  $r'$ , incluindo os respectivos arcos de ligação (figura 3.5).

$Z'$  é formada pelas rede  $Z$  e  $S$ , incluindo as transições  $t'$  e  $r'$ .

$M_0^Z$ : Marcação inicial de  $Z$ ;

$M_0^S$ : Marcação inicial de  $S$ ;

$M_0^{S^*}$ : Marcação inicial de  $S^*$ .

$R(RP)$ : conjunto de alcançabilidade da rede  $RP$  a partir de  $M_0$ .

$R(RPAM)$ : conjunto de alcançabilidade da rede  $RPAM$  a partir de  $M_0$ .

$f$ : falhas (figura 2.3).

$d$ : distúrbios

ruido nas medições.

variações dos parâmetros.

incertezas no modelo.

$u$ : entradas.

$y$ : saídas.

$P = \{p_1, p_2, \dots, p_m\}$ : conjunto de LUGARES da rede, onde  $m \in \mathbb{N}$  é o número total de

LUGARES. (Seção 3.3).

$T = \{t_1, \dots, t_n\}$ : conjunto de TRANSIÇÕES, onde  $n \in \mathbb{N}$  é o número total de TRANSIÇÕES.

$M_0$ : marcação inicial.

$I(p_j, t_i) = f(p_j)$ , onde  $f$  é uma função que especifica a relação entre  $p_j$  e  $t_i$  e depende da

marcação do LUGAR  $p_j$  ( $f(p_j) \in \mathbb{N}$ ).

$O(t_i, p_j) = g(p_j)$ , onde  $g$  é uma função que especifica a relação entre  $t_i$  e  $p_j$  e depende da

marcação do LUGAR  $p_j$  ( $g(p_j) \in \mathbb{N}$ ).

$k_{ij}$ : peso do arco (valor constante) que liga o elemento  $j$  com o elemento  $i$ .

$M_k$ : vetor coluna ( $m \times 1$ ) que representa a marcação imediatamente alcançável a partir de

$M_{k-1}$  após o disparo da TRANSIÇÃO  $t_i$  (identificada em  $u_k$ ), (equação 3.1).

$u_k$ : vetor ( $n \times 1$ ) com grandeza de 1 na posição  $i$  que representa o disparo da TRANSIÇÃO  $t_i$  e

zero (0) nas outras posições.

$A$ : matriz de incidência ( $n \times m$ ) definida por  $A = O(T, P) - I(P, T)$ .

## Resumo

O crescimento na complexidade de sistemas automatizados de produção demanda a utilização de novas tecnologias para sua modelagem e análise. Exemplos destes sistemas são sistemas flexíveis de manufatura (FMS), sistemas antropocêntricos de produção (APS) e sistemas de automação balanceada (BAS), que são classificados como sistemas dinâmicos a eventos discretos (DEDS), pois seus estados evoluem em função de eventos discretos e específicos. Neste contexto, a Rede de Petri (RP) se destaca como uma ferramenta para modelagem e análise de DEDS que apresentam processos concorrentes, assíncronos, etc. Entretanto, a modelagem dos processos considerando falhas/erros inerentes de sistemas flexíveis, antropocêntricos e com automação balanceada tem sido considerada relativamente complexa para ser representada pelas RP tradicionalmente estudada pelos pesquisadores. Assim, este trabalho introduz uma variação da RP, a Rede de Petri Auto-Modificável (RPAM), junto com novos elementos estruturais, como uma ferramenta para modelagem, análise e controle de sistemas automatizados de produção, que considera o tratamento de falhas e recuperação de erros.

## Palavras chaves

Sistemas de manufatura, sistemas dinâmicos a eventos discretos, redes de Petri, redes auto-modificáveis, tratamento de falhas e erros.

## Abstract

The growth in the complexity of automated production systems demand the utilization of novel technologies for their modeling and analyzing. Examples of such systems are flexible manufacturing systems (FMS), anthropocentric production systems (APS) and balanced automation systems (BAS). They are classified as discrete event dynamic systems (DEDS), where their states changes as a function of specific and discrete events. In this context, the Petri Net (PN) detaches as a tool for modeling and analyzing DEDS which presents concurrent, asynchronous process, etc. However, the modeling of the process considering failures/errors inherent of flexible, anthropocentric and with balanced automation systems have been considered relatively complex to be represented for PN, traditionally studied for researches. Therefore, this work introduces a variation of PN, called Self-Modifying Petri Net (SMPN), together with new structure elements, as a tool for modeling, analyzing and control of automated production systems, which considers the failure treatment and mistakes recovery.

## Keywords

Manufacturing systems, discrete event dynamic systems, Petri net, self-modifying net, failure and error treatment.

**INTRODUÇÃO**

**Capítulo 1**



**Capítulo 3**

**REDE DE PETRI (RP)**

entre outros.

“hardware” de sistemas computacionais [Chiola, 93], sistemas de transporte [Berge, 94], comunicação [Catania, 93], sistemas de distribuição de energia elétrica [Bose, 94], manufatura [Zurawski, 94], sistemas antropocêntricos de produção [Wobbe, 96], redes de assincronamente. Como exemplo de tais sistemas pode-se citar sistemas automatizados de eventos discretos, que muitas vezes consistem de múltiplos subsistemas operando simbólicos [Levis, 87]. Um exemplo são os sistemas cuja dinâmica é definida pela ação de abordagens analíticas e numéricas, considerem também a utilização de dados qualitativos e sistema. Além disso, devem ser consideradas outras classes de modelos que, junto com as modelados de forma incompleta ou considerando incertezas no modelo e no projeto do horizontes e ampliar seus conceitos e metodologias para serem aplicáveis em sistemas exato. Assim, as teorias de controle e automação estão evoluindo para expandir seus conhecido na sua totalidade) e certamente seu modelo físico-matemático dificilmente será. Devemos considerar que o modelo de um sistema real poucas vezes é completo (isto é, elemento ou sistema, geralmente utilizado para sua análise, caracterização, avaliação, etc. Um modelo é uma forma de representar algumas características relevantes de um

O controle automático, a informática, a mecatrônica e outras tecnologias têm representado um papel primordial no avanço da engenharia e dos sistemas produtivos, mas o crescente aumento na complexidade de sistemas integrados por computador demanda a utilização de novas técnicas formais de modelagem e de análise, fundamentais para avaliar alternativas de projeto e políticas de operação de um sistema [Levis, 87].

## 1.1 MOTIVAÇÃO E JUSTIFICATIVA

## INTRODUÇÃO

## Capítulo 1

Os Sistemas Flexíveis de Manufatura (FMS) são sistemas de produção nos quais requisitos como aumento na produtividade e qualidade são alcançados através da automação de tarefas tradicionalmente realizadas por operadores humanos e dispositivos que asseguram a flexibilidade operacional das partes do sistema. Este incremento na automação e flexibilidade deve respeitar regras básicas para atingir um equilíbrio na integração de atividades desempenhadas por máquinas e por operadores humanos. Assim, devem ser considerados, também, conceitos como os Sistemas Antropocêntricos de Produção (APS) [Wobbe, 96], definidos como um sistema de produção centrado nas atividades, necessidades e potencialidades do homem; e os Sistemas de Produção Balançada (BAS) [Camarinha, 95], que procuram manter um nível adequado de produção, automação e flexibilidade para atingir um equilíbrio entre fatores como: recursos humanos, recursos tecnológicos e ambiente sócio-econômico. Esta evolução de sistemas produtivos reforçam o fato de que falhas/erros (humanos e técnicos, por exemplo) são inerentes ao sistema.

Neste contexto, a competitividade tem forçado muitas empresas a adaptar suas plantas de produção segundo as condições de mercado. Este problema motiva a concepção e incremento da flexibilidade nos sistemas de manufatura. No entanto, esta flexibilidade é evidentemente restrita para cada sistema (somente são produzidos elementos similares ou da mesma família). Para produtos não similares é pouco provável que a planta possa ser adaptada rapidamente sem realizar mudanças nos equipamentos e no arranjo físico do sistema. Outro fator que deve ser considerado para assegurar uma alta flexibilidade é o procedimento para tratamentos de situações anormais como falhas, erros de operação e tarefas de manutenção, situações que, de fato, acontecem na operação do sistema e que devem ser previstas desde as etapas de modelagem e projeto [Frank, 92].

O tratamento de falhas, erros de operação e procedimentos de manutenção são fundamentais para que um sistema de manufatura possua maior autonomia e maior flexibilidade para a execução dos seus processos. Geralmente o tratamento destes tipos de situação tem sido desempenhado totalmente por operadores humanos e ainda são necessários muitos desenvolvimentos relacionados com a recuperação automática de falhas/erros [Zhou, 93]. A maior parte dos sistemas mecatrônicos atualmente em operação que consideram estas situações limitam-se a realizar um diagnóstico aproximado para sugerir uma lista de possíveis causas da falha/erro que é utilizada por um operador para correção da falha/erro.

Por outro lado, um FMS contém fluxos de processos concorrentes referentes aos diferentes produtos (inclusive os relativos às falhas/erros), interagindo de maneira assíncrona e muitas vezes necessitando do compartilhamento de recursos para reduzir os custos de produção [Hasegawa, 96]. Além disso, um controle inadequado da ocorrência dos eventos pode produzir bloqueios ou sobrecargas no sistema. Assim, métodos que permitam a solução do problema de alocação de recursos e de bloqueios são fundamentais. Este compartilhamento de recursos, junto com a variedade de rotas dos processos no sistema, dificulta sua descrição e análise através de ferramentas convencionais baseadas em Pesquisa Operacional [Miyagi, 96]. A Rede de Petri (RP) foi assim concebida principalmente para modelar e analisar sistemas com estas características e comprovou ser uma ferramenta efetiva para a modelagem e análise de FMS através de uma perspectiva de Sistemas Dinâmicos a Eventos Discretos (DEDS) [Zurawski, 94].

## 1.2 OBJETIVOS

O objetivo da pesquisa é investigar aplicações de rede de Petri (RP) (ou suas extensões em sistemas de manufatura: flexíveis, antropocêntricos e com automação

balançada) que considerem a detecção e tratamento de situações anormais (e/ou de exceção como falhas e erros de operação) e procedimentos de manutenção.

O presente trabalho pretende, assim, estabelecer uma metodologia de tratamento de falhas e erros que permita que um sistema de manufatura projetado originalmente sem estas características possa tratá-los e assegurá-los com isto, sua autonomia. Esta metodologia será baseada em:

1. Introdução de macro-elementos (ou sub-redes) que modelam especificamente as situações de exceção como falhas/erros.

2. Introdução de novos elementos estruturais no modelo como arcos inibidores e arcos auto-modificáveis que permitam a um sistema ajustar-se às novas especificações de operação que incluem o tratamento de situações anormais.

### 1.3 ORGANIZAÇÃO DO TEXTO

O segundo capítulo inicia-se com uma descrição das principais características dos Sistemas Flexíveis de Manufatura (FMS). Também aborda os Sistemas Antropocêntricos de Produção (APS) e os Sistemas Balanceados de Produção (BAS) como novas tendências que consideram a integração e a complementaridade dos aspectos tecnológicos, humanos e sócio-econômicos no projeto de um sistema de manufatura. Na segunda parte do capítulo são abordados os fundamentos das estratégias de detecção e isolamento de falhas e as considerações necessárias para o tratamento destas situações em sistemas de manufatura ou componentes destes sistemas.

No terceiro capítulo são abordadas características de RP que possibilitam a manutenção das propriedades procedurais e da dinâmica do modelo quando são introduzidas sub-redes para o tratamento de situações anormais em sistemas inicialmente

projetados sem estas considerações. Também são introduzidas redes de Petri auto-modificáveis (RPAM) como uma extensão de RP que permita explicitar claramente as condições de funcionamento de sistemas nos quais foram adicionadas novas especificações de operação, tais como tratamento de situações de falhas e erros. Também são apresentados exemplos de utilização de RPAM em sistemas já modelados através de abordagens tradicionais de RP, ilustrando assim a equivalência entre as condições restritivas da RP e da RPAM.

No capítulo 4 são apresentados e analisados estudos de casos de sistemas de manufatura enfocados no tratamento de situações anormais (como falhas/erros) e procedimentos de manutenção, permitindo assim ilustrar as vantagens da RPAM e a complementaridade que oferece em relação a modelos tradicionais de RP na modelagem e análise de sistemas com estas características. Também apresenta os modelos utilizados no simulador de RP desenvolvido no Departamento de Engenharia Mecânica/Mecatrônica da EPUSP, esta simulação foi utilizada com propósitos de validação e verificação dos modelos desenvolvidos para o presente estudo.

No capítulo 5, são apresentados os comentários finais e conclusões obtidas com o desenvolvimento do trabalho, principais resultados obtidos e sugestões de tópicos a serem pesquisados no futuro.

Nos anexos são apresentados conceitos relacionados com a utilização de RP no tratamento de situações anormais. Inicialmente é introduzido o conceito de Rede de Petri Procedural (BPN), que ilustra a possibilidade de realizar o diagnóstico de um sistema utilizando o formalismo da RP. Também é apresentado um modelo de um sistema de manufatura através de RP colorida, que introduz uma equivalência entre a RPAM e RP colorida na modelagem e tratamento de situações anormais, como falhas e erros.

Finalmente, nos apêndices são apresentados alguns dos conceitos básicos de temas abordados neste trabalho e que também foram considerados no desenvolvimento da metodologia proposta, mas que são desnecessários para leitores familiarizados com estes temas: Sistemas Dinâmicos e Eventos Discretos (DEDS), sistemas de manufatura e Rede de Petri (RP).

**SISTEMAS DE MANUFATURA.**

**Capítulo 2**



## Capítulo 2 DETECÇÃO E TRATAMENTO DE FALHAS EM SISTEMAS DE MANUFATURA

As abordagens para tratamento de falhas e erros de operação serão focalizadas em sistemas de manufatura automatizados com características de autonomia e flexibilidade e que considerem uma adequada integração e complementaridade dos aspectos tecnológicos, humanos e sócio-económicos.

### 2.1 SISTEMAS FLEXÍVEIS DE MANUFATURA (FMS)

Um Sistema Flexível de Manufatura (FMS) [Rembold, 94] é formado de máquinas-ferramentas com CNC, interconectadas para transferência de informações (por um sistema de rede de comunicações) e manipulação de materiais (por um sistema de transporte como esteiras, veículos transportadores, etc. e um sistema automático de troca de peças/ferramentas). Um FMS se distingue de outras formas de manufatura automatizada por considerar a diversidade de produtos que se deseja produzir (flexibilidade do produto), as características adaptativas das máquinas (flexibilidade dos equipamentos) e as propriedades de similaridade de processos (flexibilidade do processo) com implicações nas relações de custo/benefício e produtividade/qualidade do sistema. Tal diversidade pode exigir do processo a versatilidade para lidar com diferentes combinações de máquinas, ferramentas e seqüências de processos.

Os FMS são sistemas distribuídos formados por vários subsistemas e, geralmente, são descritos através de uma estrutura de controle hierárquico. A estrutura de um controle hierárquico depende da aplicação, filosofia de gerenciamento, disponibilidade de "software" e equipamentos computadorizados, grau de automação, etc. Um exemplo de hierarquia de

níveis de controle baseada nos trabalhos de [Chintamani, 88; Hasegawa, 96] é

apresentada a seguir:

- Nível de Fábrica.
- Nível de Estação de Trabalho.
- Nível de Equipamentos.
- Nível de Operação.

Cada um dos níveis no sistema de manufatura possui diferentes exigências de informação e controle. Assim, as operações nos níveis inferiores são as mais críticas em relação às sincronizações temporais, pois aqui materiais e informações são processados simultaneamente e as restrições de controle em tempo real são imprescindíveis. De outro lado, nos níveis superiores, as exigências de tempo real são menos críticas, as instruções são de alto nível e o controle é geralmente de tipo qualitativo envolvendo uma grande quantidade e variedade de informações. Assim, na etapa de projeto, as características que se desejam modelar de cada sistema dependem do nível focado.

### 2.1.1 Sistemas Antropocêntricos de Produção (APS)

Uma tentativa de incrementar a flexibilidade na manufatura é através da automação e integração por computador das linhas de produção utilizando ferramentas como CAD, CAM, CAP e máquinas CNC. Entretanto, mesmo nestes sistemas, deve-se considerar as questões de “*start-up*”, manutenção e ocorrência de distúrbios (envolvendo máquinas e operadores) que acontecem frequentemente numa fábrica, as quais não podem ser tratadas automaticamente na sua totalidade. Estas tarefas envolvem, ainda, a supervisão de um operador humano. Baseado neste fato, existem muitos estudos que consideram o planejamento em conjunto de recursos humanos, tecnológicos e organizacionais de sistemas de manufatura.

Neste contexto, um sistema antropocêntrico de produção (APS) é um conceito introduzido por alguns pesquisadores como um sistema de produção centrado no homem, em contraposição a sistemas centrados na tecnologia (veja uma comparação ilustrativa na tabela 2.1). Os componentes essenciais deste sistema são:

- Automação flexível
- Organização descentralizada, com delegação de funções e responsabilidades especialmente no chão de fábrica.
- Divisão reduzida de trabalho.
- Contínua orientação de atividades humanas para o produto.

Tabela 2.1 Comparação dos conceitos de produção centrado na tecnologia e centrado no homem.

Centrado na tecnologia	Centrado no homem
Introdução de novas tecnologias para reduzir o número de operadores no chão de fábrica.	Introdução de novas tecnologias para complementar capacidades humanas específicas; incremento funcional da flexibilidade, qualidade e "working life".
Substituir as habilidades humanas por tecnologia.	Melhorar a qualidade e estabilidade dos recursos humanos para melhorar a exploração do potencial oferecido por novas tecnologias.
Soluções técnicas centralizadas.	Soluções técnicas descentralizadas.
Hierarquia rígida (pouca liberdade) na tomada de decisões.	Participação na tomada de decisões.
Função operacional passiva: simples execução de tarefas.	Autonomia para desempenhar tarefas diferentes e complexas, resolver problemas individualmente ou em grupo.
Integração de unidades de fabricação, decisão e controle da fábrica por meio de centralização auxiliada por computador.	Integração de unidades para treinamento, socialização, cooperação e acesso na informação, participação em tomada de decisões e autocontrole.

A combinação de tecnologia de ponta e trabalho experiente dos operadores numa organização descentralizada e orientada ao produto possibilita um sistema de manufatura "inteligente", capaz de adequar-se rapidamente à alta tecnologia, mudanças do mercado, diversificação e eficiência na utilização de recursos. Esta abordagem visa desenvolver a habilidade dos sistemas para adaptar-se aos produtos segundo as demandas de mercado, para incrementar sua variabilidade e qualidade e reduzir tempos de entrega são fatores de muita importância para a competitividade do sistema produtivo.

A estratégia centrada na tecnologia procura incrementar a produtividade de sistemas de produção aumentando a automatização dos processos para realizar funções tradicionalmente desempenhadas por operadores humanos e centralizando a coordenação destas operações. Do ponto de vista tecnológico, o projeto de sistemas homem-máquina é muitas vezes limitado ao projeto de interfaces (por exemplo: interfaces amigáveis para o usuário), ao invés de sistemas que permitam uma complementaridade entre grupos de trabalho e máquinas [Roshardt, 95].

Por outro lado, a flexibilidade é também alcançada utilizando intuição, experiências individuais e coletivas, habilidade para tratamento de situações inesperadas, etc., as quais são baseadas em informações que não podem ser formalizadas nem estruturadas facilmente. Os propósitos de um sistema centrado no homem são:

- Utilização das habilidades humanas incluindo conhecimento tácito.
- Criação de condições favoráveis para contínuo desenvolvimento e aprendizado.
- Maior coletivização do trabalho.
- Programas de educação e treinamento permanentes.

O conceito de APS pode ser comparado com o conceito "lean production". Neste último a ênfase está na relação entre a empresa e os fornecedores/clientes. Estes conceitos podem ser comparados pela similaridade dos seguintes pontos:

- Objetivos: - Incremento da produtividade.
- Modernização industrial.
- Baseado em organização e recursos humanos (centrado no homem).
- Estrutura hierárquica e
- Organização: - Trabalho em grupo.
- Integração dos grupos de trabalho.

- Delegação de responsabilidade a nível de execução.
- Colaboração entre diferentes departamentos.

Entretanto em relação à aplicação, o conceito "*lean production*" tem ficado restrito apenas ao setor automobilístico e o relacionamento profissional é caracterizado por uma relação de liderança, enquanto no APS a relação é de participação [Kovács, 95].

Para avaliar os benefícios de um APS, devem ser considerados fatores como: tipo de produção, tamanho da empresa, tipo de gerenciamento e forma de avaliar os recursos humanos em relação a cultura, habilidades e qualificação. Em particular o modelo APS tende a ser mais apropriado quando a variedade na produção é alta e a quantidade é baixa [Kovács, 95].

## 2.1.2 Sistemas de Automação Balançada (BAS).

Neste contexto, a automação balancçada surge como um conceito mais amplo que engloba a ideia de um nível apropriado de automação tecnológica em oposição às abordagens isoladas de sistemas centrados exclusivamente no ser humano ou exclusivamente na automação total das máquinas [Camarinha, 95]. As indústrias de manufatura estão enfrentando consideráveis mudanças como consequência da globalização e abertura dos mercados. Para beneficiar-se dos avanços tecnológicos, acordos entre indústrias e redes de fornecedores de componentes e entre indústrias e universidades/instituições de pesquisa têm-se consolidado. Entretanto, num cenário típico, pequenas e médias empresas podem ter dificuldades para acessar os benefícios do Estado ou da tecnologia de ponta devido às suas limitações de recursos humanos, materiais e financeiros. Além disso, o mercado tende a impor ciclos de vida dos produtos mais curtos e incremento

da demanda de produtos por encomenda, além de uma maior variedade de produtos que exigem uma maior flexibilidade/agilidade dos processos de manufatura.

O conceito de automação balanceada explora a idéia de um nível apropriado de automação ao abordar pontos como flexibilidade, melhoramento da qualidade do produto e produtividade considerando, também, aspectos sócio-econômicos [Camarinha, 96]. Um BAS é caracterizado por uma correta combinação de atividades automatizadas e atividades manuais, procurando obter o maior proveito possível em relação à produtividade e sem perdas de qualidade no produto.

Devido à introdução de computadores e máquinas "inteligentes" nos sistemas de manufatura, o papel do operador tem mudado continuamente. Muitas das tarefas têm sido automatizadas, deslocando os operadores para o pré e pós-processamento das máquinas. Uma razão para isto é a relativa redução dos custos de computadores e das máquinas automatizadas comparadas com os custos de operadores qualificados ou experientes. No entanto, o conhecimento e a experiência das habilidades dos operadores são muito difíceis e custosas de estruturar e/ou reproduzir. Assim, ao projetar um BAS, devem ser considerados aspectos como custo de produção, nível requerido de flexibilidade ou qualidade desejada do produto, máximo reaproveitamento dos equipamentos já instalados, etc.

Em [Adlerno, 96], descrevem-se 2 exemplos para se obter um nível adequado de automação em células de manufatura. Primeiro, são definidos os níveis de controle e automação apropriados para obter um sistema balanceado considerando fatores como produtividade, custo, segurança, flexibilidade e qualidade numa célula de usinagem para produção de eixos de caminhões e ônibus. No segundo exemplo, descreve-se uma célula que produz conjuntos de pequenas peças plásticas como parafusos, porcas e pinos. Esta célula de montagem é considerada não tradicional devido ao número de grupos de

montagens paralelos com ciclos de tempo muito longos e ao sistema de alimentação de material, o qual é de produção em massa. Estes 2 exemplos são apresentados para identificar quais atividades podem ser melhor desempenhadas por operadores humanos e quais por sistemas automáticos, a fim de desenvolver um BAS.

### 2.1.3 Falhas e erros em Sistemas de Manufatura.

Falhas e erros (envolvendo máquinas e operadores) são fatos que acontecem na etapa de operação de um sistema de manufatura indiferente do nível de automação implementado. No entanto, não existem muitos trabalhos sobre uma metodologia que permita o tratamento destas situações anormais (além de procedimentos de manutenção) de modo que seja assegurada uma relativa autonomia e flexibilidade do sistema quando da ocorrência destas situações.

Com a evolução dos sistemas de manufatura descrita nas seções anteriores esperamos ter evidenciado que a consideração de fatores humanos e sócio-técnicos comprovam que a ocorrência de falhas e erros são inerente ao sistema produtivo e que a tendência de considerar estas características no projeto de sistemas produtivos tem aumentado consideravelmente. Apesar dos objetivos deste estudo não envolverem a especificação de uma metodologia para detecção de falhas, apresenta-se a seguir os principais fundamentos da detecção e isolamento de falhas e erros.

## 2.2 DETECÇÃO DE FALHAS E ERROS EM SISTEMAS DE MANUFATURA.

A detecção e o tratamento de falhas, erros de operação e procedimentos de manutenção representam um elemento fundamental para o projeto de sistemas de manufatura. Pode-se considerar 2 níveis de falhas: no nível de elementos de equipamentos

podem ser consideradas falhas nos sensores ou atuadores e, no nível de decisão podem ser considerados outros tipo de falhas tais como falhas no “*software*” ou no “*hardware*” computacional [Levis, 87]. As falhas do primeiro nível podem ser detectadas diretamente pela monitorização dos dispositivos, enquanto as do segundo nível envolvem uma estratégia melhor elaborada. O princípio fundamental de uma estratégia de detecção e isolamento de falhas (DIF), pode ser dividida em duas categorias principais: 1.) a *redundância analítica*, que utiliza modelos matemáticos analíticos, e 2.) *abordagem baseada no conhecimento*, que utiliza modelos qualitativos associados com o raciocínio heurístico.

Na **redundância analítica** [Frank, 92] deve-se ter um conhecimento suficientemente detalhado dos processos dinâmicos da planta física (objeto de controle), com informações precisas das grandezas envolvidas e do grau de confiança de cada variável relevante do processo. Assim, pode-se detectar, localizar e identificar falhas no sistema (planta) de modo praticamente imediato, comparando dados quantitativos coletados com dados do modelo matemático funcional da planta (ou previsões destes dados). No entanto, estes pré-requisitos geralmente não são totalmente satisfeitos num processo real. De fato, as variáveis medidas são insuficientes ou o grau de confiança destas é desconhecido. Entretanto, até um certo ponto, podem ser toleradas incertezas e medidas incompletas e, dependendo do grau destas incertezas, estratégias adaptativas ou robustas podem ser utilizadas.

Uma abordagem **baseada no conhecimento** é principalmente considerada no caso onde os dados sensoriais são insuficientes e somente modelos imprecisos são disponíveis. Técnicas de Inteligência Artificial (AI) [Charniak, 87] permitem explorar o conhecimento e a experiência disponível (ou acumulada) do sistema. Em muitos destes casos onde as incertezas são consideráveis, sistemas especialistas, redes neurais e lógica fuzzy [Bose, 94], estão sendo estudadas para a solução do problema de detecção e isolamento de falhas. Ao



mesmo tempo, estas técnicas, eventualmente combinadas com redes de Petri, podem ser utilizadas para implementar o supervisor global do sistema [Frank, 92].

A busca heurística é a principal consideração dos sistemas de AI como no sistema especialista. Trata-se de uma estratégia para tentar identificar mais rapidamente a causa da falha/erro e seu tratamento. Basicamente, um sistema especialista é definido como um sistema capaz de raciocinar e deduzir fatos e é formado por um banco de rotinas computacionais organizadas adequadamente, que formam a chamada *máquina de inferência* e que tenta emular a perícia humana<sup>[1]</sup> num certo domínio [Bose, 94] a partir de informações armazenadas numa base de conhecimento, utilizando técnicas baseadas na lógica matemática. A estrutura deste sistema é ilustrada na figura 2.1.

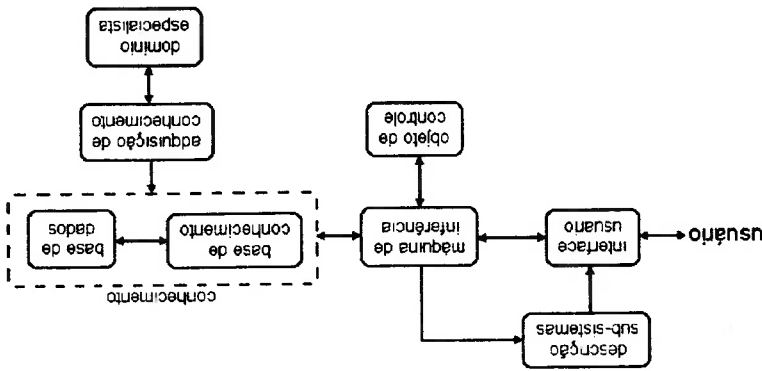


Fig. 2.1 - Estrutura básica de um sistema especialista.

A parte principal do sistema especialista é a representação do conhecimento adquirida por um especialista humano. Na *base de conhecimento* está armazenada a informação que permite resolver o problema, sob uma forma fracionada (parcial). A representação do conhecimento combina estruturas de dados, onde é armazenada a informação simbólica, e procedimentos que permitem, a partir da informação fornecida, manipular de forma

[1] Considere um engenheiro ou técnico que aprendeu ou adquiriu o conhecimento por educação e experiência num prolongado período de tempo em diagnóstico de falhas de sistemas de manufatura.

“inteligente” estas estruturas para realizar inferências até obter a solução. O conteúdo da base de conhecimento pode ser alterado, atualizado ou ampliado com relativa facilidade (como no caso de sistemas com aprendizado). As três formas mais importantes de representar o conhecimento nesta área são baseadas em:

- Regras do tipo IF-THEN.
- Redes semânticas: declarações formais bem definidas.
- Estruturas (“frames”): base de conhecimento estruturada na forma de árvore.

A máquina de inferência testa as regras em sequência ou executa uma busca em árvore para realizar inferências ou desenvolver conclusões.

Em muitas situações práticas, uma combinação das metodologias *analítica e baseada no conhecimento* pode fornecer uma solução apropriada para o problema de DIF (veja figura 2.2). Evidentemente, incluir um modelo analítico contribui para superar as dificuldades para completar a *base de conhecimento* e simplifica o raciocínio da *máquina de inferência* [Frank, 92].

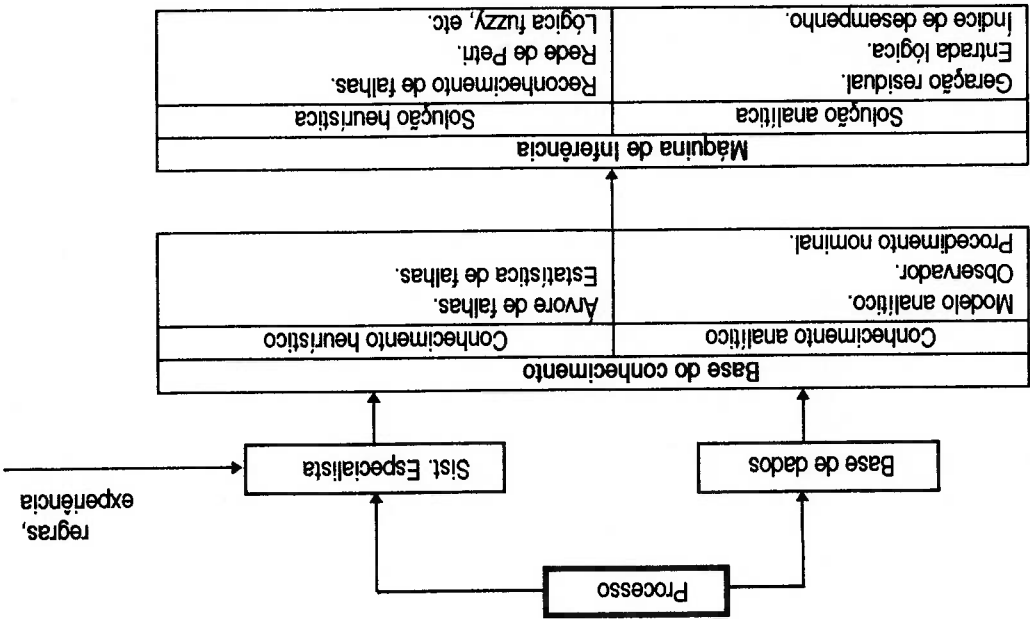


Fig. 2.2 - DIF combinando as metodologias analítica e baseada no conhecimento.

Do ponto de vista teórico, a redundância analítica tem alcançado um grau de amadurecimento maior, especialmente em sistemas lineares com pequenas incertezas no modelo. Esta abordagem é aplicada principalmente em subsistemas do sistema de manufatura (subsistemas elétricos, mecânicos, pneumáticos e hidráulicos) [Frank, 92]. O primeiro passo de uma estratégia DIF baseada no modelo analítico, consiste em gerar estimativas das saídas (e/ou parâmetros) do *processo real*. Estas estimativas são comparadas com as grandezas correspondentes do *modelo nominal*, gerando assim, sinais residuais (e/ou erros nos parâmetros). O passo final é o *diagnóstico, localização e, se for possível, correção da falha*.

Esta geração de sinais residuais pode ser classificada em duas categorias:

1. Abordagem por *identificação de parâmetros físicos* do sistema (figura 2.3), baseada na estimação dos parâmetros do modelo matemático. No entanto, o problema da identificação pode ser de difícil solução no caso de sistemas de ordens superiores, difíceis de desacoplar ou no caso de sistemas com parâmetros não-lineares.

2. A outra categoria é baseada no *observador*, que avalia o resíduo, reconstruindo a saída do processo para uma decisão lógica a respeito da falha.

Assim o procedimento de detecção de falhas fica constituído por 2 passos:

- Geração de resíduos através da observação da saída.
- Avaliação do resíduo para uma decisão lógica.

Na figura 2.4 é ilustrada a configuração de um observador baseado no modelo do processo. O elemento H representa o ganho de realimentação. Na tarefa de isolamento da falha é necessário estruturar os resíduos gerados. Isto pode ser realizado por múltiplos observadores onde cada um é projetado para ser sensível a alguns tipos de falhas e insensível a outros através de filtros que desacoplam os efeitos de diferentes falhas no

processo. Isto pode ser acompanhado pela conexão dos observadores a diferentes conjuntos de entradas ou saídas do processo, dependendo do tipo de detecção desejado.

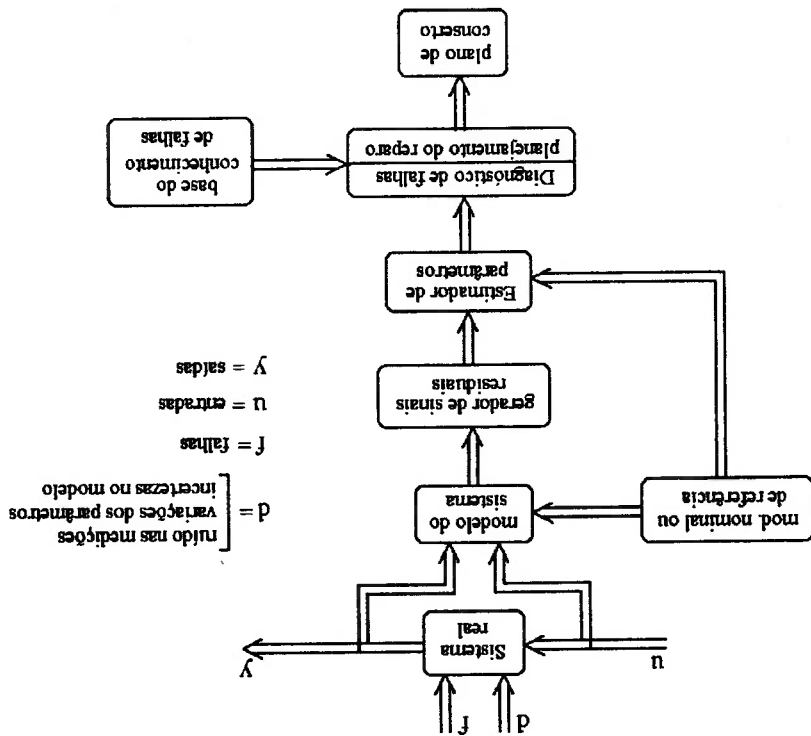


Fig. 2.3 - Estrutura básica DIF baseada no modelo analítico do sistema.

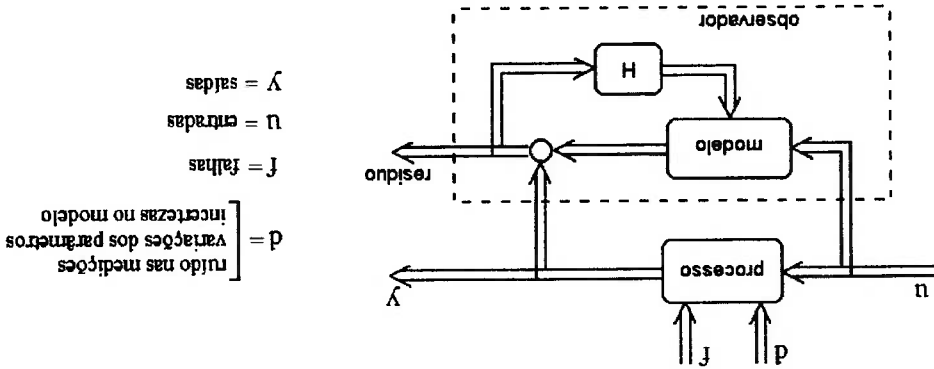


Fig. 2.4 - Conceito básico de um sistema DIF baseado no observador.

### 2.3 TRATAMENTO DE FALHAS E ERROS EM SISTEMAS DE MANUFATURA.

Processos de tratamento de falhas ou erros de operação são difíceis de serem automatizados, já que não podem ser antecipados com precisão quais falhas ou erros ocorrerão numa máquina e ainda estas falhas ou erros nem sempre são as mesmas ou semelhantes.

Em FMS já implementados, mais de 90% dos esforços são dispendidos na implementação de códigos de controle para situações de exceção ou recuperação automática de erros [Zhou, 89]. Na etapa de projeto, tenta-se assim, antecipar erros comuns e codificar rotinas para permitir que o sistema possa recuperar-se automaticamente. Entretanto, podem acontecer duas situações indesejáveis:

- São codificadas rotinas para manipular erros que, de fato, no tempo de vida do programa, nunca acontecem.

- Uma vez que o sistema esteja em operação, ocorrem erros que não foram antecipados, exigindo alterar ou acrescentar rotinas de controle para tratamento de ocorrências futuras dos mesmos erros. Estas alterações após a etapa de projeto, além de reduzir a eficiência do "software", podem criar efeitos colaterais indesejáveis ou aumentar os problemas de manutenção do "software".

Para tratamento de uma falha ou erro que ocorre num equipamento, são consideradas duas abordagens:

1. A primeira inclui sistemas onde existe apenas a capacidade de informar quando o equipamento não consegue executar satisfatoriamente alguma das funções necessárias para desempenhar suas tarefas. Neste caso, são definidos procedimentos e intervenções humanas para eliminar a falha ou erro.

2. A segunda abordagem é considerar desde a etapa de projeto de um equipamento a capacidade para manter, por si mesmo, suas funções sem necessitar da intervenção humana caso aconteça uma falha ou erro. Este equipamento deve ter as seguintes funções [Umeda, 95]:

- **Monitorização** permanente do seu próprio estado.
- **Julgamento** se está num estado normal ou num estado de falha/erro, baseado nos dados coletados na monitorização.
- **Diagnóstico** da falha/erro existente e do componente responsável.
- **Proposição** de um plano de conserto baseado no resultado do diagnóstico.
- Finalmente, deve ter a capacidade de **execução** do plano de conserto proposto.

A **monitorização** e a **execução** da correção são realizadas pelos sensores e atuadores, respectivamente, enquanto o **juízo**, **diagnóstico** e **proposição** de planos de conserto são realizados pelo controlador (ver o esquema na figura 2.5). Note que, nesta figura, não é considerada nenhuma interface com o usuário, pois considera-se aqui que o equipamento é totalmente autónomo.

Uma vez executado o plano de conserto, a monitorização do objeto de controle deve fornecer dados necessários para verificar se o conserto produz a correção da falha e/ou do erro que ocorreu.

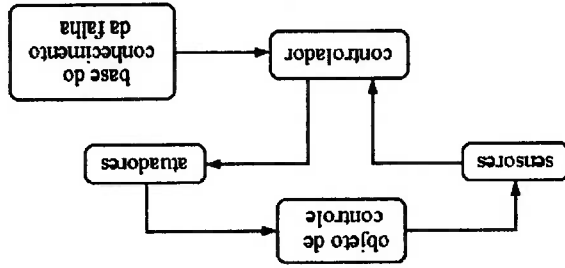


Fig. 2.5 - Estrutura de um equipamento para auto-correção de falhas.

Para descrever melhor o processo de diagnóstico de falhas<sup>[2]</sup>, é necessário definir o que são *simtoma* e *causa*. *Simtoma* é definido como uma função que não é mais desempenhada devido a uma falha na máquina e *causa* é definida como um fenómeno que produz a falha. Assim, todo simtoma está relacionado com alguma causa. O diagnóstico é assim realizado em três passos:

1. Identificação dos simtomas. Quando é detectado um estado de falha, o controlador verifica os dados dos sensores à respeito do funcionamento do objeto de controle para detectar funções que não estejam mais sendo desempenhadas na máquina, comparando os dados dos sensores com os do modelo nominal do sistema.

2. Identificação de candidatos da causa da falha. A máquina de inferência identifica relações entre as alterações nos valores dos parâmetros e os simtomas da falha e através disto, identifica as funções que produzem as alterações destes parâmetros (candidatos a falha), através da pesquisa na base de conhecimento.

3. Identificação de inconsistências entre as causas e os dados dos sensores, até obter, se for possível, a identificação de uma única causa.

Para a proposição de um plano de conserto a partir do diagnóstico da falha, é também necessário executar vários passos. O objetivo do reparo é eliminar os simtomas detectados antes de eliminar as falhas propriamente (objetivos de um reparo tradicional). Em outras palavras, não é necessário o retorno para o estado inicial (sem falhas), em geral: o importante é a manutenção do sistema desempenhando as funções que lhe são atribuídas. Este procedimento consiste de 4 passos:

<sup>[2]</sup> Podemos definir falha como uma alteração prejudicial nos atributos ou na estrutura do objeto de controle.

1. Seleccionadas as causas, a máquina de inferência estabelece os objetivos do reparo, indicando quais ajustes nos parâmetros eliminariam as causas diagnosticadas.

2. Identificação dos candidatos de reparo que satisfazam estes objetivos através do modelo analítico do sistema ou da informação contida na base de conhecimento.

3. Seleção pela máquina de inferência do método de conserto que realize o reparo ou recupere as funções perdidas ou desempenhadas deficientemente/erroneamente pelo objeto de controle, em função de vários critérios como:

- Eliminação de candidatos que envolvam a alteração de grandezas dos parâmetros além dos limites de segurança.
- Preferência por reparos com menor número de efeitos colaterais.

4. Reparo dos efeitos colaterais: se o candidato escolhido gera efeitos colaterais, o sistema de raciocínio deve gerar planos de reparo para cada efeito, executando o planeamento do reparo recursivamente. Assim o plano de conserto proposto pode gerar uma sequência de um ou mais procedimentos de reparo.

Para executar o conserto que foi proposto existem dois tipos de estratégias, dependendo dos recursos utilizados para executá-lo:

1. Na primeira estratégia o controle executa o ajuste dos parâmetros sem alterar ou reorganizar a estrutura lógica do equipamento. Este tipo de reparo é realizado através do controle de atuadores para recuperar totalmente, ou, em parte, as funções requeridas. Em [Umada, 95; Shimomura, 95] apresenta-se o desenvolvimento de um sistema com estas características. Neste sistema, uma fotocopadora (objeto de controle) é conectada num microcomputador do tipo IBM-PC com sistema operacional



MS-DOS através de um conversor AD/DA e um densímetro (medidor de densidade da imagem de saída). O microcomputador coleta os dados (monitorização) e converte em dados quantitativos em qualitativos da imagem de saída. Este microcomputador está ligado a um outro (tipo Macintosh) que “raciocina” a respeito do diagnóstico e realiza a proposição de um plano de reparo como descrito anteriormente, através de simulações (diagnóstico/ proposição de reparo). A figura 2.6 esquematiza o algoritmo para executar a correção de erros num sistema autónomo.

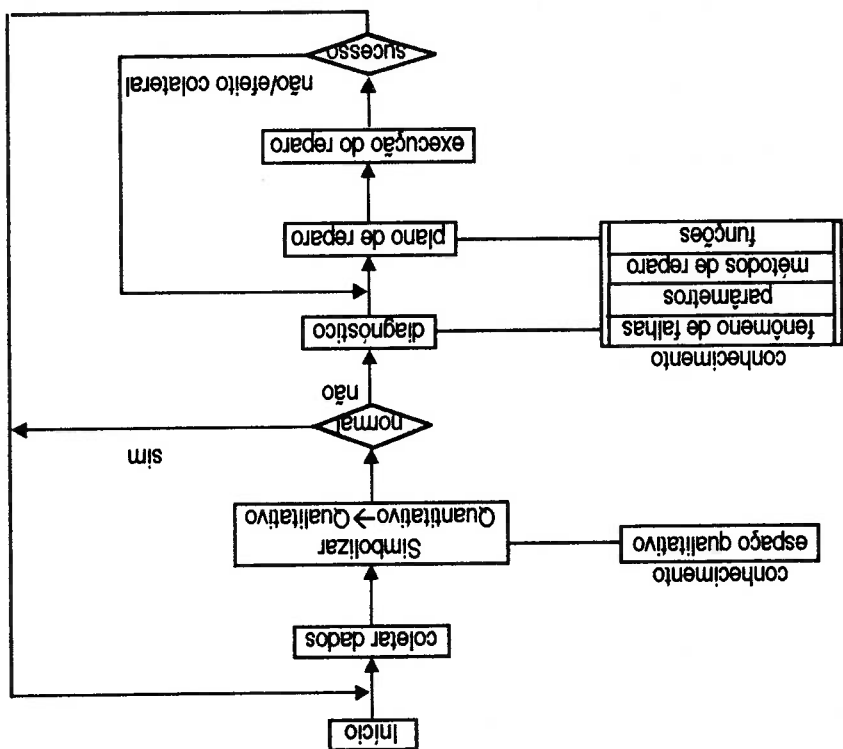


Fig. 2.6 - Algoritmo para execução de reparos num sistema autónomo.

2. Na segunda estratégia de execução, o equipamento possui mecanismos que possibilitam alterar sua própria estrutura original. Tradicionalmente, isto é possível com componentes adicionais ou partes redundantes. Este tipo de estratégia chamada de *redundância funcional*, pode envolver custos, dimensões, desempenho e complexidade indesejáveis no equipamento.

Considerando ainda que mais do que 1/3 das falhas nos sistemas automatizados são devidas ao mau funcionamento do “*software*”, o incremento da confiabilidade no “*software*” é também um assunto chave para o desenvolvimento de um sistema flexível de manufatura [Tohma, 94]. Para avaliar se um “*software*” é confiável devem ser considerados 2 conceitos (da mesma forma que para o “*hardware*”):

- Tolerância a falhas.
- Eliminação de falhas.

Geralmente, um “*software*” tolerante a falhas tende a ser menos eficiente e requer maior tempo computacional para ser processado do que um “*software*” sem estas características. A eliminação de falhas por sua vez envolve metodologias para a prova formal do funcionamento do “*software*” e testes de validação.

## 2.4 PROCEDIMENTOS DE MANUTENÇÃO.

A manutenção de sistemas de manufatura como executada habitualmente consiste simplesmente em substituir peças ou componentes que se desgastam e que resultam em falhas em um ou em vários componentes e uma consequente parada do sistema.

No presente estudo, é considerada uma manutenção **preditiva**, aquela em que a manutenção é executada num momento adequado antes da ocorrência de quebras ou falhas. Esta pode consistir na substituição de peças, limpeza da máquina, lubrificação e outras operações em períodos regulares que são determinados previamente ou quando é diagnosticada uma falha na qual um sistema não tenha condições para executar sua própria correção. Para isto é necessário realizar um diagnóstico que identifique o estado de um determinado componente, quando o mesmo apresentará falha e como programar sua substituição antes da ocorrência da situação crítica que leva à quebra e consequente parada

do sistema. Esta manutenção é baseada na monitorização dos parâmetros do sistema: se o diagnóstico da alteração destes parâmetros acusa a deterioração de algum componente, então o sistema **executa** o procedimento de manutenção ou, pelo menos, **informa** a respeito da manutenção necessária.

A denominação do tipo de manutenção depende do enfoque considerado. Em [Nepomuceno, 89], apresenta-se a seguinte classificação:

1. Manutenção corretiva, que consiste em eliminar uma falha que ocorreu de maneira inesperada.
2. Manutenção periódica, que consiste em interromper a operação do sistema para verificar o estado dos seus componentes.

Na mesma obra o autor defende uma outra classificação dividida em 3 níveis:

- A manutenção de nível 1 é a mais simples e corresponde a manter o sistema funcionando.
- A manutenção de nível 2 é conhecida como manutenção preventiva que acrescenta a substituição de partes do sistema em períodos regulares.
- A manutenção de nível 3, que consiste na manutenção preditiva do sistema.

## 2.5 NECESSIDADE DE MODELO.

Conforme apresentado neste capítulo, os sistemas de manufatura envolvem vários componentes que interagem entre si para produzir algum produto ou partes deste. As características destes sistemas permitem uma abordagem sob uma perspectiva de DEDS (Apêndice I) entretanto, as abordagens tradicionalmente utilizadas no projeto destes sistemas consideram o funcionamento ideal ou sem falhas, ou seja, admite-se que os equipamentos que compõem o sistema estão sempre operando corretamente ou estão

plenamente disponíveis quando forem necessários. Neste caso, os modelos são desenvolvidos para avaliar alternativas de projeto e políticas de operação e não exploram o estudo para um tratamento adequado das situações anormais nos sistemas.

Uma ferramenta que permita o desenvolvimento de sistemas com estas considerações e que ainda possa ser efetiva na análise de propriedades do modelo é evidentemente fundamental para o projeto e implementação de sistemas de manufatura. Neste contexto, diversos trabalhos têm destacado o potencial da Rede de Petri (RP) como uma ferramenta para a modelagem e análise de DEDS, por exemplo [Cardoso, 97; Miyagi, 96; Molloy, 82; Peterson, 81; Zhou, 93; Zurawski, 94b]. Uma das vantagens que diferencia a RP de outras técnicas de modelagem de DEDS é que o mesmo modelo utilizado na análise das propriedades procedimentais e avaliação do desempenho do sistema pode, também, ser utilizado na especificação das estratégias do controlador ou simulação do sistema de manufatura.

### Capítulo 3

#### REDE DE PETRI (RP)

Neste capítulo, são apresentados os principais conceitos de rede de Petri (RP) necessários para a modelagem e o tratamento de falhas em sistemas de manufatura. Para os interessados, no apêndice III encontram-se os fundamentos básicos de RP para um melhor entendimento desta técnica ou de suas extensões. A partir da seção 3.3 é introduzido uma parte do formalismo matemático e ilustrada a equivalência que existe entre as condições restritas presentes nas redes de Petri auto-modificáveis (RPAM) e nos outros tipo de RP, como são RP, com ARCOS inibidores.

Desde que a rede de Petri (RP) foi introduzida por Carl Adam Petri em 1962 na sua tese de doutorado para o estudo de protocolos de comunicação entre componentes assíncronos de um sistema computacional, a RP tem sido amplamente utilizada para modelagem e análise de diferentes aspectos dos sistemas de manufatura. No entanto, até metade da década de '80 as pesquisas eram principalmente de caráter acadêmico, com poucas implicações para implementações práticas. Entretanto com a disponibilidade de "hardware" e "software" de maior capacidade e menores custos e com o desenvolvimento do conceito de Manufatura Integrada por Computador (CIM) o potencial da RP foi sendo reconhecido como uma técnica efetiva para o projeto de sistemas de manufatura.

O projeto e operação destes sistemas necessitam de uma modelagem e análise a fim de selecionar alternativas ótimas de projeto, políticas de operação e dados qualitativos e quantitativos da dinâmica do sistema. Assim a RP foi redescoberta por profissionais da indústria como uma ferramenta versátil e eficiente para a modelagem e análise de sistemas produtivos. A RP, suas extensões e variações, comprovaram sua efetividade como uma

representação gráfica e uniforme para a descrição, análise, projeto e controle de sistemas de manufatura.

Para modelagem de um sistema produtivo através de uma representação em RP, uma metodologia comumente adotada é:

- Identificação de recursos, operações ou atividades no sistema.
- Estabelecimento da sequência das atividades em cada processo.
- Representação das atividades, geralmente, por LUGARES que são conectados à TRANSIÇÕES, as quais indicam o início e o término destas atividades.
- Definição dos recursos como LUGARES e conexão destes às TRANSIÇÕES de início e término de cada operação.
- Especificação da marcação inicial (estado inicial).

O controle de sistemas de manufatura utilizando RP também já foi abordado por vários pesquisadores [Zhou, 93; Zurawski, 94a; Hasegawa, 96]. Entretanto, nestes trabalhos, a ênfase foi colocada em como analisar e controlar sistemas em condições de operação normal. Poucos trabalhos têm sido publicados sobre a aplicação de RP no problema de tratamento e recuperação automática de falhas/erros. O problema fundamental está na estrutura da RP, que seria muito complexa<sup>[1]</sup> para representar todos os processos que poderiam ocorrer nestas circunstâncias além dos procedimentos normais, que geralmente já possuem um elevado grau de variabilidade, como no caso de sistemas flexíveis de produção.

Neste sentido, identificou-se extensões de RP adequadas para o tratamento de falhas, erros e alterações drásticas<sup>[2]</sup> a nível estrutural da rede e efetivas no estudo de sistemas que envolvem o tratamento de falhas/erros.

[1] Complexa em função da explosão do número de elementos (LUGARES/TRANSIÇÕES) e respectivos ARCOS.  
 [2] Drástica em função da re-estruturação de fluxos e inclusão de sub-redes.

### 3.1 VANTAGENS DA REDE DE PETRI

Dentre as técnicas existentes para estudar o problema de tratamento de falhas, erros de operação e procedimentos de manutenção, optou-se pelas Rede de Petri (RP) em função do seu potencial como uma técnica uniforme para modelagem, análise formal e projeto de sistemas de manufatura. Segundo [Hasegawa, 96; Zhou, 93], dentre as características positivas da RP pode-se citar:

1. Habilidade para representar de modo natural a sincronização de processos, eventos concorrentes, causalidade, compartilhamento de recursos (principal elemento para ocorrência de bloqueios), presença de conflitos, bloqueios, etc. que são características próprias dos sistemas de manufatura.

2. Identificação clara e explícita de estados e ações, que permite a monitorização do sistema em tempo real.

3. Possibilidade de escolha de um nível apropriado de abstração e desenvolvimento de uma representação adequada das características essenciais do sistema. De fato, permite aumentar ou reduzir o nível de detalhamento segundo os requerimentos de modelagem, permitindo a construção de modelos “*top-down*” e “*bottom-up*” através de técnicas como refinamento e modularização.

4. Interpretabilidade, podendo associar elementos de diferentes significados numa mesma representação ou segundo o propósito do modelo (validação, avaliação do desempenho, “*scheduling*”, implementação do controle, etc.). Por exemplo, algumas interpretações de TRANSIÇÃO e seus LUGARES de entrada e saída são ilustradas na tabela 3.1.

Tabela 3.1 Algumas interpretações típicas de LUGARES e TRANSIÇÕES.

Lugares de entrada	Transições	Lugares de saída
Pré-condição	Evento	Pós-condição
Dado de entrada	Passo de computação	Dado de saída
Sinal de entrada	Processador de sinais	Sinal de saída
Recurso necessitado	Tarefa ou trabalho	Recurso liberado
Condição	"Clause in logic"	Conclusão
"Buffer"	Processador	"Buffer"

5. Como ferramenta gráfica, a RP permite uma comunicação visual, facilitando a interação entre as pessoas envolvidas no projeto (projetista, operador, etc.), evitando formas complexas<sup>[3]</sup> de especificação utilizando descrições textuais ambíguas ou notações matemáticas específicas.

6. Como ferramenta formal, um modelo em RP pode ser descrito matematicamente, por exemplo, por matrizes. Pode ser utilizado para especificar uma semântica formal do sistema e como ferramenta no estudo de propriedades tais como bloqueios, limitação, etc. Além disso, pode ser utilizado para implementar ou gerar códigos para controle ou simulação. De fato, o mesmo modelo empregado na análise das propriedades procedurais e avaliação do desempenho pode também ser utilizado na programação de um simulador e/ou controlador [Zurawaski, 94a].

Lembramos, no entanto, que a simulação para propósitos de validação de um modelo produz um conjunto limitado de estados do sistema modelado e, assim, a simulação é efetiva para identificar presença, mas não ausência de erros no modelo. Considerando que esta característica da análise de sistemas por simulação envolve necessariamente um estudo da estratégia de obtenção dos dados visando a generalização dos resultados.

[3] O termo *complexo* é aqui utilizado em função da necessidade de envolver conhecimento específico de uma área concreta e/ou formalismo matemático especializado.



### 3.2 MODELAGEM DO TRATAMENTO DE FALHAS ATRAVÉS DE RP

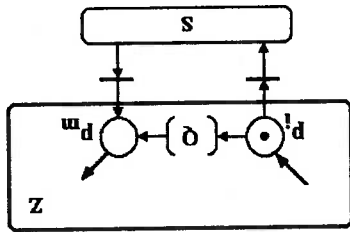
No tratamento de falhas e erros, geralmente são introduzidas sub-redes tais que, enquanto o sistema opera normalmente, a sequência dos processos representada na RP original é executada e, se ocorre uma falha ou erro, a parte que foi acrescentada é executada para recuperar o sistema desta situação. Estas sub-redes realizam a recuperação de falhas/erros que foram detectados com base no julgamento do estado de processamento (normal ou em estado de falha/erro) baseado nos dados coletados na monitorização do sistema (veja seção 2.3).

Existem 4 métodos básicos para alteração do procedimento original utilizando uma sub-rede adicional que permita a recuperação de falhas/erros no sistema [Zhuo, 89]: método da entrada condicionada, método da rota alternativa e métodos de recuperação “*Forward*” e “*Backward*”.

- *Método da entrada condicionada*: neste caso, quando é detectado um estado anormal ou de falha/erro num LUGAR da rede que representa um processo ou estado de um sistema de manufatura, este pode voltar a ser um processo ou estado em situação normal se for realizada uma ação corretora no sistema. A figura 3.1-a) esquematiza a recuperação de falhas por este método, onde Z representa a RP original e S é um bloco de RP adicional representando o procedimento de recuperação de falhas. As figuras 3.1-b) e 3.1-c) esquematizam dois casos particulares deste método: o primeiro (figura 3.1-b)) ilustra o caso no qual somente alguns parâmetros de operação necessitam ser alterados (por exemplo, a velocidade de corte ou incremento de força) e o outro (figura 3.1-c)), modela um processo de recuperação representado apenas por um LUGAR.

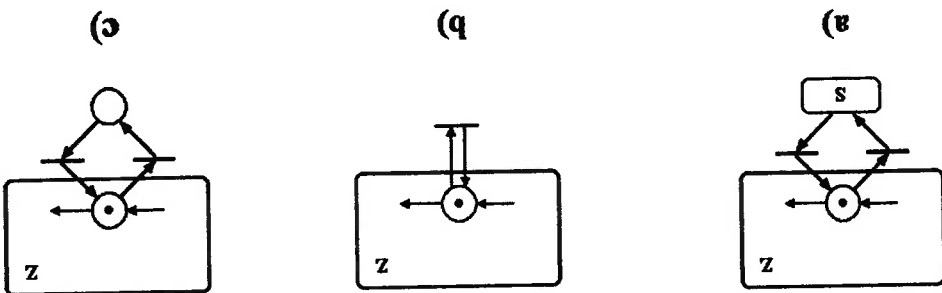
- Recuperação de erros "Backward": neste método, após a execução da atividade  $Q$ , é detectado no LUGAR  $P_m$  um estado de falha/erro. Este problema é solucionado com a execução das operações da sub-rede  $S$  que resulta numa re-execução de  $Q$  (figura 3.3). Esta metodologia é equivalente ao reprocessamento de elementos: por exemplo, quando uma peça fica com um acabamento aquém do exigido (por exemplo: acabamento superficial numa usinagem ou pintura), isto pode ser resolvido através de um reprocessamento.

Fig. 3.2 - Esquema de recuperação pelo método da rota alternativa.



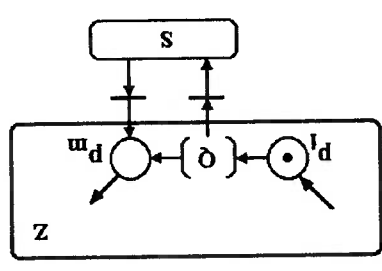
- Método da rota alternativa: neste caso, um estado anormal detectado no LUGAR  $P_i$  é tratado e o problema é solucionado gerando-se uma rota alternativa de  $P_i$  diretamente para  $P_m$  através da sub-rede  $S$ , onde serão gerados outros estados que garantem a recuperação do sistema e evitando a execução da atividade  $Q$  (figura 3.2). Esta metodologia é equivalente à utilização de recursos redundantes no sistema que substituem as funções realizadas em  $Q$  (veja redundância funcional na seção 2.3).

Fig. 3.1 - Esquema de recuperação pelo método da entrada condicionada.



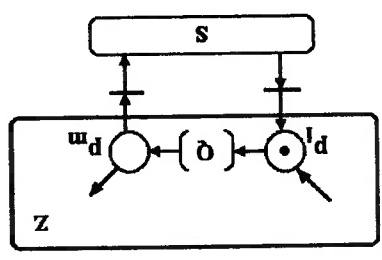
Estes métodos incrementam o número de nós da rede original, os quais tornam o modelo em RP muito mais complexo para representar todos os processos possíveis (normais + situações anormais). Também criam conflitos em todos os LUGARES onde será decidida a continuidade nos processos normais (rede Z) ou o início da execução da sub-rede para tratamento de situações anormais (rede S).

Fig. 3.4 - Esquema de recuperação de erros "Forward".



- **Recuperação de erros "Forward":** no caso de detecção de um estado de falha/erro no processo Q, o sistema pode retornar a um estado normal executando-se a sub-rede S. Esta técnica é utilizada para recuperação de uma falha/erro que é detectada apenas durante o processo Q (figura 3.4).

Fig. 3.3 - Esquema de recuperação de erros "Backward".



### 3.2.1 Manutenção das propriedades da RP

A análise qualitativa de uma RP inclui a análise de propriedades como limitação, segurança, vivacidade (ausência de bloqueios no sistema) e reversibilidade (re-inicialização). Acrescentar sub-redes da forma descrita anteriormente, garante em princípio, a conservação das propriedades da RP original. Os teoremas que demonstram a manutenção destas propriedades serão apresentados a seguir (detalhes sobre a prova destes teoremas encontra-se em [Zhou, 89]).

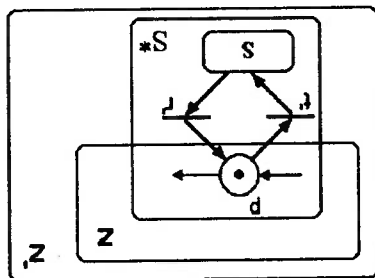


Fig. 3.5 - Rede de Petri associada (S\*).

#### Definição 3.1:

S\* é a rede formada pela sub-rede S, o lugar p e as TRANSIÇÕES r e r', incluindo os respectivos ARCOS de ligação (figura 3.5).  
 Z' é formada pelas rede Z e S, incluindo as TRANSIÇÕES r e r'.  
 M<sub>0</sub>: Marcação inicial de Z'.  
 M<sub>0</sub>: Marcação inicial de Z.  
 M\*<sub>0</sub>: Marcação inicial de S\*.

Teorema 3.1: Z' com respeito a M\*<sub>0</sub> é limitada (ou segura) se e somente se Z com respeito a M<sub>0</sub> e S\* com respeito a M\*<sub>0</sub> forem limitadas (ou seguras).

**Teorema 3.2:**  $Z'$  com respeito a  $M'_0$  é viva se e somente se  $Z$  com respeito a  $M_0$  e  $S^*$

com respeito a  $M^*_0$  forem vivas.

**Teorema 3.3:** Se  $Z$  for reversível e  $S^*$  for reversível, caso  $P$  (figura 3.5) tenha MARCA,

$Z'$  é reversível.

### 3.3 REDE DE PETRI AUTO-MODIFICÁVEL (RPAM) PARA TRATAMENTO DE FALHAS/

#### ERROS EM SISTEMAS DE MANUFATURA.

A Rede de Petri Auto-Modificável (RPAM) [Valk, 78; Sasaki, 96] é uma extensão da RP, sendo um **multigráfico bipartido direcionado** que tem por característica a capacidade de alterar sua própria estrutura em função do estado atual do sistema, o que é conseguido através de ARCOS cujos pesos ( $k$ ), ao invés de terem valores constantes, são função do número de MARCAS de LUGARES específicos definidos previamente. Para explicitar esta relação, rotula-se o ARCO (peso do ARCO) com o nome do LUGAR ( $p_i$ ) ao qual esta associado, conforme é ilustrado na figura 3.6.

**Definição 3.2:** Formalmente, pode-se definir a estrutura de uma RPAM como uma quintupla  $RPAM=(P, T, I, O, M_0)$ , onde:

- $P = \{p_1, p_2, \dots, p_m\}$  é o conjunto de LUGARES da rede, onde  $m \in \mathbf{N}$  é o número total de LUGARES da rede.
- $T = \{t_1, \dots, t_n\}$  é o conjunto de TRANSIÇÕES, onde  $n \in \mathbf{N}$  é o número total de TRANSIÇÕES da rede.
- $M_0$  é a marcação inicial.
- $I(p_j, t_i) = f(p_j)$ , onde  $f$  é uma função que especifica a relação entre  $p_j$  e  $t_i$  e depende da marcação do LUGAR  $p_q$  ( $f(p_q) \in \mathbf{N}$ ).

- $O(t_i, p_j) = g(p_u)$ , onde  $g$  é uma função que especifica a relação entre  $t_i$  e  $p_j$  e depende da marcação do LUGAR  $p_u$  ( $g(p_u) \in \mathbb{N}$ ).

Esta definição não impõe limites aos pesos dos ARCOS auto-modificáveis. Assim, além da marcação de um LUGAR  $p_q$ , por exemplo, pode-se definir expressões como  $I(p_j, t_i) = 2 + M(p_q)$  para o peso dos ARCOS. No entanto, nesta pesquisa e na bibliografia consultada, somente foram abordados casos nos quais  $I(p_j, t_i) = M(p_q)$ . Assim, na representação gráfica, simplesmente coloca-se a inscrição  $p_q$  sobre o ARCO auto-modificável (em azul) que conecta  $p_j$  a  $t_i$  (figura 3.6).

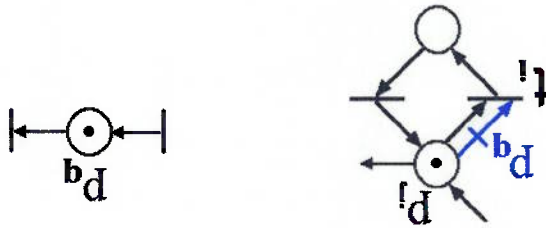


Fig. 3.6 - RP com ARCO auto-modificável.

A regra de disparo de uma TRANSIÇÃO numa RPAM segue à princípio, as mesmas regras de uma RP; isto é, uma TRANSIÇÃO ( $t_i \in T$ ) fica habilitada para disparar se cada LUGAR de entrada  $p_j$  (tal que  $I(p_j, t_i) > 0$ ) contém pelo menos um número de MARCAS igual ao peso do ARCO  $I(p_j, t_i)$ . Basicamente uma TRANSIÇÃO  $t_i \in T$  está habilitada se e somente se

$$M(p_j) \geq I(p_j, t_i) = \begin{cases} f(p_q) & \text{se } I(p_j, t_i) \text{ for ARCO auto-modificável.} \\ k_{ij} & \text{se } I(p_j, t_i) \text{ for um ARCO normal (Apêndice III).} \end{cases}$$

Assim, na evolução dos estados numa RPAM, os pesos desses ARCOS podem alterar-se à medida que a marcação se altera. A alteração do peso de um ARCO pode ser temporária ou permanente e pode, inclusive, assumir o valor zero, equivalente à inexistência do ARCO.

Tem-se, assim, uma variante de RP cuja estrutura pode ser dinamicamente alterada, podendo operar em modos alternativos que dependem do estado do sistema. Isto permite que o sistema descrito através de RPAM adapte sua estrutura às ocorrências imprevistas que podem surgir durante o funcionamento normal do sistema. Estas situações imprevistas na modelagem de sistemas de manufatura envolvem a quebra de equipamentos, erros de operação, manutenções não periódicas; enfim, tudo que resulta em degradação das operações ou parada de parte ou de todo o sistema.

A definição matricial da RP é útil para representar formalmente o comportamento dinâmico do sistema. Em [Valk, 78], é apresentada uma definição matricial de RPAM similar à definição de RP [Zurawski, 94]. A equação de estado representa uma alteração na distribuição das MARCAS dos LUGARES como resultado do disparo de uma TRANSIÇÃO. A equação de estado é definida da seguinte forma:

$$M_k = M_{k-1} + A^T u_k \quad (3.1)$$

onde:

- $M_k$  é um vetor coluna ( $m \times 1$ ) que representa a marcação imediatamente alcançável a partir de  $M_{k-1}$  após o disparo da TRANSIÇÃO  $t_i$  (identificada em  $u_k$ ).
- $u_k$  é um vetor ( $n \times 1$ ) com grandeza de 1 na posição  $i$  que representa o disparo da TRANSIÇÃO  $t_i$  e zero (0) nas outras posições.
- $A$  é a matriz de incidência ( $n \times m$ ) definida por  $A = O(T, P) - I(P, T)$ . No entanto, a matriz  $A$  não é constante, já que  $I(P, T)$  é uma função que pode alterar-se em função da marcação  $M_i$ .

Em [Valk, 78] são apresentados também, demonstrações a respeito das propriedades de alcançabilidade, limitação e vivacidade (ausência de bloqueios) nas RPAM, tratados pelos teoremas a seguir.

Teorema 3.4: Se numa RP limitada e viva forem acrescentados ARCOS auto-modificáveis, esta nova rede (agora uma RPAM) será também limitada e viva.

Teorema 3.5:  $R(RP) \subseteq R(RPAM)$ , onde:

$R(RP)$  é o conjunto de alcançabilidade da rede RP a partir de  $M_0$ .

$R(RPAM)$  é o conjunto de alcançabilidade da rede RPAM a partir de  $M_0$ .

O conjunto de alcançabilidade obtido da RP original a partir de  $M_0$  está contido no conjunto de alcançabilidade obtido para a respectiva RPAM a partir de  $M_0$ .

A demonstração destes teoremas não está no escopo do presente trabalho e assim é omitida. Uma discussão mais detalhada encontra-se em [Valk, 78].

Pode-se estabelecer assim, um paralelo entre a modelagem de sistemas de manufatura através de RPAM com sua capacidade de alterar a estrutura do grafo e o controle adaptativo realimentado de sistemas dinâmicos com variáveis contínuas (CVDS), que alteram os parâmetros de funcionamento da planta em tempo real, como uma forma de adaptar-se às diferentes condições de operação do sistema [Levis, 87].

### 3.3.1 Exemplo de Aplicação

Como um exemplo de utilização da RPAM, apresenta-se o modelo dos processos de leitura e escrita compartilhando um arquivo comum.

Existem diferentes variações do problema leitor/escritor, mas as suas estruturas básicas são similares. Seus processos são de 2 tipos: processo de leitura e processo de escrita. Ambos os processos compartilham um arquivo comum de dados. Os processos de leitura não ocorrem enquanto um processo de escrita manipula o arquivo comum. Assim, os processos de escrita devem impedir a execução de outros processos (todas as leituras e



outros de escrita), apesar de múltiplas leituras poderem ser executadas simultaneamente. Na figura 3.7, está ilustrada uma solução em RP com um limite de  $n$  processos de leitura que podem ser executados simultaneamente, o número inicial de leitores e escritores são  $s$  e  $t$ , respectivamente.

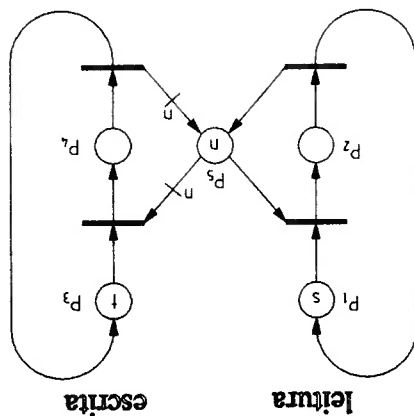
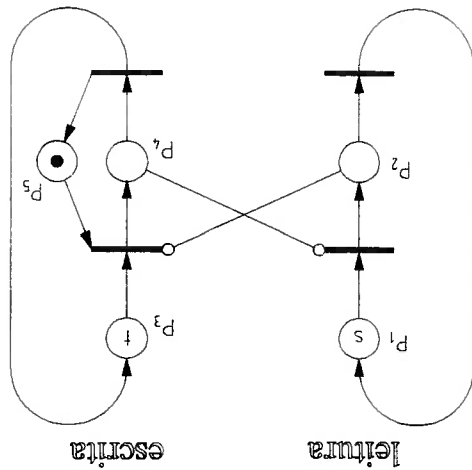


Fig. 3.7 - Modelagem da solução dos processos de leitura/escrita.

Lugar	Descrição
$P_1$	leitores aguardando
$P_2$	contador de leitores
$P_3$	escritores aguardando
$P_4$	contador de escritores
$P_5$	No. limite de processos de leitura

Segundo [Peterson, 81], se neste problema o número de processos de leitura sendo executados simultaneamente for limitado a  $n$ , é possível garantir que não ocorra qualquer processo de leitura ou outro processo de escrita. Isto é possível definindo-se na rede ARCOS com peso igual ao número de processos de leitura ( $I(p_i, t_i) = n$ ) na TRANSIÇÃO que inicia o processo de escrita (modelado no LUGAR  $P_4$ ). Entretanto, com um número ilimitado de processos de leitura podendo ser executados simultaneamente, o problema não teria solução através deste tipo de RP pela dificuldade de avaliar que nenhum processo de leitura/escrita esteja sendo executado. No entanto, o modelo em RP utilizando ARCOS inibidores (apêndice III.2), ilustrado na figura 3.8, fornece uma solução do problema para um número ilimitado de processos de leitura possíveis de executar simultaneamente. Os ARCOS inibidores

asseguram que não exista nenhuma MARCA nos LUGARES  $p_2$  e  $p_4$  no caso de disparo de  $p_1$  que inicia o processo de escrita. TRANSIÇÃO que inicia o processo de escrita.



Lugar	Descrição
$p_1$	processo de leitura aguardando
$p_2$	contador de leitores
$p_3$	processo de escrita aguardando
$p_4$	contador de escritores
$p_5$	escritor livre

Fig. 3.8 - Modelagem da solução dos processos de leitura/escrita utilizando ARCOS inibidores.

O mesmo problema é agora abordado através de RPAM utilizando ARCOS auto-modificáveis num sistema com 2 leitores ( $s=2$ ) e 2 escritores ( $t=2$ ), (conforme ilustrado na figura 3.9). Para cada leitor (ou escritor) é necessário delinear uma sequência de processos. Assim, se  $s$  (ou  $t$ ) for maior, os números de nós e de ARCOS no modelo multiplicam-se. Isto acontece pelo fato dos ARCOS auto-modificáveis serem utilizados como uma condição restritiva para inibir, em certas circunstâncias, o disparo de TRANSIÇÕES nas quais estes ARCOS "entram". Assim, em geral, um LUGAR ( $p_j$ ) conectado a uma TRANSIÇÃO através de um ARCO auto-modificável deve ter um número de MARCAS limitado  $M(p_j) \leq 1$ .

Da mesma forma que nos modelos das figuras 3.7 e 3.8, a RPAM soluciona o problema verificando a execução dos processos de leitura ou outro processo de escrita.

Entretanto, neste tipo de rede, este intterramento pode ser realizado através de ARCOS auto-modificáveis que não necessitam de uma ligação gráfica explícita, podendo “separar”, assim, os processos de leitura e escrita em duas redes “disjuntas”.

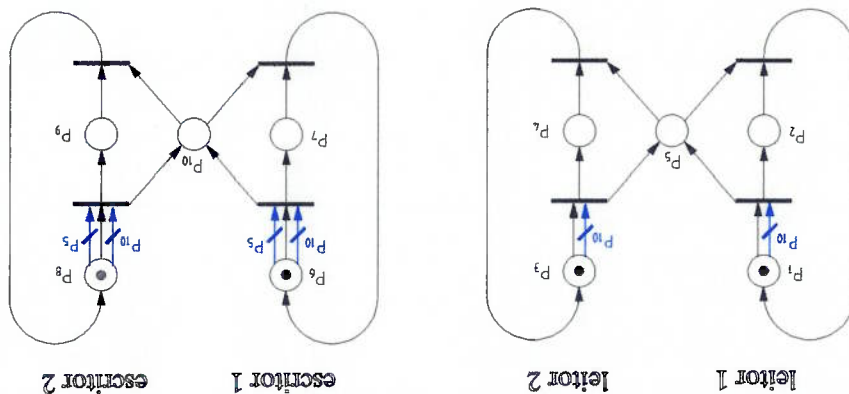


Fig. 3.9 - Solução do problema dos processos de leitura/escrita utilizando RPAM.

As figuras 3.7, 3.8 e 3.9 modelam um sistema com as mesmas características utilizando 3 elementos diferentes, RP, RP com ARCOS inibidores e RPAM, respectivamente. Este exemplo ilustra a equivalência entre as representações e, ao mesmo tempo, apresenta as diferentes representações gráficas.

### 3.3.2 Equivalência entre as Condições Restritivas das RP.

Pelas definições apresentadas, pode-se estabelecer uma equivalência entre as condições habilitadoras e inibidoras presentes nas várias classes de RP. Os ARCOS inibidores

(e habilitadores), citados anteriormente, foram concebidos para testar a ausência (ou presença) de MARCAS e permittem, assim, relacionar o estado de elementos internos ao sistema (ou da parte de interesse descrito pelo modelo). Isso ocorre, por exemplo, no caso de dispositivos sensores e dispositivos de comando de operações cujos sinais são diretamente utilizados para o controle de outros equipamentos [Hasegawa, 96]. Estes ARCOS são de natureza distinta dos ARCOS auto-modificáveis, que foram concebidos, inicialmente, para alterar a estrutura do grafo através do estado atual do sistema.

A figura 3.10 ilustra um tipo de equivalência entre estas condições habilitadoras e inibidoras aplicadas nas TRANSIÇÕES  $t_i$  e  $t_h$ . Em todos os casos  $t_h$  é uma TRANSIÇÃO inibidoras para disparar, enquanto que  $t_i$  é uma TRANSIÇÃO inibida se a condição  $p$  for verificada (MARCA no LUGAR  $p$ ).

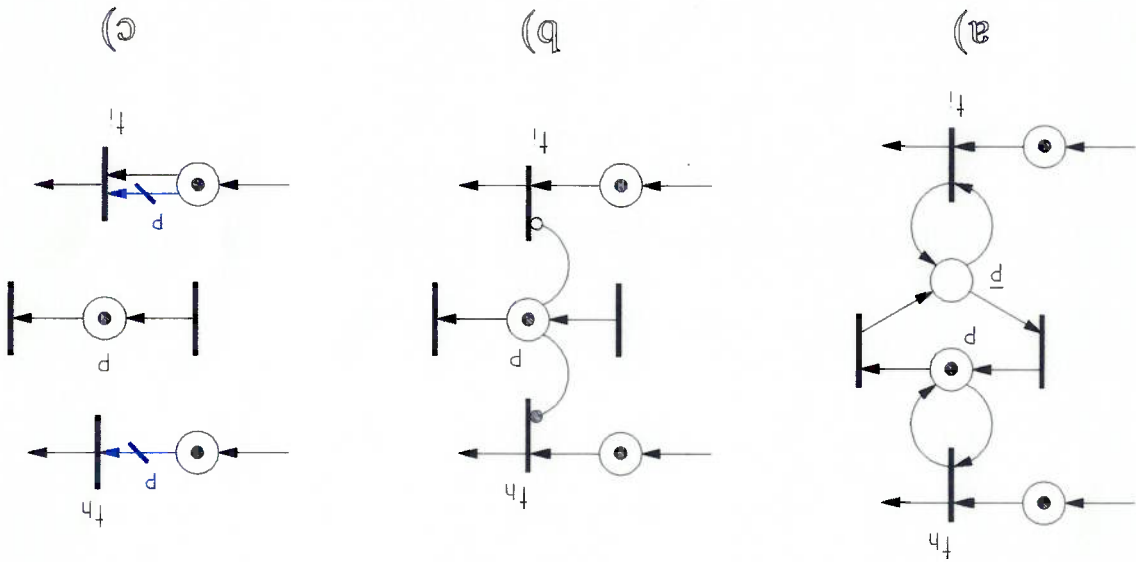


Fig. 3.10 - Equivalência entre várias condições habilitadoras e inibidoras nas RP.

Na figura 3.10-a) é representado o esquema inicialmente utilizado em RP. Neste caso a marcação no LUGAR  $p$  representa uma negação da marcação no LUGAR  $p$ , que obviamente é a negação de uma situação lógica (sim/não, 1/0, verdadeiro/falso, etc.).

característica de RP condição/evento (C/E) [Miyagi, 96]. Na figura 3.10-b) a mesma situação é modelada através de ARCOS habilitadores e inibidores restringindo o disparo das respectivas TRANSIÇÕES ( $t_h$ ,  $t_i$ ). A figura 3.10-c) ilustra o mesmo esquema utilizando ARCOS auto-modificáveis condicionados à marcação do LUGAR p.

A RPAM também considera a existência de interações de redes graficamente “disjuntas” como no caso da figura 3.10-c), onde as sub-redes superior (sub-rede com a TRANSIÇÃO  $t_h$ ) e inferior (com  $t_i$ ) podem não necessariamente pertencer à sub-rede que contém o LUGAR p. Um outro aspecto é que, em todos os casos, as condições que habilitam as TRANSIÇÕES  $t_h$  e  $t_i$  estão relacionadas com a marcação do LUGAR p e, neste sentido, a RPAM é uma representação graficamente mais compacta.

Cabe indicar que, nas figuras 3.7, 3.8 e 3.9, foi estabelecida esta equivalência, solucionando o problema de leitura e escrita utilizando um arquivo comum através de uma representação em RP, RP com ARCOS inibidores e RPAM, respectivamente.

Nos casos estudados no próximo capítulo através da modelagem com RPAM, serão também representados sistemas com as mesmas características através de RP com ARCOS inibidores, permitindo visualizar melhor a equivalência descrita nesta seção.

### 3.3.3 Sistema de Processos Espontâneos com Restrições

Existe uma situação particular na qual se todos os ARCOS de entrada numa TRANSIÇÃO forem ARCOS auto-modificáveis e, se por acaso, simultaneamente, todos os LUGARES relacionados aos respectivos ARCOS não tiverem MARCAS ( $M(p_j)=0$ ), então isto seria equivalente à inexistência dos ARCOS de entrada e, assim, esta TRANSIÇÃO não teria nenhuma condição restritiva de entrada (pré-condição) limitando o seu disparo. Esta

TRANSIÇÃO poderia, então, disparar indefinidamente. Isto é uma propriedade característica de RPAM que a distingue de outras variações de RP.

Para assegurar modelos com a equivalência descrita na seção anterior, define-se um Sistema de Processos Espontâneos com Restrições (SPSC ou *"spontaneous process system with constraints"*).

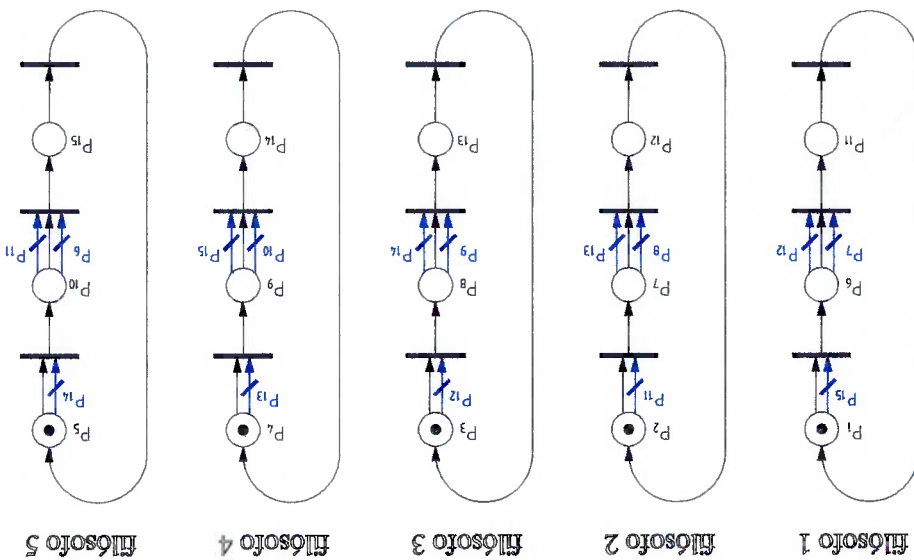
**Definição 3.3:** uma RPAM é dita espontânea se, para cada marcação alcançável, existir uma TRANSIÇÃO  $t_i$  habilitada a partir da marcação atual e que remove pelo menos uma MARCA do LUGAR  $p_j$ . Portanto, o termo *espontânea* refere-se à vivacidade do tipo *"L4-life"* [Murata, 89].

**Definição 3.4** [Valk, 78]: se uma TRANSIÇÃO numa RPAM estiver inibida ou desabilitada, mas na RP equivalente (sem os ARCOS auto-modificáveis) estiver habilitada, então este é um SPSC.

**Definição 3.5:** Um SPSC é uma RP na qual foram acrescentados ARCOS auto-modificáveis para aumentar o número de restrições que deverão ser satisfeitas para disparar uma TRANSIÇÃO  $t_i$ , geralmente envolvida em algum conflito, com a consequente solução deste.

Considera-se, agora, o clássico problema dos 5 filósofos chineses jantando. Este problema, proposto em 1968 [Peterson, 81], consta de 5 filósofos sentados a uma mesa que, alternadamente, meditam e se alimentam. Entre cada filósofo, existe um palito, mas para se alimentar cada filósofo necessita de 2 palitos: assim, se cada filósofo pegar, por exemplo, o palito do lado esquerdo e então aguardar que o palito da direita seja liberado, eles ficarão sempre aguardando (uma condição de bloqueio).

Na figura 3.11 é apresentado um modelo do problema usando RPAM do tipo SPSC constituída por 5 redes disjuntas, cada uma representando o estado de cada filósofo (meditando, pegando o garfo da direita ou jantando) com 3-LUGARES e 3-TRANSIÇÕES cada. Como esperado, estas redes apresentarão uma situação de bloqueio se a marcação dos LUGARES  $p_6, p_7, \dots, p_{10}$  for 1.



Lugar	Descrição
$p_1, \dots, p_5$	Filósofo / meditando
$p_6, \dots, p_{10}$	Filósofo / pegando o palito da esquerda
$p_{11}, \dots, p_{15}$	Filósofo / pegando o palito da direita

Fig. 3.11 - RPAM que modela o problema dos 5 filósofos chineses.

Um outro exemplo de RPAM do tipo SPSC é a rede que modela o problema dos processos de leitura e escrita utilizando um arquivo comum, discutido anteriormente.

No próximo capítulo as RPAM do tipo SPSC serão aplicadas para o estudo de sistemas de manufatura, onde suas vantagens para a modelagem e análise de situações de falha/erro serão evidenciadas.



**ESTUDO DE CASOS**

**Capítulo 4**



## Capítulo 4

### ESTUDO DE CASOS

Neste capítulo, são ilustrados através de exemplos a modelagem utilizando a RPAM para o tratamento de situações anormais como falhas ou erros de operação em sistemas de manufatura ou em seus componentes baseados nas metodologias abordadas nos capítulos anteriores, confirmando, assim, a funcionalidade desta abordagem. Nos modelos dos sistemas estudados, foram consideradas algumas simplificações que têm como propósito concentrar a análise sobre algumas características particulares de interesse para o presente estudo.

#### 4.1 CASO I

Um modelo de um sistema flexível de produção é utilizado para ilustrar as propriedades da RPAM. O sistema consiste de duas máquinas-ferramentas (M1 e M2) que executam operações de usinagem, um robô manipulador e duas esteiras transportadoras. As esteiras são utilizadas para introduzir e retirar peças nas máquinas. O robô alimenta as máquinas com três tipos de peças ( $w_1, w_2$  e  $w_3$ ) transportadas pelas esteiras, tais que:

- $w_1$  são peças que somente serão processadas por M1,
- $w_2$  são peças que inicialmente são processadas por M1 e posteriormente por M2,
- $w_3$  são peças que serão processadas somente em M2.

O esquema do sistema é representado na figura 4.1.

Os tempos dispendidos nas operações de carregamento e descarregamento de peças, executadas pelo robô, são relativamente menores que as outras operações, assim, desconsidera-se o tempo de execução destas operações de modo que são representadas apenas por uma TRANSIÇÃO simples (atividade instantânea). Caso a suposição anterior não

seja válida e/ou exista a necessidade de maior detalhamento, esta atividade (TRANSIÇÃO) pode ser decomposta em várias outras, como será ilustrado no próximo estudo de caso. O fato desta atividade ser assumida como instantânea permite também assumir que não existem solicitações simultâneas do recurso compartilhado **R** (robô manipulador) e, desta forma, tal solicitação pode ser eliminada, simplificando o modelo.

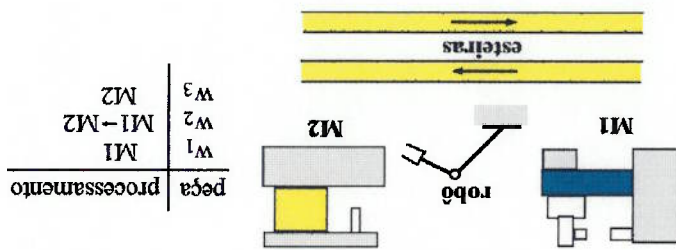


Fig. 4.1 - FMS constituído por 2 máquinas, 1 robô e 2 esteiras.

Neste exemplo, não foram consideradas falhas, erros ou tarefas de manutenção do robô e das esteiras pois, considera-se que os principais problemas ocorrem na usinagem executada pelas máquinas (M1 e M2). Assume-se, também, que as falhas de natureza inesperada nas máquinas ocorrem quando alguma peça está sendo processada. A tarefa de manutenção é considerada correta num certo sentido, mas como estas falhas sempre podem ocorrer com uma relativa previsibilidade, uma abordagem preditiva pode ser considerada e, assim, sua execução pode ser melhor planificada, de modo que ela seja executada quando a máquina estiver livre de peças.

No caso em que a necessidade deste tipo de manutenção for diagnosticada enquanto a máquina estiver processando uma peça, a execução desta tarefa de manutenção deve aguardar até que a peça seja descartada. Qualquer fato anormal que ocorra nas máquinas que não possa aguardar o descarregamento da peça para ser corrigido, será considerado uma falha.

As figuras 4.2 e 4.3 [Riascos, 97] representam o modelo do sistema produtivo através de uma RPAM do tipo SPSC. A sub-rede da figura 4.2 foi baseada no fluxo dos processos de usinagem das peças nos equipamentos: 3 processos utilizando 2 máquinas (ver apêndice III.3 para abordagem alternativa). A figura 4.3 refere-se às funções de monitorização e diagnóstico, descritas adiante.

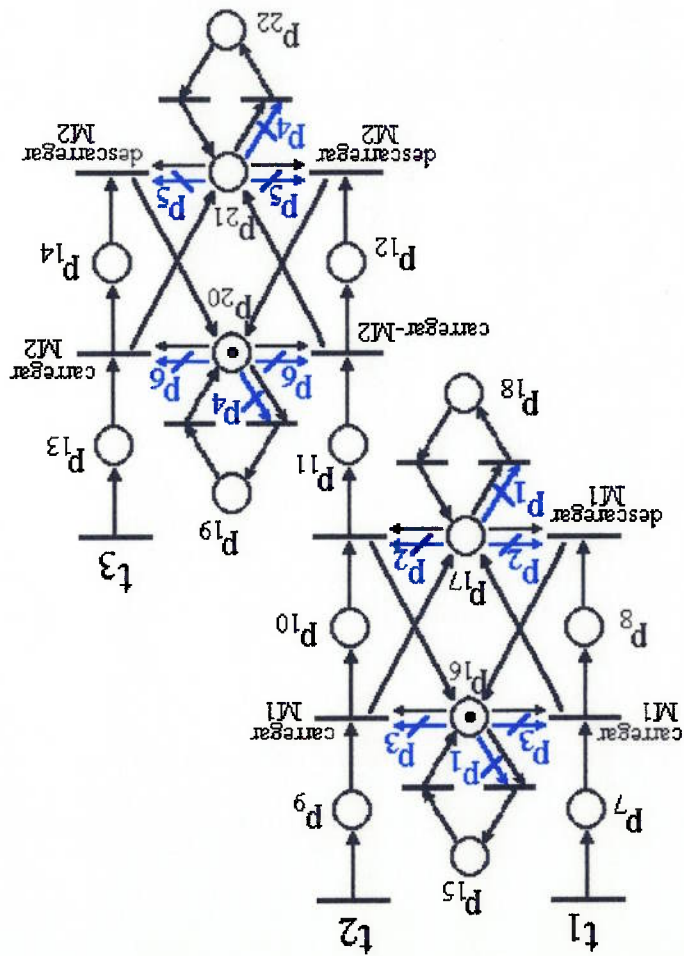
A sub-rede da figura 4.2 modela o estado de cada equipamento (livre, operando, em conserto ou em manutenção) e o tipo de peça que é processado. O disparo das TRANSIÇÕES  $t_1$ ,  $t_2$  e  $t_3$ , indicam a chegada de uma peça tipo  $w_1$ ,  $w_2$  ou  $w_3$ , respectivamente. O processamento destas peças é modelado numa sequência vertical que se inicia com o disparo destas TRANSIÇÕES e finaliza quando se descarrega a peça da respectiva máquina, isto é, colocando a peça na esteira, que a encaminha a sua saída. Estas peças aguardam processamento nos LUGARES  $p_7$ ,  $p_9$  e  $p_{13}$ , respectivamente (que modelam os "buffers" de entrada das máquinas), até que os recursos do sistema estejam disponíveis para processá-las. O LUGAR  $p_{11}$  representa um "buffer" intermediário que comporta as peças tipo  $w_2$  que já foram processadas por  $M1$  e que aguardam processamento em  $M2$ . Os LUGARES  $p_{15}$ ,  $p_{16}$ ,  $p_{17}$ , ... ,  $p_{22}$  modelam o estado dos recursos  $M1$  e  $M2$  ("M1 EM MANUTENÇÃO", "M1 LIVRE", "M1 OPERANDO NORMALMENTE", ... , "M2 EM CONSERTO", respectivamente).

As TRANSIÇÕES que iniciam e finalizam o processo de "CORREÇÃO DE ERROS" (LUGARES  $p_{18}$  e  $p_{22}$ ) e ("MANUTENÇÃO" ( $p_{15}$  e  $p_{19}$ )) são conectadas aos LUGARES "MÁQUINA TRABALHANDO" (LUGARES  $p_{12}$  e  $p_{21}$ ) e "MÁQUINA LIVRE" ( $p_{16}$  e  $p_{20}$ ), para representar simplificadaamente, a forma de recuperação do sistema após a execução da ação corretora. Esta ação pode ser executada interrompendo o processamento de uma peça (no caso de diagnóstico de falha/erros) ou aguardando até a conclusão deste processamento (diagnóstico de manutenção).

O disparo de todas as TRANSIÇÕES exceto as TRANSIÇÕES de "INICIAR/FINALIZAR-CONSERVO/MANUTENÇÃO" requerem a disponibilidade do robô para estarem habilitadas.

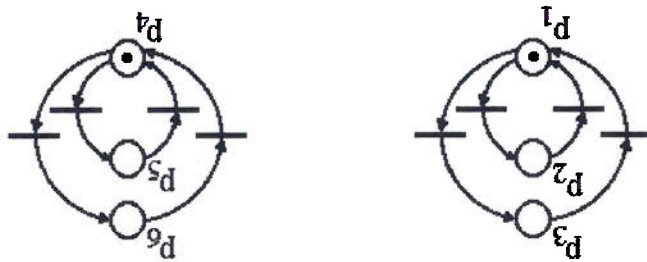
Fig. 4.2 - Modelo em RPAM dos processos de fabricação do sistema produtivo.

Lugar	Descrição
$p_1$	peça $w_1$ aguardando ser carregada em M1
$p_2$	processamento da peça $w_1$ em M1
$p_3$	peça $w_2$ aguardando ser carregada em M1
$p_4$	processamento da peça $w_2$ em M1
$p_5$	peça $w_3$ aguardando ser carregada em M2
$p_6$	processamento da peça $w_2$ em M2
$p_7$	peça $w_3$ aguardando ser carregada em M2
$p_8$	processamento da peça $w_2$ em M2
$p_9$	peça $w_3$ aguardando ser carregada em M2
$p_{10}$	processamento da peça $w_2$ em M2
$p_{11}$	peça $w_2$ aguardando ser carregada em M2
$p_{12}$	processamento da peça $w_2$ em M2
$p_{13}$	peça $w_3$ aguardando ser carregada em M2
$p_{14}$	processamento da peça $w_3$ em M2
$p_{15}$	M1 em manutenção
$p_{16}$	M1 livre
$p_{17}$	M1 processando
$p_{18}$	M1 em conserto
$p_{19}$	M2 em manutenção
$p_{20}$	M2 livre
$p_{21}$	M2 processando
$p_{22}$	M2 em conserto



Assim, considerar o requerimento deste recurso só incrementa a complexidade do modelo e dificulta a focalização do presente estudo de tratamento de falhas/erros, por esta razão foi omitido no modelo.

A figura 4.3 modela a estrutura lógica que relaciona os três estados que podem ser monitorados/diagnosticados nos equipamentos do sistema produtivo. As duas sub-redes desta figura modelam os estados monitorados baseados nos dados dos sensores vinculados a cada máquina e/ou o diagnóstico fornecido por algum outro subsistema, por exemplo, utilizando técnicas de Inteligência Artificial e/ou um modelo analítico ou utilizando uma RP-procedural (BPN) descrita no anexo A.1.



Lugar	Descrição
p <sub>1</sub>	M1 operando normalmente
p <sub>2</sub>	M1 com falha
p <sub>3</sub>	M1 requer manutenção
p <sub>4</sub>	M2 operando normalmente
p <sub>5</sub>	M2 com falha
p <sub>6</sub>	M2 requer manutenção

Fig. 4.3 - Modelo em RPA de estado de operação das máquinas.

Os estados de cada equipamento ("USINAGEM EM ESTADO NORMAL" ou "NECESSÁRIO CONserto/MANUTENÇÃO") representados na figura 4.3 foram modelados em RP para facilitar a integração com o modelo do sistema produtivo da figura 4.2, já que estas sub-redes modelam os estados de operação de cada equipamento e uma representação do mesmo tipo simplifica a implementação de todo o conjunto (sistema de monitorização/



diagnóstico de falhas + sistema de controle de fluxo de processos) num simulador, bem como facilita o desenvolvimento de um simulador específico, para propósitos de análise e validação.

#### 4.1.1 Modos Alternativos de Operação das RPAM

Os ARCOS auto-modificáveis associados às marcações dos LUGARES  $p_1, p_2, \dots, p_6$  ("MI-NORMAL", "MI-COM FALHA", ... e "M2-PRECISA MANUTENÇÃO", respectivamente, descritos na tabela da figura 4.3) eliminam os conflitos nos LUGARES  $p_{16}, p_{17}, p_{20}$  e  $p_{21}$  da figura 4.2, restringindo o disparo das respectivas TRANSIÇÕES, através do aumento do número de condições necessárias para dispará-las; estas condições são estabelecidas segundo o estado monitorado/diagnosticado em cada máquina. Por exemplo, se M1 estiver operando normalmente (MARCA em  $p_1$ ), as TRANSIÇÕES "CARREGAR-M1" e "DESCARREGAR-M1" poderão ficar habilitadas e as TRANSIÇÕES "INICIAR MANUTENÇÃO/CONSRTO EM M1" (ligadas aos LUGARES  $p_{16}$  e  $p_{17}$ , respectivamente) ficam inibidas (veja figura 4.4, onde os ARCOS auto-modificáveis relacionados com a marcação dos LUGARES  $p_2$  e  $p_3$  ficam apagados).

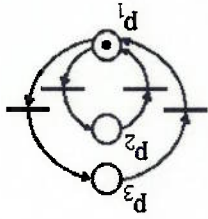
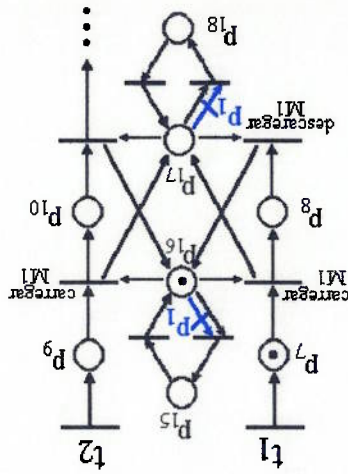


Fig. 4.4 - Modo alternativo de operação do modelo da figura 4.2 (M1 operando normalmente).

No caso de acontecer uma falha em M1 (MARCA em p<sub>2</sub>), a estrutura do modelo em RPAM é alterada para outro modo de operação, interrompendo o processamento da peça e fazendo com que a peça não seja descarregada (TRANSIÇÃO "DESCARRREGAR-M1" inibida) e que seja iniciado o conserto (TRANSIÇÃO "INICIAR CONSERTO" habilitada). Este modo de operação é ilustrado na figura 4.5-a) (com os ARCOS auto-modificáveis p<sub>1</sub> e p<sub>3</sub> apagados). A figura 4.5-b) (com os ARCOS auto-modificáveis p<sub>1</sub> e p<sub>2</sub> apagados) ilustra uma nova estrutura do modelo caso seja diagnosticada a necessidade de uma manutenção em M1.

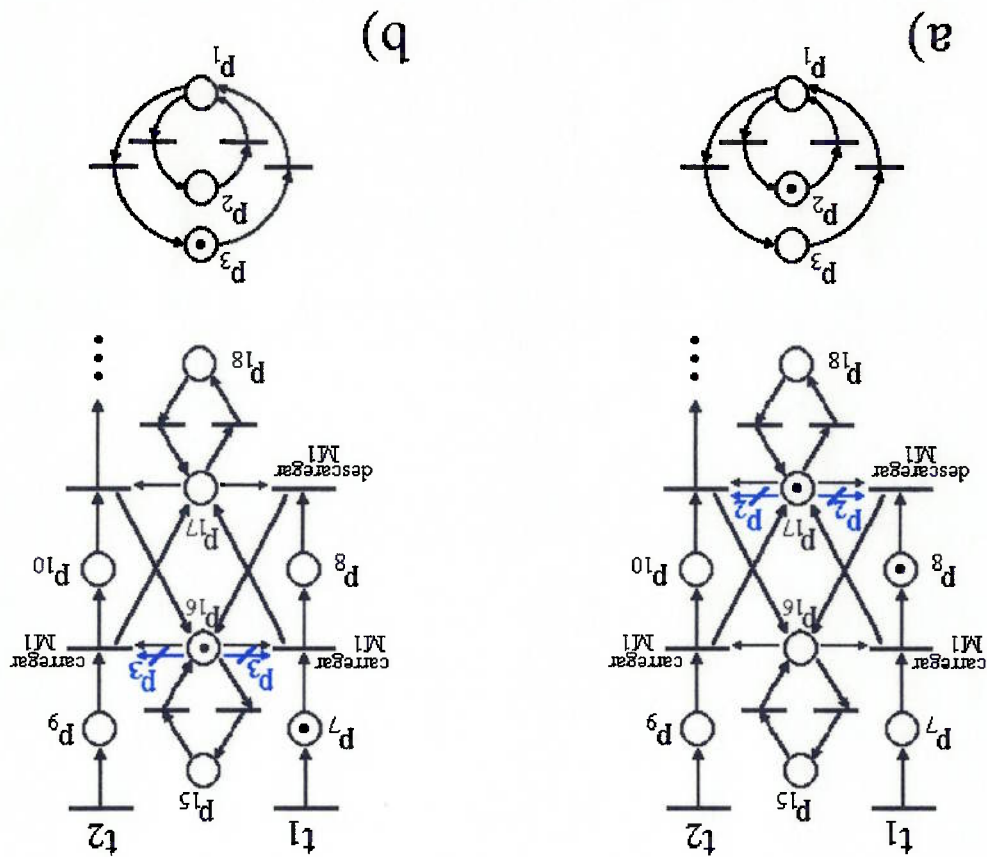


Fig. 4.5 - Modo alternativo de operação. a) M1 com falha. b) M1 requer manutenção.

O processo de conserto e recuperação dos equipamentos é modelado na figura 4.2 através do método da entrada condicionada (ver seção 3.2), cada um representado por um LUGAR apenas (LUGARES p<sub>18</sub> e p<sub>22</sub> no caso de conserto, p<sub>15</sub> e p<sub>19</sub> no caso de manutenção),

porém, tal processo não especifica qual a estratégia de execução de reparo que se utilizaria caso ocorresse uma falha/erro conforme discutido na seção 2.3, por exemplo:

1. Através de um agente externo no equipamento, por exemplo: um operador que identifica e elimina a falha.
2. Um equipamento autônomo para tratamento de falhas/erros, que executa o reparo através de:

- Ajuste de parâmetros
- Redundância funcional

#### 4.1.2 Equivalência da RPAM com outros tipos de RP

Na seção 3.3.2 é apresentada uma equivalência das condições inibidoras e habilitadoras nos vários tipos de RP. Como exemplo de um problema prático, ilustra-se na figura 4.6 outro modelo equivalente, utilizando uma RP com ARCOS inibidores com as mesmas características do sistema modelado em RPAM representado nas figuras 4.4 e 4.5. Nesta RP a marcação dos LUGARES que indicam os estados de operação dos equipamentos (LUGARES  $p_9$ ,  $p_{10}$  e  $p_{11}$ ) condicionam o disparo de TRANSIÇÕES, permitindo, assim, a continuidade dos processos de usinagem (condicionadas por ARCOS inibidores) ou permitindo o conserto/manutenção das máquinas (condicionadas por ARCOS habilitadores).

Este tipo de RP não permite a “disjunção”<sup>[1]</sup> do modelo do sistema ou a representação da monitorização/diagnóstico do estado dos equipamentos, como acontece nas RPAM. A representação destes estados pode tornar-se muito mais complexa que a ilustrada na figura 4.3, dificultando, assim, o acompanhamento do estado atual de cada equipamento, já que este é função dos estados individuais de cada parte do sistema.

[1] A disjunção da RPAM é gráfica, mas as sub-redes são acopladas proceduralmente.

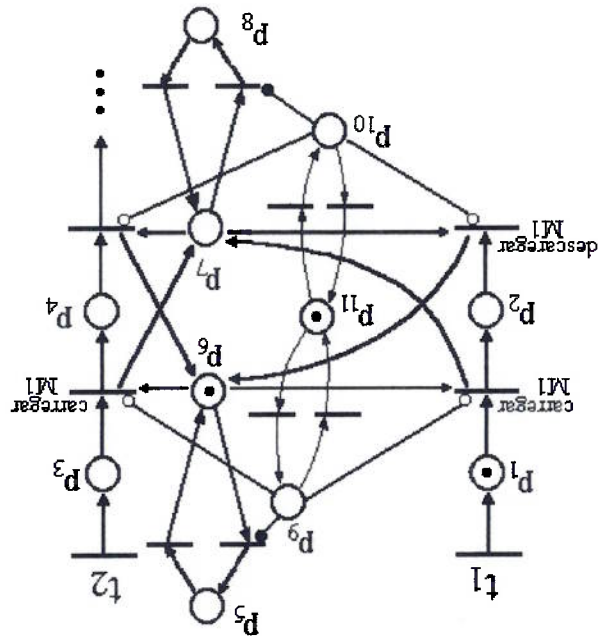


## 4.2 CASO 2

Como um segundo exemplo, considera-se agora um modelo que integra a utilização de ARCOS auto-modificáveis e ARCOS inibidores na RP. O objetivo aqui é confirmar que a integração das várias extensões na RP pode fornecer um modelo mais compacto e relativamente mais legível e que permita a análise do sistema e o desenvolvimento de testes num simulador de RP.

Fig. 4.6 - Modelo em RP com ARCOS inibidores do sistema da figura 4.1

Lugar	Descrição
$p_1$	peça $w_1$ aguardando
$p_2$	M1 processando peça $w_1$
$p_3$	peça $w_2$ aguardando
$p_4$	M1 processando peça $w_2$
$p_5$	M1 em manutenção
$p_6$	M1 livre
$p_7$	M1 processando
$p_8$	M1 em concerto
$p_9$	M1 requer manutenção
$p_{10}$	M1 requer concerto
$p_{11}$	M1 operando normalmente



Uma das atividades que com maior frequência é objeto de automação é a troca de peças e ferramentas (apêndice II). O exemplo aqui considerado ilustra como as RPAM podem ser utilizadas na descrição do tratamento de falhas/erros que ocorrem com maior frequência num sistema de montagem/desmontagem de peças composto por 1 robô manipulador e uma máquina CNC. Na figura 4.7, ilustra-se o sistema produtivo considerado, composto por um robô IIRB-6 ASEA (hoje ABB) de 6 graus de liberdade de rotação com uma garra com 2 pinças que permitem segurar e transportar peças que são usinadas na máquina CNC do tipo MAZATROL T32-2.

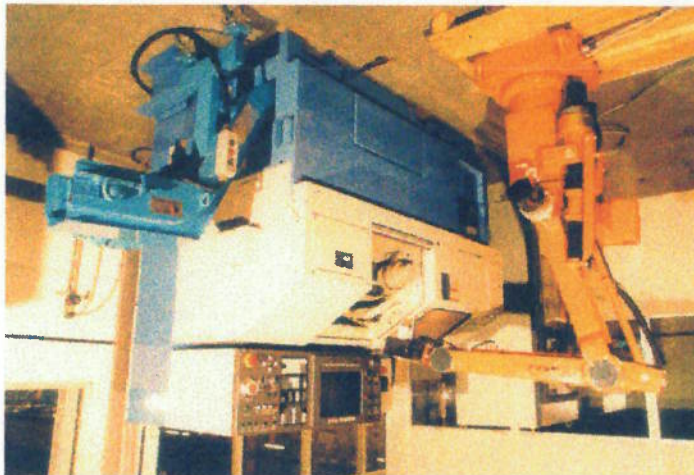


Fig. 4.7 - Sistema de manufatura composto por um Robô e uma máquina CNC.

A figura 4.8 [Riascos, 98] e a tabela 4.1 ilustram um modelo em RPAM do tipo SPSC utilizando ARCOS inibidores que modelam o estado de processamento de uma peça numa sequência vertical que inicia quando o robô agarra uma matéria-prima que é posicionada na máquina e finaliza quando o robô descarrega a peça já processada (LUGARES  $P_5$  e  $P_{11}$ , respectivamente). Os processos destes 2 equipamentos (robô e máquina CNC) são desenvolvidos em 3 etapas:

1. O robô R carrega as peças na máquina M.
2. A máquina M usina as peças.

3. R descarrega as peças já processadas da máquina M. Não foram consideradas falhas nos "buffers" de entrada e de saída para simplificar o modelo estudado.

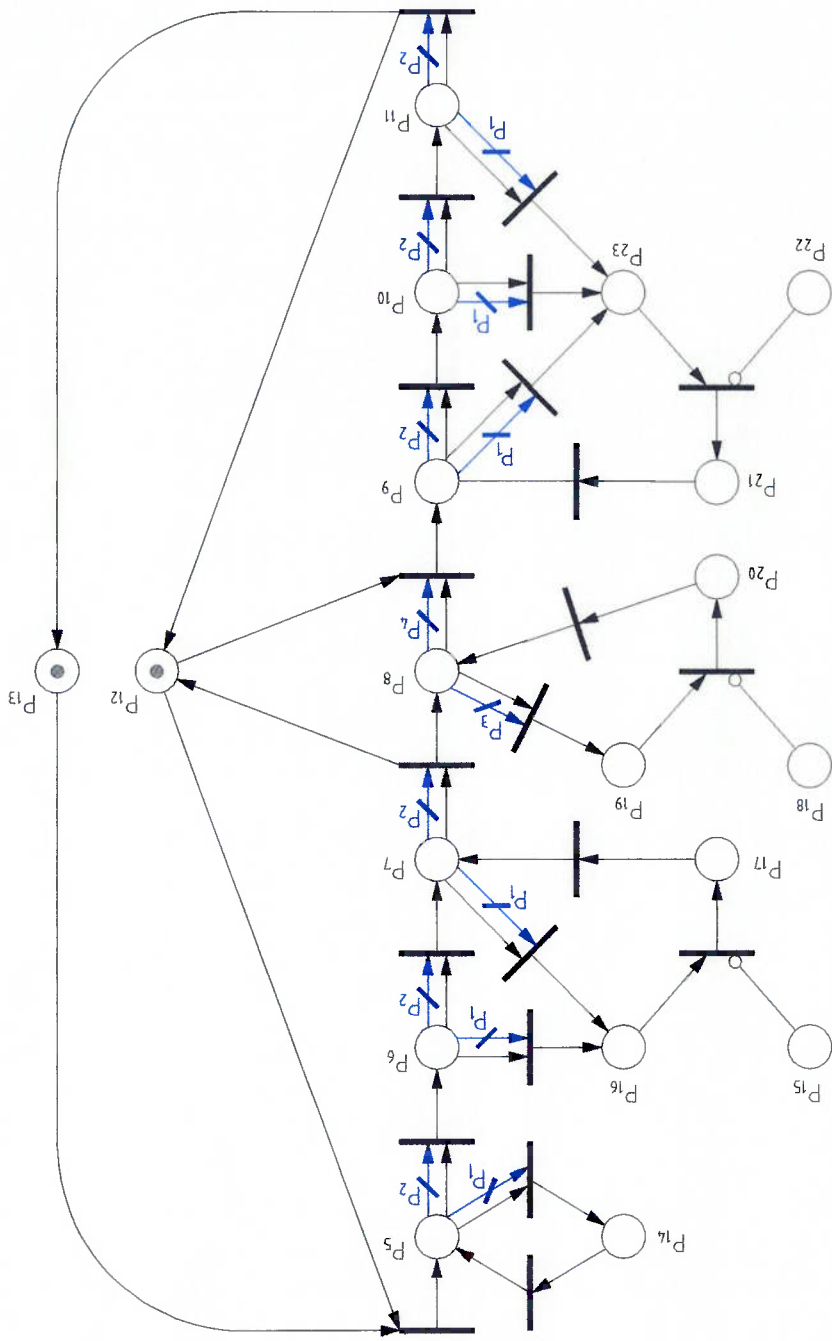


Fig.4.8 - Modelo em RPA do sistema robô/máquina

A rede modela o processo de carregamento (e descarregamento) de peças numa sequência de três atividades (agarrando, transportando e descarregando). Neste exemplo, somente foram consideradas falhas/erros por escorregamento parcial e/ou perda total da peça, que podem ocorrer por diversas causas, entre elas:

- Posicionamento incorreto da peça, o que não permite um agarramento estável.
- Escorregamento da peça por aplicação de uma força inadequada na garra e/ou na peça.
- Colisões inesperadas entre a peça/garra com algum obstáculo que possa produzir um escorregamento parcial ou completo da peça.
- Posicionamento incorreto da peça entre a placa de castanha e o contraponto da máquina que não permita a usinagem da peça e exija o seu reposicionamento.

No processo de usinagem na máquina **M** (modelado por um macro LUGAR,  $p_8$ ) foi considerada a falha de quebra de ferramenta.

Lugar	Transição
$p_5$	R agarrando matéria-prima
$p_6$	R transportando matéria-prima
$p_7$	R descarregando matéria-prima em M
$p_8$	M processando
$p_9$	R agarrando peça pronta
$p_{10}$	R transportando peça pronta
$p_{11}$	R descarregando peça pronta
$p_{12}$	R livre
$p_{13}$	M livre
$p_{14}$	redireccionando a garra de R
$p_{15}$	acesso impossível até a peça
$p_{16}$	verificando nova posição da matéria-prima
$p_{17}$	executando nova trajetória
$p_{18}$	ferramenta não disponível
$p_{19}$	descartando velha ferramenta
$p_{20}$	colocando nova ferramenta
$p_{21}$	executando nova trajetória
$p_{22}$	acesso impossível até a peça
$p_{23}$	verificando nova posição da peça

Tabela 4.1 - Descrição dos LUGARES da figura 4.8.



Os LUGARES  $p_5, p_6, \dots, p_{11}$  modelam cada estado da sequência de processamento da peça. Para cada um destes LUGARES, foi considerada uma estratégia de recuperação (entrada condicionada, rota alternativa, etc.) que representa a forma adequada de tratamento segundo o tipo de erro que ocorrer. Os LUGARES  $p_{14}, p_{15}, \dots, p_{23}$  modelam os estados do sistema no caso de execução de um processo de recuperação de falhas/erros. A tabela 4.2 descreve o esquema de recuperação da falha/erro utilizado em cada LUGAR.

A pré-definição da forma de tratamento de cada falha/erro evita a necessidade de execução de um diagnóstico da causa da falha/erro e da proposição de um plano de conserto em cada caso (veja seção 2.3). Assim, são considerados a monitorização [Srinivasan, 93], o julgamento do estado de operação (normal/falha) e a execução do procedimento de recuperação da falha/erro, segundo o método proposto.

Tabela 4.2 - Esquema de recuperação utilizado.

Etapa	Lugar	Esquema utilizado
Colocar peça	- transportando	- entrada condicionada
	- descarregando	- rota alternativa
Usinagem	- usinando	- entrada condicionada
	- agarrando	- entrada condicionada
Retirar peça	- transportando	- "backward"
	- descarregando	- "backward"

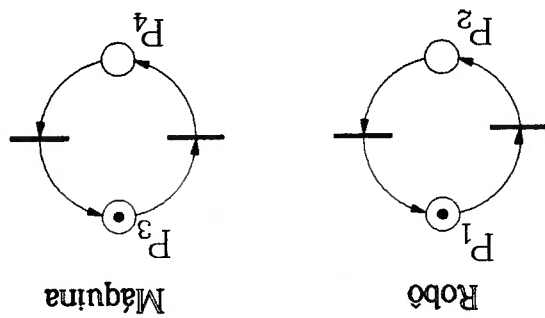
As sub-redes da figura 4.9 modelam os estados de operação dos equipamentos. A marcação dos LUGARES destas sub-redes define o peso dos ARCOS auto-modificáveis da rede da figura 4.8, eliminando os conflitos nos LUGARES correspondentes, como  $p_5, p_6, \dots, p_{11}$ . Da mesma forma que no caso anteriormente estudado, estas sub-redes modelam o estado monitorado pelos sensores dos equipamentos, que também podem ser auxiliados, por exemplo, por um sistema especialista para detecção e isolamento de uma falha, permitindo que a rede da figura 4.8 possua modos alternativos de operação baseados nos estados de cada equipamento.

A seqüência dos processos modelados pelos LUGARES  $P_5, \dots, P_{11}$  da figura 4.8, representa a seqüência dos processos tradicionalmente modelados nos sistemas de manufatura, ou seja, considerando o funcionamento de um sistema ideal, sem nenhum tipo de falha. Para cada estado de funcionamento "normal", foi considerado um estado "anormal", Assim, para cada estado de funcionamento "normal", foi considerado um estado "anormal", como pode verificado (figura 4.8).

Para implementar um sistema com estas características é necessário considerar um dispositivo adicional aos sensores do robô e da máquina CNC, por exemplo, um dispositivo de visão artificial (com uma câmera CCD) que localize a peça caso esta caia da garra e verifique se a posição final pode ser atingida pela garra do robô.

Fig. 4.9 - Modelo em RP do estado dos equipamentos.

Lugar	Descrição
$P_1$	Robô operando normalmente
$P_2$	Robô segurando peça incorretamente
$P_3$	Máquina operando normalmente
$P_4$	Falha na ferramenta



#### 4.3 IMPLEMENTAÇÃO E TESTE NO SIMULADOR DE RPAM.

Foi utilizado um simulador de RPAM desenvolvido no departamento de Engenharia Mecânica/Mecatrônica da EPUSP baseado num programa desenvolvido em linguagem C para ambiente de DOS/WINDOWS em microcomputador tipo IBM-PC. Este simulador permite trabalhar com várias sub-redes disjuntas, cada uma das quais modelando uma característica do sistema e permitindo estabelecer um vínculo entre a marcação dos LUGARES de uma rede e o peso de ARCOS das outras redes desenhadas. Assim, o programa permite simular simultaneamente várias redes para propósitos de análise e validação.

Para analisar um modelo neste simulador, primeiro desenha-se a rede utilizando os recursos gráficos disponíveis. No modelo desenhado a análise é realizada disparando (automaticamente ou manualmente) TRANSIÇÕES habilitadas da rede, simulando, assim, o comportamento dinâmico do sistema e permitindo, com isto, verificar graficamente a possibilidade de atingir certos estados. Na forma automática, estas TRANSIÇÕES podem ser disparadas utilizando taxas aleatórias ou determinísticas. Outra vantagem oferecida é que a simulação pode ser temporariamente interrompida para alterar arbitrariamente a marcação e verificar outras alternativas na evolução dos estados do sistema.

Para ilustrar a utilização deste simulador, considera-se o processo de leitura e escrita utilizando um arquivo comum de dados, descrito na seção 3.3.1. A figura 4.10 ilustra um "hard copy" da tela apresentada no monitor do microcomputador. A sub-rede à esquerda modela os processos de leitura e a sub-rede à direita, os processos de escrita. Este modelo apresentou na simulação um comportamento adequado, confirmando a dinâmica esperada pelo sistema.

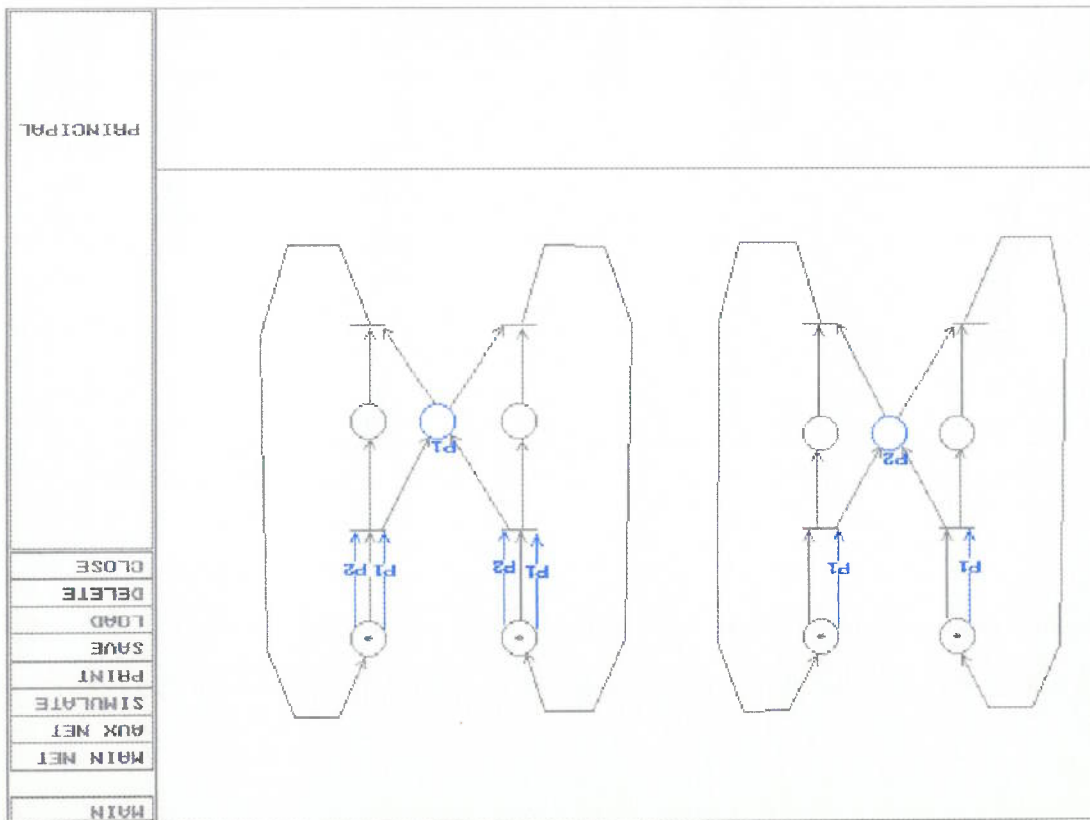


Fig. 4.10 - RPAM modelando os processos de leitura e escrita.

Considera-se agora o caso estudado na seção 4.1. Na figura 4.11, é apresentado um "hard copy" da tela com a RPAM do tipo SPSC que modela os processos de usinagem e as sub-redes que modelam os estados de operação dos equipamentos e cuja marcação define os pesos de ARCOS da rede principal (SPSC) da mesma forma que nas figuras 4.2 e 4.3.

A figura 4.12 ilustra o caso estudado na seção 4.2 e apresenta um "hard copy" da tela com o modelo do sistema (modelando a estrutura do sistema) e as sub-redes que modelam o estado de operação de seus equipamentos, da mesma forma que as figuras 4.8 e 4.9.



Fig. 4.12 - RPAM do sistema da figura 4.7.

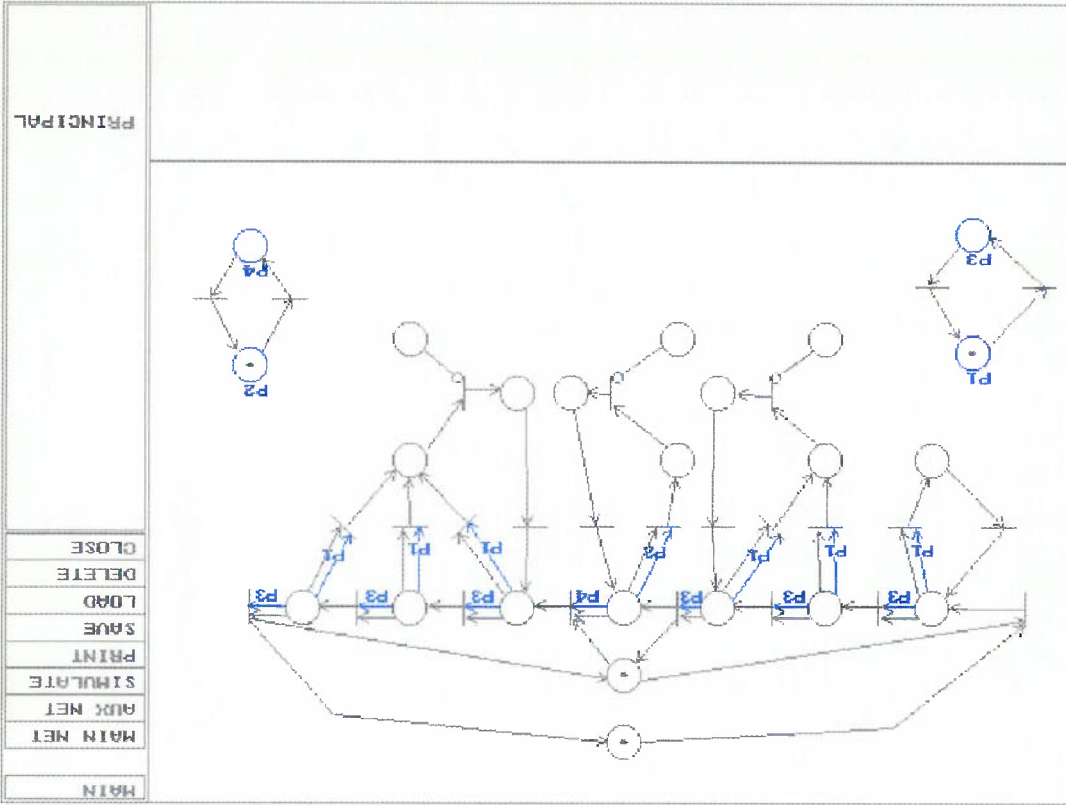
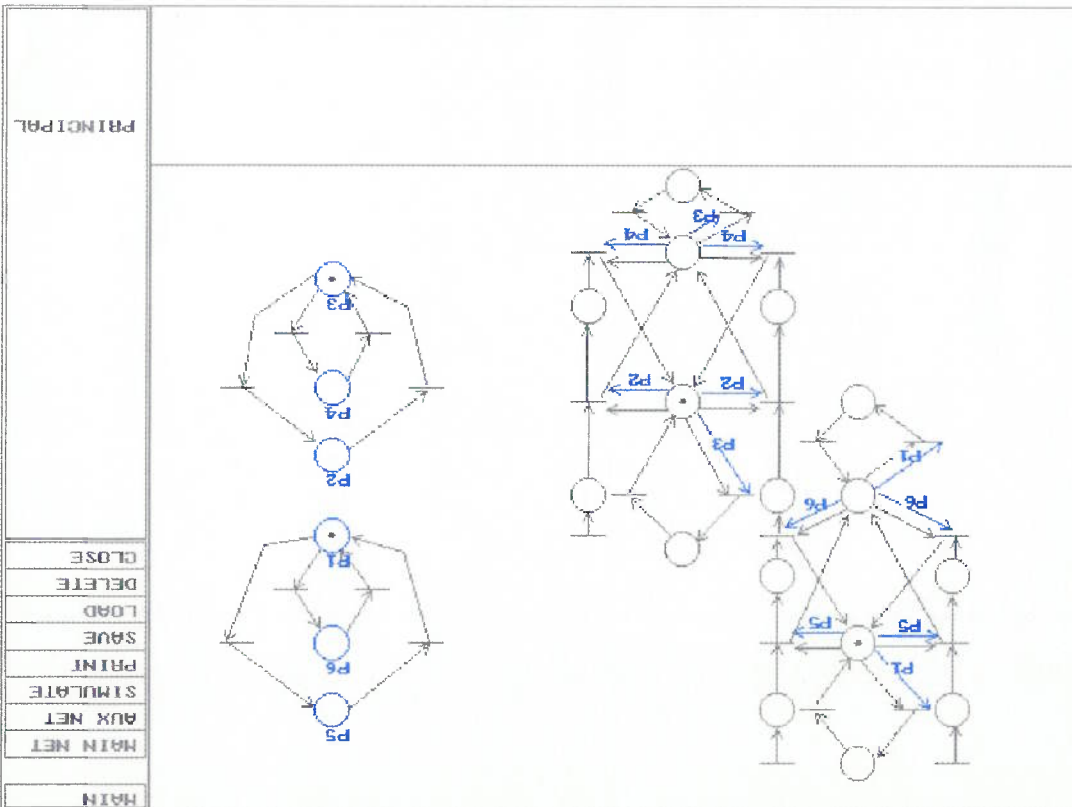


Fig. 4.11 - RPAM modelando os processos do sistema da figura 4.1



#### 4.4 ANÁLISE DE RESULTADOS

Neste capítulo, foram apresentados exemplos de sistemas de manufatura modelados através de RPAM, nos quais foi dada especial ênfase à representação do tratamento de falhas/erros e procedimentos de manutenção baseados nos métodos apresentados na seção 3.2. Estes métodos garantem a manutenção das propriedades do modelo (segurança, vivacidade, etc.) quando do acréscimo de sub-redes para tratar situações anormais (como falhas/erros) num modelo que originalmente foi projetado sem estas características.

A RPAM mostrou ser adequada para modelar sistemas com estas características e apresenta vantagens para resolver conflitos, como no caso da introdução de múltiplas escolhas nas seqüências dos processos (por exemplo: continuidade no processamento de uma peça ou inicialização do tratamento de falha/erros). Já que a introdução de ARCOS auto-modificáveis foi realizada para aumentar o número de condições necessárias para disparar uma TRANSIÇÃO. Assim, são projetados modelos livres de escolha.

Também foram apresentados os modelos utilizados para simulação, realizada com propósito de verificação da estrutura lógica das redes desenvolvidas para a análise dos sistemas modelados e que também confirmaram a legibilidade e acompanhamento da evolução do sistema através da descrição dos estados pela marcação do grafo.

Foi observado no modelo da figura 4.11 que, conforme esperado, quando disparadas todas as TRANSIÇÕES possíveis (habilitadas) de forma aleatória, então saturam-se rapidamente os "buffers" de entrada (LUGARES  $P_7$ ,  $P_9$ ,  $P_{11}$  e  $P_{13}$ ), já que as TRANSIÇÕES de entrada destes LUGARES não possuem nenhuma condição restritiva, contrariamente às outras TRANSIÇÕES que dependem do estado de operação do respectivo equipamento.

No modelo da figura 4.12, a situação anteriormente descrita não ocorre, uma vez que o disparo da TRANSIÇÃO que inicia o processamento requer a disponibilidade do recurso da máquina-ferramenta (M, recurso único) que somente é liberado quando é finalizado o processamento (robô descarregou uma peça já processada). Assim, somente é permitido o processamento de uma peça de cada vez.

A metodologia utilizada mostrou ser adequada para modelar especificações de operação adicionais (tratamento de falhas/erros) num modelo de um sistema que inicialmente foi projetado sem considerar estas características. Neste contexto desta parte do trabalho desenvolveu-se uma metodologia para tratamento de tipos específicos de falhas/erros em sistemas de manufatura (por exemplo: aquelas que acontecem com maior frequência). A metodologia também mostrou ser efetiva para tratar falhas em diferentes níveis de detalhamento de um modelo em RP, como no estudo de caso da seção 4.2 (veja figura 4.8), onde as atividades modeladas na RP podem ser assumidas como uma decomposição de atividades desenvolvidas no modelo estudado na seção 4.1 (figura 4.2). Isto permite utilizar metodologias como “*top-down*” e “*bottom-up*” para modelagem do tratamento de falhas/erros em sistemas de manufatura complexos e de grande porte.

**COMENTÁRIOS FINAIS E CONCLUSÕES**

**Capítulo 5**

## Capítulo 5

### COMENTÁRIOS FINAIS E CONCLUSÕES

Este trabalho introduz uma técnica (RPAM) para modelagem de sistemas de manufatura através de uma abordagem de Sistemas Dinâmicos a Eventos Discretos (DEDS) e de redes de Petri (RP) considerando a ocorrência de falhas e erros de operação e os respectivos métodos de recuperação.

Os sistemas aos quais foi dedicada especial ênfase foram os sistemas de manufatura nos quais requisitos como aumento na produtividade e qualidade tentam ser atendidos através da automação de tarefas repetitivas tradicionalmente realizadas por operadores humanos sem que isto implique em perda de flexibilidade. Entretanto, este incremento na automação deve respeitar regras básicas para manter um equilíbrio com respeito à integração de atividades desempenhadas por máquinas e por operadores humanos. Assim foram também abordados os Sistemas Antropocêntricos de Produção (APS) como um conceito de sistemas de produção centrados no homem, e os Sistemas de Produção Balançada (BAS), que mantêm um nível adequado de produção/flexibilidade e o equilíbrio entre os recursos humanos, tecnológicos e sócio-econômicos. Assim é evidenciado que, principalmente com a inclusão dos diferentes aspectos técnicos e humanos, a ocorrência de falhas e erros é inerente ao sistema.

Nestes sistemas de manufatura, foi abordado o problema de ocorrência e tratamento de falhas e erros de operação, pois a modelagem destas situações geralmente não são consideradas por técnicas convencionais. Mesmo na abordagem de DEDS e das técnicas de redes de Petri, estas situações exigem incremento do número de elementos no modelo devido ao acréscimo de sub-redes que descrevem estas situações anormais, aumentando custos de futuras análises, além da possibilidade de criação de conflitos em alguns lugares,

especialmente onde é decidida a continuidade nos processos normais ou a execução das sub-redes que tratam e corrigem as falhas/erros.

A RP tradicional demonstra ser deficiente para representar e analisar sistemas complexos de manufatura, onde procedimentos relativos a falhas e erros são necessários. Assim foram introduzidos os conceitos de ARCOS auto-modificáveis, procurando incrementar o poder de modelagem da RP.

Esta nova rede, denominada de RPAM, foi então explorada para modelagem de sistemas de manufatura com ênfase na descrição do tratamento de falhas/erros e procedimentos de manutenção. A construção de um modelo em RPAM com estas considerações assegura a manutenção das propriedades do modelo global do sistema quando associadas à sub-rede para tratamento de situações anormais (como falhas/erros) num modelo que originalmente foi projetado sem estas características.

Utilizando o conceito de RPAM para a descrição de sistemas de manufatura, observou-se os seguintes pontos:

1. A inclusão de ARCOS com seus pesos relacionados à marcação de determinados LUGARES reduz a necessidade de definir novas condições (LUGARES), o que simplifica o grafo.

2. A utilização da marcação de LUGARES específicos para definir o peso ( $k$ ) de determinados ARCOS facilita a sincronização de atividades entre os recursos do sistema e a solução de conflitos, necessária quando existem casos de múltipla escolha. São passos indispensáveis no estudo e projeto destes sistemas.

3. A utilização de RPAM para a especificação do controle de sistemas considerando procedimentos de manutenção e tratamento de falhas/erros pode ser realizada de forma relativamente natural, evitando, assim, a explosão do número de nós necessários para modelar estes procedimentos.

4. A utilização de ARCOS auto-modificáveis permite resolver os conflitos criados nos LUGARES onde deve ser realizada uma escolha. Nos casos práticos como por exemplo os analisados no presente trabalho, a escolha de uma opção é realizada segundo o julgamento do estado de operação dos equipamentos (estado de operação normal ou de falha) e decidida a continuidade da execução dos processos normais ou a iniciação da execução da sub-rede com os procedimentos que recuperam a falha/erro.

5. A RPAM permite trabalhar redes graficamente “disjuntas” (mas acopladas proceduralmente), como por exemplo nos casos das redes que modelam os sistemas estudados no capítulo 4, onde uma rede é utilizada para modelar a sequência lógica dos processos executados pelos equipamentos e outras redes para modelar o estado de operação de cada equipamento. Assim, a legibilidade de várias sub-redes pode ser relativamente mais fácil que a legibilidade de uma rede complexa que modela todo o sistema.

6. A utilização de redes exclusivamente para modelar o estado de operação dos equipamentos facilita a monitorização/diagnóstico dos mesmos, que podem ser executados, por exemplo, por um sistema especialista ou por uma RP-procedural (BPN) (Anexo A.1).

7. A utilização do mesmo formalismo para representar o estado de operação de cada equipamento e os processos do sistema facilita a integração dos modelos. Esta

integração é fundamental na implementação de todo o conjunto (sistema de monitorização/diagnóstico de falhas + sistema de controle de fluxo de processo) num simulador, para propósitos de análise e validação como realizado na secção 4.3.

Como foi confirmado no presente trabalho, a RPAM demonstra grande potencial para modelagem de sistemas sujeitos a alterações estruturais, como no caso de ocorrência de falhas e erros de operação. Também se pode estabelecer um paralelo entre a modelagem de DEDS através de RPAM, com sua capacidade de alterar a estrutura do grafo, e o controle adaptativo realimentado de CVDS, que alteram em tempo real os parâmetros de funcionamento da planta como uma forma de adaptar-se às diferentes condições de operação do sistema. Esta analogia permite guiar o desenvolvimento de futuras pesquisas em DEDS baseadas em estudos, já realizados, em CVDS.

Prende-se continuar o estudo desta metodologia para desenvolver técnicas de análise e projeto destes sistemas vinculando a detecção e o diagnóstico de falhas através de modelos também desenvolvidos em RP ou RPAM.

Não foi objeto principal do presente trabalho mas, os estudos desenvolvidos em RPAM, consideram também os modelos em RP exclusivamente com ARCOS inibidores/habilitadores (sem ARCOS auto-modificáveis) e RP colorida (Anexo A.2) para modelagem das mesmas características dos sistemas apresentados nos capítulos 3 e 4. A equivalência estabelecida pode ser útil para uma futura análise de RPAM vinculadas a outro tipo de RP ou a utilização de ferramentas analíticas ou de simulação desenvolvidas para estes tipos de RP.



Também pretende-se continuar a pesquisa com o propósito de desenvolver controladores de propósito geral cujas rotinas de controle sejam geradas a partir de modelos de RPAM.

**ANEXO**

Este anexo apresenta inicialmente uma extensão da RP que permite modelar o diagnóstico de um sistema utilizando o formalismo da rede, ampliando, assim, os paradigmas estabelecidos e ilustrando outras possibilidades na utilização da RP para o tratamento de falhas, erros e procedimentos de manutenção em sistemas de manufatura.

#### A.1 RP-PROCEDURAL PARA DETECÇÃO E DIAGNÓSTICO (BPN).

Uma rede de Petri pode ser utilizada para modelar o procedimento de um sistema a ser diagnosticado desde que nenhum processo recursivo necessite ser considerado. Instâncias particulares de um estado podem ser representadas por LUGARES, e relações de causa/efeito entre estas instâncias podem ser representadas por TRANSIÇÕES. Assim, uma RP pode ser utilizada para definir a estrutura do processo de raciocínio num sistema que relaciona causas e efeitos. De fato, uma base de conhecimento pode ser utilizada para resolver o problema do diagnóstico através da utilização da árvore ou do grato de alcançabilidade da rede.

Em [Portinale, 97] é apresentada uma abordagem para realizar o diagnóstico de um sistema baseado numa representação através de RP. Para isto utiliza-se uma rede condição/evento (segura) na qual são acrescentados novos elementos, definindo  $BPN = \{P, T, I, O\}$ , onde:

- P é um conjunto finito de LUGARES. O estado inicial do sistema está representado nos LUGARES iniciais e as conclusões nos LUGARES finais. Entretanto, é necessário que todo LUGAR da rede ( $p_i \in P$ ) tenha apenas um ARCO de entrada e um ARCO de saída.

$\forall p \in P$ , tem-se:  $|p| \leq 1$  e  $|p'| \leq 1$ , onde:  $p(p')$  são as pré-transições (pós-transições) de  $p$ .

$\forall p_1, p_2 \in P$ , se  $p_1 = p_2$  e  $p_1' = p_2'$  então  $p_1 = p_2$ .

• I : função que define os ARCOS de entrada das TRANSIÇÕES.

• O : função que define os ARCOS de saída das TRANSIÇÕES.

•  $T = T_n \cup T_{or}$ , onde  $T_n \cap T_{or} = \emptyset$ .  $T_n$  é uma TRANSIÇÃO como normalmente definida

numa RP e  $T_{or}$  é um novo tipo de TRANSIÇÃO que modela situações na qual uma

instância particular pode ser determinada por causas alternativas, enquanto é mantida a

segurança da rede. A equivalência entre as TRANSIÇÕES  $T_{or}$  e  $T_n$  é ilustrada na figura

A.1.1. Nesta representação não são permitidas TRANSIÇÕES ( $T_n$  ou  $T_{or}$ ) com múltiplas

entradas e múltiplas saídas, simultaneamente, isto é:

$\forall t \in T_n$ , tem-se: se  $|t| = 1$  então  $|t'| \geq 1$  ou se  $|t'| \geq 1$  então  $|t| = 1$ .

$\forall t \in T_{or}$ , tem-se:  $|t| = 2$  e  $|t'| = 1$ .

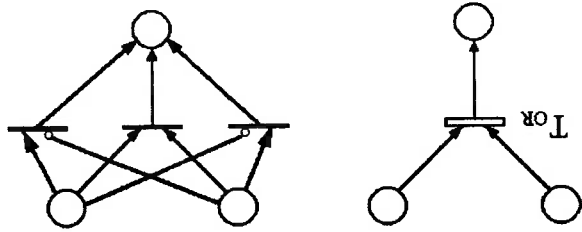


Fig. A.1.1 - Equivalência da TRANSIÇÃO  $T_{or}$  e  $T_n$ .

O problema de diagnóstico é basicamente um problema abductivo em LUGAR de um problema de predição, ou seja, deve-se realizar um processo inverso à análise de alcançabilidade. A partir de uma marcação ( $M_i$ ) obtém-se a marcação ( $M_{i-1}$ ) da qual foi alcançada, até que se cumpra o objetivo do processo que é encontrar a marcação inicial ( $M_0$ ). Para tal, além de MARCAS normais, é necessário definir um novo tipo de MARCA:

- MARCA "preta" normal ( $\bullet$ ), que representa uma condição satisfeita.
- MARCA inibidora ( $\circ$ ), que representa condições não satisfeitas. Quando um LUGAR é

marcado com este tipo de MARCA, a instância representada pelo LUGAR é ausente ou negada.

A regra de disparo inversa das TRANSIÇÕES Tor e Tn incluindo os 2 tipos de MARCAS

é ilustrada na figura A.1.2.

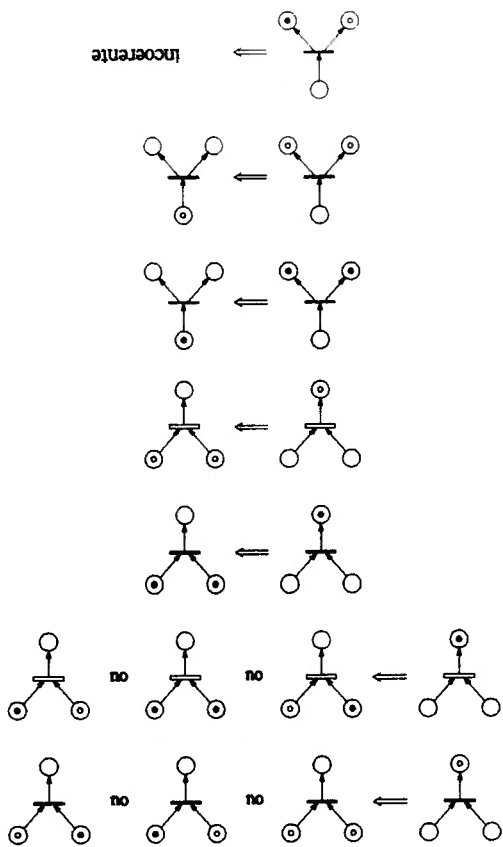


Fig. A.1.2 - Regra de disparo inversa em RP para diagnóstico.

Uma das vantagens de utilizar BPN para modelar o procedimento de um sistema a ser diagnosticado é a possibilidade de explorar o paralelismo inerente do modelo. Experiências têm comprovado uma vantagem relevante do método utilizando o diagnóstico de um sistema baseado em invariantes quando comparado com o raciocínio tradicional de lógica abductiva [Portinale, 97].

## ANEXO 2

Neste anexo são apresentados modelos em RP colorida (com ARCOS com inscrições fixas e inscrições variáveis) representando as mesmas características dos modelos das figuras 4.8 e 4.9. A equivalência existente entre estes tipos de redes possibilita uma futura análise de RPAM vinculada a RP colorida e o uso de ferramentas analíticas ou de simulação desenvolvidas para este tipo de RP.

### A.2 EQUIVALÊNCIA DA RPAM COM RP-COLORIDA.

No sistema de manufatura modelado através de RPAM, citado na seção 4.2, pode-se também verificar equivalência com modelos de outros tipos de RP com características de funcionamento similares aos múltiplos modos de operação que fornece a RPAM. Uma equivalência pode ser estabelecida e verificada com RP colorida, sendo útil para uma futura análise de RPAM vinculadas a este tipo de RP ou para utilizar ferramentas analíticas ou de simulação desenvolvidas para estas redes. Entretanto, observa-se que existem modelos de RP colorida com ARCOS com inscrições fixas e redes com ARCOS com inscrições variáveis [Miyagi, 96]. Para cada um destes casos é ilustrada, a seguir, a respectiva equivalência.

A figura A.2.1 ilustra uma RP com ARCOS com inscrições fixas equivalente à RPAM da figura 4.8 considerando os mesmos tipos de falha/erro. Nesta rede, a alteração do modo de operação é realizado pelo disparo das TRANSIÇÕES “ACONTECEU-FALHA” ou “QUEBROU-FERRAMENTA” que alteram o tipo de MARCA existente no respectivo LUGAR. A MARCA original só retorna após a execução do processo de recuperação da falha.

A figura A.2.2 ilustra um modelo equivalente através de RP colorida (RP com ARCOS com inscrições variáveis), onde a alteração do modo de operação é realizada de acordo com as regras especificadas numa tabela para os seguintes casos:

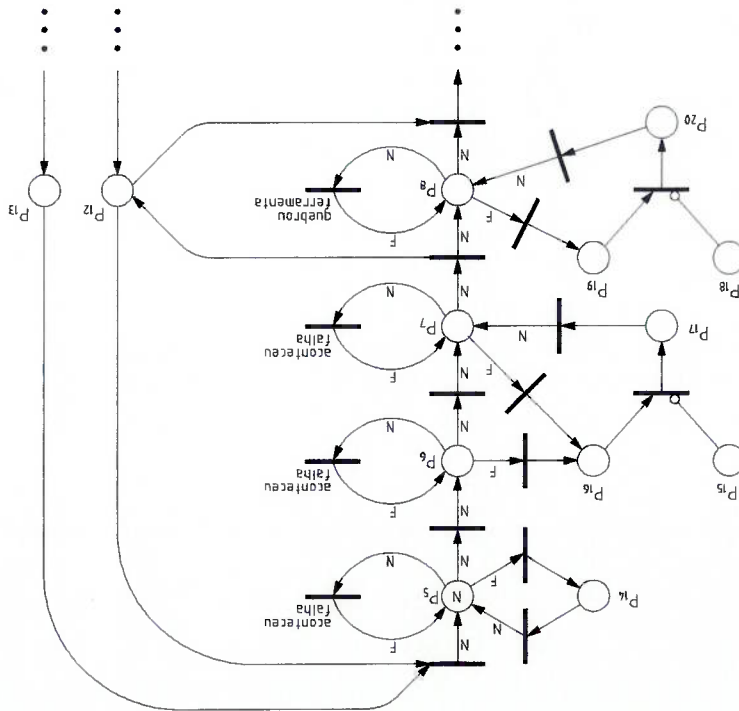
Lugar	Descrição
p <sub>5</sub>	R agarrando matéria prima
p <sub>6</sub>	R transportando matéria prima
p <sub>7</sub>	R descarregando matéria prima em M
p <sub>8</sub>	M processando
p <sub>12</sub>	R livre
p <sub>13</sub>	M livre
p <sub>14</sub>	redireccionando a garra de R
p <sub>15</sub>	acesso impossível até a peça
p <sub>16</sub>	verificando nova posição da matéria prima
p <sub>17</sub>	executando nova trajetória
p <sub>18</sub>	ferramenta não disponível
p <sub>19</sub>	trocando velha ferramenta
p <sub>20</sub>	colocando nova ferramenta

Tabela A.2.1 - Descrição dos LUGARES da figura A.2.1 e A.2.2

descrição dos LUGARES da figura 4.8.

A tabela A.2.1 fornece a descrição dos LUGARES das figuras A.2.1 e A.2.2, igual à

Fig. A.2.1 - Modelo de RP com ARCOS com inscrições fixas do sistema da figura 4.7.



- Modo de operação normal.
  - Modo de operação de falha quando a garra segura a peça.
  - Modo de operação de falha quando ocorre quebra da ferramenta.
- Estas regras são especificadas na tabela A.2.2.

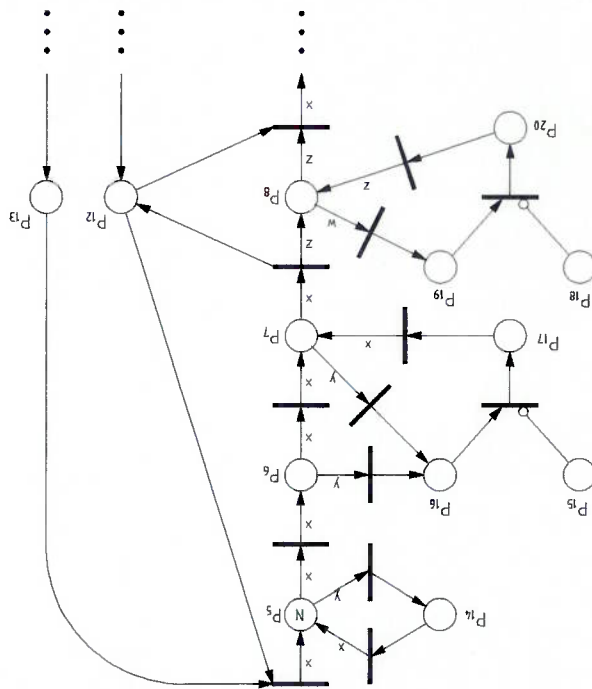


Fig. A.2.2 - Modelo em RP colorida do sistema da figura 4.7.

Tabela A.2.2 - Definição das inscrições (cores) da rede da figura A.2.2.

Inscrição	$x = N$ $y = F$ $z = N$ $w = F$
R e M operando normalmente	$x = N$ $y = F$ $z = N$ $w = F$
R apresenta falha, e M operando normalmente	$x = F$ $y = N$ $z = N$ $w = F$
R operando normalmente, e M apresenta falha	$x = N$ $y = F$ $z = F$ $w = N$



**REFERÊNCIAS BIBLIOGRÁFICAS**

## REFERÊNCIAS BIBLIOGRÁFICAS

- ADLEMO, Anders; ANDREÅSSON, Sven-Arne. Balanced automation in flexible manufacturing systems. *Studies in Informatics and Control*, with emphasis on useful applications of advanced technology, v. 5, n. 2, p. 179-187, June, 1996.
- BERGE, N.; SAMAN, M.; JUANOLE, G.; ATAMNA, J. Methodology for LAN modelling and analysis using Petri net based models. In: INTERNATIONAL WORKSHOP ON MODELLING, ANALYSIS AND SIMULATION IN TELECOMMUNICATION SYSTEMS. Durham, 1994. MASCOT '94: Proceedings. 1994. p. 335-342.
- BOSE, Bimal K. Expert system, fuzzy logic, and neural network applications in power electronics and motion control. *Proceedings of the IEEE*, v. 82, n. 8, p. 1303-1323, August, 1994.
- CAMARINHA-MATOS, Luis M.; AFSARMANESH, Hamidsh. Towards balanced automation: introduction. In: INTERNATIONAL CONFERENCE ON ARCHITECTURES AND DESIGN METHODS FOR BALANCED AUTOMATION SYSTEMS. Portugal, 1995. *Balanced Automation Systems: BASYS '95. Proceedings*. London, Chapman & Hall, 1995. p. xi-xiv.
- CAMARINHA-MATOS, Luis M. Balanced automation systems: guest editorial. *Studies in Informatics and Control*, with emphasis on useful applications of advanced technology, v. 5, n. 2, p. 87-88, June, 1996.
- CARDOSO, Janette; VALETTE, Robert. *Redes de Petri*. Florianópolis. Editora da Universidade Federal de Santa Catarina, 1997.
- CATANIA, V.; PULIAFITO, A.; VITA, L. A modeling framework to evaluate performance parameters in gracefully degrading systems. *IEEE Transactions Industrial Electronics*, v. 40, n. 5, p. 461-472, 1993.
- CHARNIAK, Eugene; MCDERMOTT, Drew. *Introductions to Artificial Intelligence*. s.l., Addison Wesley, 1987.
- CHINTAMANENI, Prasad R.; JALOTE, Pankaj; SHIEH, Yuan-Bao; TRIPATHI, Satish K. On fault tolerance in manufacturing systems. *IEEE Network*, v. 2, n. 3, p. 32-39, May, 1988.
- CHIOLA, G.; DUTHEILLLET, C.; FRANCESCINIS, G.; HADDAD, S. Stochastic well-formed coloured nets and symmetric modelling applications. *IEEE Transactions on Computers*, v. 42, n. 11, p. 1343-1360, 1993.

FRANK, Paul M. Principles of model-based fault detection. In: INTERNATIONAL SYMPOSIUM ON AI IN REAL-TIME CONTROL, Delhi, 1992. Proceedings. s.l., IFAC/IFIP/MACS, 1992. p. 363-370.

HASEGAWA, Kensuke. Modeling, control and deadlock avoidance of flexible manufacturing systems. In: CONGRESSO BRASILEIRO DE AUTOMÁTICA, CBA 11. São Paulo, 1996. Conferências Plenárias. São Paulo, SBA, 1996. p. 37-51.

KOVAČS, Ilona; MONIZ, António Brandão. Issues on the anthropocentric production systems. In: INTERNATIONAL CONFERENCE ON ARCHITECTURES AND DESIGN METHODS FOR BALANCED AUTOMATION SYSTEMS. Portugal, 1995. Balanced Automation Systems: BASYS '95. Proceedings. London, Chapman & Hall, 1995. p. 131-140.

LEVIS, Alexander H. et al. Challenges to control: a collective view. IEEE Transactions on Automatic Control, v. AC-32, n. 4, p. 275-285, April, 1987.

MIRVAGI, Paulo Eigi. Controle Programável: fundamentos do controle de sistemas a eventos discretos. São Paulo, Edgard Blücher, 1996.

MOLLOY, M. K. Performance analysis using stochastic Petri nets. IEEE Transactions on Computers, v. 3, n. 9, p. 913-917, 1982.

MURATA, Tomihito; KODOMA, Norihisa; MATSUMOTO, Kuniaki; HARUNA, Koichi. A Petri net-based controller for flexible and maintainable sequence control and its applications in factory automation. IEEE Transactions on Industrial Electronics, v. IE-33, n. 1, p. 1-8, February, 1986.

MURATA, Tadao. Petri nets: properties, analysis and applications. Proceedings of the IEEE, v. 77, n. 4, p. 541-580, April, 1989.

NEPOMUCENO, Lauro Xavier. Técnicas de Manutenção Preditiva. São Paulo, Edgard Blücher, 1989, v. 1.

PETERSON, James Lyle. Petri Net Theory and The Modeling of Systems. Englewood Cliffs, Prentice-Hall, 1981.

PORTNALE, Luigi. Behavioral Petri nets: a model for diagnostic knowledge representation and reasoning. IEEE Transactions on Systems, Man, and Cybernetics, v. 27, n. 2, p. 184-195, April, 1997. (Part B: Cybernetics).

REMBOLD Ulrich; NNAJI, B. O.; STORR, A. Computer Integrated Manufacturing And Engineering. s.l., Addison-Wesley, 1994.

RIASCOS, Luis Alberto Martínez; MIYAGI, Paulo Eigi; ARATA, Wilson Munemasa; MARUYAMA, Newton. Redes de Petri auto-modificáveis para sistemas de manufatura considerando falhas e procedimentos de manutenção. In: CONGRESSO BRASILEIRO DE ENGENHARIA MECÂNICA, 14., Bauru, 1997. COBEM '97. Proceedings. Bauru, UNESP, 1997.

RIASCOS, Luis Alberto Martínez; MIYAGI, Paulo Eigi. Modelagem do tratamento de falhas em sistemas de manufatura através de redes de Petri. In: CONGRESSO BRASILEIRO DE AUTOMÁTICA, 12., Uberlândia, 1998. Anais. São Paulo, SBA, 1998.

RIASCOS, Luis Alberto Martínez; MIYAGI, Paulo Eigi; MARUYAMA, Newton; Cozman, Fabio Gagliardi. Modeling of failures treatment in manufacturing systems through Petri nets. In: UK MECHATRONICS FORUM INTERNATIONAL CONFERENCE, 6., Skövde, 1998. Mechatronics '98. Proceedings. Skövde, 1998.

ROSHARDT, R.; UHRMAN, C.; WAFFLER, T.; WEIK, S. A complementary approach to flexible automation. In: INTERNATIONAL CONFERENCE ON ARCHITECTURES AND DESIGN METHODS FOR BALANCED AUTOMATION SYSTEMS. Portugal, 1995. Balanced Automation Systems: BASYS '95. Proceedings. London, Chapman & Hall, 1995. p. 141-148.

SASAKI, Edson Mitsuo. Aplicação de Redes de Petri Auto-Modificáveis em Sistemas de Manufatura. São Paulo, 1996. 66p. Dissertação de Mestrado - Escola Politécnica, Universidade de São Paulo.

SHIMOMURA, Yoshiki; TANIGAWA, Sadao; UMEDA, Yasushi; TOMIYAMA, Tetsuo. Development of self-maintenance photocopyers. AI Magazine, v. 16, No 4, p. 41-53, winter, 1995.

SRINIVASAN, V. S.; JAFARI, Mohsen A. Fault detection/monitoring using time Petri nets. IEEE Transactions on Systems, Man, and Cybernetics, v. 23, n. 4, p. 1155-1162, July/August, 1993.

TOHMA, Yoshiro. Review of fault-tolerant computing for safety critical applications in Japan. International Journal of Computer Systems Science & Engineering, v. 9, n. 1, p. 3-10, January, 1994.

UMEDA, Yasushi; TOMIYAMA, Tetsuo; YOSHIKAWA, Hiroyuki. A design methodology for self-maintenance machines. Journal of Mechanical Design, v. 117, n. 3, p. 355-362, September, 1995.

VALK, Rüdiger. Self-modifying nets: a natural extension of Petri nets. Advances in Petri Nets: lecture notes in computer science, v. 62, p. 464-476, 1978.

- WOBBE, W. Anthropocentric production systems are socio-technological innovations. In: INTERNATIONAL CONFERENCE ON ARCHITECTURES AND DESIGN METHODS FOR BALANCED AUTOMATION SYSTEMS: Implementation Challenges for Anthropocentric Manufacturing. Portugal, 1996. Balanced Automation Systems II: BASYS '96. Proceedings. London, Chapman & Hall, June, 1996. p. 17-24.
- ZHOU, MengChu; DICESARE, Frank. Adaptive design of Petri net controllers for error recovery in automated manufacturing systems. **IEEE Transactions on Systems, Man and Cybernetics**, v. 19, n. 5, p. 963-973, September/October, 1989.
- ZHOU, MengChu; DICESARE, Frank. **Petri Net Synthesis for Discrete Event Control of Manufacturing Systems**. Kluwer Academic, Boston, 1993.
- ZURAWSKI, Richard. Systematic construction of functional abstractions of Petri net model of flexible manufacturing systems. **IEEE Transactions on Industrial Electronics**, v. 41, n. 6, p. 584-601, December, 1994.
- ZURAWSKI, Richard; ZHOU, MengChu. Petri nets and industrial applications: a tutorial. **IEEE Transactions on Industrial Electronics**, v. 41, n. 6, p. 567-583, December, 1994.
- ZURAWSKI, Richard; ZHOU, MengChu. Special section on Petri nets in manufacturing: guest editorial. **IEEE Transactions on Industrial Electronics**, v. 41, n. 6, p. 565-566, December, 1994.

## APÊNDICE

1.1	1.1	MODELAGEM CLÁSSICA E MODERNA DE CVDS	1.1
1.2	1.1.1	Sistemas Robustos, Adaptativos e Tolerantes a Falhas	1.2
1.3	1.2	SISTEMAS COM VARIÁVEIS DISCRETAS E EVENTOS DISCRETOS	1.3
1.3	1.2.1	Sistemas Dinâmicos a Eventos Discretos (DEDS)	1.3
1.6	1.2.2	Considerações e Dificuldades na modelagem de DEDS	1.6
1.7	1.2.3	Modelos de DEDS	1.7
1.9		REFERÊNCIAS BIBLIOGRÁFICAS	1.9
II.1	II	SISTEMAS DE MANUFATURA	II.1
II.5		REFERÊNCIAS BIBLIOGRÁFICAS	II.5
III.3	III.1	PROPRIEDADES DAS REDES DE PETRI	III.3
III.9	III.2	ARCOS INIBIDORES E ARCOS HABILITADORES	III.9
III.10	III.3	MODELAGEM DE SISTEMAS DE MANUFATURA	III.10
III.12	III.4	REDE DE PETRI COLORIDA	III.12
III.13	III.5	OUTRAS EXTENSÕES DA REDE DE PETRI	III.13
III.16	III.6	COMPARAÇÃO DE RP E DIAGRAMA LÓGICO DE RELES (LLD)	III.16
III.20		REFERÊNCIAS BIBLIOGRÁFICAS	III.20

## APÊNDICE I

Neste apêndice são introduzidos alguns dos conceitos básicos de sistemas dinâmicos a eventos discretos (DEDS) como: as principais características, modelos de DEDS mais utilizados, dificuldades e considerações na modelagem e análise de DEDS e as principais considerações que os diferenciam dos sistemas dinâmicos de variáveis contínuas (CVDS).

### I.1 MODELAGEM DE CVDS

Um modelo matemático de um sistema artificial interagindo com a natureza pode ser um conjunto de equações diferenciais que representam a dinâmica do sistema. Tais equações são desenvolvidas utilizando as leis físicas que governam um sistema particular, por exemplo, as leis da conservação da energia e do movimento de Newton. Assim, um sistema pode ser representado de muitas maneiras diferentes e portanto, podem existir muitos modelos matemáticos do mesmo sistema, dependendo da perspectiva considerada [Ogata, 93].

Técnicas para modelagem de sistemas descritos por modelos matemáticos não-lineares são ainda objeto de estudos. Um grande avanço tem ocorrido com o desenvolvimento de técnicas que solucionam parcialmente estes problemas, técnicas como: desacoplamento dos distúrbios, desacoplamento das entradas/saídas e linearização realimentada, transformam um sistema não-linear num sistema linear equivalente [Levis, 87]. A teoria e aplicações, neste caso, estão num estado incipiente de desenvolvimento e existem muitas limitações, isto é, problemas abertos e perguntas ainda sem resposta. Os principais problemas não solucionados de sistemas não-lineares relacionam-se com a estabilidade, sendo necessárias pesquisas para melhorar as características de desempenho, sensibilidade, robustez e que delimitem regiões de estabilidade para certos tipos de sistemas,

além de desenvolver metodologias práticas de projeto com ferramentas de engenharia associadas ao suporte computacional disponível.

### **1.1.1 Sistemas Robustos, Adaptativos e Tolerantes a Falhas**

Muitos sistemas reais operam em ambientes com variações de parâmetros como a variação das matérias primas, combustíveis, níveis de carga, etc. que dificultam a descrição da dinâmica do sistema com modelos matemáticos simples. Na presença de incertezas e distúrbios difíceis de se avaliar, os sistemas realimentados devem manter as especificações de desempenho. Entretanto, os projetos exigem robustez com respeito às incertezas no modelo, garantindo a estabilidade numa larga faixa de variações de operação da planta e dos possíveis distúrbios e para adaptar-se a alterações na dinâmica do sistema.

A malha adaptativa de realimentação tenta reduzir as incertezas por uma estimação de parâmetros em tempo real e então utiliza-se os parâmetros estimados para continuamente retorná-los e adaptar a planta às condições atuais [Gildin, 98].

Uma classe mais geral é aquela que consegue adaptar-se a mudanças significativas no seu ambiente como no caso dos sistemas tolerantes a falhas. Nesta classe de problemas, admitimos que um ou vários dos componentes relevantes de um sistema físico da planta apresenta uma falha e esta falha pode ter um impacto significativo na estabilidade ou desempenho do sistema. No nível de elementos de equipamentos podem ser consideradas falhas em sensores e/ou atuadores e no nível de decisão podem ser considerados outros tipos de falhas, como falhas no “*software*” ou no “*hardware*” de controle.



## 1.2 SISTEMAS COM VARIÁVEIS DISCRETAS E EVENTOS DISCRETOS

Existem muitas situações nas quais modelos de sistemas descritos por equações diferenciais (ordinárias ou parciais) não são apropriados para representar a dinâmica do sistema. Considere-se, como primeiro exemplo, um sistema regulado por um controlador baseado em regras. Tal controlador executa ações de controle sobre os estados do sistema definidos por variáveis descritas por um conjunto de números reais e um conjunto de variáveis lógicas (ou Booleanas). Existem também outras situações, na qual o estado do sistema envolve variáveis contínuas e discretas. Isto é comum quando existe uma discretização devido a "switches", chaves, relés ou limitadores, ou em sistemas com múltiplos modos de operação, tais como controladores reconfiguráveis de aeronaves, robôs manipuladores e sistemas com múltiplos modos de tratamento de falhas.

### 1.2.1 Sistemas Dinâmicos a Eventos Discretos (DEDS)

Outra classe de sistemas discretos consiste de sistemas definidos por alguns autores como "man-made systems". Nestes sistemas a alteração dos estados é governada pela interação de eventos discretos que ocorrem em instantes discretos de tempo, sendo conhecidos como sistemas dinâmicos a eventos discretos (DEDS). Como exemplo de tais sistemas pode-se citar: Sistemas flexíveis de manufatura (FMS) [Rembold, 94], Sistemas antropocêntricos de produção (APS) [Wobbe, 96], redes de comunicação [Catania, 93], Sistemas de distribuição de energia elétrica [Bose, 94], "Hardware" de sistemas computacionais [Chiola, 93], sistemas de tráfego de veículos ou sistemas de transporte em geral.

O estado destes sistemas é alterado em instantes discretos de tempo (de forma assíncrona) em lugar de continuamente no tempo. A classificação dos estados consiste de

variáveis numéricas e variáveis discretas, tais como o número de peças esperando processamento em cada estação, a lista de trabalhos, o estado de prontidão de cada estação ou o tempo de processamento. De fato, suas leis dinâmicas são caracterizadas por regras procedurais produto do raciocínio humano<sup>[1]</sup>. Estes sistemas em geral, não podem ser descritos por equações diferenciais (ordinárias ou parciais) por sua natureza descontínua [Cassandras, 90; Ho, 87].

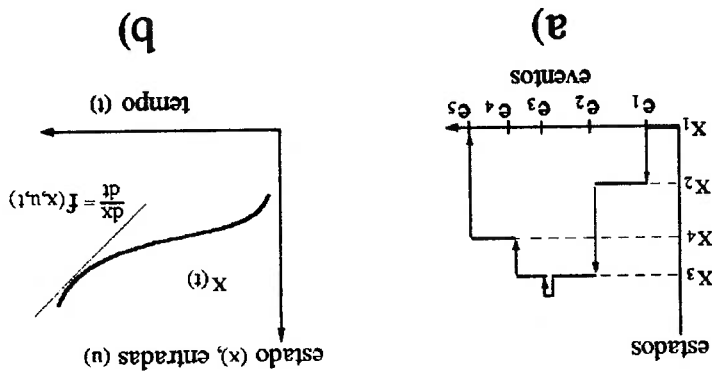
Estes sistemas contrastam-se aos sistemas dinâmicos de variáveis contínuas (CVDS), nos quais o *estado* e as *variáveis* do sistema evoluem de forma contínua no tempo e cuja dinâmica é adequadamente representada por equações integro-diferenciais. Assim, a noção de tempo como uma variável independente nos CVDS, fica substituída pelo conceito de *seqüência de eventos* nos DEDS [Ho, 89; Cao, 90]. Às vezes, sistemas contínuos são tratados como sistemas de tempo discreto. Estes sistemas têm mais em comum com os CVDS apesar da similaridade com os DEDS na forma de manipular o tempo, isto é, nos sistemas de tempo discreto cada alteração no estado ou passo é sincronizado por um “clock” global [Ogata, 93].

Para visualizar melhor a diferença na evolução dos estados entre os DEDS e os CVDS, é apresentado na figura I.1 um exemplo do contraste do conceito de “trajetória”, utilizado para definir o estado de um sistema. A figura I.1-a) ilustra a “trajetória” que representa a alteração de estados de um DEDS composta por segmentos horizontais que representam a manutenção de cada estado e onde o comprimento de cada segmento é proporcional ao tempo que o sistema permanece nesse estado. Os estados tomam valores num conjunto discreto, enquanto sua duração é geralmente contínua, podendo ser de natureza qualitativa ou quantitativa. De outro lado, a figura I.1-b) ilustra a evolução dos

[1] Daí o termo “*man-made systems*” como sinônimo de DEDS

estados de um CVDS representado por uma trajetória que muda continuamente e pode ser descrita através de equações integro-diferenciais.

Os DEDS se caracterizam por apresentar muitos componentes interagindo entre si e podem ser assíncronos e sequenciais, existindo a necessidade de sincronização de processos, tratamento de operações concorrentes, solução de conflitos, compartilhamento de recursos, etc. Além disso, um controle inadequado da ocorrência dos eventos pode produzir bloqueios<sup>[2]</sup>, "overflow"<sup>[3]</sup> e/ou reduzir o desempenho do sistema.



Geralmente na etapa de operação, os DEDS integrados (e/ou controlados) por computador, interagem mais com operadores humanos, por exemplo, através de telas de monitores de computadores. Isto tem implicações na implementação prática, além disso, interfaces com sistema de inteligência artificial (AI) e a habilidade para processar linguagens naturais, podem ser fatores mais importantes nos DEDS que nos CVDS [Cao, 90].

Outro fator que diferencia os DEDS é que estes sistemas não têm as restrições das leis físicas da natureza que limitam sua configuração, têm apenas as limitações físicas dos materiais e restrições ergonômicas, podendo ocorrer assim, uma explosão combinatoria (ou exponencial) de elementos do modelo aumentando a complexidade da análise do sistema.

[2] Paralisação parcial ou total das atividades de um sistema.

## 1.2.2 Considerações e Dificuldades na modelagem de DEDS.

Alguns dos problemas que dificultam a abstração de DEDS e que devem ser considerados numa estrutura geral de modelagem são [Ho, 89]:

1. Natureza descontínua dos Eventos Discretos: os estados físicos de um DEDS são inerentemente discretos e esta natureza não pode ser desconsiderada, apesar que às vezes ser possível analisar DEDS por aproximações contínuas via modelos de difusão ou por procedimentos baseados exclusivamente em taxas (de produção, de eficiência, etc.) nos estudos referentes a análise de desempenho [Molloy, 82].

2. Natureza contínua da maior parte das medidas de desempenho: as medidas de desempenho nos DEDS são principalmente formuladas em termos de variáveis contínuas, tais como "throughput", tempo de espera, etc. De fato, o tempo (uma variável fundamental para medidas de desempenho) é uma variável contínua e assim, não há vantagens em considerar modelos de DEDS de tempo discreto ou "dados de amostragem" de DEDS.

3. Importância da formulação probabilística: Todas as abordagens orientadas ao desempenho incorporam efeitos estocásticos como parte integral da sua modelagem [Molloy, 82]. Distúrbios não programados e/ou "breakdowns" são fatos que acontecem no tempo de operação de um DEDS mas são muito difíceis de quantificar. Outro fator que se deve considerar é a necessidade de incluir modelos de tomada de decisões e da interação humana com o sistema (não necessariamente probabilísticos) [Hess, 87].

4. A necessidade de uma análise hierárquica: já que o grau de detalhamento requerido na modelagem e controle são diferentes para cada um dos níveis de controle. Numa

[3] Sobrecarga física e/ou operacional de elementos do sistema.

hierarquia devidamente estruturada deve ser observada a propriedade de que a modelagem num nível seja independente à alterações nos componentes dos níveis inferiores.

5. A presença da dinâmica: devem considerar modelos que representem a dinâmica e os procedimentos transitórios nos DEDS em lugar de concentrar-se na análise de desempenho como é realizado nas linguagens de simulação de DEDS. Métodos como a análise de perturbações [Cao, 89] são exemplos deste caso.

6. As limitações da carga computacional: uma simples enumeração dos estados pode crescer de maneira combinatorialmente explosiva (como em máquinas de estado finito ou cadeias de Markov) ou de maneira exponencial (como em RP), conduzindo rapidamente a capacidades computacionais impossíveis para equipamentos convencionais. Assim, é necessário distinguir entre modelos conceituais e computacionais.

### 1.2.3 Modelos de DEDS.

Na bibliografia consultada [Cao, 90; Ho, 89], os pesquisadores concordam que ainda não existe uma estrutura universal de modelagem para os DEDS comparada com a bem sucedida estrutura ou paradigmas criados nos CVDS utilizando equações diferenciais. No entanto, algumas técnicas têm sido utilizadas com sucesso para modelar algumas classes de DEDS. As técnicas mais conhecidas são [Cao, 90]:

- Modelos de cadeias de Markov, modelos de redes de filas.
- Rede de Petri<sup>[4]</sup>, máquina de estados finitos.
- Álgebra Min-Max.

<sup>[4]</sup> Inicialmente foi definida a Rede de Petri C/E, redes condigão/evento, posteriormente foram desenvolvidas várias extensões.

- Processos recursivamente finitos ou processos sequenciais em comunicação.
- Linguagens de simulação de eventos discretos. Nesta categoria incluímos os Processos Generalizados Semi-Markov (GSM).

Neste contexto, existem três tipos de classificação para os modelos de DEDS, veja

tabela I.1 [Ho, 89]:

1. Modelos temporizados ou não-temporizados. Nos modelos temporizados a ocorrência dos eventos depende do tempo, incorporando o conceito do tempo como parte integral do modelo.

2. Modelos lógicos, algébricos ou para avaliação do desempenho. Nos modelos lógicos os problemas são de natureza lógica ou qualitativa, por exemplo manipulação de variáveis como: *sim/não, on/off, etc.* Os algébricos permitem capturar a descrição da trajetória de DEDS em termos de um conjunto de operações algébricas em função dos seus estados e eventos. Os modelos para avaliação do desempenho visam avaliar as taxas de eficiência, de produtividade e de confiabilidade.

3. Modelos estocásticos e determinísticos. A teoria de sistemas estocásticos é o estudo de sistemas dinâmicos nos quais os distúrbios do ambiente são modelados como processos probabilísticos e/ou aleatórios, em contraposição aos modelos determinísticos.

Tabela I.1 - Classificação de modelos de DEDS.

	Temporizados	Não-temporizados	
Lógicos		Rede de Petri Máquinas de estado finito	
Algébricos	Algebra min-max	Processos recursivamente finitos. Processos sequências de comunicação.	
Desempenho	Cadeias de Markov Redes de Filas. GSM/simulação		↔ Determinísticos
			↔ Estocásticos

## REFERÊNCIAS BIBLIOGRÁFICAS

- BOSE, Bimal K. Expert system, fuzzy logic, and neural network applications in power electronics and motion control. *Proceedings of the IEEE*, v. 82, n. 8, p. 1303-1323, August, 1994.
- CAO, Xi-Ren. A comparison of the dynamics of continuous and discrete event systems. *Proceedings of the IEEE*, v. 77, n. 1, p. 7-13, January, 1989.
- CAO, Xi-Ren; HO, Yu-Chi. Models of discrete event dynamics systems. *IEEE Control Systems Magazine*, v. 10, n. 4, p. 69-76, June, 1990.
- CASSANDRAS, Christos G.; RAMADGE, Peter J. Toward a control theory for discrete event systems. *IEEE Control Systems Magazine*, v. 10, n. 4, p. 66-68, June, 1990.
- CATANIA, V.; PULIAFITO, A.; VITA, L. A modeling framework to evaluate performance parameters in gracefully degrading systems. *IEEE Transactions Industrial Electronics*, v. 40, n. 5, p. 461-472, 1993.
- CHOLA, G.; DUTHEILLBT, C.; FRANCESCHINIS, G.; HADDAD, S. Stochastic well-formed coloured nets and symmetric modelling applications. *IEEE Transactions on Computers*, v. 42, n. 11, p. 1343-1360, 1993.
- GILDIN, Eduardo. Desenvolvimento de um Controlador Adaptativo para Manipuladores Flexíveis com Incertezas de Cargas. São Paulo, 1998. 110p. Dissertação de Mestrado - Escola Politécnica, Universidade de São Paulo.
- HESS, Ronald A. A qualitative model of human interaction with complex dynamic system. *IEEE Transactions on System, Man, and Cybernetics*, v. SMC-17, n. 1, p. 33-51, January/February, 1987.
- HO, Yu-Chi. Scanning the issue, dynamics of discrete event systems. *Proceedings of the IEEE*, v. 77, n. 1, p. 3-6, January, 1989.
- HO, Yu-Chi. Basic research, manufacturing automation, and putting the cart before the horse: editorial. *IEEE Transactions on Automatic Control*, v. AC-32, n. 12, p. 1042-1043, December, 1987.
- LEVIS, Alexander H. et al. Challenges to control: a collective view. *IEEE Transactions on Automatic Control*, v. AC-32, n. 4, p. 275-285, April, 1987.
- MOLLOY, M. K. Performance analysis using stochastic Petri nets. *IEEE Transactions on Computers*, v. 3, n. 9, p. 913-917, 1982.

OGATA, Katsuhiko. *Engenharia de Controle Moderno*. 2 ed. Rio de Janeiro, Prentice-Hall do Brasil, 1993.

WOBBE, W. Anthropocentric production systems are socio-technological innovations. In: INTERNATIONAL CONFERENCE ON ARCHITECTURES AND DESIGN METHODS FOR BALANCED AUTOMATION SYSTEMS: Implementation Challenges for Anthropocentric Manufacturing. Portugal, 1996. *Balanced Automation Systems II: BASYS '96. Proceedings*. London, Chapman & Hall, June, 1996. p. 17-24.



## APÊNDICE II

Neste apêndice são abordados alguns dos conceitos dos sistemas automatizados de manufatura classificando-os em função de sua flexibilidade e sua produtividade, permitindo localizar assim, os sistemas flexíveis de manufatura (FMS) junto a outros sistemas de produção. Também define o papel desempenhado pela troca automática de peças e ferramentas nos vários sistemas de manufatura.

### Sistemas de Manufatura.

Basicamente os sistemas de manufatura podem ser classificados em cinco grupos, onde cada grupo possui uma faixa ótima de operação em função da relação: custo de produção de cada peça versus o número de peças produzido numa unidade de tempo.

A figura II.1 [Rembold, 94], ilustra graficamente a distribuição destes grupos em função da flexibilidade relacionada com o número de tipos de peças que podem ser processadas por cada sistema e dependendo da produtividade, que está relacionada com as taxas de produção ou ao número de peças produzidas em cada lote.

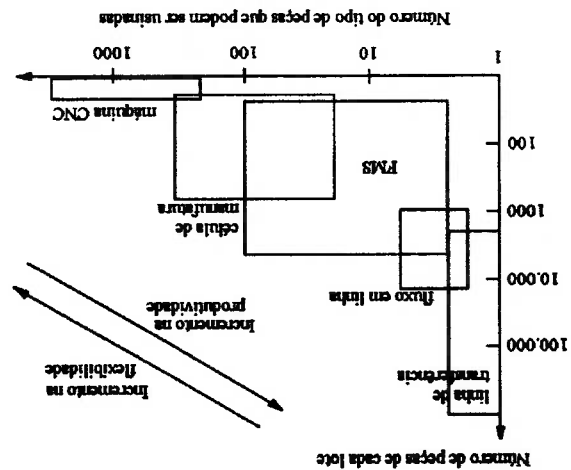


Fig. II.1 - Classificação de Sistemas Automatizados de Manufatura.

As grandezas indicadas nos eixos horizontal e vertical da figura II.1, são valores aproximados. Estas grandezas variam em função de fatores como: taxa de produção, tamanho do lote, complexidade, similaridade das peças, etc.

As principais características destes 5 grupos [Ferre, 90; Alvarez, 90; Rembold, 94] são:

1. Linha de transferência: utiliza o princípio de produção em massa. Requer um investimento relativamente alto para ser reconfigurado e diversificar sua produção, entretanto, os custos de produção de cada item são relativamente baixos. Uma exceção desta rigidez são as atuais linhas de montagem de veículos. Numa mesma linha podem ser montados diferentes versões de um veículo em função de uma relativa flexibilidade dos equipamentos que compõem a linha (alguns autores classificam este tipo de linha num outro grupo como linha flexível). A figura II.2 ilustra uma linha de montagem de veículos [Alvarez, 90].

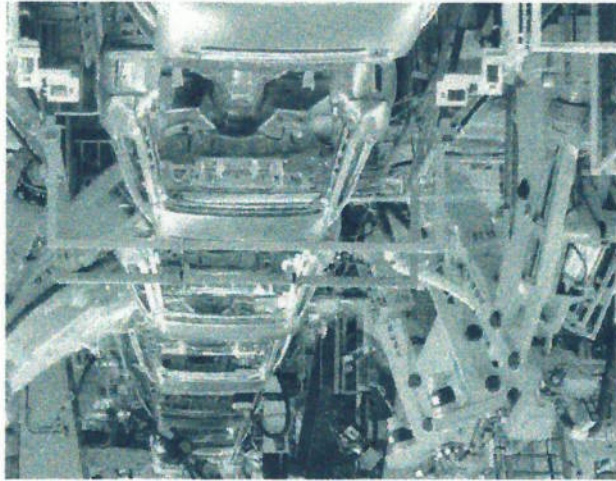


Fig. II.2 - Linha de montagem de veículos.

1. Fluxo em linha (em grupos): segue em princípio, as características de produção da linha de transferência, entretanto aqui o conjunto de peças são organizadas em grupos para minimizar tempos de montagem e troca de ferramentas.

2. Sistema Flexível de Manufatura (FMS): é composto por várias máquinas-ferramentas com CNC, as quais estão interconectadas para transferência de informações (por um sistema de rede de comunicação) e manipulação de materiais (por um sistema de transporte como esteiras, veículos transportadores, etc. e um sistema de troca de ferramentas e peças, geralmente montadas em "pallets").

3. Célula de manufatura: normalmente constituída por uma ou várias máquinas CNC ou centros de usinagem dotados de um sistema automático de troca de peças e ferramentas que permitem trabalhar sem a presença contínua do operador humano. Salvo em casos de robotização muito avançados para tarefas como carregar e descarregar peças das "pallets" e para reposição do depósito de ferramentas, a presença de operador é ainda indispensável [Ferre, 90]. A figura II.3 ilustra uma célula flexível de manufatura da DECKEL composta por 2 centros de usinagem DC-40 com um dispositivo alimentador de "pallets" em linha [Ferre, 90].

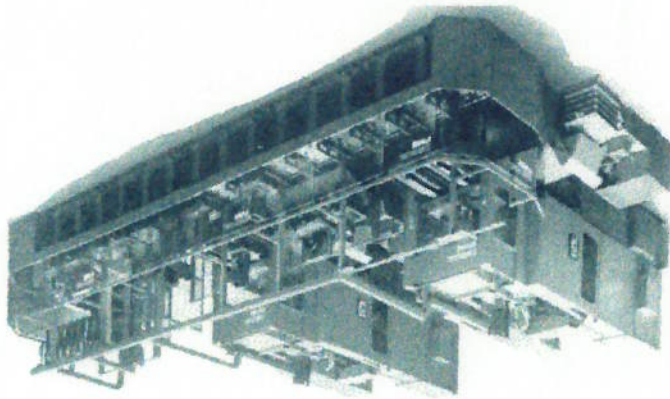


Fig. II.3 - Célula flexível de manufatura.

Existe alguma confusão entre os termos centro de usinagem e célula flexível: os centros de usinagem são equipamentos "standard" oferecidos pelos fabricantes e nas suas versões mais completas oferecem a funcionalidade de uma célula de manufatura. Um centro de usinagem se baseia no mesmo princípio de uma fresadora (peça fixa e



ferramenta móvel) e permite a troca automática, mas limitada em número de ferramentas e peças possibilitando ao centro trabalhar várias horas sem a intervenção do operador. Na figura II.4 é ilustrado um centro de usinagem D24 da DECKEL com dispositivo alimentador giratório de "pallets" PS-8 [Ferre, 90].

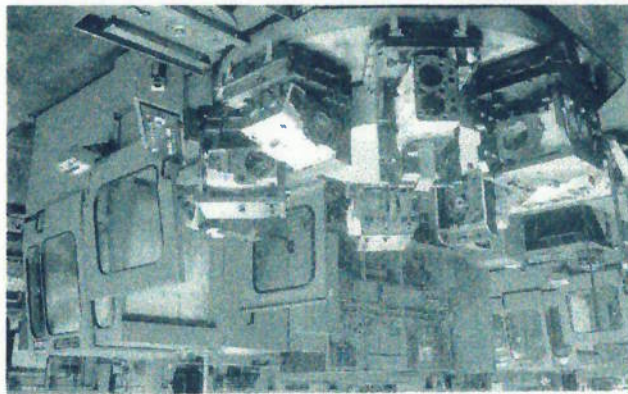


Fig. II.4 - Centro de usinagem com sistema de alimentação múltiplo de peças.

5. Máquinas CNC: fazem operações tradicionais de usinagem (torneamento/fresagem) com uma relativa flexibilidade. Entretanto, devido a seu tempo de "set-up" relativamente alto, este sistema de produção é adequado para lotes com número reduzido de peças [Ferre, 90]. Na figura II.5 é ilustrada uma máquina CNC TNS-26 da TRUB para torneamento de tarugos com 10 eixos e sistema de controle de 32 eixos que permite usinar peças a partir de tarugos de até 1" de diâmetro e possui 2 porta-ferramentas de tipo revolver que alojam até 12 ferramentas cada.

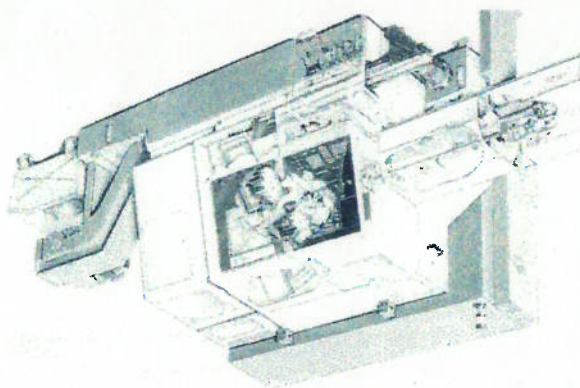


Fig. II.5 - Máquina ferramenta CNC para torneamento.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ALVAREZ, Ramiro. Máquina-herramienta de control numérico. *Revista de Gestión de las Nuevas Tecnologías de Producción para Directivos, CIM*, n. 16, p. 43-44, Junio-Julio, 1990.
- FERRÉ Massip, Rafael. Centros de mecanizado. *Revista de Gestión de las Nuevas Tecnologías de Producción para Directivos, CIM*, n. 16, p. 46-52, Junio-Julio, 1990.
- REMBOLD Ulrich; NMAJ, B. O.; STORR, A. *Computer Integrated Manufacturing And Engineering*. s.l., Addison-Wesley, 1994.

### APÊNDICE III

#### REDE DE PETRI (RP).

Uma rede de Petri (RP) e uma técnica que permite modelar graficamente ou matematicamente um sistema. Uma representação *gráfica* de uma estrutura de RP [Peterson, 81; Murata, 89] consiste de nós conectados por segmentos orientados de ligação chamados ARCOS. Existem dois tipos de nós, LUGARES (P) representados por círculos (O) e TRANSIÇÕES (T) representadas por barras (|). Os ARCOS, representados por setas ( $\rightarrow$ ), conectam LUGARES a TRANSIÇÕES ou TRANSIÇÕES a LUGARES (nunca LUGAR  $\rightarrow$  LUGAR ou TRANSIÇÃO  $\rightarrow$  TRANSIÇÃO). Uma RP é um **multigrafo**, já que permite múltiplos ARCOS de um nó do grafo até outro. É **bipartido**, já que os nós do grafo são particionados em dois conjuntos (LUGARES e TRANSIÇÕES) e os ARCOS conecta, nós de diferentes conjuntos. É **direcionado**, já que os ARCOS são orientados.

Uma estrutura de rede de Petri pode ser definida formalmente como uma quadrupla

$(P, T, I, O)$  onde:

- $P = \{p_1, p_2, \dots, p_m\}$  é o conjunto de LUGARES da rede, onde  $m \in \mathbb{N}$  é o número total de

LUGARES da rede.

- $T = \{t_1, t_2, \dots, t_n\}$  é o conjunto de TRANSIÇÕES,  $n \in \mathbb{N}$  é o número de TRANSIÇÕES da rede.

- $I : (P, T)$  é a função que define os ARCOS de entrada em relação às TRANSIÇÕES. Se

$I(p_j, t_i) = k$ , existem  $k \in \mathbb{N}$  ARCOS orientados do LUGAR  $p_j$  para a TRANSIÇÃO  $t_i$ , e quando

$I(p_j, t_i) = 0$ , não existe ARCO orientado do LUGAR  $p_j$  para a TRANSIÇÃO  $t_i$ .

- $O : (T, P)$  é a função que define os ARCOS de saída em relação às TRANSIÇÕES.

Geralmente, na representação gráfica, ARCOS múltiplos conectando LUGARES e

TRANSIÇÕES são representados de forma compacta por um único ARCO rotulado com seu

peso ou multiplicidade  $k$  (veja figura III.1).

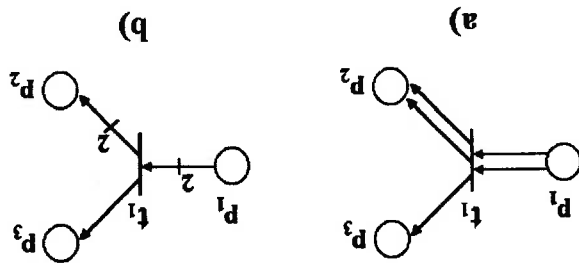


Fig. III.1 - a) Arcos múltiplos. b) Representação compacta.

Uma RP com MARCAS associadas a seus LUGARES é chamada de Rede de Petri Marcada RP=(P,T,I,O,M<sub>0</sub>), onde M<sub>0</sub> é a marcação inicial. Uma distribuição particular (M) das MARCAS nos LUGARES da rede representa um estado do sistema. Graficamente estas MARCAS são modeladas por pontos pretos no interior dos LUGARES (●).  $M = (M(p_1), M(p_2), \dots, M(p_m))$  onde  $m \in \mathbf{N}$  é o número total de LUGARES.  $M(p_j)$  é o número de MARCAS do LUGAR  $p_j$  ( $M(p_j) \in \mathbf{N}$ ).

O disparo de uma TRANSIÇÃO na RP corresponde a ocorrência de um evento que altera o estado do sistema, isto é, altera a marcação atual (M<sub>i</sub>) para uma nova marcação (M<sub>i+1</sub>). Uma TRANSIÇÃO ( $t_i \in T$ ) fica habilitada para disparar se e somente se cada LUGAR de entrada  $p_j$  (tal que  $I(p_j, t_i) > 0$ ) contém pelo menos um número de MARCAS igual ao peso do ARCO ( $p_j, t_i$ ), ou seja:  $M(p_j) \geq I(p_j, t_i)$  para qualquer LUGAR  $p_j \in P$ .

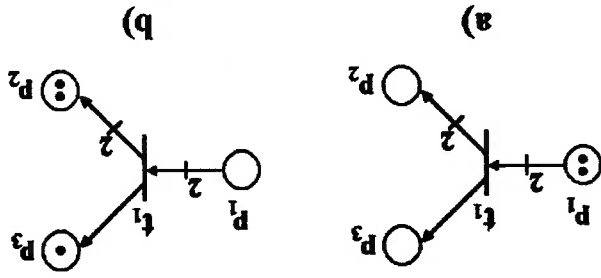


Fig. III.2 - a) TRANSIÇÃO habilitada. b) Marcação após o disparo.

O disparo de uma TRANSIÇÃO  $t_i$  habilitada remove de cada LUGAR de entrada  $p_j$  um número de MARCAS igual ao peso do ARCO  $I(p_j, t_i)$  que conecta  $p_j$  a  $t_i$ . Depositando, também, em cada LUGAR de saída  $p_q$  um número de MARCAS igual ao peso do ARCO  $O(t_i, p_q)$  que conecta  $t_i$  a  $p_q$ . A figura III.2 ilustra a alteração da marcação após o disparo de uma TRANSIÇÃO.

### III.1 PROPRIEDADES DAS REDES DE PETRI

O estudo das propriedades da RP é muito importante, já que permite uma análise detalhada do sistema modelado. Dois tipos de propriedades na RP podem ser distinguidas:

- Propriedades procedurais, as quais dependem do estado ou marcação inicial da RP.
- Propriedades estruturais, que dependem da topologia ou estrutura da RP.

Entre as principais propriedades úteis no estudo dos FMS encontram-se: alcançabilidade, limitação, segurança, vivacidade e reversibilidade. A descrição de outras propriedades tais como conservação, cobertura, persistência, distância síncrona e "fairness" podem ser encontradas em [Murata, 89], além de uma descrição mais detalhada das propriedades aqui abordadas.

- A *alcançabilidade* de uma RP refere-se a possibilidade de um sistema atingir um determinado estado. Uma marcação  $M_i$  é *alcançável* a partir de  $M_0$  se existir uma sequência de disparos de TRANSIÇÕES que leve uma rede com marcação  $M_0$  a uma outra com marcação  $M_i$ . O conjunto de alcançabilidade  $R(RP)$ , é o conjunto de todas as marcações alcançáveis a partir de  $M_0$  através de sequências de disparo das TRANSIÇÕES.

- Uma RP é *limitada* se e somente se o número de MARCAS de todos os LUGARES da rede não superam um limite definido  $M(p_j) \leq B$  para qualquer marcação *alcançável* ( $B \in \mathbb{N}$ ).



Um caso especial de uma rede limitada é quando  $B=1$ , no qual a RP é dita *segura*. Num ambiente de manufatura, a limitação (ou segurança) de uma RP indicam ausência de “*overflow*” (caso o LUGAR represente áreas de armazenamento como por exemplo: “*buffers*”) ou a limitação dos recursos.

- Uma RP é *viva* se e somente se para toda TRANSIÇÃO ( $t_i$ ) e para qualquer marcação ( $M_i$ ), existe uma sequência de disparos de TRANSIÇÕES a partir de  $M_i$  que habilitam  $t_i$ . A vivacidade de uma RP indica a ausência da possibilidade de bloqueios. Além disso, a vivacidade garante que todos os processos modelados possam ser executados. Uma das principais razões para que ocorra um bloqueio é o compartilhamento de recursos, pode-se demonstrar que quatro condições devem ser observadas simultaneamente para que ocorra um bloqueio, ou em outras palavras, para evitar a ocorrência de um bloqueio é suficiente garantir que pelo menos 1 destas condições nunca seja satisfeita [Hasegawa, 96]:

- Exclusão mútua por compartilhamento de recursos.
- Ocupação do recurso enquanto aguarda a liberação de outros recursos para executar um processo que requer todos estes recursos conjuntamente.
- Não liberação de um recurso utilizado num processo por outro processo.
- Existência de um caminho circular.

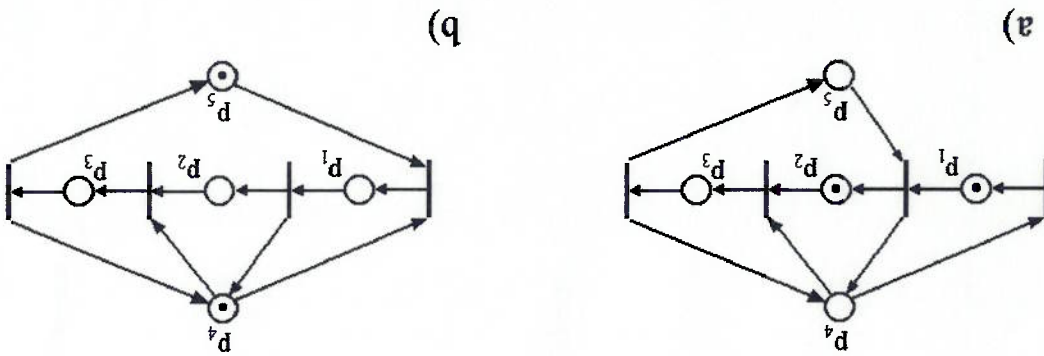
Um exemplo de bloqueio é ilustrado no processo de carregamento e descarregamento de peças realizado pelo robô R na máquina M, figura III.3. O modelo em RP deste sistema é apresentado na figura III.4.

Considerando o modelo da figura III.4-a) existe um bloqueio em  $M=(1,1,0,0,0)$ . O robô é um recurso compartilhado sequencialmente para, carregamento e descarregamento da mesma máquina. Esta marcação representa o estado no qual o robô segura uma peça para ser carregada em M, mas M não esta disponível, para que M fique disponível é necessário

que robô descarregue **M**, mas **R** não está disponível. Nota-se que as 4 condições citadas são satisfeitas.



Fig. III.3 - Sistema composto por uma máquina-ferramenta e um robô.



Lugar	Descrição
$p_1$	Robô carregando M.
$p_2$	M processando
$p_3$	Robô descarregando M
$p_4$	Robô disponível
$p_5$	M disponível

Fig. III.4 - a) RP do sistema da figura III.3 apresentando um bloqueio. b) RP sem bloqueio.

Para solucionar o problema de bloqueios podem ser consideradas 3 abordagens.

1. Detecção e correção do bloqueio [Zhou, 93].
2. Prevenção do bloqueio [Zhou, 93]: a rede da figura III.4-b) previne um bloqueio eliminando a condição "b" (ocupação de um recurso enquanto aguarda a liberação...).

3. Eliminação estrutural de situação de bloqueio, em [Hasegawa, 96] é apresentada esta abordagem a qual elimina a condição "d" (eliminação do caminho circular).

- Uma RP é *reversível* se e somente se a marcação inicial ( $M_0$ ) for *alcançável* a partir de qualquer marcação  $M_i \in R(RP)$ . Esta propriedade está relacionada com a capacidade de *reincializar*, ou seja, que o sistema possa retornar ao seu estado inicial a partir de qualquer estado (por exemplo, um estado de falha).

Uma vantagem de aplicar RP ao estudo de sistemas a eventos discretos é a sua capacidade de formalização matemática para verificação das propriedades de um sistema modelado. A verificação destas propriedades é implementada através de 4 métodos

[Zurawski, 94a]:

1. Geração da árvore ou grafo de alcançabilidade.
2. Análise da equação de estado.
3. Simulação.
4. Técnicas de redução.

1. Geração da árvore e grafo de alcançabilidade [Murata, 89]: este método envolve a enumeração de todas as marcações possíveis ou alcançáveis a partir da marcação inicial  $M_0$ , a qual é a raiz da árvore, cada nó representa uma marcação alcançável e cada ARCO representa o disparo de uma TRANSIÇÃO.

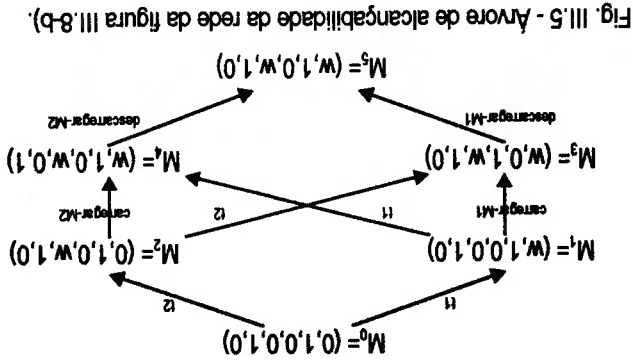


Fig. III.5 - Árvore de alcançabilidade da rede da figura III.8-b).

Um grande número de propriedades procedurais podem ser estudadas utilizando este método, entretanto, por ser um método exaustivo, torna-se pouco efetivo para a análise de redes com um número considerável de LUGARES e TRANSIÇÕES.

2. Análise da equação de estado [Zurawski, 94a]. Nesta abordagem uma matriz de incidência que representa as conexões entre LUGARES e TRANSIÇÕES é utilizada para representar a estrutura do sistema que caracteriza o comportamento dinâmico da rede. A equação de estado representa uma alteração na distribuição das MARCAS como resultado do disparo de uma TRANSIÇÃO. A equação é definida da seguinte forma:

$$M_k = M_{k-1} + A^T u_k \quad \text{(III.1)}$$

Onde:

- $M_k$  é um vetor coluna ( $m \times 1$ ) que representa a marcação imediatamente alcançável a partir de  $M_{k-1}$  após o disparo da TRANSIÇÃO  $t_k$ .

- $u_k$  é um vetor ( $n \times 1$ ) com grandeza de 1 na posição  $i$  que representa o disparo da TRANSIÇÃO  $t_i$  e zero (0) nas outras posições.

- $A$  é a matriz de incidência ( $n \times m$ ) definida por  $A = O(P,T) - I(P,T)$

Para o estudo das propriedades estruturais 2 conceitos relacionados à matriz de incidência são especialmente úteis, os  $T$ -invariantes e  $P$ -invariantes [Murata, 89].

3. Simulação. A simulação é efetiva quando o sistema é relativamente complexo e intratável utilizando métodos analíticos. Também é efetiva no estágio final do projeto para propósitos de verificação. A maior parte das análises por simulação são realizadas para estudos de desempenho, entretanto, também têm sido a base para o desenvolvimento de estruturas de controle lógico para sistemas de tempo real em sistemas de manufatura [Cheng, 91; Tang, 92] ou para detectar conflitos nas rotas de veículos transportadores em sistemas de manufatura [Zeng, 91].

Um outro exemplo está em [Jeng, 97] onde é apresentada uma técnica que sintetiza uma RP para modelagem de sistemas de manufatura. Tal técnica adota a abordagem "bottom-up" ou de composição modular. Cada módulo é uma rede para controle de recursos (RCN) a qual representa um subsistema que controla um tipo de recurso num sistema de manufatura. Pode-se demonstrar que a rede total obtida pela fusão dos módulos é conservativa e, então, é limitada. Há também, um algoritmo desenvolvido para detectar

estados do sistema.

Um exemplo é uma abordagem que reduz a representação de cada tarefa antes de desenvolver o grafo de alcançabilidade [Dwyer, 96]. Aqui é utilizado o grafo de interação por tarefas (TIG). Num TIG, os nós representam tarefas que compõem o modelo do sistema que por sua vez pode ser otimizado ou reduzido. Esta é a abordagem adotada quando o grafo de alcançabilidade é derivado de uma representação em RP do sistema total e cada tarefa é representada em TIG. O resultado é uma representação compacta, mas diferente de muitas outras técnicas reduzidas, pois a redução é realizada apenas para garantir a alcançabilidade de uma marcação desconsiderando a representação de todos os

a conservação das propriedades a serem analisadas.

4. Técnicas de redução: Para facilitar a análise de modelos relativamente grandes, podem ser utilizadas transformações que reduzem a dimensão do grafo mas que assegurem

em modelos complexos.

No entanto, a simulação só pode assegurar resultados para os estados simulados. Assim, para garantir uma generalização, é necessário considerar uma estratégia de simulação/teste que evite a necessidade de simular praticamente todos os estados que resultaria no problema da capacidade de computação e da alcançabilidade que seria, inviável

condições suficientes para vivacidade estrutural da rede. O algoritmo é mais eficiente que as técnicas de enumeração de estados baseadas na árvore de alcançabilidade.

### III.2 ARCOS INIBIDORES E ARCOS HABILITADORES.

A possibilidade de testar a ausência de MARCAS num LUGAR é feita pela introdução de

um novo elemento, o ARCO inibidor [Zurawski, 94a]. A presença de um ARCO inibidor conectando um LUGAR de entrada numa TRANSIÇÃO altera a condição de habilitação da TRANSIÇÃO e incrementa o poder de modelagem de uma RP. Este ARCO é representado graficamente por uma linha terminada num pequeno círculo. Na presença de um ARCO inibidor, uma TRANSIÇÃO é considerada habilitada se não existirem MARCAS no LUGAR de entrada conectado à TRANSIÇÃO, enquanto que para os LUGARES conectados com setas (ARCOS normais) a regra de disparo continua igual. No entanto, o disparo não altera a marcação do LUGAR conectado por um ARCO inibidor.

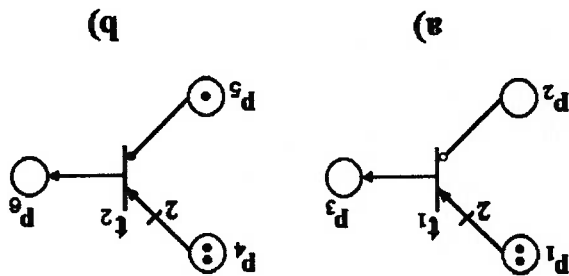


Fig. III.6 - a) RP com ARCO inibidor. b) RP com ARCO habilitador.

Da mesma forma que foi introduzido o conceito de ARCO inibidor em RP, podemos introduzir o conceito de ARCO habilitador como foi abordado para a extensão denominada de "Mark Flow Graph" (MFG) em [Miyagi, 96]. Este ARCO permite o disparo da TRANSIÇÃO caso no LUGAR de entrada exista MARCA, e é representado por um ARCO terminado num círculo preto. A figura III.6 ilustra a representação gráfica de um ARCO

inibidor e um ARCO habilitador, onde ambas as TRANSIÇÕES ( $t_1$  e  $t_2$ ) estão habilitadas para disparar.

### III.3 MODELAGEM DE SISTEMAS DE MANUFATURA

Num sistema de manufatura, a presença de uma MARCA num LUGAR pode ser interpretada como a indicação da execução de uma atividade num processo, algum estado de um equipamento, o estado de prontidão de uma parte para o próximo processo ou uma peça sendo processada. Por exemplo, num LUGAR relativo a um recurso, a presença de MARCAS pode indicar a disponibilidade do recurso ou a ocorrência de problemas enquanto que, se o LUGAR representar um estado de processo, a presença de MARCAS pode indicar que o processo está sendo executado ou suspenso temporariamente.

Também deve-se considerar que um mesmo sistema de manufatura pode ser modelado de várias formas utilizando RP. Por exemplo, podemos modelar um sistema por processos (ou fluxo de material) ou por recursos (seqüência de processos que acontece em cada recurso). Para exemplificar esta característica, considera-se o sistema de manufatura representado na figura III.7 composto por duas máquinas (M1 e M2) que processam 3 tipos de peças ( $w_1$ ,  $w_2$  e  $w_3$ ).

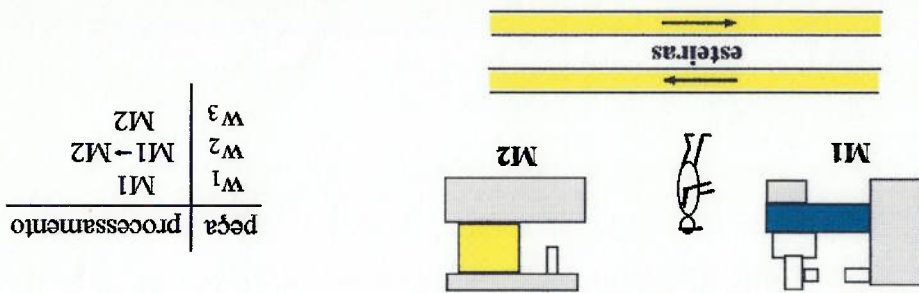
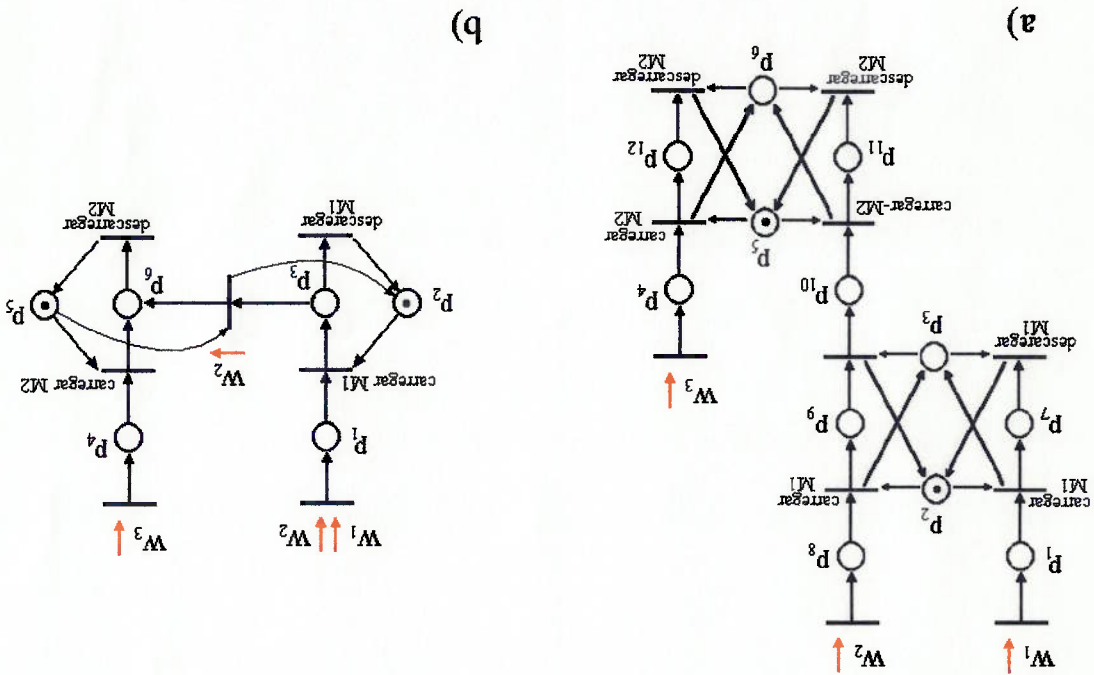


FIG. III.7 - Sistema de manufatura com 2 máquinas-ferramentas.



- $w_1$  são peças somente processadas por M1.
- $w_2$  são peças inicialmente processadas por M1 e posteriormente por M2.
- $w_3$  são peças processadas por M2.

O sistema é modelado na figura III.8-a) por uma RP segundo o fluxo de materiais (peças), enquanto que na figura III.8-b) o sistema é modelado por uma RP baseada nos processos que envolvem os recursos M1 e M2. Cada modelo representa características particulares do sistemas e sua escolha depende das especificações de projeto que serão analisadas.



Lugar	Descrição
$p_1$	$w_1$ aguardando ser carregada em M1
$p_2$	M1 livre
$p_3$	M1 processando
$p_4$	$w_3$ aguardando ser carregada em M2
$p_5$	M2 livre
$p_6$	M2 processando
$p_7$	$w_1$ sendo processada em M1
$p_8$	$w_2$ aguardando ser carregada em M2
$p_9$	$w_2$ sendo processada em M1
$p_{10}$	$w_2$ aguardando ser carregada em M2
$p_{11}$	$w_2$ sendo processada em M2
$p_{12}$	$w_3$ sendo processada em M1

Fig. III.8 - a) modelo em RP dos processos. b) modelo em RP baseado nos recursos.



III.4 REDE DE PETRI COLORIDA.

A rede de Petri colorida tenta manter uma representação compacta em casos nos quais deve ser especificado o fluxo de diversos itens na mesma sequência de processos. Assim, estes itens (ou elementos) não são mais representados por MARCAS de um mesmo tipo. Ao contrário, cada elemento no modelo deve ser representado por um tipo específico de MARCA. Da mesma forma, pode-se definir ARCOS com inscrições fixas, onde se estabelece que um tipo de MARCA pode "fluir por um ARCO" somente se esta MARCA corresponde ao tipo indicado na inscrição do ARCO. O fato de diversos ARCOS que entram e saem de uma TRANSIÇÃO terem inscrições, indica por onde as MARCAS fluirão.

A figura III.9 ilustra o princípio do fluxo de MARCAS pelo disparo de uma TRANSIÇÃO numa rede com MARCAS individuais e ARCOS com inscrições fixas.

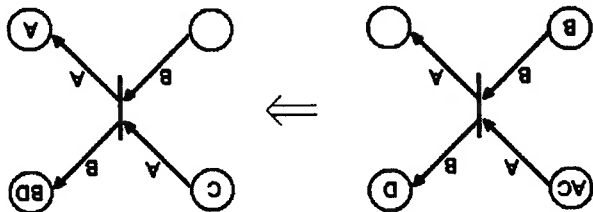


Fig. III.9 - Rede com ARCOS com inscrições fixas.

Entretanto, o aumento no número de itens no sistema, equivalente ao número de tipos de MARCAS resulta no aumento do número de TRANSIÇÕES necessárias para definir o fluxo de cada MARCA separadamente [Miyagi, 96].

No caso de TRANSIÇÕES com os mesmos LUGARES de entrada e saída (pré- e pós-condições), os ARCOS podem possuir inscrições variáveis que podem ser alteradas para indicar um fluxo de uma MARCA em particular.

Uma vez conseguido isto, o próximo passo é a geração dos códigos e rotinas para implementação de controladores programáveis (do tipo PLC), onde os programas tradicionalmente têm sido desenvolvidos com diagramas lógicos de reles (LDR) ou linguagens procedurais (como assembler). No entanto, o "software" escrito nestas linguagens de programação não têm suficiente flexibilidade e comprometem sua manutenção, já que compreender as especificações de operação num programa deste tipo é relativamente difícil para pessoas sem uma considerável experiência.

*reversível.*

*Dadas as especificações de um sistema de manufatura, modelar o sistema utilizando RP de forma que sua estrutura e sua marcação inicial definam uma rede limitada, viva e*

RP pode ser colocado como:

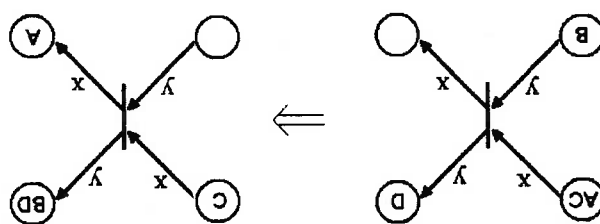
O problema geral para modelagem e análise de um sistema de manufatura utilizando

### III.5 OUTRAS EXTENSÕES DA REDE DE PETRI.

precisa ser modificada.

Assim, torna-se possível introduzir a representação de mais itens, de tal forma que cada novo elemento é tratado como um novo tipo de MARCA e a estrutura da rede não

Fig. III.10 - Rede com ARCOS de inscrições variáveis, quando  $x=A$  e  $y=B$ .



Para resolver estas dificuldades, representações derivadas da RP têm sido propostas.

Entre estas se destacam:

- GRAFCET, proposto na França em 1977 como uma interpretação de RP para programação de autómatos como um método de representação padronizada da estrutura de controle para controladores programáveis [Murata, 86].

- C-NET, uma aplicação desta interpretação de RP foi descrita em [Murata, 86]. Esta rede é definida por  $CN = \{P, T, I, O, \delta, \phi, \eta, U, V, M_0\}$ , onde:

$P$  = conjunto finito de LUGARES da rede.

$T$  = conjunto finito de TRANSIÇÕES

$I$  = função que define os ARCOS de entrada em relação às TRANSIÇÕES.

$O$  = função que define os ARCOS de saída em relação às TRANSIÇÕES.

$M_0$  = marcação inicial.

$\delta(p) = x_i$  é o sinal de controle produzido pela chegada de uma MARCA no LUGAR  $p_i$  que executará uma ação no equipamento especificado.

$\phi(p) = \{y_{k1}, \dots, y_{kK}\}$  é o sinal de resposta indicando que a ação foi finalizada,  $k$  é o número de tipos de sinais dependendo da ação executada.

$\eta(t) = y_{ik}$  é o sinal utilizado para habilitar uma atividade ou TRANSIÇÃO ( $t$ ),

usualmente 0/1.

$$U = \begin{cases} 0 & \text{se for detectado } x_i \in \delta \\ \theta_k & \text{se for detectado } y_{ik} \in \phi \end{cases}$$

$$V = \begin{cases} 0 & \text{se } t \text{ não estiver habilitada} \\ 1 & \text{se } t \text{ estiver habilitada.} \end{cases}$$

Gráficamente, estes novos elementos podem ser representados conforme ilustrado na

figura III.11

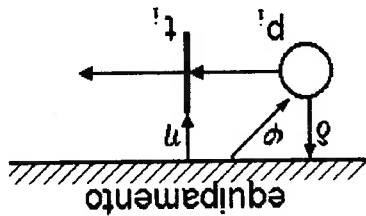


Fig. III.11 - Representação dos elementos de uma C-NET.

Esta C-NET foi instalada num microcomputador com seu respectivo editor/monitor, tabelas de controle e interfaces, formando uma estação controladora (SCR) utilizada em aplicações específicas ou pode ser auxiliada por uma linguagem baseada em regras (por exemplo IF-THEN, do tipo Pascal ou C++) para projetar controladores de propósito geral para sistemas de manufatura, assim reduziu-se o número de horas-homens necessárias para projetar o "software" de controle e as horas-homens necessárias para manutenção. Controladores baseados neste tipo de representação foram produzidos comercialmente pela HITACHI CO.

- "Mark Flow Graph" (MFG), é uma representação interpretada da RP na qual foram inseridas algumas alterações que simplificam a interface entre diferentes dispositivos do sistema. Em [Miyagi, 88] propõe-se uma metodologia para descrição dos processos num sistema de manufatura baseado num grafo ("Production Flow Schema" PFS) para desenvolver uma estrutura hierárquica seguindo uma abordagem "top-down", que permite o detalhamento do sistema desde o nível conceptual até o nível de implementação e inclui especificações das interações com sensores e atuadores. Também, foi desenvolvida uma linguagem de programação, que descreve textualmente um modelo de um sistema baseado em PFS/MFG, no qual as atividades paralelas e assíncronas são descritas hierarquicamente. Assim, a partir da representação gráfica em

PFS/MFG, tem-se uma descrição hierárquica do sistema que pode ser carregada e executada em simuladores ou controladores baseados em MFG já existentes [Miyagi, 88].

### III.6 COMPARAÇÃO DE RP E DIAGRAMA LÓGICO DE RELES (LTD)

Tradicionalmente, um diagrama lógico de reles (LTD) é utilizado para capturar a sequência das operações executadas pelo “*software*” do sistema, especificando os procedimentos de entrada/saída (I/O) para um controlador programável que executa as ações sobre os processos do sistema.

Sua utilização é limitada para o controle e não é feita para análise e avaliação das características qualitativas e de desempenho. Além de que o incremento na complexidade e necessidades de flexibilidade dos modernos sistemas de manufatura são condições que implicam numa constante “manutenção”, difícil de ser implementadas nesta técnica.

considerando uma comparação entre as representações gráficas de condições lógicas como: “AND”, “OR” e outros conceitos como concorrência, “*delay*” e sincronização, isoladamente, a complexidade na representação é similar, veja tabela III.1.

Assim, se a complexidade gráfica for principalmente determinada pelo número de nós e elementos de ligação, a representação destas condições aparentemente, seria equivalente nas duas metodologias (RP do tipo C/E e LTD). Em [Vankatesh, 94] são apresentados exemplos de comparação, nos quais se pode concluir que para sistemas simples a representação por RP fica aproximadamente 20% menor que a representação por LTD. Na medida que são incrementadas especificações no sistema, o modelo por RP pode ficar até um 50% menor.

Tabela III.1 Representação da construção lógica por RP e LLD.

Construção lógica	Rede de Petri	Diagrama lógico de relés
AND IF A=1 and B=1 and C=1 THEN D=1		
OR IF A=1 or B=1 THEN D=1		
Concorrência IF A=1 and B=1 THEN C=1 and D=1 and E=1		
Delay IF A=1 THEN Delay B=1		
Sincronização IF A=1 THEN D=1 IF B=1 and C=1 THEN E=1 IF D=1 and E=1 THEN F=1		

Entretanto, em relação à adaptabilidade para mudanças nas especificações, a RP é evidentemente superior, já que quando baseadas num modelo inicial, a RP requer um incremento menor do número de nós e ARCOS de ligação para atingir novas especificações. No entanto, estas características são influenciadas por fatores como experiência do projetista com cada metodologia, especificações de operação e número de passos dinâmicos para codificar ou mudar o programa [Vankatesh, 94]. Na análise qualitativa de LLD, não é possível revisar propriedades, tais como ausência de bloqueios e reversibilidade antes que o LLD seja simulado ou implementado.

A partir de um modelo já desenvolvido, em RP consegue-se incluir novas especificações, como por exemplo a manipulação de falhas, de uma maneira relativamente fácil e direta, já que em muitos casos o tratamento de falhas e erros é realizado através de sub-redes acrescentadas num modelo originalmente projetado sem estas características.

## LISTA DE SIMBOLOS

$P = \{p_1, p_2, \dots, p_m\}$  é o conjunto de LUGARES da rede, onde  $m \in \mathbf{N}$  é o número total de LUGARES da rede.

$T = \{t_1, t_2, \dots, t_n\}$  é o conjunto de TRANSIÇÕES,  $n \in \mathbf{N}$  é o número de TRANSIÇÕES da rede.

$I : (P, T)$  é a função que define os ARCOS de entrada em relação às TRANSIÇÕES. Se  $I(p_j, t_i) = k_i$ , existem  $k_i \in \mathbf{N}$  ARCOS orientados do LUGAR  $p_j$  para a TRANSIÇÃO  $t_i$ , e quando  $I(p_j, t_i) = 0$ , não existe ARCO orientado do LUGAR  $p_j$  para a TRANSIÇÃO  $t_i$ .

$O : (T, P)$  é a função que define os ARCOS de saída em relação às TRANSIÇÕES.

$M = (M(p_1), M(p_2), \dots, M(p_m))$  onde  $m \in \mathbf{N}$  é o número total de LUGARES.  $M(p_i)$  é o número de MARCAS do LUGAR  $p_i$  ( $M(p_i) \in \mathbf{N}$ ).

$M_k$ : vetor coluna ( $m \times 1$ ) que representa a marcação imediatamente alcançável a partir de  $M_{k-1}$  após o disparo da TRANSIÇÃO  $t_i$  (equação III.1).

$u_k$ : vetor ( $n \times 1$ ) com grandeza de 1 na posição  $i$  que representa o disparo da TRANSIÇÃO  $t_i$  e zero (0) nas outras posições.

$A$ : matriz de incidência ( $n \times m$ ) definida por  $A = O(P, T) - I(P, T)$

$\delta(p_i) = x_i$ , é o sinal de controle produzido pela chegada de uma MARCA no LUGAR  $p_i$  e que executará uma ação no equipamento especificado.

$\phi(p) = \{y_{k1}, \dots, y_{kk}\}$  é o sinal de resposta indicando que a ação foi finalizada,  $k$  é o número de tipos de sinais dependendo da ação executada.

$\eta(t_i) = y_{ki}$ , é o sinal utilizado para habilitar uma atividade ou TRANSIÇÃO ( $t_i$ ), usualmente 0/1.

$$U = \begin{cases} 0 & \text{se for detectado } x_i \in \delta \\ \theta_k & \text{se for detectado } y_{ki} \in \phi \end{cases}$$

$$V = \begin{cases} 0 & \text{se } t_i \text{ não estiver habilitada} \\ 1 & \text{se } t_i \text{ estiver habilitada.} \end{cases}$$



## REFERÊNCIAS BIBLIOGRÁFICAS

- CHEN, Y.; WONGLADKOWN C. A real-time control simulator design for automated manufacturing systems using Petri nets. In: INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATIONS. Sacramento, CA, 1991. Proceedings of IEEE, s.l., p. 2542-2547.
- DWYER, Mathew B.; CLARKE, Lori A. A compact Petri net representation and its implications for analysis. IEEE Transactions on Software Engineering, v. 22, n. 11, p. 794-811, November, 1996.
- HASEGAWA, Kensuke. Modeling, control and deadlock avoidance of flexible manufacturing systems. In: CONGRESSO BRASILEIRO DE AUTOMÁTICA, CBA 11. São Paulo, 1996. Conferências Plenárias. São Paulo, SBA, 1996. p. 37-51.
- JENG, Mu Der. A Petri net synthesis theory for modeling flexible manufacturing systems. IEEE Transactions on Systems, Man, and Cybernetics, v. 27, n. 2, p. 169-183, April, 1997. (Part B: Cybernetics).
- MIYAGI, Paulo Eigi. Control System Design, Programming and Implementation for Discrete Event Production Systems by using Mark Flow Graph. Tokyo, 1988. 90p. Doctor thesis - Tokyo Institute of Technology.
- MIYAGI, Paulo Eigi. Controle Programável: fundamentos do controle de sistemas a eventos discretos. São Paulo, Edgard Blücher, 1996.
- MURATA, Tomihiro; KODOMA, Norihisa; MATSUMOTO, Kuniaki; HARUNA, Koichi. A Petri net-based controller for flexible and maintainable sequence control and its applications in factory automation. IEEE Transactions on Industrial Electronics, v. IE-33, n. 1, p. 1-8, February, 1986.
- MURATA, Tadao. Petri nets: properties, analysis and applications. Proceedings of the IEEE, v. 77, n. 4, p. 541-580, April, 1989.
- PETERSON, James Lyle. Petri Net Theory and The Modeling of Systems. Englewood Cliffs, Prentice-Hall, 1981.
- TANG, T. N.; ZHOU, MengChu. Development and applications of a Petri net simulation tool for manufacturing systems. In: JAPAN USA SYMPOSIUM ON FLEXIBLE AUTOMATIONS. San Francisco, 1992. Proceedings. San Francisco, 1992. p. 1199-1205

VANKATESH, Kurapati; ZHOU, MengChu; CAUDILL, Reggie J. Comparing ladder logic diagrams and Petri nets for sequence controller design through a discrete manufacturing systems. *IEEE Transactions on Industrial Electronics*, v. 41, n. 6, p. 611-619, December, 1994

ZENG, L.; WANG, H. P. N. Conflict detection of automated guided vehicles: a Petri net approach. *International Journal Production Research*, v. 29, n. 5, p. 856-879, 1991.

ZHOU, MengChu; DICESARE, Frank. *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems*. Kluwer Academic, Boston, 1993.

ZURAWSKI, Richard; ZHOU, MengChu. Petri nets and industrial applications: a tutorial. *IEEE Transactions on Industrial Electronics*, v. 41, n. 6, p. 567-583, December, 1994.