

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
PRODUÇÃO**

Reynaldo Chile Palomino

**Uma Abordagem para a Modelagem, Análise e Controle de
Sistemas de Produção Utilizando Redes de Petri**

Dissertação sub-metida à Universidade Federal de Santa Catarina para
a obtenção do Grau de Mestre em Engenharia



0.243.732-9


UFSC-BU

**Florianópolis
Agosto de 1995**

**"Uma Abordagem para a Modelagem, Análise e Controle de Sistemas de
Produção Utilizando Redes de Petri"**

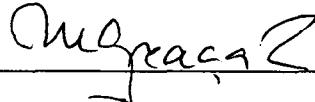
Reynaldo Chile Palomino

Esta dissertação foi julgada adequada para a obtenção do título de "Mestre em Engenharia"
Especialidade em Engenharia de Produção e aprovada em sua forma final pelo programa de
Pós-Graduação em Engenharia de Produção.

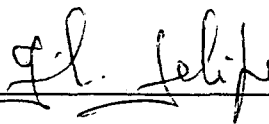


Prof. Osmar Possamai, Dr.
Coordenador do Curso


Banca Examinadora:



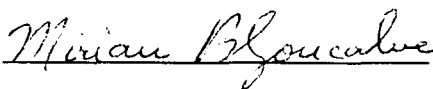
Prof.^a. Maria da Graça Pereira, Dr.
Orientadora



Prof. Julio Felipe Szeremeta, M. Sc.
Co-Orientador



Prof. Dalvio Ferrari Tubino, Dr.



Prof.^a. Mirian Buss Gonçalves, Dr.

Dedico este trabalho com muito carinho aos meus pais Justino e Mercedes pelo apoio recebido a cada momento, à minha irmã Gloria e à minha futura esposa Daniela pela imensa paciência, incentivo e amor dedicado.

AGRADECIMENTOS

Expresso aqui os meus agradecimentos:

- Ao professor Plinio Stange, pela orientação e incentivo no desenvolvimento inicial deste trabalho.
- À professora Maria da Graça Pereira, pela orientação na fase final deste trabalho.
- Ao professor Julio Felipe Szeremeta, pelo seu apoio, comentários e sugestões que permitiram o aperfeiçoamento do trabalho,
- Ao colega Luis Alberto Alfaro Casas pela amizade e estímulo durante o curso.
- Aos membros da banca examinadora por suas sugestões.
- Aos demais professores, funcionários e colegas do Programa de Pós-Graduação em Engenharia de Produção pela colaboração prestada.
- A Cristina por sua ajuda desinteressada na correção do português.
- À Universidade Federal de Santa Catarina e ao Conselho Nacional de Pesquisa e Desenvolvimento- CNPq que proporcionaram os meios para a realização deste trabalho.

SUMÁRIO

CAPÍTULO 1.- INTRODUÇÃO	1
1.1.- Considerações Gerais	1
1.2.- Evolução dos Sistemas de Manufatura.....	2
1.3.- Objetivos	3
1.4.- Importância do Trabalho	4
1.5.- Conteúdo do Trabalho.....	4
CAPÍTULO 2 - TÉCNICAS DE MODELAGEM DE SISTEMAS DE PRODUÇÃO	6
2.1.- Sistema	6
2.1.1 - Componentes de um sistema	6
2.2 - Produção.....	7
2.3 - Sistema de Produção	8
2.4 - Componentes do sistema de produção	9
2.5 - Tipos de sistemas de produção.....	10
2.5.1 - Sistema de Produção Contínua	11
2.5.2 - Sistema de Produção Intermitente	11
2.6.- Comparativo entre os Três Sistemas de Produção	12
2.7 - Modelos de Sistemas de Produção	13
2.7.1 - Representação dos Sistemas: Os Modelos.....	13
2.7.2.- Linguagens de modelagem de sistemas.....	15
2.7.3 - Tipos de Modelos.....	15
2.8 - Sistemas Discretos e Contínuos	16
2.8.1 - Sistemas Contínuos	17
2.8.2.- Comportamento dinâmico dos Sistemas de Variáveis Contínuas (SVC)	17
2.8.3 - Sistemas Discretos	17
2.8.4.- Comportamento dinâmico dos sistemas de eventos discretos (SED)....	19

2.9.- Sistemas de Manufatura	22
2.10.- Flexibilidade e o processo de manufatura	25
2.11.- Sistemas Flexíveis de Manufatura (FMS)	26
2.12.- Hierarquia de decisão e de controle	28
CAPÍTULO 3 - REDES DE PETRI	32
3.1.- Introdução.....	32
3.2.- Conceituação formal e gráfica	33
3.2.1.- Rede de Petri Ordinária	33
3.2.2.- Rede de Petri Marcada	34
3.3.- Representação Gráfica	35
3.4.- Representação matricial.....	35
3.5.- Funcionamento de uma Rede de Petri.....	37
3.5.1.- Sensibilização de uma transição	37
3.5.2.- Disparo de uma transição	37
3.5.3.- Seqüência de disparo.....	38
3.6.- Grafo de marcações acessíveis.....	39
3.7.- Principais Propriedades das Redes de Petri	40
3.7.1.- Propriedades Básicas	41
3.7.2.- Propriedades específicas.....	44
3.7.3. - Invariantes	48
3.7.4.- Exemplo de propriedades de uma Rede de Petri	51
3.8.- Técnicas de análise das Redes de Petri.....	52
3.8.1 - Análise por Enumeração das Marcações.....	53
3.8.1.1.- Árvore de Cobertura	53
3.8.1.2.- Análise pela árvore	55
3.8.1.3.- Limitações da árvore de alcançabilidade na análise das RdP	56
3.8.2.- Análise estrutural das Redes de Petri	57

3.8.2.1.- Verificação das propriedades.....	59
3.8.2.2.- Verificação das propriedades das RdP através dos invariantes.....	62
3.8.3- Análise por redução	63
CAPÍTULO 4 - ABREVIACÕES, EXTENSÕES E ESTRUTURA PARTICULAR DAS REDES DE	
PETRI.....	65
4.1.- Introdução.....	65
4.2.- Abreviações	67
4.3.- Extensões.....	71
4.4.- Relação entre as Redes de Petri temporais.....	80
4.4.1.- Relação com o modelo de Ranchandani	80
4.4.2.- Relação com o modelo de sifakis	80
4.5.- Conclusão.....	82
CAPÍTULO 5.- AS REDES DE PETRI E SUA RELAÇÃO COM OS SISTEMAS DE PRODUÇÃO	
.....	83
5.1 - Introdução.....	83
5.1.1.- Modelagem e análise	86
5.1.2.- Avaliação da performance de Sistemas de Manufatura.....	87
5.1.3.- Implementação de controle em tempo real.....	87
5.1.4.- Simulação e ferramentas de software.....	88
5.2.- Aplicação das Redes de Petri a Sistemas Flexíveis de Manufatura.....	89
5.3- Ligação entre as Redes de Petri e o grafo PERT.....	91
5.4 - Alocação de recursos em Redes de Petri.....	94
5.5- Detecção de situações de deadlock utilizando Redes de Petri	98
5.6- Significado das fichas num sistema de Montagem	99
5.7.- Síntese de Redes de Petri par FMS	101
5.7.1. - Metodologia para síntese dos modelos de Redes de Petri	102

5.7.2.- Módulos de desenho básico	104
5.7.3.- Descrição do processo de sínteses.....	105
5.7.4.- Desenho de modelos de RdP de primeiro nível.....	106
5.8.- Análise estrutural dos Grafos de Eventos de Sistemas de Produção.....	108
5.8.1.- Avaliação da Performance.....	109
5.8.2.- Funcionamento em velocidade máxima	111
CAPÍTULO 6.- APLICAÇÃO DAS REDES DE PETRI: MODELAGEM E ANÁLISE DE UM SISTEMA DE MONTAGEM COM REDES DE PETRI TEMPORIZADA	115
6.1.- Introdução.....	115
6.2.- aplicação.....	117
6.3.- Conclusões.....	132
CAPÍTULO 7 - CONCLUSÕES E RECOMENDAÇÕES	134
7.1.- Conclusões.....	134
7.2.- Recomendações para futuros trabalhos	136

LISTA DE FIGURAS

Fig. 2.1- Exemplo de um Sistema Produtivo. Qualquer parte desse sistema constitui um subsistema.....	7
Fig. 2.2- Esquema de um Sistema de produção.....	9
Fig. 2.3- As Interdependências do Almojarifado, produção e depósito no sistema de produção.....	9
Fig. 2.4- Gradações que Apresentam os Três Sistemas de Produção.....	13
Fig. 2.5- Fases das Atividades no Processo de Modelagem de Sistemas.....	14
Fig. 2.6- Trajetória de um Sistema Dinâmico de Variável Contínua (SDVC).....	17
Fig. 2.7- Trajetória de um Sistema Dinâmico de Eventos Discretos (SDED).....	18
Fig. 2.8- Diversos Tipos de Sistemas Discretos.....	19
Fig. 2.9- Linha de Manufatura Flexível.....	23
Fig. 2.10- Representação de uma Estação de Trabalho.....	23
Fig. 2.11- Representação de uma Célula de Manufatura.....	24
Fig. 2.12- Representação de um Sistema Integrado de Manufatura.....	24
Fig. 3.1- Relações de Causalidade Modeladas por Redes de Petri.....	30
Fig. 3.2- Rede de Petri Marcada.....	35
Fig. 3.3- Disparo de uma transição.....	38
Fig. 3.4- Grafo de marcações Acessíveis.....	40
Fig. 3.5- Rede de Petri Marcada com k-limitada.....	42
Fig. 3.6- Exemplo de uma Rede de Petri não Reinicializável.....	43
Fig. 3.7- Grafo de Marcações Acessíveis da Rede não Reinicializável.....	43
Fig. 3.8- Uma Rede de Petri Representando uma Atividade Paralela.....	44
Fig. 3.9- Representação de uma Situação de Conflito.....	45
Fig. 3.10- Dois Tipos de Confusão a) Confusão Simétrica e b) confusão Assimétrica.....	46
Fig. 3.11- Transição não Primitiva.....	47
Fig. 3.12- Hierarquia numa Rede de Petri.....	48
Fig. 3.13- Exemplo de Invariantes numa Rede de Petri.....	51

Fig. 3.14- Exemplos de Redes de Petri	51
Fig. 3.15- Rede de Petri Marcada com Possível Deadlock.....	55
Fig. 3.16- a) Árvore de Alcançabilidade e b) Grafo de Marcações Acessíveis para a Figura 3.15	55
Fig. 3.17- Rede de Petri sem Deadlock	57
Fig. 3.18- Rede de Petri	58
Fig. 3.19- Rede de Petri não Conservativa	60
Fig. 3.20- Rede de Petri mostrando condição necessária mas não suficiente.....	62
Fig. 4.1- Grafo de Estado Fortemente Conectado.....	67
Fig. 4.2- Rede de Petri: a) Sistema FIFO, b) RdP Ordinária, c) RdP Colorida	70
Fig. 4.3- Rede de Petri com Capacidade Finita	71
Fig. 4.4- Rede de Petri com Arco Inibidor	71
Fig. 4.5- RdP Continua.....	72
Fig. 4.6- Rede de Petri Híbrida.....	74
Fig. 4.7- Rede de Petri Sincronizada	75
Fig. 4.8- Rede de Petri Temporizada	76
Fig. 4.9- Rede de Petri Temporal	78
Fig. 4.10- Transformação de uma Transição Temporizada	80
Fig. 4.11- Transformação das Redes de Petri.....	81
Fig. 5.1- Rede PERT A-on-A.....	91
Fig. 5.2- Rede PERT A-on-N.....	92
Fig. 5.3- Rede de Petri equivalente a uma Rede PERT	93
Fig. 5.4- Alocação de Recursos em Redes de Petri.....	95
Fig. 5.5- Compartilhamento de Recursos a) Exclusão Mútua Sequencial, b) Exclusão Mútua Paralela	96
Fig. 5.6- Alocação de um único Recurso em RdP	97
Fig. 5.7- Ocorrência de um Deadlock	99
Fig. 5.8- Definição de uma Seqüência PN ($n > 0$).....	104

Fig. 5.9- Dois modelos de RdPs de primeiro nível. a) Rede com um unica lugar e transição b) Rede com um unico Recurso e Operação c) Caso Geral.....	107
Fig. 5.10- RdP de Primeiro Nivel para Sistemas com uma Estrutura Sincronização-Escolha .	108
Fig. 5.11- Grafo a Eventos do Sistema e a Evolução da sua Execução a partir do Estado Inicial.	113
Fig. 6.1- Linha de Montagem com 10 Máquinas.....	116
Fig. 6.2- Linha de Montagem com 4 Maquinas.....	117
Fig. 6.3- Layout do Sistema de Manufatura Automatizado da figura 6.2.....	118
Fig. 6.4- Rede de Petri de Primeiro Nivel para o Sistema de Montagem	120
Fig. 6.5- Rede de Petri após o Refinamento de 4 Lugares e 1 Transição.....	121
Fig. 6.6- Rede de Petri após a Adição de Recursos não Compartilhados.....	122
Fig. 6.7- Modelo de Rede de Petri Final para o Sistema de Montagem	125

LISTA DE TABELAS

Tab. 2.1- Exemplos de Sistemas de Produção.....	8
Tab. 2.2- Classificação dos modelos de DEDS.....	21
Tab. 6.1- Descrição dos Lugares e Transições da RdP da Figura 6.4.	120
Tab. 6.2- Significado dos Lugares na Rede de Petri Final.....	124
Tab. 6.3- L-invariantes da Rede de Petri para a Rede da Figura 6.6.....	126
Tab. 6.4- Tempo de Ciclo para cada Circuito Elementar.....	128
Tab. 6.5- Tempo de Ciclo para cada Circuito da Rede da Figura 6.6	129

RESUMO

As Redes de Petri estão sendo atualmente utilizadas em Sistemas de Manufatura. Neste sentido, o presente trabalho tem como propósito apresentar os diferentes conceitos relativos às Redes de Petri e suas várias classes de modelos derivados, os quais podem ser utilizados para modelar Sistemas (dinâmicos) de Eventos Discretos de qualquer tipo. Esta ferramenta nos permite visualizar com facilidade situações tais como concorrência, sincronização e compartilhamento de recursos os quais são próprios de um Sistema de Manufatura, além do mais, nos permite uma análise tanto qualitativa como quantitativa. Cada um dos modelos derivados tais como RdP Temporizadas, RdP Estocásticas, RdP Colorida, etc, têm seu próprio caráter específico e campo de aplicação privilegiado.

ABSTRACT

Nowadays Petri Nets has been applied in Manufacturing System. Concerning to this area, the objectives of the present research is to introduce the differents concepts related to Petri Nets and its various classes of derived model which can be used for modeling (dynamic) discret events system of any type. This tool help us to easily visualize situations such as concurrency, synchronization and sharing of resource, which are peculiar to a FMS. In addition to, this tool allows us to enables qualitative and quantitative analysis. Each one of the derived models such as Timed, Stochastic, coloured Petri Net, has its own specified character and privileged fields of application.

CAPÍTULO 1.- INTRODUÇÃO

1.1.- Considerações Gerais

A evolução do processo de manufatura tem passado nestes últimos tempos por radicais modificações, a tal ponto que as teorias de administração e organização da manufatura que vieram com a revolução industrial começam a perder espaço para uma nova forma de organização: os Sistemas Flexíveis de Manufatura.

Até o final dos anos setenta a automação fabril desenvolveu-se basicamente através de linhas de produção do tipo flow-shop (ou produção em massa), que por muito tempo foi um dos fatores principais da geração de riqueza dos países altamente industrializados.

O advento do computador nos anos quarenta, acompanhado de uma rápida evolução do mercado com clientes (público consumidor) cada vez mais exigentes e variados, possibilitou o surgimento de sistemas de manufatura altamente automatizados e flexíveis, com a capacidade de tratar uma variedade de produtos de volume médio e admitir rápidas alterações no processo alcançando alta produtividade sem perder a qualidade dos produtos.

Os Sistemas Flexíveis de Manufatura correspondem a esta tentativa de satisfazer tais exigências do mercado, constituindo-se assim na mais moderna concepção para sistemas de produção integrados por computador com capacidade de gerar produtos de volume médio com elevada qualidade e baixo custo.

É assim que nestas últimas décadas o rápido desenvolvimento tecnológico da automação industrial trouxe como consequência um considerável aumento do grau de complexidade de sistemas de manufatura nos quais o número de atividades que ocorrem de forma paralela ou concorrente é muito grande. Conseqüentemente elas devem ser controladas apropriadamente para assim garantir o funcionamento correto do sistema como um todo.

1.2.- Evolução dos Sistemas de Manufatura

Até a criação da máquina a vapor a produção caracterizou-se pela atividade individual assim como pelo uso da energia muscular em lugar da mecânica.

No século XVIII com o uso da energia gerada pelo vapor, as condições de trabalho mudaram rapidamente com prejuízo para o operário porque houve uma efetiva redução do elemento humano como fonte de força. Logo com o advento das máquinas ferramentas que passaram a realizavam grande parte do trabalho manual, o impacto causado grande, uma vez que estas apenas necessitavam de um operador para executar tarefas relacionadas ao setup inicial, carga e descarga dos pallets, etc. Os problemas relacionados ao funcionamento da máquina atingiram diretamente os operadores de usinagem. Outro passo significativo que afetou a interface homem-máquina foi o desenvolvimento de controle de máquinas por sofisticados servomecanismos (dos Santos F^o 1993), concedendo um maior grau de autonomia a estas unidades de produção, exigindo nesta etapa maior capacitação e conhecimento por parte dos operadores para lidar com uma quantidade muito maior de informações com vistas à especificação e manutenção do controle das respectivas máquinas. Neste ponto, o desenvolvimento dos computadores eletrônicos digitais proporcionou uma alternativa para a solução deste problema de tratamento de dados. Logo, a evolução dos computadores e uma redução dos custos permitiram o desenvolvimento das chamadas "Máquinas ferramentas de Controle Numérico" (Máquinas N.C.) operando com trocas automáticas de ferramentas. Os operadores, embora em número reduzido, passam a ter um maior grau de responsabilidade operacional, sendo que começam a operar diferentes máquinas, tornando-se responsáveis pela sua operação e acompanhamento de todas as fases de fabricação do produto. A responsabilidade direta do elemento humano no ambiente de manufatura passa a ser portanto mais restrita e substancialmente reduzida. Desse modo, a participação do homem nos processos de manufatura evoluiu de um simples operador de máquinas para um supervisor de produção, realizando operações básicas relacionadas à monitoração global dos processos.

O próximo passo foi o desenvolvimento de sistemas de manufatura totalmente automatizados a partir da integração de máquinas ferramentas e centros de usinagem controlados por um computador central. Para a realização do fluxo de peças, material bruto e ferramentas, utilizam-se robôs, veículos

guiados automaticamente (AGV), esteiras, e principalmente os computadores e sistemas de monitoramento e controle de processos, os quais devem ter suas funções perfeitamente sincronizadas e definidas.

Um fato importante que se pode observar nos Sistemas de Manufatura Automatizados é que o comportamento das atividades possui uma natureza predominantemente discreta, como será visto posteriormente, onde os estados do sistema evoluem conforme a ocorrência de um ou mais eventos que podem ou não estarem mutuamente relacionados (Miyagi 1989). Estas características fazem com que os sistemas complexos sejam praticamente impossíveis de serem projetados, previamente analisados e implementados através de técnicas tradicionais de controle. As principais dificuldades estão relacionadas com a inadequação dos modelos para tratar sistemas concorrentes e assíncronos com alto grau de complexidade. Desta forma, necessita-se uma técnica de modelagem que possa ser útil para a representação das características de um Sistema Flexível de Manufatura capaz de contribuir para o planejamento e realização das atividades como: supervisão do sistema, diagnose e reparo de falhas no sistema, carga e descarga de pallets, monitoramento do sistema, preparação das máquinas e ferramentas, etc. Neste sentido as Redes de Petri são uma ferramenta de modelagem de eventos discretos capazes de representar com muita simplicidade atividades paralelas e/ou concorrentes assim como assíncronas, sendo essa a razão da sua importância na automação industrial.

1.3.- Objetivos

O objetivo da presente dissertação é apresentar os diferentes conceitos relativos às Redes de Petri assim como suas várias classes de modelos derivados, os quais podem ser usados para modelar, analisar e controlar sistemas dinâmicos e avaliar o desempenho dos mesmos, mostrando ao mesmo tempo as vantagens a respeito de outras ferramentas de modelagem que, com o desenvolvimento da automação industrial e das fábricas flexíveis, começaram a ter pouca utilidade por não ter capacidade para representar situações complexas, como é o caso dos Sistemas Flexíveis de Manufatura.

1.4.- Importância do Trabalho

Desde a sua criação em 1962, a Rede de Petri têm sido utilizadas como ferramenta de especificação de sistemas utilizados em diversas áreas do conhecimento, em especial nos sistemas computacionais. Ela tem tido crescente aplicação em protocolos de comunicação, desenvolvimento de software e hardware, sistemas distribuídos, etc (Peterson 1981),(Murata 1989). Posteriormente, com a inclusão do tempo no modelo original, ampliou-se o campo de aplicações que, junto com sua capacidade de representar atividades concorrentes e assíncronas, permitem a modelagem e análise de sistemas complexos de produção, os quais resultam difíceis de serem modelados por técnicas tradicionais como Filas de Espera, Cadeias de Markov, Redes Pert, etc.

Para enriquecer o seu poder de modelagem muitas extensões foram propostas, todas baseadas nas Redes de Petri, mas com diferenças na maneira de associar informações às fichas e de representação do tempo. É deste modo que as Redes de Petri vêm cada vez mais ganhando importância na análise e avaliação do desempenho de sistemas.

O desenvolvimento recente da automação fabril e a criação das fábricas flexíveis têm conduzido a diversos estudos relativos à modelagem, concepção e análise destes sistemas. Atualmente diversas pesquisas estão em andamento para a especificação de novas ferramentas de modelagem com vistas ao aperfeiçoamento de modelos existentes. Neste sentido é que apresentamos uma abordagem relativa às Redes de Petri para a modelagem, análise e controle de Sistemas de Manufatura Flexível.

1.5.- Conteúdo do Trabalho

No capítulo 2 será feita uma revisão sobre os diferentes sistemas de produção e a importância de se utilizar modelos para a sua representação e análise. Também será vista a natureza discreta dos Sistemas Flexíveis de Manufatura, a qual facilita sua modelagem com Redes de Petri.

No capítulo 3 são apresentadas as Redes de Petri no modelo original, assim como aspectos relacionados à sua modelagem, propriedades e técnicas de análise. Já no capítulo 4 serão apresentadas

as diversas classes de Redes de Petri, abreviações e extensões feitas ao modelo original para aumentar o seu poder de representação.

O capítulo 5 apresenta a utilização das Redes de Petri como uma ferramenta de modelagem, análise e controle dos Sistemas Flexíveis de Manufatura e sua relação com outros modelos, destacando as vantagens de sua utilização. Será apresentada também uma metodologia híbrida para síntese de modelos de Redes de Petri em Sistemas de Manufatura que, se aplicada corretamente, pode evitar o cálculo das propriedades básicas de uma Rede de Petri (vivacidade, limitação e reversibilidade), já que estas propriedades tornam-se inerentes à Rede final obtida.

No capítulo 6 será apresentada uma aplicação das Redes de Petri para a modelagem e análise de uma linha de montagem utilizando-se primeiro a metodologia híbrida apresentada no capítulo anterior e verificando-se posteriormente as propriedades da Rede, utilizando-se para isso o analisador de Redes de Petri (ARP), desenvolvido no Laboratório de Controle e Microinformática do Departamento de Engenharia Elétrica da UFSC.

Finalmente o capítulo 7 será reservado para as conclusões e recomendações deste trabalho.

CAPÍTULO 2 - TÉCNICAS DE MODELAGEM DE SISTEMAS DE PRODUÇÃO

2.1.- Sistema

Para modelar um sistema é necessário entender o conceito de sistema assim como o limite do mesmo. A seguir definiremos o que se entende por sistema de um modo geral.

Segundo (Elmaghraby 1966; Riggs 1993; Stafford 1971), sistema é qualquer coleção de interação de elementos que funciona para alcançar um objetivo comum e que evoluiu no tempo.

A definição acima indica pois que aquilo que pode ser definido como sistema num contexto, pode ser apenas um componente em outro dando origem deste modo aos subsistemas. Por exemplo: para um projetista de automóveis, o automóvel é um sistema mecânico composto de eixos de manivelas, bloco de cilindros, pistões, etc. Já para um engenheiro civil, projetando um sistema de estrada, o automóvel é apenas um componente desse sistema o qual inclui caminhões, ônibus e outros.

Assim o universo parece estar formado de conjuntos de sistemas cada qual contido em outro ainda maior.

Já para Greene (1968). Sistema é um conjunto complexo de coisas diversas que ordenadamente relacionadas entre si, contribuem para determinado objetivo ou propósito.

Sob um ponto de vista mais prático (Chiavaneto 1983), define um sistema como um conjunto de elementos dinamicamente relacionados entre si, formando uma atividade para atingir um objetivo, operando sobre entradas (informação, energia ou matéria) e fornecendo saídas (informação, energia ou matéria) processadas.

2.1.1 - Componentes de um Sistema

Todo sistema apresenta os seguintes componentes listados a seguir e podem ser vistos melhor na figura 2.1.

- a) **Fronteiras:** São os limites do sistema, que podem ter existência física ou apenas uma delimitação imaginária para efeito de estudo.
- b) **Subsistemas:** São os elementos que compõem o sistema.
- c) **Entradas (input):** Representam os insumos ou variáveis independentes do sistema.
- d) **Saídas (outputs):** Representam os produtos ou variáveis dependentes do sistema.
- e) **Processamento:** Engloba as atividades desenvolvidas pelos subsistemas que interagem entre si para converter as entradas em saídas.
- f) **Retroação (feedback):** É a influência que as saídas do sistema exercem sobre as suas entradas no sentido de ajustá-las ou regulá-las ao funcionamento do sistema.

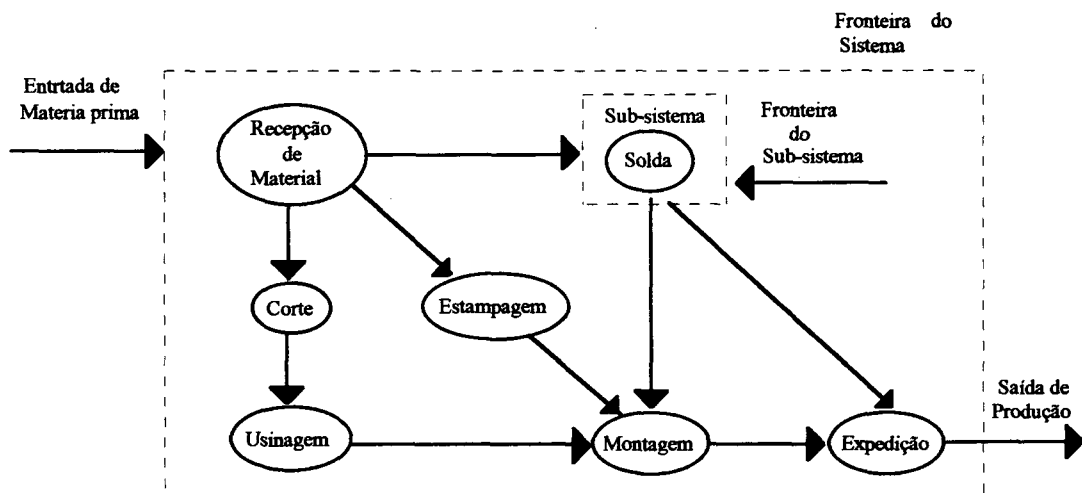


Fig. 2.1.- Exemplo de um sistema produtivo. Qualquer parte desse sistema um subsistema (Lida 1978)

2.2 - Produção

Segundo Riggs (1993), produção é o ato intencional de produzir alguma coisa útil.

(Cenip 1988; Elmaghraby 1966) definem produção como o incremento da utilidade de um objeto ou um serviço. Entendemos produção de outro modo como a adição ou incremento de valor a um bem (produto ou serviço) por efeito de uma transformação (ver Figura 2.2).

Ampliando um pouco mais o conceito de produção, ela pode ser definida como um processo que permite a transformação de algum insumo num determinado produto, o qual por sua vez pode resultar sendo insumo de outro processo. Para permitir esta transformação usa-se um conjunto de elementos (máquinas, mão de obra, ferramentas, meios de transporte, etc), que em conjunto vêm a compor o chamado sistema de produção.

A produção, segundo sua natureza, pode classificar-se em três tipos a saber: Produção primária (ou extrativa) secundária (ou de transformação) e terceárias (ou prestadoras de serviço) (Chiavenato 1991)

2.3 - Sistema de Produção

Toda empresa, para poder funcionar, adota um sistema de produção com vistas a realizar suas operações e produzir seus produtos ou serviços da melhor maneira possível, e com isto garantir sua eficiência e eficácia.

Assim Riggs (1993) define um sistema de produção como: Um processo planejado pelo qual os elementos são transformados em produtos úteis, isto é um procedimento organizado para se conseguir a conversão de entradas (insumos) em saídas (produtos acabados), como mostrado na tabela 2.1.

ENTRADA	PROCESSO DE CONVERSÃO	SAÍDAS
Insumos	Transformação	Produtos úteis
Materiais	Máquinas	Produtos
Dados	Interpretação	Conhecimento
Custos variáveis	Custos fixos	Lucro
Queda de água	Usina	Energia elétrica

Tabela 2.1 - Exemplos de sistemas de produção

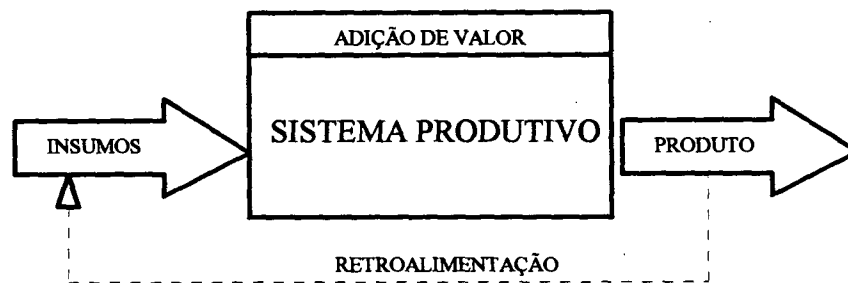


Fig. 2.2- Esquema de um sistema de produção

Um Sistema de produção (figura 2.2), portanto, começa a tomar forma desde que se formula um objetivo e se elege o produto que se vai comercializar. Deste modo, toda empresa, como um sistema de produção, tem por finalidade organizar todos os setores que fazem parte da mesma para realizar suas operações de produção, adotando uma interdependência lógica entre todas as etapas do processo produtivo, desde que as matérias primas ou materiais deixam o almoxarifado até chegar posteriormente (depois de sofrer uma alteração) ao depósito de produtos acabados na qualidade de produto final, conforme pode-se observar na figura 2.3.

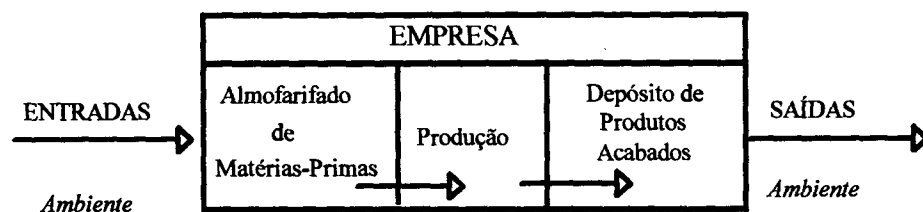


Fig. 2.3- As interdependências do almoxarifado, produção e depósito no sistema de produção.

2.4 - Componentes do Sistema de Produção

Os sistemas de produção são compostos de elementos (peças a fabricar, máquinas etc) e das relações entre eles (interconexões físicas, operações, etc).

A idéia de elemento ou de objeto físico pode ser associada a um conjunto de atributos mensuráveis pertinentes ao fim para o qual foi previsto o referido elemento. Estes atributos podem ser constantes ou variáveis.

Atributos constantes: são os que definem a natureza perene do elemento (tipo de elemento, parâmetros,...) e suas características (taxa de falhas, dimensões,...).

Atributos variáveis: são aqueles que evoluem com o tempo (estado de ocupação de uma máquina, estado operacional de uma peça no posto de trabalho em uma certa posição, período no stand by,...).

Os elementos da mesma natureza podem ser agrupados em lotes e o número de elementos destes lotes pode ser avaliado.

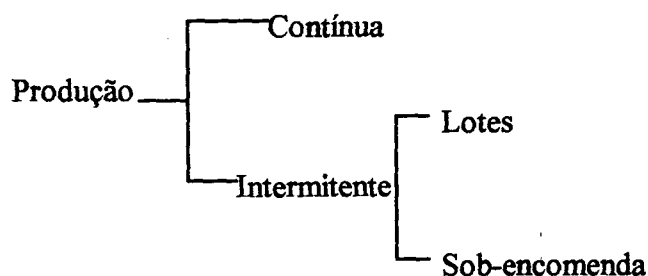
As relações são descritas com o auxílio dos atributos constantes que informam as características da relação (tempo de operação de cada peça na máquina,...).

Caso o sistema modelado permaneça o mesmo por um determinado intervalo de tempo, as relações possuirão atributos constantes, já que estas serão solicitadas com uma periodicidade constante neste intervalo de tempo.

Conseqüentemente o modelo de um dado sistema de produção poderá ser considerado como a representação do conjunto de elementos desse sistema e das relações qualificadas por seus atributos (Gomes 1990).

2.5 - Tipos de Sistemas de Produção

O desenvolvimento neste campo até agora resultou na divisão da produção em duas grandes classes: Produção contínua e Produção intermitente (que por sua vez se divide em produção em lotes e produção sob encomenda).



2.5.1 - Sistema de Produção Contínua

Este tipo de produção é caracterizado pelo grande volume de produção, produto padronizado e produção de grandes lotes de cada vez. O ritmo de produção é acelerado e as operações são executadas sem interrupção ou mudança. Como o produto é sempre o mesmo ao longo do tempo e o processo produtivo não sofre mudanças, o sistema pode ser aperfeiçoado continuamente. Assim o que se procura com este tipo de produção é precisamente alcançar o baixo custo, grande volume e produção em massa.

Na prática os modelos contínuos estão representados por linhas de montagem, fabricação de produtos químicos e refinação de petróleo, enfim, produtos que são mantidos em linha por um longo tempo e sem modificação.

2.5.2 - Sistema de Produção Intermitente

As situações intermitentes de produção são aquelas nas quais as instalações devem ser suficientemente flexíveis para manejar uma ampla variedade de produtos e tamanhos, ou onde a natureza básica da atividade impõe mudanças importantes dos insumos.

2.5.2.1 - Sistema de Produção em Lotes

É caracterizada por produzir uma quantidade limitada de um tipo de produto de cada vez (denominada lote de produção). Cada lote é previamente dimensionado para assim poder atender a um determinado volume de vendas previsto para um dado período de tempo. Desse modo, os lotes de produção são produzidos um a seguir do outro. Neste tipo de produção o plano de produção é feito antecipadamente, podendo assim a empresa melhor aproveitar seus recursos com maior grau de liberdade, ao contrário do que ocorre no sistema de produção sob encomenda, no qual o plano de produção é feito após o recebimento do pedido ou encomenda.

Este tipo de produção em lotes é utilizado por uma infinidade de indústrias a saber: têxteis, de cerâmica, de eletrodomésticos, de materiais elétricos, etc.

2.5.2.2 - Sistema de Produção sob Encomenda

Este tipo de fabricação contratada ou feita sob encomenda é produzida especialmente a pedido de um freguês como turbinas, ferramentas e matrizes, maquinaria especial, navios, etc. Os pedidos são em geral de natureza não repetitiva e as quantidades podem variar de uma a centenas de unidades. Neste tipo de produção, cada pedido usualmente acarreta uma grande variedade de operações, e o andamento em geral não segue nenhum plano padronizado ou rotineiro. É pois a encomenda ou o pedido efetuado que vai definir como a produção deverá ser planejada e controlada, sendo portanto esta etapa do planejamento e controle de produção muito complexa.

2.6.- Comparativo entre os Três Sistemas de Produção

Como pode-se observar no descrito acima, o tipo de produto que vai ser produzido é que determina o sistema de produção a ser adotado pela empresa, em muitos casos dada a diversidade de produtos que uma empresa fabrica ou produz, estas empresas apresentam misturas desses sistemas de produção. Se o produto é de grande porte e depende da encomenda do cliente, então o sistema adotado será a produção sob encomenda. Se por outro lado, há uma grande variedade de produtos que entram e saem da produção, e que a empresa vende após estocar, então o sistema adotado será a produção em lotes. Já se há um ou mais produtos que permanecem em produção por um longo tempo e que a empresa os vende após estocá-los, certamente o sistema adotado será de produção contínua.

Obviamente a adoção de um determinado sistema de produção apresenta certas vantagens como desvantagens.

Assim, a produção por encomenda é o sistema onde ocorre maior descontinuidade na produção, enquanto a produção contínua é o sistema onde há maior continuidade no processo produtivo, já a produção por lotes representa o sistema intermediário, onde a continuidade e a descontinuidade se alternam, como mostrado pela figura 2.4.



Fig. 2.4- Gradações que apresentam os três sistemas de produção

2.7 - Modelos de Sistemas de Produção

Antes de ver os diferentes modelos de sistemas de produção, é bom primeiro compreender o significado do que é um modelo, o mesmo que está ligado à cibernética.

Cibernética: é uma ciência da comunicação e do controle, seja no animal (homem, seres vivos) seja na máquina.

A cibernética compreende: os processos; os sistemas de transformação da informação e sua concretização em processos físicos, fisiológicos, psicológicos, etc.

Segundo Chiavenato (1983) Bertalanffy salienta que a cibernética é uma teoria dos sistemas de controle baseados na comunicação (transferência da informação) entre o sistema e o meio, e dentro do sistema e do controle (retroação) da função dos sistemas com respeito ao ambiente.

2.7.1 - Representação dos Sistemas: Os Modelos

Um dos grandes problemas da cibernética é justamente a representação de sistemas originais através de outros sistemas comparáveis, denominados modelos. No sentido literal da palavra, modelo é a representação de alguma coisa. Pode ser definido também como a representação simplificada de um sistema com o propósito de estudar o mesmo.

Para Riggs (1993) um modelo é uma réplica ou uma abstração da característica essencial de um processo.

Assim, problemas que desobedecem a soluções diretas por causa do tamanho, complexidade ou estrutura, são freqüentemente avaliados através de modelos de simulação.

Como podemos perceber, os modelos são muito importantes na representação de sistemas, podendo ser eles de diferentes tipos, como veremos posteriormente.

Modelo, então, vem a ser uma representação simplificada de alguma parte da realidade, sendo a natureza do problema aquela que vai determinar que tipo de modelo será o mais apropriado.

Segundo Chiavenato (1991) Starr aponta três razões para a utilização de modelos:

- a) a manipulação de entidades reais (pessoas ou organizações) é socialmente inaceitável ou legalmente proibido;
- b) o volume de incerteza com que a administração está lidando, cresce rapidamente e aumenta desproporcionalmente as conseqüências dos erros;
- c) a capacidade de construir modelos que constituem boas representações da realidade aumentam enormemente.

Sendo assim, uma das principais razões para o uso de modelos é a possibilidade de estudar um determinado sistema sem o custo, perigo, inconveniência; e até mesmo a possibilidade de observar o comportamento do sistema real visando observar propriedades ainda não identificadas para avaliar e posteriormente sugerir mudanças ou melhorias no sistema modelado. Assim a modelagem de um sistema envolve uma série de atividades as quais podem ser organizadas visando auxiliar este processo como pode ser visto na figura 2.5.

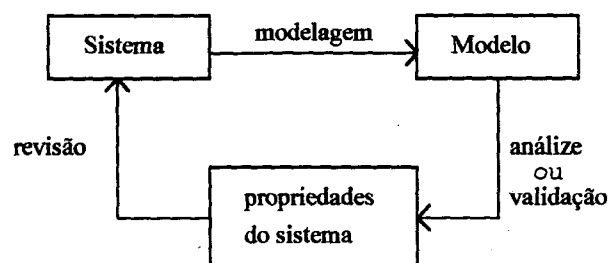


Fig. 2.5- Fases das atividades no processo de modelagem de sistemas

Neste processo, após a utilização de uma linguagem de modelagem para modelar as características mais relevantes do sistema, obtém-se o modelo o qual é válido/ou analisado para verificar se o mesmo possui todas as características. Caso o modelo não apresente problemas ou propriedades inaceitáveis, ele passa por uma revisão, onde são feitas alterações até torná-lo um sistema consistente (Ramamorthy 1980).

2.7.2.- Linguagens de Modelagem de Sistemas

Uma linguagem de modelagem é o meio pelo qual se expressam modelos, tendo como principal objetivo a descrição de sistemas (o mesmo que deve possuir uma série de características orientadas a esta atividade). Em guarda (1989) são apresentadas algumas destas características necessárias.

- a) Possuir uma base formal, visando obter uma interpretação exata e precisa;
- b) Clareza, visando facilitar a comunicação entre todos os envolvidos numa modelagem, e;
- c) Possibilitar a construção de modelos que preencham os requisitos de conceitualização (contendo apenas as propriedades desejadas do sistema modelado) e de totalidade(todas as propriedades desejadas do sistema modelado).

Porém é difícil que uma linguagem tenha todas as características, mesmo porque algumas delas são conflitantes.

É comum nos sistemas encontrar componentes que apresentem atividades concorrentes ou paralelas. Neste sentido, as Redes de Petri são uma linguagem de modelagem que foi desenvolvida especificamente para modelar sistemas discretos que possuem componentes que interagem concorrentemente (Peterson 1981; Agervala 1979).

2.7.3 - Tipos de Modelos

Como podemos observar, os modelos são muito importantes porque nos permitem representar sistemas, já que eles são réplicas ou abstrações das características essenciais de um processo, além de

mostrarem as relações entre causa e efeito, entre objetivos e restrições. Assim, os modelos podem ser classificados como sendo matemáticos ou físicos ou esquemáticos.

2.7.3.1 - Modelo Matemático

Usa notação simbólica e equações matemáticas para representar os sistemas. É fácil de manipular, o efeito das variáveis interatuantes se aprecia claramente e, sobre tudo, é um modelo preciso. Quando um modelo matemático pode construir-se para representar em forma exata a situação de um problema, suministra uma poderosa arma para seu estudo.

2.7.3.2 - Modelo Físico

São réplicas que sofrem uma mudança de escala. Os modelos microscópicos por exemplo, podem ser aumentados em escala para investigação, e estruturas enormes podem ser abaixadas em escala a um tamanho manejável. Nestas mudanças de escala, muitas vezes são perdidos certos detalhes. Numa réplica física, por exemplo, pode ser uma desvantagem onde um fator, tal como a distância, é uma consideração chave.

2.7.3.3 - Modelo Esquemático

Fazem parte deste tipo de modelo os gráficos de flutuação de preços, diagramas de atividades, redes de eventos, etc. Todos eles representam o mundo real. Por exemplo, quando queremos ver a evolução de um determinado preço, geralmente utilizamos um gráfico bidimensional para apreciar seu comportamento. Os aspectos gráficos são então úteis para propósitos de demonstração.

2.8 - Sistemas Discretos e Contínuos

Depois de termos definido o conceito de sistema em forma geral, podemos agora dividi-los segundo sua natureza, os mesmos que podem ser classificados como discretos e contínuos. Na prática poucos sistemas são totalmente discretos ou contínuos, porém, depois de fazer algum tipo de mudanças

para a maioria dos sistemas, será possível então classificá-los como sendo discretos ou contínuos.

2.8.1 - Sistemas Contínuos

São aqueles sistemas nos quais as variáveis de estado mudam continuamente no tempo. Na realidade, todos os sistemas são contínuos. São os modelos (representação aproximada dos sistemas) que são discretos.

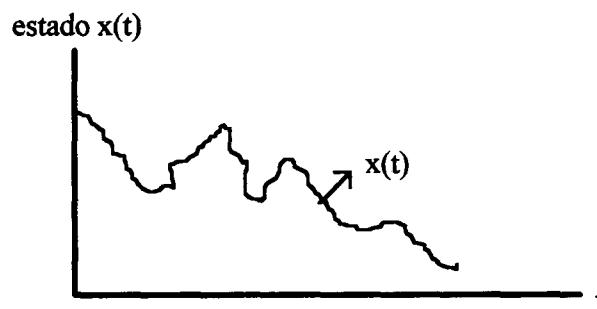


Fig. 2.6- Trajetória de um sistema dinâmico de variável contínua (SDVC)

2.8.2.- Comportamento Dinâmico dos Sistemas de Variáveis Contínuas (SVC)

As teorias de controle de sistemas evoluindo no tempo, satisfazendo basicamente leis físicas, é fundamentado basicamente em modelos matemáticos descritas por equações diferenciais (Ho 1989; Cassandras 1990).

Como visto na figura 2.6 a noção do tempo é uma variável independente, a qual é substituído por uma seqüência de eventos num SDED. A trajetória de um sistema dinâmico de variável contínua (SDVC) está constantemente mudando com o estado, tomando valores em \mathbb{R}^n , sendo capaz de representar o comportamento do sistema em um instante qualquer a partir de um instante inicial.

2.8.3 - Sistemas Discretos

São sistemas nos quais as variáveis de estado mudam só num conjunto discreto de pontos no

tempo. Por exemplo: O banco é um exemplo de um sistema discreto desde que a variável de estado, o número de clientes no banco, muda só quando um cliente chega ou quando o serviço prestado a um cliente é completado.

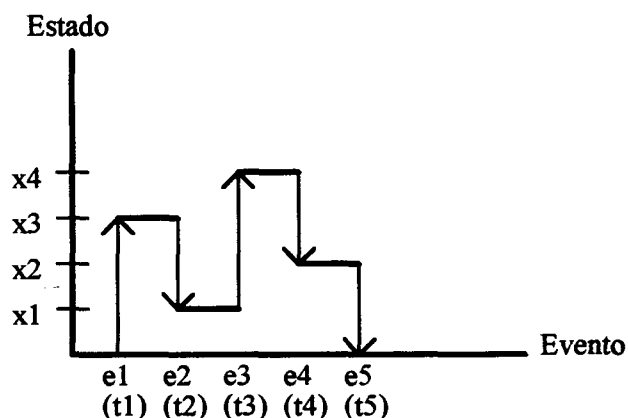


Fig. 2.7 - Trajetória de um sistema dinâmico de eventos discretos (SDED)

Como vimos anteriormente, a maioria dos sistemas são contínuos, mas para casos de estudo, eles podem se tornar discretos. Assim, temos os seguintes tipos de sistemas discretos, os quais são representados na figura 2.7 (Cardoso 1993).

2.8.3.1 - Sistemas Discretizados

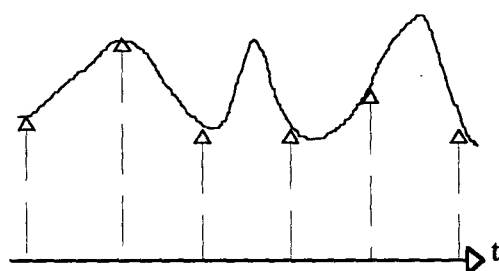
São sistemas estudados somente em instantes precisos. Trata-se, portanto, de sistemas contínuos observados em instantes discretos (sistemas amostrados), as variáveis de estado evoluem de maneira contínua, sem mudança brutal de comportamento, mas é somente a instantes discretos que há um interesse no seu valor.

2.8.3.2 - Sistemas Discretos

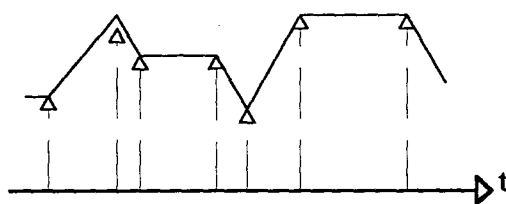
São sistemas para os quais as variáveis de estado, ou ao menos algumas delas, variam brutalmente a certos instantes. Entretanto, estes instantes podem necessariamente ser previstos, e o conhecimento do estado a um instante dado não permite que, sem cálculo se conheça o estado seguinte.

2.8.3.3 - Sistemas a eventos discretos

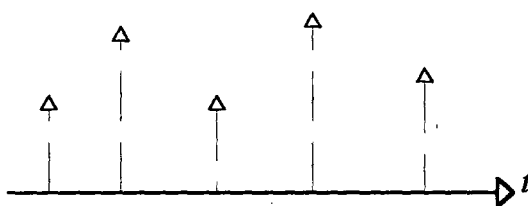
São sistemas modelados de tal maneira que as variáveis de estado variam brutalmente em instantes determinados e que os valores das variáveis nos estados seguintes podem ser calculados diretamente a partir dos valores precedentes e sem ter que considerar o tempo entre estes dois instantes.



a) Sistemas Discretizados



b) sistemas discretos



c) sistemas a eventos discretos

Fig. 2.8 - Diversos tipos de sistemas discretos (Cardoso 1994)

2.8.4.- Comportamento Dinâmico dos Sistemas de Eventos Discretos (SED)

Comparado com os (SVC) estes sistemas (SED) apresentam um comportamento diferenciado, isto é: A evolução dinâmica do sistema como visto na figura. 2.8 está subordinada à ocorrência de

eventos, os quais não podem ser representados por equações diferenciais ordinárias. Exemplos de tais sistemas (SED) são linhas de montagem, redes de comunicação, sistemas de tráfego, sistemas computacionais, e sistemas de manufatura em geral, que são sistemas dinâmicos criados pelo próprio homem (Ho 1989), onde a evolução do sistema no tempo depende das interações complexas da cronometragem de vários eventos discretos, tais como a chegada ou saída de uma trabalho ou o início ou fim de uma tarefa ou mensagem. O estado de tais sistemas dinâmicos muda só nesses instantes discretos de tempo em lugar de continuamente. Ao contrário dos SDVC do mundo físico que são descritos por equações diferenciais, estes sistemas estabelecidos pelo próprio homem são chamados Sistemas Dinâmicos de Eventos Discretos (SDED) (Ho 1989). Eles tendem a agir mais com o homem do que com a natureza.

Um fato importante a ser destacado é que no comportamento dinâmico desses sistemas (SDED) a evolução do estado global do sistema correspondente ao vetor de estados $X(t)$ nos (SDVC) ocorre de forma descontínua, representando assim a natureza discreta dos estados.

Segundo Cassandras (1990) há duas maneiras de representar sequências na modelagem dos (SDED): O primeiro deles é limitado para modelos não temporizados, baseados em uma simples ordenação dos eventos $\{e_1, e_2, \dots, e_n\}$, e o segundo é para representar modelos temporizados nos quais se associa a cada evento o instante de sua ocorrência $\{(e_1, t_1), (e_2, t_2), \dots\}$. Por exemplo quando queremos medir o desempenho, o tempo é um fator importante para descrever o modelo do comportamento do sistema, enquanto que se queremos realizar apenas o controle de um SDED, necessitaremos ferramentas de modelagem capazes de representar as condições para a evolução do sistema que passa a depender da ocorrência dos eventos, sem precisar do fator tempo.

Os modelos de SDED mais conhecidos são: Cadeias de Markov, Redes de filas, Autômatos e Máquinas de estados finitos, Redes de Petri, e modelos algébricos. Em Cao (1990) e Miyagi (1989) é feita uma comparação dos diferentes modelos, mostrando sua aplicabilidade em sistemas de manufatura para os mesmos. A tabela 2.2 mostra uma classificação das técnicas descritas anteriormente utilizadas para a modelagem de SDED (Ho 1989).

	Temporizado	Não temporizado
Lógico	Lógica temporal	Máquinas de estados finitas Redes de Petri
Algébrico	Álgebra Min-Max	Processos sequenciais Processos recursivos
Análise de Performance	<div style="background-color: #cccccc; border: 1px solid black; padding: 2px;"> Cadeias de Markov Redes de filas Redes de Petri estocásticas </div>	
<div style="display: inline-block; width: 15px; height: 10px; background-color: #cccccc; border: 1px solid black; margin-right: 5px;"></div> Estocástico		<div style="display: inline-block; width: 15px; height: 10px; background-color: white; border: 1px solid black; margin-right: 5px;"></div> Determinístico

Tab. 2.2- Classificação dos modelos de DEDS

No presente trabalho trataremos do último tipo de sistema discreto (2.8.3.3), devido a que as Redes de Petri são uma ferramenta de representação formal de eventos discretos. Tal representação (evolução do estado do sistema) é reproduzido passo a passo e evento por evento, precisando portanto ter um conhecimento dos seguintes conceitos.

Eventos: geralmente representam o início ou fim das atividades em determinados instantes de um sistema, no qual não são consumidos tempo nem recursos (são por tanto instantes de observação e de mudança de estado do sistema que inicia uma atividade que não estava em curso antes). Em geral sua ocorrência depende da evolução do sistema e portanto de outras atividades anteriores.

Atividades (ou tarefa): é a execução efetiva da operação, isto é, a sucessão de um número finito de estados de um elemento consumindo tempo e/ou recursos. São, pois, as chamadas caixas pretas utilizadas para esconder a evolução do sistema físico entre dois eventos.

Processo: É uma seqüência finita de eventos e atividades interdependentes que permitem facilitar a descrição de um sistema.

Assim, para modelar um sistema de produção existem diferentes técnicas clássicas, que podem ser vistas melhor em Gomes (1990). As técnicas são:

- a) representação dos sistemas de produção através de modelos de eventos discretos;
- b) modelos analíticos para sistemas de produção;

- b.1) modelos estocásticos de sistemas de produção;
- b.2) modelagem de processos de produção pela álgebra de díoides;
- c) modelos de eventos discretos para simulação.

2.9.- Sistemas de Manufatura

Um sistema de manufatura (sistema produtivo) como visto na figura 2.1, pode ser considerado como um sistema composto de vários subsistemas (processos produtivos) como transportadores, robôs, células de produção, máquinas operatrizes, etc, no qual cada subsistema realiza uma determinada operação produtiva, a qual pode ser uma transformação física (ex. usinagem, corte, estampagem,....) ou um deslocamento físico (carregamento, ou descarregamento de peças). Uma característica básica destes subsistemas é que por eles fluem diversos itens como peças, materiais, recursos, etc, que passam sequencialmente pelos subsistemas, onde transformações apropriadas são realizadas até obter o produto final requerido (de acordo com especificações técnicas, desde geometria até controle de qualidade).

Assim o processo produtivo como um todo (Sistema) é caracterizado pelo inter-relacionamento complexo de suas atividades e condições. Baseado na natureza discreta do inter-relacionamento entre atividades e condições, estes sistemas de manufatura podem ser definidos como sistemas de eventos discretos (Ho 1989) ou sistemas de produção discreta (Silva 1989).

Dada a natureza discreta das atividades envolvidas nos FMS, é portanto conveniente utilizar modelos discretos para o estudo destes sistemas

Uma classificação comumente conhecida da organização de um sistema de manufatura é baseado em sistemas do tipo flow-shop, job-shop e as células de manufatura.

Um sistema produtivo do tipo flow-shop (figura 2.9) denominada linha de manufatura flexível, é projetado para tratar um volume de produção muito grande de uma única família de produtos, os quais diferem levemente um do outro. O grau de flexibilidade neste tipo de sistema é dado a partir da configuração das máquinas ferramentas que compõem a linha. Por outro lado, o tipo de sistema job-

shop apresenta uma grande flexibilidade mas o volume de produção é pequeno e a variedade de produtos muito grande. Em virtude da grande complexidade de gerenciamento do fluxo de materiais e informações (já que para produto uma nova ordem de produção tem que ser feita) que este sistema apresenta, seu índice de produtividade é muito baixo se comparado com uma linha de produção do tipo flow-shop.

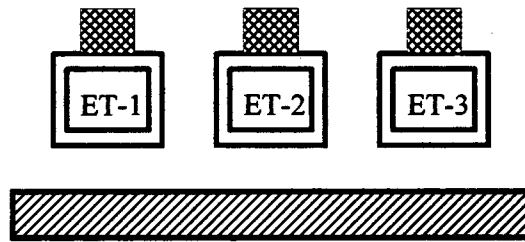


Fig. 2.9- Linha de manufatura flexível

As células de manufatura encontram-se numa situação intermediária entre os dois sistemas citados anteriormente, já que as mesmas permitem o agrupamento das peças em famílias de peças, segundo suas características de projeto. As células representam uma solução, no sentido de obter-se um sistema com bom índice de produtividade, associado com um maior grau de flexibilidade, se comparado com sistemas do tipo flow-shop (Dos Santos F^o 1993). As células de manufatura podem ser definidas como sendo uma configuração de estações de trabalho (figura 2.10) e de um sistema de manuseio e transporte de materiais (figura 2.11), sendo a mesma controlada por um computador para manufaturar eficientemente mais de um tipo de produto em pequenas e médias quantidades. As estações de trabalho são por ex. máquinas de comando numérico, robôs industriais, dispositivos especiais para operações de montagem, etc, ou combinações destes equipamentos.

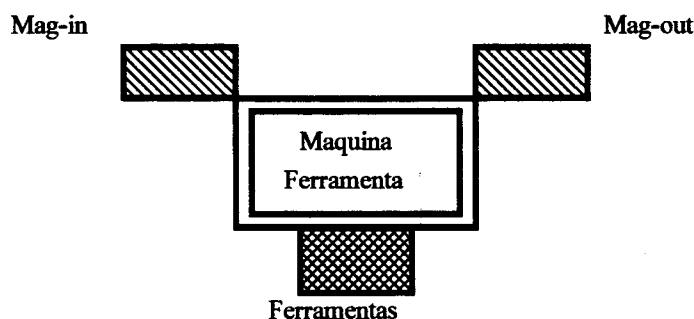


Fig.2.10- Representação de uma estação de trabalho

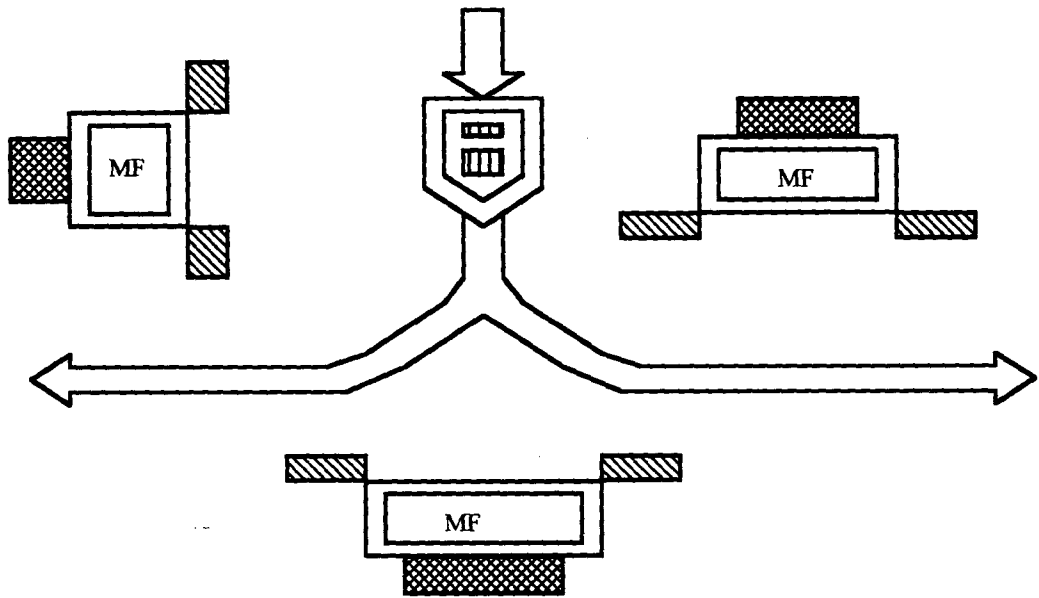


Fig.2.11- Representação de uma célula de manufatura

Cada um dos arranjos mostrados nas figuras anteriores são modelos básicos de organização para a realização de processos de manufatura. A união desses modelos (figura 2.12) vai representar a planta do sistema produtivo capaz de realizar uma variedade de processos de forma independente, a partir de um controle adequado de fluxo de materiais ligados ao sistema.

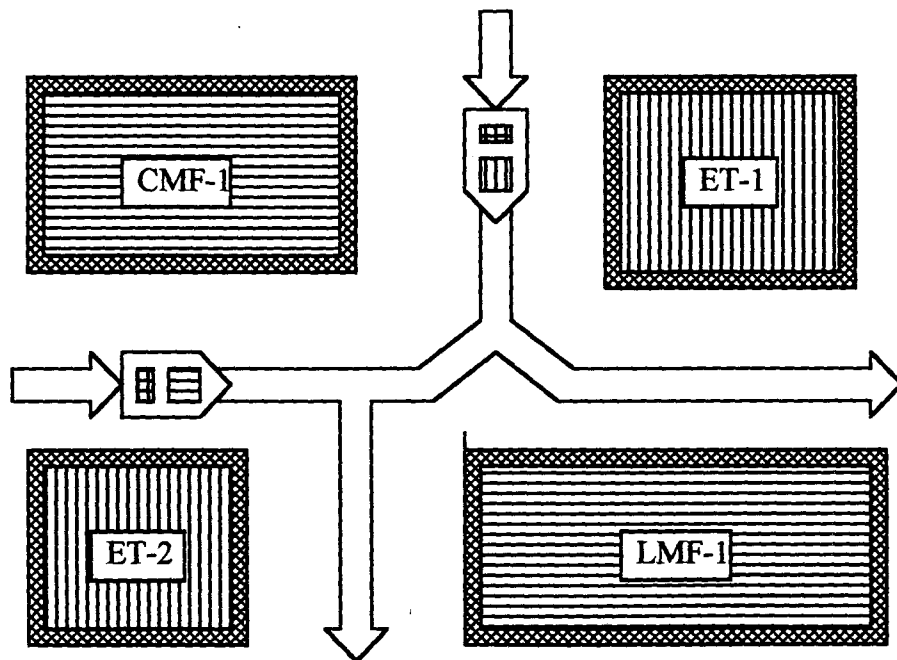


Fig.2.12- Representação de um sistema integrado de manufatura (Dos Santos 1993)

2.10.- Flexibilidade e o Processo de Manufatura

A flexibilidade é uma característica essencial nos sistemas produtivos atuais, dando maior competitividade às empresas, obtendo assim uma rápida resposta as exigências do mercado.

Na literatura geralmente são distinguidos dois tipos de flexibilidade (Barad 1988; Silva 1989):

- a) Flexibilidade de longo prazo ou flexibilidade potencial: é aquela capaz de enfrentar um universo indefinido de mudanças, ou seja, a possibilidade de introduzir novos produtos no sistema de manufatura durante sua operação e com pouco esforço;
- b) Flexibilidade de curto prazo: que corresponde à possibilidade de manipular concorrentemente uma grande variedade de famílias de produtos num tempo dado no sistema de manufatura.

Segundo Dos Santos F^o (1993) Groover ressalta que existe uma relação entre o conceito de flexibilidade e a forma de realizar a automação de um sistema. Partindo dessa premissa, um sistema de produção automatizado é classificado em três tipos básicos: Automação Fixa, Automação Programável e Automação Flexível (Lorini 1993). Num sistema de automação fixa a seqüência de operações é limitada pela configuração dos equipamentos, envolvendo altas taxas de produção e baixa flexibilidade quanto à adaptação a alterações do produto. Nos sistemas de Automação Programável, os equipamentos que os constituem são programáveis, permitindo a reprogramação de novos processos.

No caso de um sistema de automação flexível, existe a capacidade de produzir-se uma variedade de produtos sem que haja perda de tempo em adaptar-se o sistema às alterações de um produto em relação ao próximo produto a ser processado, diferenciando-o da automação programável por não envolver perdas de tempo com reprogramação e setup das máquinas.

A partir da fusão dos termos manufatura, automação e flexibilidade, surge o termo de manufatura automatizada, que engloba uma série de sistemas produtivos encontrando-se entre eles os Sistemas Flexíveis de Manufatura (FMS). A rápida evolução do mercado nestes últimos anos requer das empresas maior competitividade para assim satisfazer as exigências cada vez mais sofisticadas dos consumidores, associada à diminuição do ciclo de vida do produto e desenvolvimento de novos

produtos. É necessário então elaborar-se sistemas de manufatura altamente automatizados como as linhas de produção do tipo flow-shop, que tenham ao mesmo tempo a capacidade das fábricas do tipo job-shop. Os FMS correspondem a uma tentativa para satisfazer essas exigências contraditórias (Valette 1990).

2.11.- Sistemas Flexíveis de Manufatura (FMS)

FMS são, portanto, uma classe de sistemas de manufatura automatizada cuja característica principal, que as diferencia dos outros sistemas é sua flexibilidade para processar eficientemente mais de um tipo de peça em pequenas e médias quantidades. Aqui, o conceito de flexibilidade vai se referir não apenas à possibilidade de processar mais de um tipo de produto ou peça, senão também à capacidade de mudar rapidamente o tipo de peça a ser produzido, além da possibilidade de manufaturar simultaneamente diferentes tipos de peças. Assim os FMS surgiram como uma tentativa para reconciliar a eficiência da linha de produção do tipo flow-shop, com a flexibilidade de um sistema do tipo job-shop com a finalidade de satisfazer uma demanda versátil a um baixo custo. A alta produtividade neste tipo de sistema é alcançada portanto pela incorporação eficiente de princípios de tecnologia de grupo, controle total de qualidade e estratégias de produção, tais como MRP II, JIT (Viswanadham 1987).

Desse modo para que um FMS possa atender aos requisitos anteriormente citados, o mesmo é composto de:

- a) Um conjunto de máquinas ferramentas flexíveis de comando numérico;
- b) Um sistema de transporte e manuseio de peças e ferramentas automáticos;
- c) Um sofisticado sistema de tomada de decisão para decidir a cada instante quais são as operações que devem ser executadas nas máquinas.

Uma máquina flexível tem a capacidade de executar vários tipos de operações. O sistema de transporte automático por sua vez é requerido para transportar as partes para a máquina onde a próxima operação será executada (usam-se em geral os Veículos Guiados Automaticamente (AGV)),

cujos circuitos são realizados por fios elétricos implantados no chão da fábrica e percorridos por uma corrente de uma determinada frequência através de contatos. Desse modo, quando um veículo atravessa um contato ele bem pode receber uma ordem de parar ou continuar, ou de seguir numa outra frequência. Dessa forma os veículos podem movimentar-se ao longo de toda a fábrica.

É por essas características e por outras que os FMS têm sido considerados como um tipo de sistemas produtivos singulares, capazes de corresponder às expectativas do mercado em relação as necessidades atuais e futuras, o que implica em constantes alterações no sistema de controle. Tais sistemas são complexos e cada vez mais tendem a aumentar o seu nível de complexidade na medida em que a ele são incorporados novas funções. Assim, qualquer método que se proponha a descrever este tipo de sistema, deve permitir a representação do sistema em diferentes níveis de abstração e facilitar a sua decomposição em subsistemas funcionais, com uma clara definição das inter-relações entre estes subsistemas (Rillo 1988).

Atualmente percebe-se a evolução destes sistemas em dois sentidos (dos Santos F^o 1993).

- a) no sentido de estabelecer o amadurecimento dos sistemas presentes;
- b) no sentido de procurar materializar prováveis concepções de sistemas para o futuro.

Em ambos os sentidos há uma necessidade premente por metodologias consistentes para auxiliar durante todo o ciclo de vida de um sistema de manufatura (Leung 1990).

Os sistemas de manufatura têm sido freqüentemente modelados como sendo sistemas discretos, os quais devido a sua complexidade não podem ser modelados por um modelo que utilize o conceito de estado total (filas de espera por ex.), pois a partir de um estado pode acontecer muitos eventos provocando diversas alterações de estado. Portanto, faz-se necessário para modelar FMS procurar métodos que possuam de modo inerente o conceito de estado parcial. Outro aspecto importante é o fato de que durante a vida útil dos FMSs estão envolvidos especialistas de diferentes formações. Portanto, para que a contribuição destas pessoas seja aproveitada eficientemente a metodologia e a linguagem de

programação devem ser de fácil aprendizado e compreensão, além de possuírem boa legibilidade através de representações gráficas simples (Rillo 1988).

Resumindo todos os conceitos anteriores, pode-se considerar um FMS como um sistema dinâmico de eventos discretos (DEDS), do qual, as partes a serem processadas de vários tipos ou classes de trabalho entram no sistema de uma maneira assíncrona, e são processados concorrentemente, compartilhando os recursos limitados (estações de trabalho, robôs, sistemas de manuseio de materiais buffers, etc).

2.12- Hierarquia de Decisão e de Controle

Como visto anteriormente, um sistema integrado de manufatura (SIM) é formado por vários elementos, "Células de Manufatura Flexíveis" (CMF), Estações de trabalho (ET) e linhas de manufatura flexíveis (LMF). Apesar de cada elemento ser bem definido com métodos de modelagem e de análise própria, não é possível evitar completamente uma abordagem conjunta devido a sua complexidade de integração. Portanto para realizar de maneira eficiente a descrição funcional de um SIM, é adequado uma abordagem hierárquica capaz de organizar a diversidade associada à capacidade operacional de seus elementos em conjuntos homogêneos subordinados (Santos F^o 1993). Assim um sistema de manufatura pode ser dividido numa hierarquia de cinco níveis (Gershwin 1989; Valette 1990).

- a) Planejamento de longo prazo;
- b) Planejamento e escalonamento de curto prazo;
- c) Supervisão Global;
- d) Supervisão de subsistemas ou controle de estações de trabalho;
- e) Controle local.

Onde cada um dos níveis recebe comandos do nível imediatamente superior e ao mesmo tempo recebe informações sobre o estado do nível imediatamente inferior.

Planejamento de longo prazo: Opera a partir da demanda estimada (ou encomenda) calculando o tempo necessário para a produção assim como a quantidade de matéria prima necessária. Este nível também inclui o planejamento de alocação de recursos para a produção além das datas de lançamento e prazos de entrega.

Escalonamento: Este nível é caracterizado por uma explosão combinatória do número de alternativas de escolha de uma máquina para executar uma determinada operação porque cada máquina pode realizar diversos tipos de operações. Assim este nível encarrega-se de ordenar a ordem de execução das operações em cada máquina e de calcular um intervalo de tempo necessário para sua execução.

Supervisão global: Implementa várias funções.

- Atualiza após cada evento a representação do estado da fabrica
- Toma decisões em tempo real com respeito ao lançamento das operações, e verifica que eles sejam compatíveis com o calendário de manufatura.
- Monitora e coordena o comportamento global da fábrica para detectar falhas dos elementos físicos e as violações do calendário de manufatura.

Supervisão de subsistemas: Realiza a coordenação de subsistemas tais como: sistemas de transporte, estações de trabalho ou células de manufatura. Ela opera no nível de coordenação global supervisionando o comportamento dos subsistemas, e monitorando a execução das operações planejadas mas numa maneira mais detalhada. Por ex., enquanto a supervisão global considera a operação carregar uma peça na máquina, a supervisão do centro de usinagem correspondente trata-a como um programa coordenando um robô, uma esteira, e a máquina ferramenta correspondente.

Controle local: Executa o controle em tempo real das máquinas, sendo o único nível que está em contato direto com os elementos físicos da fábrica. A partir das informações fornecidas pelos sensores controla-se em tempo real os sistemas através dos atuadores.

As metodologias mais comuns utilizadas atualmente para o modelamento dos níveis citados acima são os fluxogramas, diagramas de estado, diagramas de tempo. Por outro lado as linguagens de programação mais comuns são as linguagens procedurais clássicas, e aquelas utilizadas em controladores lógicos programáveis. É importante também que a metodologia de modelamento e a linguagem de programação permitam a representação de diversos aspectos estáticos e dinâmicos do sistema, através de um modo claro e uniforme (Rillo 1988).

Devido ao elevado custo da implementação física dos FMSs, é requerido a utilização de sistemas de simulação ou métodos de análise que serão usados em todas as fases do projeto para sua validação, e quando haja necessidade de mudanças que não podem ser testados diretamente no sistema real. Neste sentido é importante que as ferramentas de simulação não sejam muito distantes das metodologias e linguagens utilizadas para a especificação e implementação do sistema de controle (Rillo 1988).

Um aspecto importante dos sistemas de controle em tempo real é que o tempo de execução de uma tarefa que o sistema de controle deve realizar seja irrelevante em relação à necessidade de execução da tarefa. Além de evitar que estes realizem tarefas desnecessárias, já que desse modo só se consegue aumentar o tempo de execução da tarefa realmente necessária.

Assim, para lograr uma eficiente especificação e implementação de sistemas de controle os métodos a serem utilizados devem ter as seguintes características:

- a) Permitir a representação do sistema em diferentes níveis de abstração;
- b) Facilitar a decomposição do sistema em subsistemas funcionais com uma clara definição das inter-relações;
- c) Conter explicitamente o conceito de estado parcial;
- d) Devem ser acessíveis a especialistas de diferentes formações;
- e) Facilitar as constantes alterações às quais o sistema de controle estará sujeito;

- f) Facilitar a representação de concorrência entre tarefas as quais são executadas simultaneamente assim como a sincronização;
- h) Permitir a representação da variável tempo;
- i) Devem ser simples para permitir que as empresas tenham completo conhecimento sobre o sistema, facilitando assim a integração de novos elementos;
- j) Permitir a descrição dos aspectos estáticos e dinâmicos do sistema a ser representado.

Após determinar as características necessárias, um modelo apropriado é então necessário para propósitos de análise e controle. Mas, como escolher um modelo (o melhor) para analisar as atividades complexas de um FMS?. Um modelo de Rede de Petri seria uma boa escolha.

CAPÍTULO 3 - REDES DE PETRI

3.1.- Introdução

A Rede de Petri introduzida por Cari Adam Petri em sua tese intitulada "Comunicação com autômatos" é uma ferramenta gráfica e algébrica que apresenta um bom nível de abstração em comparação com outros modelos gráficos (Peterson 1981). Sendo um modelo do tipo estado-evento, onde cada evento possui pré-condições que vão permitir sua ocorrência e pós-condições decorrentes desta, as quais são por sua vez pré-condições de outros eventos posteriores.

Uma Rede de Petri é vista também como um tipo particular de grafo orientado que permite modelar as propriedades estáticas de um sistema a eventos discretos, constituído de dois tipos de nós: as transições (que correspondem aos eventos que caracterizam as mudanças de estado do sistema), e os lugares (que correspondem às condições que devem ser certificadas para os eventos acontecerem) interligados por arcos direcionados ponderados (Peterson 1981).

A Rede de Petri é, portanto, um formalismo que permite a modelagem de sistemas dinâmicos discretos com grande poder de expressividade, permitindo representar com facilidade todas as relações de causalidade entre processos em situações de: sequencialidade (figura 3.1.a), conflito (figura 3.1.b), concorrência (figura 3.1.c) e a sincronização (figura 3.1.d).

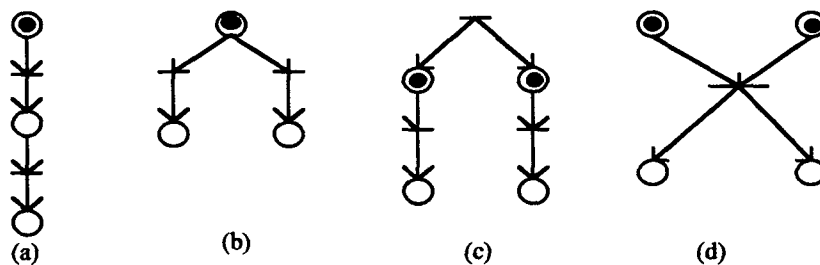


Fig. 3.1.- Relações de causalidade modeladas por Redes de Petri

Os eventos do sistema são então representados por transições (barras) correspondendo assim, a cada evento, uma transição cujo disparo indica sua ocorrência. Relacionam-se os eventos às condições através de arcos direcionados que interligam as transições aos lugares.

Rede de Petri é, portanto, um modelo formal de especificação e controle do fluxo de informações de um sistema discreto qualquer.

3.2.- Conceituação Formal e Gráfica

3.2.1.- Rede de Petri

Uma Rede de Petri (simples ou Autônoma) é composta de quatro partes : Um conjunto de lugares P, um conjunto de transições T, uma aplicação de entrada I ou Pre, e uma aplicação de saída O ou Pos. As funções de entrada e saída relacionam transições e posições. Sendo assim a estrutura das Redes de Petri é definida por suas posições, transições, a função de entrada I (ou Pré), e a função de saída O (ou Pós).

Definição Formal:

Uma Rede de Petri (RdP) é uma quádrupla

$$R = (P, T, \text{Pré}, \text{Pós}) \quad (3.1)$$

onde:

$P = \{p_1, p_2, \dots, p_n\}$ é um conjunto de lugares, $n > 0$.

$T = \{t_1, t_2, \dots, t_m\}$ é um conjunto de transições, $m > 0$.

$\text{Pré} : P \times T \rightarrow \mathbb{N}$ é a aplicação de arcos de entrada das transições (lugares precedentes).

$\text{Pós} : T \times P \rightarrow \mathbb{N}$ é a aplicação de arcos de saída das transições (lugares posteriores).

$$P \cap T = \emptyset.$$

N : é o conjunto dos números naturais.

3.2.2.- Rede de Petri Marcada

Uma Rede de Petri marcada é uma dupla

$$N = (R, M_0) \quad (3.2)$$

onde :

-R é uma Rede de Petri

- M_0 é a marcação inicial dada pela aplicação

$$M: P \rightarrow N \quad (3.3)$$

Sendo $M(p) \geq 0$ equivalente ao número de fichas contidas em cada lugar.

A marcação de todos os lugares é representada por um vetor n-dimencional :

$$M = [m_1, m_2, \dots, m_j]^T \quad (3.4)$$

onde:

m_j = é o número de fichas do lugar p_j

A marcação num determinado instante representa o estado da Rede de Petri, ou mais precisamente o estado do sistema descrito pela RdP, assim a evolução do estado da RdP corresponde a uma evolução da marcação, a qual é causada pelo disparo de transições como veremos posteriormente.

Uma marcação M é, portanto, uma aplicação que associa a cada lugar na RdP um inteiro não negativo chamado de ficha. Pode-se dizer também que M é a distribuição das fichas nos lugares "p" ou o número de fichas nos lugares, conforme pode ser visto na Figura. 3.2, onde a marcação M_0 está dada por $M = [0, 3, 0]^T$.

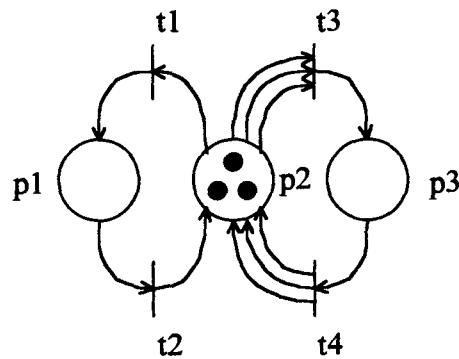


Fig. 3.2.- Rede de Petri marcada

3.3.- Representação Gráfica

Uma Rede de Petri pode ser vista como um multigrafo bipartito e orientado definido por $R = (P, T, \text{Pré}, \text{Pós}, M_0)$, onde P é um conjunto de lugares representado por círculos e T é um conjunto de transições representado por barras, Pré e Pós são as relações de precedência e poscedência aplicados sobre $P \times T$ e $T \times P$ respectivamente, e M representa a marcação dos lugares através de fichas (pontos pretos) no interior de cada lugar.

$\text{Pré}(p_i, t_j) > 0$ Se existe um arco do lugar p_i para a transição t_j ;

$\text{Pré}(p_i, t_j) = 0$ Caso contrário;

$\text{Pós}(t_m, p_a) > 0$ Se existe um arco da transição t_m para o lugar p_a ;

$\text{Pós}(t_m, p_a) = 0$ Caso contrário.

3.4.- Representação Matricial

Aqui vamos representar algebricamente a estrutura gráfica de uma Rede de Petri.

A informação contida na representação gráfica pode também ser organizada numa *matriz de incidência* $C = (c_{ij})_{n \times m}$ definida a partir de Pré (p, t) (ou C^-) e Pós (t, p) (ou C^+).

$$C = \text{Pós} - \text{Pré} \quad (3.5)$$

ou :

$$(p_i, t_j) \rightarrow c_{ij} \text{ Pós}(t_j, p_i) \text{ Pré}(p_i, t_j) \quad (3.6)$$

ou

$$C = C^+ - C^- \quad (3.6.a)$$

Considerando a RdP da figura 3.2 ela pode ser definida algebricamente da seguinte maneira:

$$P = \{p_1, p_2, p_3\}$$

$$T = \{t_1, t_2, t_3, t_4\}$$

$$\text{Pre}(p_i, t_j) = \begin{array}{cccc} & t_1 & t_2 & t_3 & t_4 \\ \begin{array}{l} p_1 \\ p_2 \\ p_3 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

$$\text{Post}(p_i, t_j) = \begin{array}{cccc} & t_1 & t_2 & t_3 & t_4 \\ \begin{array}{l} p_1 \\ p_2 \\ p_3 \end{array} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

tem-se então:

$$C(p_i, t_j) = \begin{array}{cccc} & t_1 & t_2 & t_3 & t_4 \\ \begin{array}{l} p_1 \\ p_2 \\ p_3 \end{array} & \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & -3 & 3 \\ 0 & 0 & 1 & -1 \end{bmatrix} \end{array}$$

e a marcação inicial é:

$$M_0 = \begin{array}{l} \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} p_1 \\ p_2 \\ p_3 \end{array}$$

A matriz de incidência associada a uma RdP corresponde então a sua estrutura, independente da marcação.

Na matriz de incidência, cada coluna vai corresponder à modificação de uma marcação quando a transição associada é disparada. Por exemplo a primeira coluna da matriz anterior vai significar que quando t_1 é disparada, uma ficha é movida desde p_2 , enquanto uma ficha é depositada em p_1 e assim sucessivamente .

3.5.- Funcionamento de uma Rede de Petri

3.5.1.- Sensibilização de uma Transição

Uma transição t é sensibilizada por uma marcação M , se e somente se:

$$\forall p_i \in P, M(p_i) \geq \text{Pré}(p_i, t_j) \quad (3.7)$$

Por exemplo, para a Rede de Petri da figura 3.2 para uma marcação inicial $M_0 = [0, 3, 0]^T$ as transições t_1 e t_3 estão sensibilizadas, pois :

$$\text{Pré}(\dots, t_1) = [0, 1, 0]^T \text{ e } \text{Pré}(\dots, t_3) = [0, 3, 0]^T$$

e portanto:

$$M > \text{Pré}(\dots, t_1) \text{ e } M = \text{Pré}(\dots, t_3)$$

3.5.2.- Disparo de uma Transição

Numa Rede de Petri, $t_j \in T$ só pode disparar se estiver sensibilizada. Disparando uma transição sensibilizada t_j , uma nova marcação M' é obtida, tal que:

$$M'(p) = M_0(p) - \text{Pré}(p, t_j) + \text{Pós}(t_j, p), \forall p \in P \quad (3.8)$$

a figura dada a seguir mostra diversas situações possíveis de disparo.

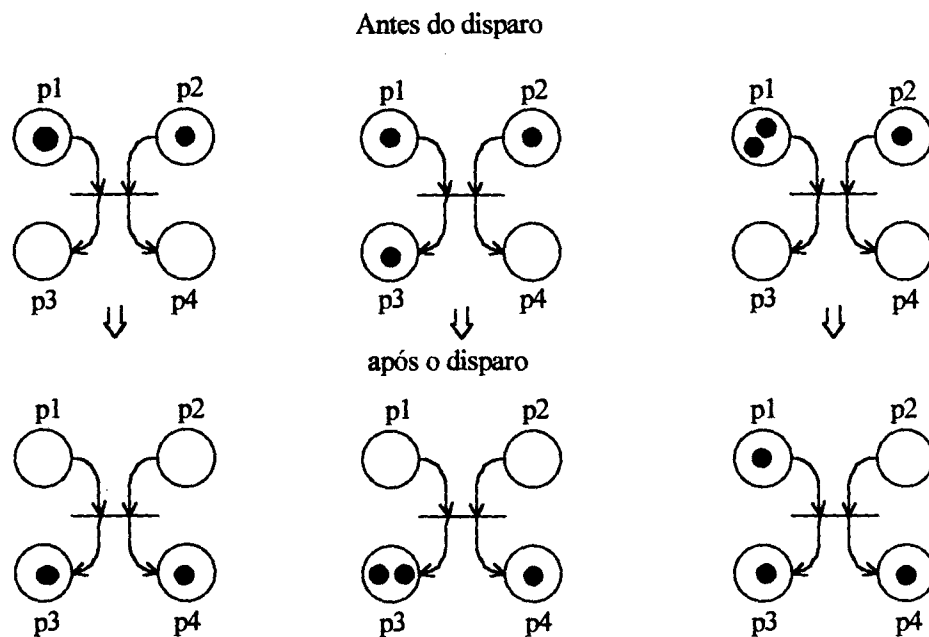


Fig. 3.3.- disparo de uma transição

Ex. Seja a Rede de Petri da figura 3.2, após o disparo da transição t_1 sensibilizada, a partir da marcação inicial M_0 obtém-se a marcação seguinte:

$$M'(p) = [0, 3, 0]^T - [0, 1, 0]^T + [1, 0, 0]^T = [1, 2, 0]^T$$

3.5.3.- Seqüência de Disparo

Uma seqüência de transições "s" que devem disparar para atingir uma marca M' partindo de M é denominada de seqüência de disparo.

Se $M_0 \xrightarrow{t_1} M_1$ e $M_1 \xrightarrow{t_2} M_2$, diz-se que a seqüência $s = t_1 t_2$ é disparável a partir de

M_0 com a seguinte notação:

$$M_0 \xrightarrow{t_1 t_2} M_2 \quad (3.9)$$

Com uma seqüência "s" é associado um vetor característico $S = t_j k$, onde o j indica quais as transições disparadas e o k o número de vezes que ela disparou. Sua dimensão é igual ao número de transições da rede.

Ex. na RdP da figura 3.2 para alcançar a marcação $M = (0 \ 0 \ 1) = [0, 0, 1]^T$ partindo de $M = (0 \ 3 \ 0)$, deve-se que disparar a seqüência $s = t_1 \ t_2 \ t_1 \ t_2 \ t_3 = t_{12} \ t_{22} \ t_{31}$

$$\begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} \xrightarrow{t_1} \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \xrightarrow{t_2} \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} \xrightarrow{t_1} \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \xrightarrow{t_2} \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} \xrightarrow{t_3} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

cujo vetor característico é $S = (2, 2, 1, 0) = [2, 2, 1, 0]^T$

Um vetor característico pode corresponder portanto a diversas seqüências de disparo por exemplo $(2, 1, 0, 0)$ corresponde a ambas as seqüências $t_1 \ t_2 \ t_1$ e $t_1 \ t_1 \ t_2$, observando-se assim uma perda de informação na evolução da rede, pois em S não está representada a ordem de disparo das transições.

Observa-se também que da execução da RdP resulta dois tipos de seqüências:

- Seqüência de disparo (s);
- Seqüência de marcações (M_0, M_1, M_2, \dots).

Assim, a evolução da marcação de uma RdP para uma seqüência " s " são dadas então pela equação:

$$M' = M - \text{Pré } S + \text{Pós } S \quad (3.10)$$

que segundo (3.5), torna-se:

$$M' = M + C.S \text{ com } M \geq 0, e, S \geq 0 \quad (3.11)$$

esta equação é chamada de "*equação fundamental*" da RdP

3.6.- Grafo de Marcações Acessíveis

O comportamento dinâmico de uma RdP é definido pelo multiconjunto de marcações acessíveis $A(R, M)$ que se pode atingir a partir da marcação inicial, através de uma seqüência de disparo " s ".

$$A(R; M) = \{M_i, \exists s M \xrightarrow{s} M_i\} \quad (3.12)$$

Este conjunto de marcações pode ser representado na forma de um grafo $GA(R;M)$, o qual possui como conjunto de nós, ou conjunto de marcações acessíveis $A(R;M)$. Um arco orientado liga dois nós M_i e M_j , se existe uma transição t sensibilizada que permite passar de uma marcação M_i a uma outra M_j .

$$M_i \xrightarrow{t_j} M_j \quad (3.13)$$

Os arcos do grafo são rotulados pelas transições correspondentes. A figura 3.4 mostra o grafo de marcações acessíveis para a RdP da figura 3.2.

Observa-se que neste grafo a noção de processo desaparece, pois não se pode fazer mais a diferença entre as transições paralelas e as transições em conflito para uma dada marcação.

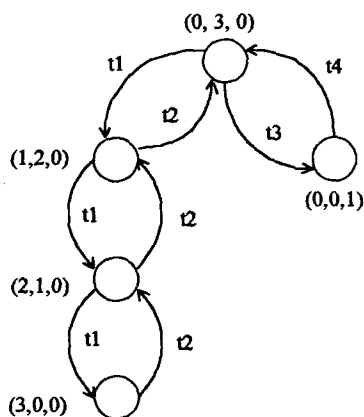


Fig. 3.4- Grafo de marcações acessíveis

3.7.- Principais Propriedades das Redes de Petri

Dentro das principais propriedades para um modelo de Redes de Petri podemos diferenciar dois tipos:

a) as relativas à RdP marcada ou propriedades dinâmicas (chamadas também de boas propriedades);

b) as relativas às RdP não marcadas, conhecidas como propriedades estruturais (Murata 1989; Cardoso 1993).

Propriedades Dinâmicas.- São as que dependem da marcação inicial e estão ligadas à evolução da rede. Sua verificação se faz geralmente pela construção do grafo de marcações acessíveis.

Propriedades Estruturais.- São aquelas que dependem da estrutura topológica das RdPs. Elas são independentes da marcação inicial M_0 . Assim sua análise está baseada na teoria da álgebra linear, e essas propriedades podem ser caracterizadas em termos da matriz de incidência "C".

É bom frisar que ambas propriedades estão inter-ligadas, já que a dinâmica de uma RdP vai depender diretamente de sua estrutura.

3.7.1.- Propriedades Básicas

As principais propriedades das Redes de Petri que possibilitam a análise do sistema modelado são:

a) **Alcançabilidade.**- A alcançabilidade é fundamental e básica para estudar as propriedades dinâmicas de qualquer sistema. O disparo de uma transição habilitada mudará a distribuição das fichas (marcação) na rede de acordo com as regras descritas anteriormente.

Definição.- Na RdP $(P, T, Pré, Pós, M_0)$ denomina-se de alcançabilidade de uma marcação M (representada por $A(R; M)$) ao multi conjunto de todas as marcações geradas a partir de M .

Daí $A(R, M) =$ grafo de marcações

Assim, uma marcação M_n é dita ser alcançável desde uma marcação M_0 , se existe uma seqüência de disparos que transforma M_0 em M_n . Um disparo ou seqüência de ocorrência é denotada por $s = M_0 t_1 M_1 t_2 M_2 \dots t_n M_n$, ou simplesmente $s = t_1 t_2 t_3 \dots$. Neste caso M_n é alcançável desde M_0 por "s", podendo-se escrever então:

$$M_0 [s > M_n \tag{3.14}$$

b) Limitação:

Definição. - Uma RdP marcada $N = (P, T, Pré, Pós, M_0)$ é dita ser k -limitada ou simplesmente limitada se o número de fichas em cada lugar não excede um número finito para qualquer marcação alcançável desde M_0 , isto é:

$$\exists k \in \mathbb{N} / M(p) \leq k, \forall M \in A(R, M_0) \quad (3.15)$$

Se $k = 1$ diz-se que o lugar é seguro, salvo ou binário. Por exemplo na rede da figura 3.5.a, cada marcação M' , a qual pode ser alcançável desde M_0 , tem no máximo uma ficha em p .

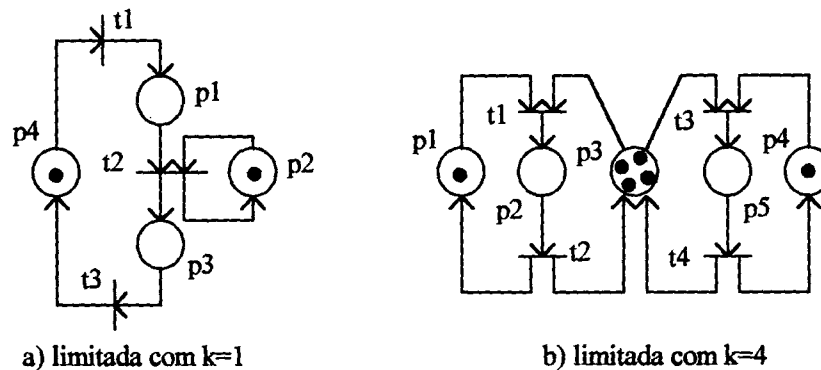


Fig. 3.5.- Rede de Petri marcada e k -limitada

Uma rede marcada N é k -limitada se e somente se todos os seus lugares são k -limitados.

Uma rede marcada N é segura se e somente se todos os seus lugares são seguros, diz-se

também *salva ou binaria*.

Se consideramos a RdP da figura 3.2 vê-se que para a marcação inicial $M = 3 * p_2$, o lugar p_3 é binário enquanto os lugares p_1 e p_2 são 3-limitados.

c) Vivacidade. - O conceito de vivacidade está relacionado com a total ausência de deadlock (bloqueios) na operação do sistema.

Definição. - Uma transição $t \in T$ é sempre viva se:

$$\forall M \in A(R, M_0), \exists M' \in A(R, M) / t \text{ é habilitada por } M' \quad (3.16)$$

Uma RdP marcada $N(R, M)$ é viva se e somente se todas as suas transições são vivas.

As transições podem ser vivas se for possível sempre encontrar uma seqüência de disparos que as contém, ou quase viva se for somente disparável uma vez (Cardoso 1993).

d) Reinicialização

Definição. - Uma RdP marcada $N = (R, M)$ é reinicializável para toda marcação se e somente se seu grafo de marcações acessíveis $GA(R, M)$ é fortemente conexo, isto é:

$$\forall M' \in A(R, M) \exists s / M' \xrightarrow{s} M_0 \quad (3.17)$$

Exemplo: Considerando a RdP da figura 3.6 cujo grafo de marcações é dado pela figura 3.7 ela é não reinicializável, pois não existe nenhuma seqüência que permite voltar à marcação inicial $M_0 = p_1 p_4$ após o disparo da transição t_1 .

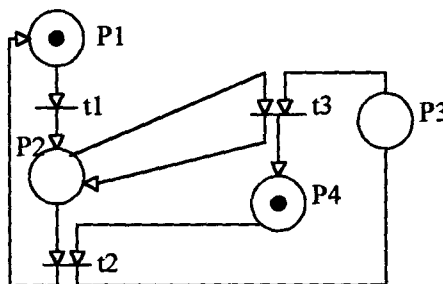


Fig. 3.6.- Exemplo de uma Rede de Petri não reinicializável

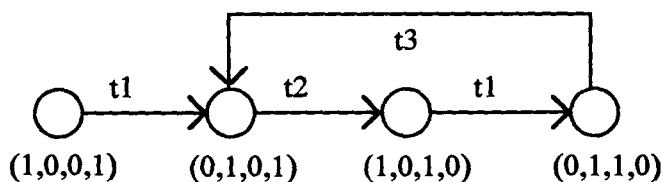


Fig. 3.7.- Grafo de Marcações Acessíveis da rede não reinicializável

Portanto se se considera para a rede da figura 3.6 a marcação $M_0 = p_2 p_4$, ela é ao mesmo tempo viva e reinicializável. Assim, conclui-se que a Reinicialização depende da marcação inicial e da estrutura da rede.

Observação: É importante salientar que as propriedades que foram definidas são fortemente ligadas à marcação inicial.

3.7.2.- Propriedades Específicas

a).- **Paralelismo.**- Atividades paralelas ou concorrentes podem ser facilmente expressadas em termos de Redes de Petri. Por exemplo, na figura 3.8, as atividades paralelas ou concorrentes são representadas pelas transições t_2 e t_3 . Em geral duas transições são ditas concorrentes se elas ocorrem independentemente, isto é, uma transição pode disparar antes ou depois ou em paralelo com a outra, como no caso de t_2 e t_3

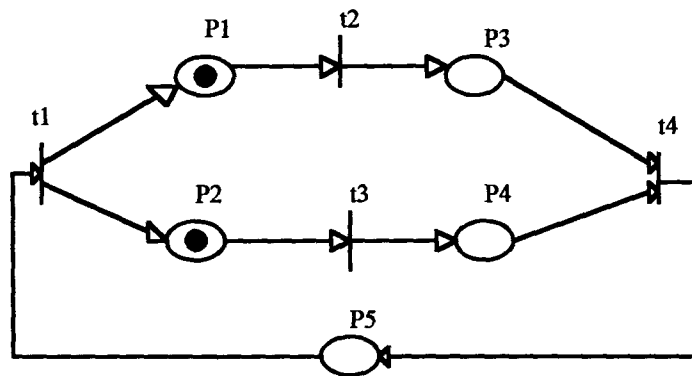


Fig. 3.8- Uma Rede de Petri representando uma atividade paralela

O paralelismo pode ser:

a.1) **Paralelismo Estrutural.**- Duas transições t_i e t_j são paralelas estruturalmente se:

$$\text{Pré}(p, t_i) \cap \text{Pré}(p, t_j) = \emptyset \quad (3.18)$$

isto quer dizer que elas (por ex. t_2 e t_3 na figura 3.8) não têm nenhum lugar de entrada em comum.

a.2) **Paralelismo Efetivo.**- Duas transições t_i e t_j são efetivamente paralelas para uma marcação dada M , se e somente se elas são paralelas estruturalmente e:

$$M \geq \text{Pré}(p, t_i) \quad (3.19)$$

$$M \geq \text{Pré}(p, t_j) \quad (3.20)$$

ex. na RdP da figura 3.8 as transições t_2 e t_3 são paralelas estruturalmente, pois:

$$\text{Pré}(p_1, t_2) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{e} \quad \text{Pré}(p_2, t_3) = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

então tem-se que $\text{Pré}(p_1, t_2) \cap \text{Pré}(p_2, t_3) = \emptyset$. Como t_2 e t_3 são paralelas estruturalmente e

$$M \geq \text{Pré}(p_1, t_2) \rightarrow [1, 1, 0, 0, 0]^T \geq [1, 0, 0, 0, 0]^T$$

$$M \geq \text{Pré}(p_2, t_3) \rightarrow [1, 1, 0, 0, 0]^T \geq [0, 1, 0, 0, 0]^T$$

então t_2 e t_3 são também efetivamente paralelas, já que elas podem disparar de forma independente entre si.

b) **Conflito.**- Um conflito (escolha ou decisão) numa Rede de Petri (figura 3.9) é uma situação que pode levar a uma disputa não determinística de recursos (fichas).

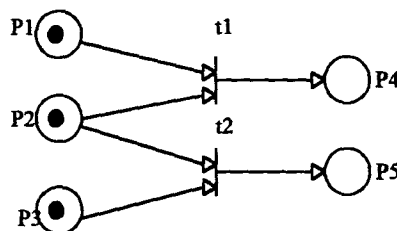


Fig. 3.9.- Representação de uma situação de conflito

O conflito pode ser:

b.1) Conflito Estrutural.- É dado quando duas transições t_j e t_k têm um lugar de entrada em comum, isto é:

$$\exists p_i / \text{Pré}(p_i, t_j) \cdot \text{Pré}(p_i, t_k) \neq 0 \quad (3.21)$$

b.2) Conflito Efetivo.- Duas transições t_j e t_k estão em conflito efetivo para uma marcação M se e somente se estão em conflito estrutural, isto é :

$$\forall M \in A(R, M_0), M(p_i) < \text{Pré}(p_i, t_j) + \text{Pré}(p_i, t_k) \quad (3.22)$$

Ex. na Rede de Petri da figura 3.2 as transições t_1 e t_3 estão em conflito estrutural pois:

$$\text{Pré}(p_2, t_1) \cdot \text{Pré}(p_2, t_3) = 3$$

e para a marcação inicial $M_0 = (0, 3, 0)^T$, as transições t_1 e t_3 estão em conflito efetivo, pois além de estarem em conflito estrutural, satisfazem a equação 3.22, isto é:

$$\begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} < \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \\ 0 \end{bmatrix}$$

Uma situação onde conflito e concorrência são misturados é chamada de confusão.

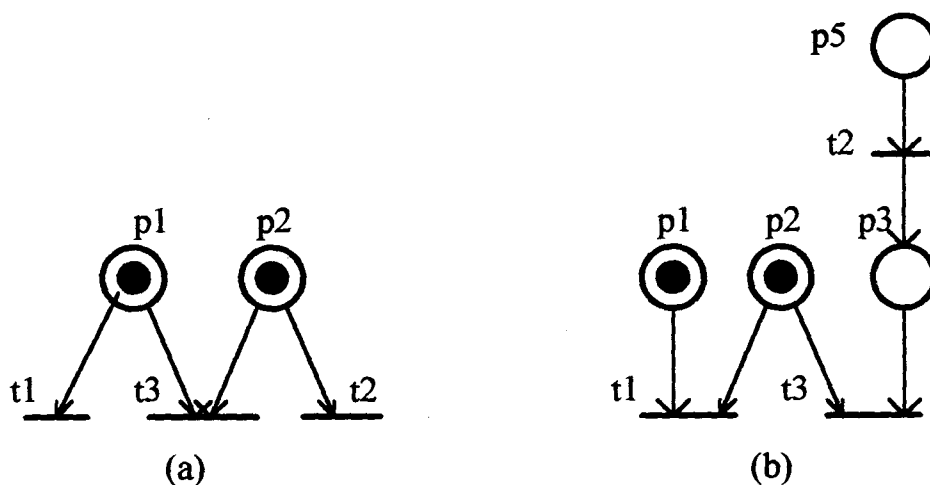


Fig. 3.10- Dois tipos de confusão a) Confusão simétrica e b) confusão assimétrica

A figura 3.10 (a) mostra uma confusão simétrica, desde que t_1 e t_2 são concorrentes e ambas as transições estão em conflito com t_3 ; por outro lado a figura 3.10 (b) mostra uma confusão assimétrica onde t_1 é concorrente com t_2 , mas estará em conflito com t_3 se t_2 disparar primeiro.

c) Natureza Assíncrona.- Não existe medida inerente ao tempo. Quando mais de uma transição está habilitada ao mesmo tempo, então qualquer uma delas pode ser a próxima a disparar, portanto o disparo de uma transição é feito de uma maneira não determinística.

Dada a natureza assíncrona do modelo de RdP clássica, a única propriedade importante do tempo está em definir uma ordem parcial da ocorrência das transições.

O disparo de uma transição pode ser considerado portanto instantâneo, garantindo desse modo que duas transições não podem disparar simultaneamente. Estes tipos de transições são também chamadas de transições primitivas. Quando uma transição não é primitiva, isto é, se sua ocorrência requer um tempo maior do que zero, ela pode ser decomposta em duas transições primitivas, uma inicial e outra final com um lugar entre elas, representando que o evento primitivo está acontecendo conforme visto na figura seguinte.

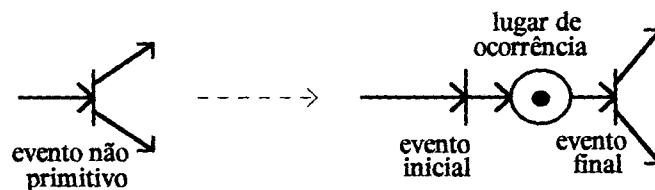


Fig. 3.11.- Transição não Primitiva

d) Possibilidade de Hierarquia

Numa rede de Petri um lugar ou uma transição pode ser substituída por uma sub-rede o que leva a uma abstração maior, ou vice-versa, permitindo assim refinar a rede (Zhou 1992 (b)) conforme visto na figura seguinte.

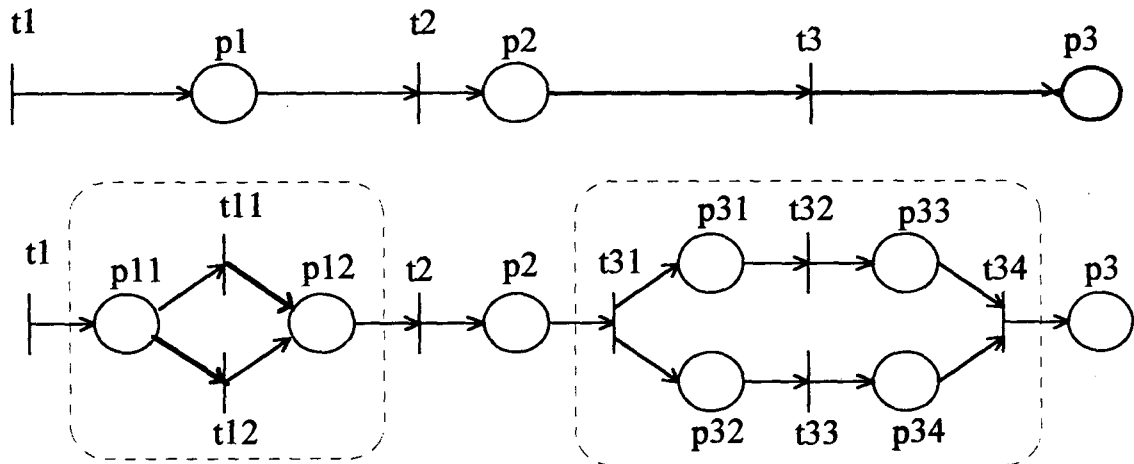


Fig 3.12.- Hierarquia numa Rede de Petri

3.7.3. - Invariantes

Os invariantes são propriedades derivadas diretamente da estrutura da RdP e que portanto não dependem de sua marcação inicial.

Definição.- Seja V um vetor de posições ponderadas de uma RdP. Se $V.C = 0$, então $V.M_k = V.M_0$ para todas as marcações M_k e M_0 , onde M_k é alcançável desde M_0 , então V é dita uma invariante (Jensen 1981).

Se uma seqüência de disparo s é aplicada desde uma marcação M_i , a marcação M_k , a qual é alcançada, é dada pela equação fundamental 3.11, $M_k = M_0 + C.S$.

Existem 2 tipos de invariantes a saber: Invariante de lugar e Invariante de transição (Cardoso 1993; Murata 1981).

a) Invariante de Lugar.- Um invariante de lugar é uma função linear de marcação dos lugares, cujo valor é uma constante que depende apenas da marcação inicial da rede.: Um vetor X é um L-invariante se:

$$X^T.C = 0 \quad (3.23)$$

(Lautenbach and Schmid, 1974). Tal vetor tem uma propriedade muito importante. De $M_k = M_0 + C \cdot S$, onde S é o vetor característico de uma seqüência de disparo s , conduzida desde a marcação inicial M_0 para a marcação alcançável M_k , assim, nós obtemos:

$$X^T \cdot M_k = X^T \cdot M_0 + X^T \cdot C \cdot S \quad (3.24)$$

Desde que $X^T \cdot C = 0$, pode-se deduzir que:

$$X^T \cdot M_k = X^T \cdot M_0 \quad (3.25)$$

para qualquer marcação alcançável M_k .

Um L-invariante é uma propriedade estrutural já que ela não depende da marcação inicial. Entretanto, o valor de $X^T \cdot M_0$ depende da marcação inicial, e ela corresponde a uma marcação invariante a qual esta associada com X .

Uma solução da equação 3.23 é chamada de componente conservativa de uma RdP, a qual sob o ponto de vista gráfico define uma sub-rede de Petri.

Se X^T é solução de 3.23, então a equação 3.25 é o invariante de lugar correspondente.

Esta equação depende de M_0 enquanto que a componente conservativa é completamente independente. Por ex. na figura 3.13, $X_2 = (1, 0, 1, 1, 0)$ é um L-invariante achado via equação 3.23 independente da marcação. Para a marcação inicial $M_0 = (1, 0, 0, 0, 0)$ a marcação invariante X_2 é igual a $m_1 + m_3 + m_4 = 1$, desde que $X^T \cdot M_0 = 1$. Isto significa que o número de fichas no conjunto de posições $\{p_1, p_3, p_4\}$ é constante.

b) Invariante de Transição.- Um invariante de transição é uma seqüência de disparo de transições que não modifica a marcação da rede. Tal invariante corresponde a uma seqüência cíclica de eventos que pode ser repetida indefinidamente.

Um vetor Y , é um T-invariante, se :

$$C.Y = 0 \quad (3.26)$$

Se existe uma seqüência de disparos desde uma marcação M_i tal que $S = qY$ (donde q é um inteiro positivo) então s leva de volta para M_i . De fato a marcação alcançada desde M_i pela seqüência de disparo s é dada pela equação fundamental.

Desde que:

$$C.S = 0 \quad (3.27)$$

então;

$$M_k = M_0 \quad (3.28)$$

Toda solução S da equação 3.27 é chamada de componente repetitiva estacionária.

Se S é o vetor característico de uma seqüência de disparos " s " de transições efetivamente disparadas a partir de uma marcação acessível, então, esta seqüência " s " é um invariante de transição.

O cálculo das componentes conservativas, assim como repetitivas são obtidas baseando-se em eliminação gaussiana como é visto em Murata (1989). Já um método simplificado e bastante fácil para calcular os invariantes é achado em Cardoso (1993). Na atualidade existem softwares que facilitam a obtenção dos invariantes (Maziero 1990).

Aplicando as formulas 3.23 e 3.26 nós achamos os invariantes para a Rede da figura 3.13, onde

Por ex.: $Y_1 = (1, 1, 0, 0, 1)$ é um T-invariante (achado via equação 3.26) para a figura 3.13 independente da marcação, isto significa que existe alguma seqüência de disparos " s " aplicável desde M_i em que cada transição t_1, t_2, t_5 aparece o mesmo número de vezes (por causa de que seus pesos são os mesmos em Y), então o disparo de " s " desde M_i leva de volta a M para a marcação inicial $M_0 = (1, 0, 0, 0, 0)$.

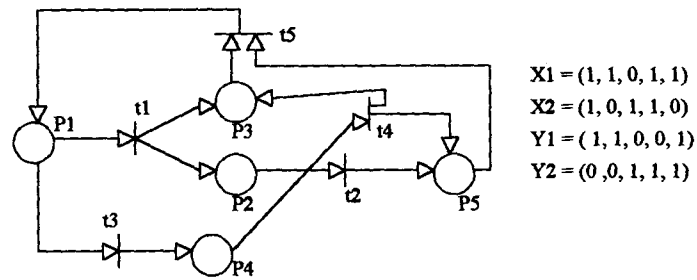


Fig. 3.13.- Exemplo de Invariantes numa Rede de Petri

3.7.4.- Exemplo de Propriedades de uma Rede de Petri

A seguir mostram-se três Redes de Petri para ilustrar alguns dos conceitos emitidos até aqui.

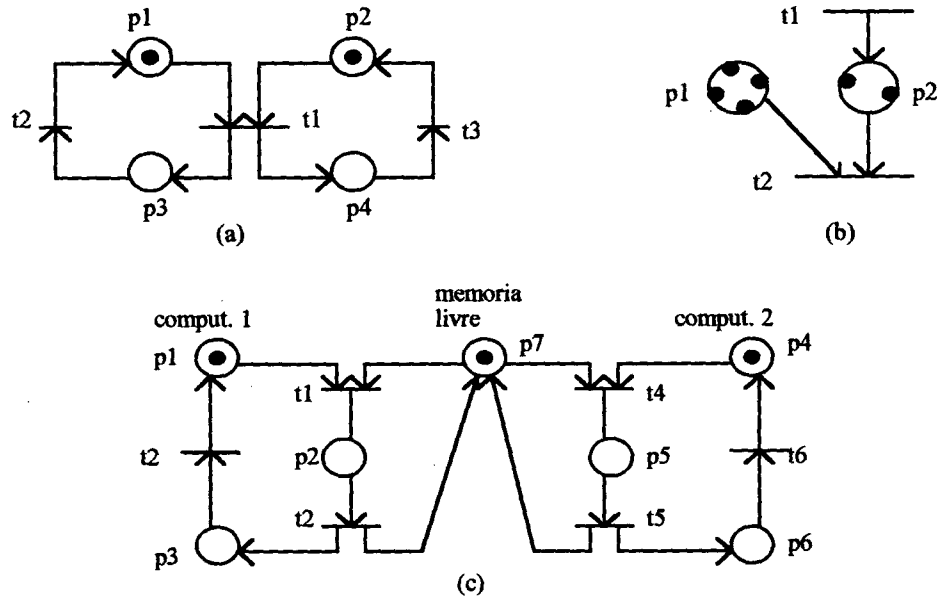


Fig. 3.14 - Exemplos de Redes de Petri

A figura (a) mostra-nos os seguintes conceitos:

- i) É uma Rede de Petri autónoma (tempo nem sincronização estão envolvidas no modelo);
- ii) A Rede é limitada
- iii) A RdP é viva e reinicializável através dos disparos t_1, t_2, t_3 ou t_1, t_3, t_2 ;
- iii) A RdP é viva e reinicializável através dos disparos t_1, t_2, t_3 ou t_1, t_3, t_2 ;
- iv) Existem duas marcações invariantes. A primeira é $M(p_1) + M(p_3) = 1$ o qual significa que para qualquer marcação alcançável M_1 , o número de fichas no conjunto de posições p_1 e p_3 é sempre igual a

1. Pode-se escrever também $m_1 + m_3 = 1$. Similarmente $m_2 + m_4 = 1$. Ambos conjuntos $\{p_1, p_3\}$ e $\{p_4, p_2\}$ são componentes conservativas;

v) A RdP põe também em evidência propriedades como: concorrência/paralelismo (através do disparo em forma independente de t_2 e t_3 quando $M_i = (0, 0, 1, 1)$ é alcançada), sincronização e difusão (o qual é ilustrada pela transição t_1).

por outro lado na figura (b) podemos verificar os seguintes conceitos:

i) t_1 é uma transição fonte (quer dizer que sempre esta habilitada);

ii) A RdP é ilimitada (unbounded) devido a que cada disparo de t_1 adiciona uma ficha a p_2 ;

iii) A RdP não é viva. Devido a que após t_2 ter disparado 4 vezes, não existem mais fichas em p_1 inabilitando assim o disparo de t_2 .

Por ultimo na fig. (c) observamos 2 conceitos muito importantes:

i) A RdP mostra a existência de um conflito, já que o disparo das transições t_1 e t_4 têm um lugar em comum de entrada que é p_7 , sendo portanto um conflito estrutural. Como existe uma ficha em cada lugar p_1, p_4, p_7 então existe um conflito efetivo entre as transições t_1 e t_4 , já que o disparo de uma delas inabilita o disparo da outra;

ii) Desde que a memória pode ser usada por dois computadores (compartilhamento de recursos), isto implica um conflito. Existe uma marcação invariante $m_2 + m_5 + m_7 = 1$ significando que se existe uma ficha em p_2 , não existirá ficha em p_5 e vice-versa, expressando que a memória não pode ser usada por ambos computadores ao mesmo tempo (exclusão mútua).

3.8.- Técnicas de Análise das Redes de Petri

Uma vez expostas algumas das principais propriedades das Redes de Petri, faz-se necessário determinar se uma RdP possui ou não tais propriedades para o qual desenvolveu-se certas metodologias

de análise (= verificação das propriedades). essas metodologias podem ser classificadas em três grandes grupos seguintes (Murata 1989; Cardoso 1993):

a) Análises por Enumeração das Marcações: (feita através da árvore de alcançabilidade/cobertura) é dinâmica e simula a execução da RdP a partir da marcação inicial.

b) Análise Estrutural: (baseada na equação fundamental das RdP), é estática, isto é, depende apenas da topologia da rede.

c) Análise por redução: Procura a eliminação de lugares e/ou transições que não afetam as propriedades das RdPs e posteriormente aplicar uma das técnicas anteriores.

3.8.1 - Análise por Enumeração das Marcações

Este método envolve essencialmente a enumeração de todas as marcações alcançáveis. Conforme visto em 3.6 o grafo de marcações acessíveis pode ser ilimitado. O algoritmo que permite decidir se uma RdP é k-limitada é baseado na construção de uma árvore chamada "árvore de cobertura".

3.8.1.1.- Árvore de Cobertura

conforme visto, o Grafo de Marcações Acessíveis de uma RdP pode ser ilimitado. Para torná-lo limitado vamos definir um ente algébrico denotado por ω e definido por:

$$\omega + a = \omega$$

$$\omega - a = \omega$$

$$a < \omega$$

$$\omega \leq \omega \quad \forall a \in \mathbb{N}$$

e construir um novo grafo denominado de *Árvore de Alcançabilidade* que terá dois tipos de nós, o nó raiz (igual a M_0) e o nó fronteira (marcação alcançada). O nó fronteira deverá ser processado e irá transformar-se num dos três tipos de nós:

- **terminal** (é um nó que não habilita nenhuma transição $t \in T$);
- **duplicado** (é um nó que já existe, isto é, que a árvore já tem outro igual);
- **interno** (é qualquer nó situado entre um nó raiz ou duplicado ou terminal).

Para construirmos a árvore podemos utilizar o algoritmo seguinte: (Murata 1989)

- considerar M_0 como nó fronteira;
- enquanto existir nó fronteira fazer:
 - seleccionar uma marcação M ;
 - se existir outra marcação igual a M na árvore, considerá-la duplicada e finalizar seu processamento;
 - se M não habilitar nenhuma $t \in T$, considera-la terminal e finalizar seu processamento;
 - para cada $t \in T$ habilitada por M fazer:
 - obter M' disparando t ;
 - se no caminho que liga M' a M_0 existir outra marcação M'' tal que M' é cobertura de M'' , isto é $M'(p) \geq M''(p)$, fazer $M'(p) = \omega \forall p \in P$, onde $M'(p) > M''(p)$;
 - considerar M' como fronteira.
- Finalizar o processamento quando todos os nós forem classificados em terminal, duplicado ou interno.

Ex. Consideremos a RdP mostrada na Figura 3.15 para a marcação inicial $M_0 = (1, 0, 0)$. Só a transição t_1 está habilitada.

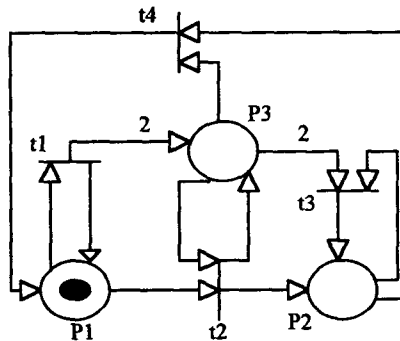


Fig. 3.15.- Rede de Petri marcada com possível Deadlock

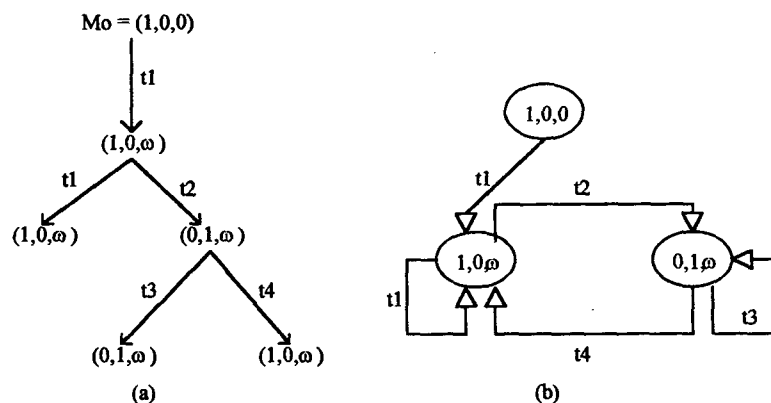


Fig.-3.16.- a) Árvore de Alcançabilidade e b) Grafo de Marcações Acessíveis para a Figura 3.15

3.8.1.2.- Análise pela Árvore

a) **Limitação.**- A RdP é limitada se e somente se o " ω " não aparecer na árvore de alcançabilidade. A aparição do símbolo " ω " como parte da árvore de cobertura significa que o número de fichas nesse lugar é potencialmente ilimitado. Assim se aparecer " ω " a rede é ilimitada.

b) **Conservação.**- Caso o " ω " não aparecer e:

b.1) $\sum M(p) = \sum M(p)$, então é conservativa pura;

b.2) Se $\exists W / \sum w_i \cdot M(p_i) = \sum w_i \cdot M_0(p_i)$, então é conservativa ponderada, e algebricamente difícil obter pela árvore.

c) **Reinicialização.**- Se não existir nó terminal e todos os nós forem duplicados iguais ao nó raiz, então a rede é reinicializável.

d) **Alcançabilidade.**- M é sempre alcançável, se aparecer explicitamente em todos os ramos da árvore de alcançabilidade.

e) **Vivacidade.**- Se não existir nós duplicados é sempre morta.

3.8.1.3.- Limitações da Árvore de Alcançabilidade na Análise das Redes de Petri

Como foi visto, a árvore de alcançabilidade pode ser usada para solucionar os problemas de cobertura, limitação e conservação, mas ela não pode em geral ser usada para solucionar os problemas de alcançabilidade ou vivacidade ou para definir ou determinar que sequências de disparo são possíveis. Esses problemas são limitados pela existência do símbolo " ω " a mesma que esconde informação muito importante sobre a RdP. Por exemplo nas Redes de Petri das Figuras 3.15 e 3.17, cuja árvore de alcançabilidade é mostrada pela Figura 3.16 (a), a mesma árvore representa essas duas RdPs similares (mas diferentes). Na RdP da Figura 3.15, o número de fichas que chega a p_3 após o disparo de t_1 é sempre um número par, ao passo que na Figura 3.17 ela pode ser um inteiro arbitrário. O símbolo " ω " não permite detectar este tipo de informação.

Um problema similar acontece para o problema de vivacidade. As Figuras 3.15 e 3.17, são duas RdPs cuja árvore de alcançabilidade é dada pela Figura 3.16 (a), entretanto a Figura 3.15 pode morrer após a sequência $s = t_1 t_2 t_3$ por exemplo, por causa da existência de um possível nó terminal (deadlock), enquanto que a RdP da Figura 3.17 não pode morrer. Uma vez mais a árvore de alcançabilidade não pode distinguir entre estes dois casos.

É importante salientar também que na mesma estrutura de uma RdP, para uma nova marcação M_0 , tem-se que construir uma nova árvore, pois as propriedades podem ser alteradas.

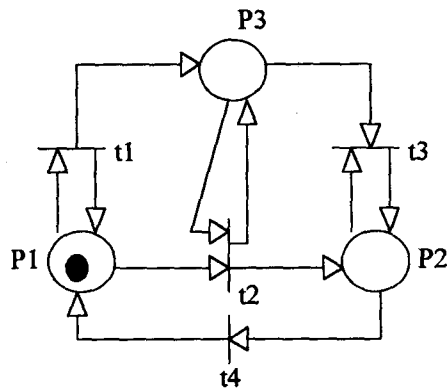


Fig.3.17.- Rede de Petri sem Deadlock

3.8.2.- Análise Estrutural das Redes de Petri

Uma segunda abordagem para análise de RdPs é baseada sobre a matriz de incidência das RdPs (equação 3.5). Assim para tentar verificar certas propriedades não obtidas na árvore, vamos usar aqui apenas a estrutura da RdP, para o qual vamos primeiro definir alguns conceitos.

a) Uma transição t_j representada algêbricamente pelo vetor unitário n-dimensional $[t_j] = [0, 0, 0, \dots, e_j, \dots, 0, 0, 0]^T$, o qual é zero em toda parte exceto na j-esima componente.

Ex.

$$[t_3] = [0 \ 0 \ 1 \ 0]^T$$

$$[t_1] = [1 \ 0 \ 0 \ 0]^T$$

b) A função próximo estado $M \xrightarrow{t} M'$ onde:

$$\delta(M, t_j) = M' = M + C^+ \cdot [t_j] - C^- \cdot [t_j]$$

$$M' = M + (C^+ - C^-) \cdot [t_j]$$

$$M' = M + C [t_j] \tag{3.32}$$

Ex: Dada a Rede de Petri da figura 3.18 com sua respectiva matriz de incidência, sabe-se que a transição t_1 esta habilitada por M_0 :

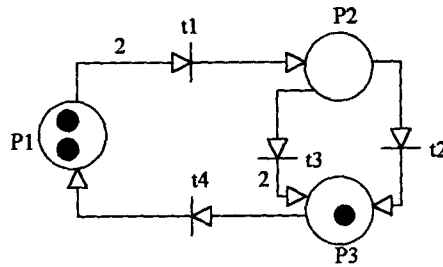


Fig. 3.18.- Rede de Petri

$$C = C^+ - C^- = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \end{bmatrix} - \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -2 & 0 & 0 & 1 \\ 1 & -1 & -1 & 0 \\ 0 & 1 & 2 & -1 \end{bmatrix}$$

como $M_0 \geq C^- [t_1]$, disparando-a obtemos:

$$M_1 = M_0 + C [t_1]$$

$$M_1 = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -2 & 0 & 0 & 1 \\ 1 & -1 & -1 & 0 \\ 0 & 1 & 2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -2 \\ 1 \\ 0 \end{bmatrix}$$

$$M_1 = [0, 1, 1]^T$$

Dai, tomando a seqüência de disparo $s = t_{jk}$ (onde j = transiço disparada, e k = nmero de vezes) e aplicando a funço prximo estado vem:

$$\delta (M,s) = M' = M + C [t_{jk}]$$

$$M' = M + C [t_{1k}] + C [t_{2k}] + \dots + C [t_{nk}]$$

$$M' = M + C ([t_{1k}] + [t_{2k}] + \dots + [t_{nk}])$$

$$M' = M + C.f(s) \tag{3.33}$$

Onde $f(s) = ([t_{1k}] + [t_{2k}] + \dots + [t_{nk}])$ é chamado de vetor característico da seqüência $t_{1k}, t_{2k}, \dots, t_{nk}$, denominado também de *vetor de Parikh* e sua j -ésima componente de $f(s)$, $f(s)_j$ indica o número de vezes que t_j dispara na seqüência $t_{1k}, t_{2k}, \dots, t_{nk}$.

Na Rede de Petri da Figura 3.18, por exemplo, o estado alcançável aplicando $f(s) = [1, 0, 1, 2]^T$ a partir de M_0 será:

Aplicando 3.33 temos:

$$M' = M_0 + C f(s) = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -2 & 0 & 0 & 1 \\ 1 & -1 & -1 & 0 \\ 0 & 1 & 2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 2 \end{bmatrix} = [2, 0, 1]^T + [0, 0, 0]^T$$

$$M' = [2, 0, 1]^T$$

3.8.2.1.- Verificação das Propriedades

a) Conservação.- Dada uma RdP marcada para mostrar a conservação é necessário achar um vetor peso $W = [w_1, w_2, \dots, w_n]^T$, para a qual a soma dos pesos em toda marcação alcançável seja constante; isto é:

$$\sum M(p_j) w_j = M_0(p_j) w_j, \forall M \in A(R; M_0) \quad (3.34)$$

ou seja :

$$M \cdot W = M_0 \cdot W, \forall M \in A(R; M) \quad (3.35)$$

Desde que M é alcançável existe uma seqüência " s " de transições disparadas que levam a rede de M_0 para M . Assim de 3.33 temos que :

$$M = M_0 + C \cdot f(s)$$

$$M^T = M_0^T + f(s)^T \cdot C^T$$

$$M^T \cdot W = M_0^T \cdot W + f(s)^T \cdot C^T W \quad (3.36)$$

Dai

$$f(s)^T \cdot C^T W = 0 \quad (3.37)$$

como

$$f(s)^T \neq 0$$

então:

$$C^T \cdot W = 0 \quad (3.38)$$

Logo, a RdP é conservativa se e somente se existir um vetor positivo W tal que $C^T \cdot W = 0$

Exemplo. Na seguinte RdP temos:

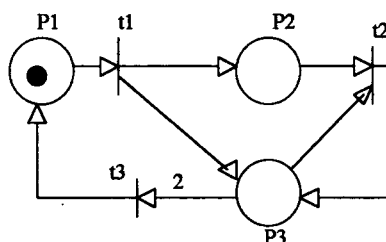


Fig. 3.19.- Rede de Petri não conservativa

$$C = \begin{bmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -2 \end{bmatrix}$$

e aplicando 3.38 temos:

$$C^T \cdot W = \begin{bmatrix} -1 & 1 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & -2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \begin{aligned} -w_1 + w_2 + w_3 &= 0 \\ -w_2 &= 0 \\ w_1 - 2w_3 &= 0 \end{aligned}$$

não é conservativa nem pura e nem ponderada.

b) **Alcançabilidade.**- Assumindo que uma marcação M' é alcançável desde uma marcação M . Então ali existe uma seqüência "s" de transições disparadas as quais levam M para M' . Isto significa que $f(s)$ é uma solução.

Em 3.33 temos que:

$$M' = M + C.f(s)$$

$$M' - M = C.f(s)$$

$$(\Delta M)^T = (C.f(s))^T$$

$$f(s)^T.C^T = (\Delta M)^T \tag{3.39}$$

que é um sistema linear.

Teorema 3.1- É condição necessária para que M' seja alcançável a partir de M que 3.39 tenha solução inteira e positiva

Na RdP anterior por exemplo vamos verificar se as marcações $M' = [0, 2, 1]^T$ e $M'' = [5, 3, 2]^T$ são alcançáveis partindo da marcação inicial $M = [2, 1, 0]^T$.

Para $M' = [0, 2, 1]^T$, temos que.

$$\Delta M = M' - M = [0, 2, 1]^T - [2, 1, 0]^T$$

$$\Delta M = [-2, 1, 1]^T$$

aplicando 3.39 temos:

$$\begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} -1 & 1 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & -2 \end{bmatrix} = \begin{bmatrix} -2 & 1 & 1 \end{bmatrix}$$

de onde resulta que $x_1 = 3$, $x_2 = 2$ e $x_3 = 1$ então pelo teorema M' pode ser alcançável (com $s = t_1 t_1 t_2 t_3 t_1 t_2$).

Para $M'' = [5, 3, 2]^T$ e $M = [3, 2, 2]^T$

aplicando 3.39 obtemos que $x_1 = -8$, $x_2 = -10$, $x_3 = -5$, portanto M'' nunca será alcançável.

Note-se que a solução da equação 3.39 nos permite conhecer apenas o número de vezes que cada $t \in T$ deve disparar, mas não nos indica a (s) ordem (s) de disparo.

Outro problema é que embora a solução da equação 3.39 é condição necessária, ela não é suficiente. Pode existir uma $f(s)$ com componentes naturais e a marca objetivo não ser alcançável.

Por exemplo, na seguinte RdP:

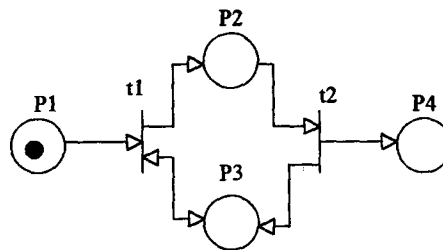


Fig. 3.20.- Rede de Petri mostrando condição necessária mas não suficiente

Se queremos determinar se $M' = [0, 0, 0, 1]^T$ é alcançável a partir de $M = [1, 0, 0, 0]^T$ aplicando a equação 3.39, temos que $f(s) = x = (1, 1)$ correspondendo a duas sequências $t_1 t_2$ ou $t_2 t_1$. Mas nenhuma dessas duas sequências de transições são possíveis, desde que nem t_1 nem t_2 estão habilitadas (a rede está em deadlock) em $(1, 0, 0, 0)$, assim a solução da equação 3.39 não é suficiente para assegurar alcançabilidade.

3.8.2.2.- Verificação das Propriedades das Redes de Petri através dos Invariantes

Como visto em 3.7.3 numa Rede de Petri $R = (P, T, \text{Pré}, \text{Pós})$ um invariante de lugar é todo vetor $X = [x_1, x_2, \dots, x_n]^T$ com $x_i = 0$; ou $x_i = 1$ que seja solução linearmente independente (LI) de $X^T \cdot C = 0$. E um invariante de transição é todo vetor $Y = [y_1, y_2, \dots, y_n]^T = (y_1, y_2, \dots, y_n)$ com $y_i \in \mathbb{N}$ que seja solução linearmente independente de $C \cdot Y = 0$

a) Conservação:

Se na PR = (P, T, Pré, Pós, M₀) ocorrer que:

$$P = X_1 \cup X_2 \cup \dots \cup X_k \text{ e } X_i \cap X_j = \emptyset, \forall i \neq j$$

então a RdP é conservativa pura.

b) Limitação:

Se $\forall p \in P$ temos que $p \in X_j$, sendo X_j um invariante de lugar, então, a Rede é limitada.

c) alcançabilidade:

visto em 3.8.2.1 (b)

d) Vivacidade:

Toda RdP viva e limitada é necessariamente coberta por invariantes de transição. Isto permite negar a vivacidade, pois se for limitada e não coberta não pode ser viva.

e) Reinicialização:

Uma RdP é reinicializável se existir invariantes de transição.

3.8.3- Análise por Redução

Quando a complexidade de sistemas do mundo real nos leva a construção de uma rede muito grande, resulta impossível analisar a rede utilizando o grafo de alcançabilidade ou métodos invariantes. Daqui que muitos pesquisadores têm desenvolvido técnicas de redução (Berthelot 1982; Lee 1985,1987), que podem servir como métodos de sínteses, se aplicados apropriadamente. Esta técnica consiste portanto em reduzir o tamanho da rede, utilizando certas regras de redução de maneira tal que

a rede inicial e a rede reduzida tenham as mesmas propriedades. Esta redução é feita retirando da RdP lugares e/ou transições que não afetam as propriedades, via regra. Sendo portanto a razão básica de aplicar métodos de redução, a de diminuir a complexidade da análise. Redes de Petri de alto nível tais como RdPs Predicado/transição, RdPs Coloridas (Jensen 1981) e algumas RdPs estendidas, podem ser usadas também para superar o problema de complexidade.

CAPÍTULO 4 - ABREVIACÕES, EXTENSÕES E ESTRUTURA PARTICULAR DAS REDES DE PETRI

4.1.- Introdução

As Redes de Petri Ordinárias como mostradas no capítulo anterior, também chamadas de primitivas ou autônomas (David 1994) possuem baixo poder de modelagem por representarem apenas relações de causa e efeito entre os eventos e as condições. A sua utilização é restringida portanto a diversos tipos de sistemas pertencentes a classe de sistemas (dinâmicos) de eventos discretos, onde sincronização externa e o tempo não intervêm. Um dos campos de aplicação mais freqüentes é protocolos de comunicação em sistemas de computador. Desde que concorrência, sincronização e compartilhamento de recursos possam ser achados na especificação de tais sistemas, as Redes de Petri são uma ferramenta muito apropriada para sua modelagem (Murata 1981; Berthelot 1982). Um campo de aplicação que tornou-se importante durante a última década são os sistemas de manufatura (Silva 1989; Valette 1990). Aqui: concorrência (duas máquinas trabalhando independentemente), sincronização (Uma máquina está livre esperando uma parte ficar pronta para ser processada pela mesma) e compartilhamento de recursos (um robô é requerido para manusear partes por duas máquinas, mas não pode servir ambas ao mesmo tempo) são características usuais de tais sistemas.

Redes de Petri são utilizadas portanto com sucesso na modelagem de diversos sistemas que podem ser vistos mais amplamente em (Murata 1989; Peterson 1981; Favrel 1985).

O objetivo do presente capítulo tem portanto a finalidade de mostrar algumas das abreviações assim como extensões mais importante das Redes de Petri que nos permitem ampliar seu horizonte de aplicações, sobre tudo em sistemas de manufatura.

Na literatura as Redes de Petri geralmente estão divididas em três principais classes: Redes de Petri Ordinárias (versão original ou modelo básico), abreviações e extensões.

Numa **RdP Ordinária** todos os arcos têm o mesmo peso o qual é 1, existindo apenas um só tipo de ficha, a capacidade de lugares é infinita (isto é, o número de fichas não é limitado pela capacidade dos lugares), o disparo de uma transição pode acontecer se cada lugar precedente tiver no mínimo uma ficha, e nenhum tempo é envolvido.

As **abreviações** correspondem a representações simplificadas que têm por finalidade facilitar a representação gráfica e possuem o mesmo poder de modelagem das RdPs. Dentro desta classe estão consideradas as RdPs Generalizadas, RdP com capacidade finita, e as RdPs Coloridas. Aqui todas as propriedades de uma RdP Ordinária são mantidas com umas poucas adaptações.

As **extensões** por outro lado correspondem a modelos para os quais as regras de funcionamento sofrem algumas variações com a finalidade de enriquecer a capacidade de representação do modelo inicial. Aqui podem-se considerar três tipos de subclasses:

- a) extensões que têm o poder de representação de máquinas turing (RdP com arcos inibidores e RdPs prioritárias);
 - b) extensões que permitem a modelagem de RdPs híbridas e RdPs contínuas;
 - c) extensões correspondentes a modelos que descrevem o funcionamento de sistemas cuja evolução vai depender de eventos externos e/ou tempo (RdP sincronizada, RdP temporizada e RdPs estocásticas).
- Aqui nem todas as propriedades de uma RdP Ordinária são mantidas.

Algumas estruturas particulares de Redes de Petri têm interessantes propriedades, por exemplo os grafos a eventos (ou grafo marcado) e os grafos de estado (ou máquina de estado), os quais são importantes de um ponto de vista prático. Ambos os tipos de grafos são Redes de Petri simples.

Um Grafo de eventos é tal que cada lugar tem exatamente uma transição de entrada e uma de saída. Por exemplo a rede da figura 3.8 é um grafo de eventos. Num grafo de eventos não existe conflito, já que cada lugar tem exatamente uma transição de saída. Em geral, existem sincronizações que correspondem a transições com diversos lugares de entrada. Sendo assim os grafos de eventos são bem adaptados para modelar sistemas cujo comportamento qualitativo é determinístico. Uma outra

característica deste tipo de grafo é que ele é fortemente conexo. Isto significa que para qualquer par de nós "u" e "v" (lugares e/ou transições) existe um caminho ou circuito de "u" para "v" que leva de volta de um nó para si mesmo. Circuitos passando no máximo uma vez por cada nó são chamados de *circuitos elementares*. Um grafo de eventos fortemente conexo tem as seguintes propriedades: a) um L-invariante é associado com cada circuito elementar; b) existe um único T-invariante com peso 1 associado com cada transição e c) um grafo de eventos fortemente conexo é vivo se e somente se existe no mínimo uma ficha em cada circuito elementar.

Um grafo de estado, por outro lado, é tal que cada transição tem exatamente um lugar de entrada e um lugar de saída. Num grafo de estado não existe sincronização e em geral existe conflitos. Para um grafo de estado fortemente conectado (figura 4.1) podemos apreciar as seguintes propriedades: a) Um T-invariante é associado com cada circuito elementar; b) existe um único L-invariante com peso 1 associado com cada lugar; c) é vivo se e somente se existe no mínimo uma ficha na RdP.

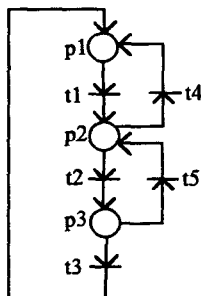


Fig. 4.1.- Grafo de estado fortemente conectado

A seguir serão dadas a conhecer as abreviações, assim como as extensões das Redes de Petri que vão nos permitir aplicá-las dependendo do tipo de sistema a ser modelado, e segundo nossos requerimentos.

4.2.- Abreviações

a) RdP Generalizada

É uma RdP onde pesos (inteiros estritamente positivos) são associados aos arcos. A Figura 3.2 representa por ex. uma RdP generalizada, já que os arcos de $p_2 \rightarrow t_3$ e de $t_4 \rightarrow p_2$ possuem peso 3.

Para os demais arcos que não esteja especificado explicitamente, o peso será 1. Quando um arco $p_i \rightarrow t_j$ tem um peso P , significa que a transição só será habilitada se o lugar p_i tiver no mínimo P fichas. Quando a transição t_j é disparada, P fichas serão então tomadas de p_i . Similarmente quando um arco $t_j \rightarrow p_k$ tem um peso P , vai significar que quando a transição t_j é disparada, P fichas serão colocadas em p_k .

Em geral todas as RdPs Generalizadas podem ser transformadas em RdPs Ordinárias, já que as propriedades das RdPs Ordinárias podem ser adaptadas para as RdPs generalizadas. Por ex. a equação fundamental 3.11 é válida para a RdP Generalizada como mostra a matriz de incidência da Figura 3.2 no capítulo anterior.

b) RdP Colorida

Quando os sistemas começam a se tornar complexos, como é o caso de um sistema de manufatura, começam a aparecer certos problemas na modelagem, onde já não é possível tais sistemas serem modelados por RdPs Ordinárias. Esta complexidade (Cardoso 1993) significa às vezes a composição de vários processos semelhantes. Neste caso, quando se utiliza a RdP Ordinária (com a marcação dos lugares dada por fichas indiferenciadas e com os lugares se comportando como contadores) tem-se duas escolhas:

- 1.-Modelar o comportamento geral sem precisar a identidade de cada processo, mas somente seu número;
- 2.-Modelar individualmente cada um dos processos que constituem o sistema, e modelar a interação existente entre eles; o que consiste muitas vezes em desdobrar o modelo que representa o comportamento geral.

No primeiro caso se obtém uma descrição compacta, mas não detalhada o suficiente: há uma falta de informação. No segundo caso o modelo obtido pode ser pouco prático de se trabalhar seja pelo tamanho da rede, seja pelo número de interações existentes. Para superar esses inconvenientes vários trabalhos foram realizados que se traduzem em diferentes modelos. Eles são chamados de Redes de

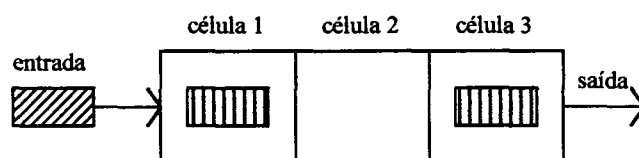
Petri de alto Nível (RdPAN) dentro dos quais se encontram as RdPs Predicado/Transição, RdPs Coloridas e as RdPs a Objetos, com variantes em suas definições.

Nas RdPs Coloridas, a cada ficha é atribuído uma cor diferente (Jensen 1980,1986). Elas formam uma categoria de Redes cuja percepção intuitiva é menos clara do que as RdPs Generalizadas. Elas são de grande valor para a modelagem de certos sistemas complexos.

Na Figura 4.2 mostra por exemplo um sistema FIFO com três células. Um objeto (peça) pode movimentar-se desde a esquerda para o lado direito sem passar pela célula precedente. Na Figura 4.2 (a), um objeto na célula 1 pode mover-se para a célula 2 desde que esta se encontre vazia. Se a célula 2 mudar de condição, então a célula 1 torna-se vazia, e um novo objeto pode entrar na célula 1. Na mesma figura é possível também que o objeto na célula 3 abandone a fila.

O comportamento deste sistema pode ser modelado pela RdP na figura 4.2 (b), onde o disparo de T_1 corresponde a transferência do objeto da célula 1 para a célula 2, similarmente ao disparo de T_2 corresponde ao passo do objeto da célula 2 para a célula 3, enquanto que o disparo de T_3 corresponde à saída do objeto da célula 3.

A figura 4.2 (c) representa a correspondente RdP Colorida, descrevendo o mesmo sistema (dobramento da rede). Aqui o lugar p_{123} vai corresponder ao conjunto de lugares $\{p_1, p_2, p_3\}$ da figura 4.2 (b), e a transição t_{12} vai corresponder ao conjunto de transições $\{t_1, t_2\}$ da Fig. 4.2 (b). Com os arcos da Fig. 4.2 (c) são associados funções. A função "id" significa identidade e é associado a diversos arcos. A função "+1" é associada então com $p'_{123} \rightarrow t_{12}$, e, $t_{12} \rightarrow p_{123}$.



(a)

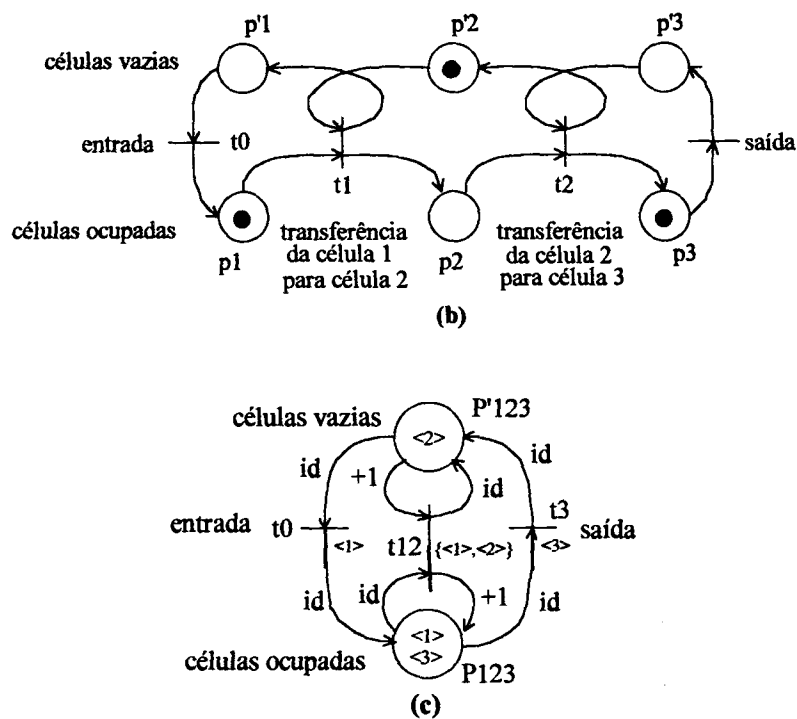


Fig. 4.2.- Rede de Petri: a) Sistema FIFO, b) RdP Ordinária, c) RdP Colorida (David 1994).

c) RdP com Capacidade Finita

É uma RdP onde a cada lugar é associado uma capacidade dada de fichas. O disparo de uma transição de entrada p_i , cuja capacidade $Cap(p_i)$ é somente possível se o disparo desta transição não resulta num número de fichas em p_i que exceda esta capacidade.

Na figura 4.3 aprecia-se que só um objeto pode estar presente numa célula i , então $M(p_i) \leq 1$ em qualquer tempo. Isto é assegurado na Figura 4.2 (b) desde que exista uma marcação invariante $M(p_i) + M(p'_i) = 1$. Assim mesmo o sistema pode ser representado pela RdP de capacidade finita na Figura 4.3. Nesta figura as transições t_1 e t_3 estarão habilitadas enquanto t_0 não estiver habilitada (embora ela apareça como uma transição fonte) porque p_i tem alcançado sua máxima capacidade.

A transformação de uma RdP de capacidade finita em uma RdP Ordinária é bastante simples, basta a cada lugar p_i que tenha uma capacidade finita $Cap(p_i)$ adicionar um lugar complementar p'_i , cuja marcação é também complementar para a capacidade de p_i . Isto quer dizer que $M(p'_i) = Cap(p_i) - M(p_i)$.

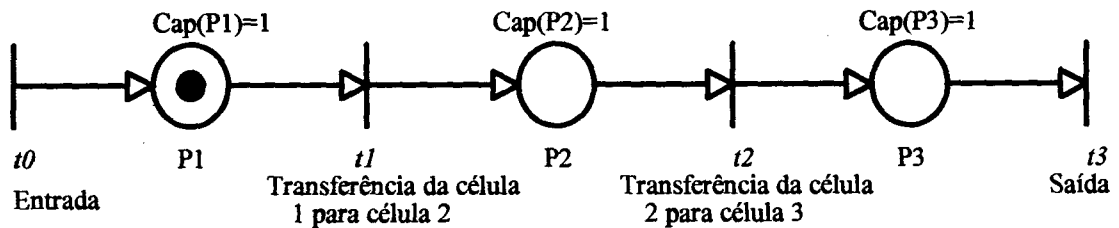


Fig. 4.3.- Rede de Petri com Capacidade Finita

4.3.- Extensões

a) RdP com Arcos Inibidores

Quando duas transições estão em conflito, a priorização é um problema comum numa RdP. Para dar solução ao mesmo, aumentando assim o poder de modelagem das RdPs (Peterson 1981) foram criados os arcos inibidores.

Um arco inibidor é um arco dirigido que une um lugar p_i a uma transição t_j . O extremo final é marcado por um círculo pequeno como mostrado na figura 4.4. O arco inibidor entre p_2 e t_4 significa que a transição t_4 pode disparar se o lugar p_2 não contém nenhuma ficha. O disparo de t_4 consiste em tomar uma ficha de cada lugar de entrada de t_4 , com exceção de p_2 , e depositar uma ficha em cada lugar de saída de t_4 . As expressões teste zero e RdPs estendidas são freqüentemente também usadas na literatura (Murata 1989) para se referir aos arcos inibidores.

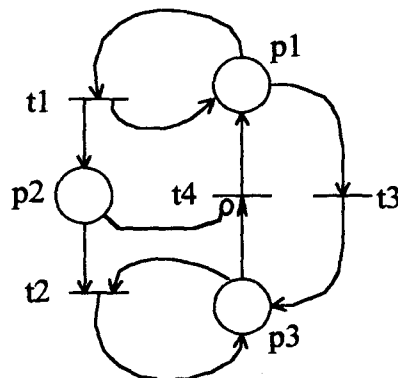


Fig. 4.4.- Rede de Petri com Arco Inibidor

b) Redes de Petri Contínuas

A característica principal em relação às RdPs é que a marcação de uma posição é um número real (positivo) e não mais um inteiro. Sendo o disparo de uma transição realizado como um fluxo contínuo. Estas redes representam sistemas que não podem ser modelados por RdPs Ordinárias, obtendo um modelo muito apropriado também quando o número de marcações de uma RdP Ordinária torna-se muito grande.

A figura 4.5 é um exemplo de RdP contínua, onde os lugares são representados por dois círculos (David 1994) e as transições por retângulos para diferencia-las dos lugares e das transições nas RdPs Ordinárias.

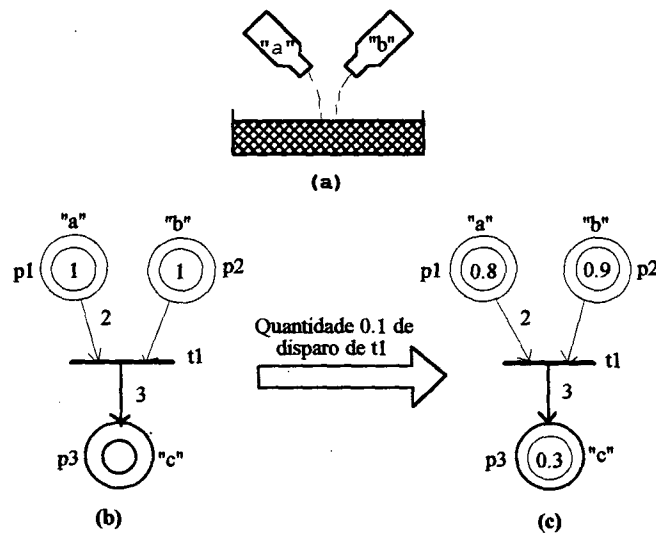


Fig. 4.5.- RdP Contínua

Consideremos que uma determinada bebida por ex. seja obtida pela mistura de outras duas na razão de 2 a 1. A Figura 4.5 (b) representa o estado inicial aonde existe 1 litro da bebida "A" e 1 litro da bebida "B", e nada da bebida final "C". O disparo da transição T_1 corresponde a mistura na razão apropriada. A transição T_1 é habilitada se $M_1 > 0$ e $M_2 > 0$. Numa RdP pode-se ter um disparo, e logo outro disparo. Numa RdP Contínua se terá uma quantidade de disparos a qual não é mais um inteiro (David 1990). Por exemplo se a quantidade de disparo é $X = 0.1$, a marcação na figura 4.5 (c) é obtida a partir da marcação na Figura 4.5 (b). A quantidade $2X = 0.2$ marcas são tomadas do lugar p1 (desde

que o peso do arco $p_1 \rightarrow t_1$ seja 2), a quantidade $X = 0.1$ é tomada do lugar p_2 , e a quantidade de $3X = 0.3$ são então depositadas em p_3 (desde que o peso do arco $t_1 \rightarrow p_3$ seja 3). A quantidade de disparo pode ser qualquer número real X tal que $X \leq M_1 / 2$ (por causa que o peso do arco $p_1 \rightarrow t_1$ é 2), e $X \leq M_2$. Esta quantidade pode ser infinitamente pequena.

Pode-se observar que existem duas marcações invariantes para a RdP contínua na Figura 4.5 (b), isto é, $M_1 / 2 + M_3 / 3 = 0.5$, e $M_2 + M_3 / 3 = 1$, já que na marcação inicial existem uma infinidade de marcações alcançáveis. Observe-se também que a marcação $M = (0, 0.5, 1.5)$ é um deadlock (já que não existe nada na bebida "A").

c) Rede de Petri Híbrida

Este é um novo modelo apresentado pela primeira vez por Le bail em 1991 (David 1994). Esta rede é formada tanto por lugares e transições discretas quanto lugares e transições contínuas.

O seguinte gráfico mostrado na figura 4.6 é tomado de David (1994), onde uma máquina realiza o revestimento de um fio metálico com plástico. O comportamento desta máquina pode então ser modelado por uma RdP contínua que compreende aos lugares p_3, p_4 e p_5 , e a transição t_3 na Figura 4.6 (b). Ora se a máquina A quebrar, a produção é interrompidas isto é, a transição t_3 não pode ser mais disparada. Sendo então este sistema modelado por uma RdP híbrida na Figura 4.6 (b), na qual os lugares p_1 e p_2 e as transições t_1 e t_2 são discretas. Na Figura 4.6 (b), a transição t_3 é habilitada se $M_3 > 0, M_4 > 0$, e exista uma ficha em p_1 . Considerando uma quantidade de disparo X desta transição: as marcações p_3, p_4 e p_5 são modificadas segundo os correspondentes pesos, e a marcação de p_1 permanece a mesma, desde que existam os arcos $p_1 \rightarrow t_3$ e $t_3 \rightarrow p_1$ com o mesmo peso. Se a máquina A quebrar, a transição t_2 é disparada (observe que isto significa a prioridade de t_2 sobre t_3). Existe agora uma ficha em p_2 , mas nenhuma ficha em p_1 . Então a transição t_3 não é mais habilitada.

As abreviações e extensões mostradas até aqui são RdPs as quais só permitem uma abordagem qualitativa. As seguintes extensões que daremos a conhecer a continuação, permitem descrever, não só o que acontece no sistema modelado, mas também quando os eventos acontecem. Estas RdPs permitem

portanto que sistemas sejam modelados quando os disparos das transições são sincronizados por eventos externos, e/ou cujas evoluções são dependentes do tempo. Este tipo de extensões são também conhecidas na literatura como RdPs não autónomas.

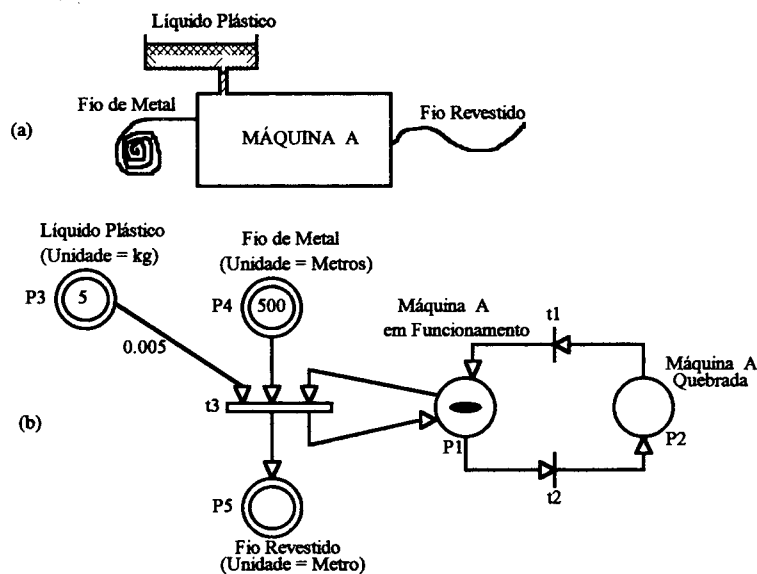


Fig. 4.6.- Rede de Petri Híbrida

Entre as principais extensões que incluem o tempo para a execução das atividades podemos citar os trabalhos de Ranchandani, relativo às Redes de Petri Temporizadas, e o de Merlin, relativo às Redes de Petri temporais.

d) Redes de Petri Sincronizadas

Numa RdP autónoma, sabe-se que uma transição pode ser disparada se ela é habilitada, mas não sabemos quando ela será disparada. Numa RdP Sincronizada, um evento é associado a cada transição, e o disparo desta transição acontecerá se a transição estiver habilitada e quando o evento associado ocorrer.

Na figura 4.7, por exemplo, representa os estados de um motor. Esta é uma RdP Sincronizada porque os disparos das transições são sincronizados sobre eventos externos (o evento externo corresponde a uma mudança no estado do mundo externo) (Moalla 1978).

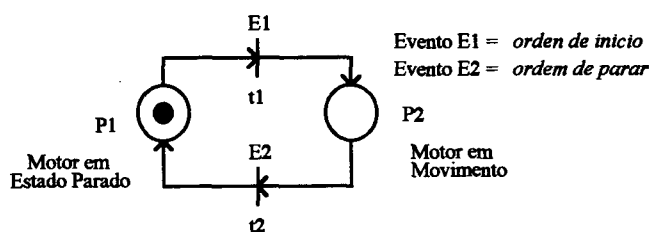


Fig. 4.7.- Rede de Petri Sincronizada

e) Rede de Petri T-Temporizada

Apresentada por Ranchandani em sua tese de Doutorado em 1973 no MIT, associa a cada transição da rede um único parâmetro temporal (sua duração de disparo).

Um tempo d_j , possivelmente de valor zero, é associado com cada transição t_j (Ranchandani 1973). Considerando o exemplo da figura 4.8, é modelada por uma RdP T-Temporizada, donde uma ficha pode ter dois estados: ou ela pode ser reservada para o disparo de uma transição t_j ou ela pode ser não-reservada. Somente fichas não reservadas são consideradas para habilitar condições. Na Figura 4.8 (b), a transição t_1 está habilitada porque todas as fichas são não-reservadas no tempo inicial. Após o disparo de t_1 existe então uma ficha não-reservada em p_2 , e uma ficha não-reservada em p_1 . A transição t_2 está habilitada. A ficha em p_2 está então reservada para o disparo de t_3 , e este disparo acontecerá 5 unidades de tempo depois, já que $d = 5$.

Noutras palavras, desde que uma transição torna-se habilitada, seu disparo absorve as fichas correspondentes desde cada uma dos seus lugares de entrada, as quais permanecem na transição durante o tempo da execução do disparo. Quando a duração do disparo termina, então as fichas são depositadas em cada lugar de saída da transição.

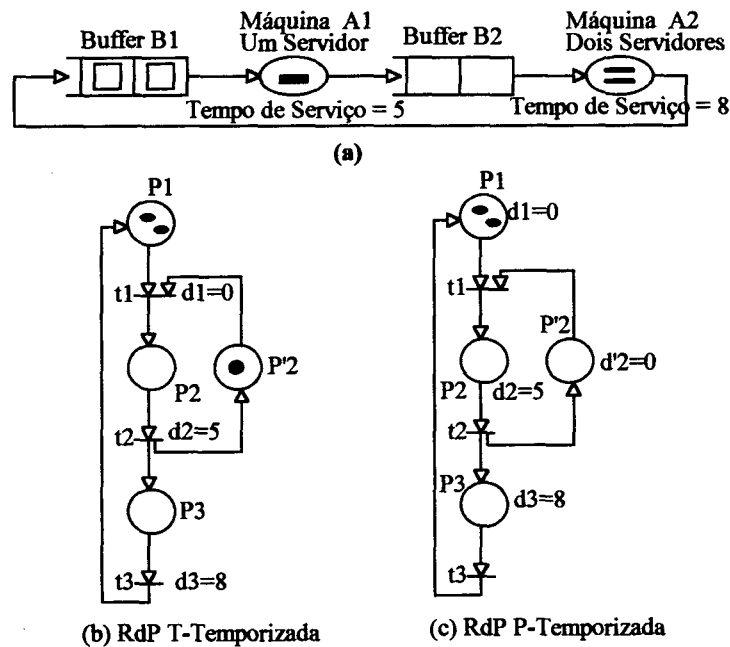


Fig. 4.8.- Rede de Petri Temporizada

f) Rede de Petri P-Temporizada

Contrário ao modelo de Ranchandani, associa a cada lugar um tempo d_i possivelmente de valor zero (Sifakis 1977).

Quando uma ficha é depositada no lugar p_i , a mesma deverá permanecer no mínimo um tempo d_i neste lugar (esta ficha é dita ser indisponível por este tempo). Quando o tempo d_i decorreu, as fichas então tornam-se disponíveis. Somente fichas disponíveis são consideradas para habilitar condições.

Na figura 4.8 (a) por exemplo existem 2 pallet (cada um contendo uma parte), as quais por sua vez passam pelas máquinas M_1 e M_2 . A máquina M_1 pode somente operar uma parte de cada vez, e seu tempo de serviço é de 5 unidades de tempo. A máquina M_2 pode operar duas partes de cada vez, e seu tempo de serviço para uma parte é de 8 unidades de tempo. O estado inicial é tal que ambos pallet estão no buffer B_1 .

A correspondente RdP P-Temporizada é mostrada na Figura 4.8 (c). Já que existem só duas partes e a máquina M_2 é capaz de operar ambas ao mesmo tempo, nunca vai existir partes esperando no buffer B_2 (portanto não existe uma necessidade de associar um lugar ao buffer B_2). Os lugares p_1, p_2

e p_3 correspondem as partes no buffer B_1 , sobre a máquina M_1 e sobre a máquina M_2 respectivamente. Um lugar p'_2 complementar a p_2 assegura que existe só uma parte sobre a máquina M_1 . Os tempos $d_2 = 5$ e $d_3 = 8$ são associados com os lugares p_2 e p_3 . As outras posições têm tempo zero.

g) Rede de Petri Estocástica

Em RdPs Temporizadas, uma duração fixa (geralmente constante, podendo ser também variável), é associada com cada lugar ou transição da rede, é o caso por exemplo dos sistemas de produção, onde o tempo de trabalho de uma máquina para realizar uma determinada operação é constante. Porém existem casos, onde ela não pode ser modelada com durações constantes; sendo este caso por ex.: o tempo de funcionamento real entre 2 Breakdowns de uma máquina. Esta duração pode ser modelada por uma variável random. Redes de Petri Estocásticas podem ser usadas neste caso (Hatono 1991; Molloy 1982,1985). Aqui um tempo random é associado com o disparo de cada transição, onde o tempo é geralmente distribuído segundo uma lei exponencial. Sendo portanto a marcação $M(t)$ da RdP Estocástica um processo markoviano homogêneo e desse modo uma cadeia de markov homogênea poderá ser associada com cada RdP estocástica.

h) Rede de Petri Temporal

Esta RdP foi criada por Merlin (1974) a qual consiste na atribuição de um intervalo de tempo $[T_{\min}, T_{\max}]$ de disparo para cada $t_i \in T$. Neste caso:

T_{\min} = Tempo mínimo de espera para t_i poder disparar após habilitar-se.

T_{\max} = Tempo máximo em que t_i pode disparar após habilitado.

Tem-se então que se $t_i \in T$ é habilitado no instante θ , ela só pode disparar entre $[T_{\min} + \theta, T_{\max} + \theta]$.

Noutras palavras, uma transição deve permanecer sensibilizada durante a espera mínima T_{\min} antes de poder ser disparada, e não pode disparar além da espera máxima T_{\max} . O disparo de uma transição tem duração nula, hipótese essencial ao funcionamento deste modelo de RdP.

Assim, uma RdP Temporal é uma RdP onde $[T_{\min}, T_{\max}] = [0, \infty)$. Na figura 4.9 por ex. t_2 só pode disparar entre 2 e 3 unidades de tempo após habilitada, isto é, após a chegada da ficha em p_2 .

O estado de uma RdP Temporal, é dad pelo par ordenado $E = (M, \delta)$. Neste caso, o disparo de $t \in T$ no instante θ e habilitad por M , leva a um novo estado $E' = (M', \delta')$ sendo, portanto, a evolução da Rede dada da seguinte maneira:

$$i) M'(p) = M(p) + \text{Pós}(t, p) - \text{Pré}(p, t), \forall p \in P;$$

$$ii) \delta'(t_i) = \emptyset \text{ se } t_i \text{ não está habilitada por } M';$$

$$iii) \delta'(t_i) = [T_{\max} \{0, T_{\min} - \theta\}; T_{\max} - \theta] \text{ se } t_i \text{ estava habilitada por } M, \text{ e continua habilitada por } M' \text{ e } t_i \text{ não estiver em conflito com } t_j;$$

$$iv) \delta'(t_j) = [T_{\min}, T_{\max}], \text{ nos demais casos.}$$

O cálculo do estado seguinte está dada pela seguinte regra:

Uma transição t_i está disponível num tempo θ após um estado $E = (M, \delta)$ se e somente se as duas condições seguintes são satisfeitas:

a) A transição é sensibilizada pela marcação M , e;

b) O intervalo de tempo θ está compreendido entre o tempo de disparo mais cedo de t_i e a menor dos tempos de disparo mais tarde das outras transições sensibilizadas.

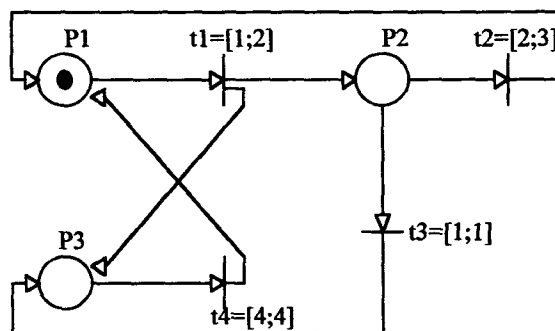


Fig. 4.9.- Rede de Petri Temporal

Na figura anterior temos por exemplo a seguinte evolução dos estados da rede:

$$E_0 = (M_0, \delta_0), \text{ com: } M_0 = [1, 0, 0]$$

$$\delta_0 = \{ [1;2], [2;3], [1;1], [4;4] \}$$

Após o disparo de t_1 com $\theta = 2$, temos:

$$E_1 = (M_1, \delta_1) \text{ com: } M_1 = [0, 1, 1]$$

$$\delta_1 = \{ [\emptyset], [2;3], [1;1], [4;4] \}$$

habilitando assim, as transições t_2 , t_3 e t_4 . No caso de t_3 (com $\theta = 1$) ser disparada, teremos o seguinte estado:

$$E_2 = (M_2, \delta_2) \text{ com: } M_2 = [0, 0, 2]$$

$$\delta_2 = \{ [0], [0], [0], [3;3] \}$$

e como neste estado t_4 resulta sendo a única transição habilitada, ela somente pode disparar no tempo $\theta = 3$, tendo por tanto o seguinte estado:

$$E_3 = (M_3, \delta_3) \text{ com: } M_3 = [1, 0, 1]$$

$$\delta_3 = \{ [1;2], [\emptyset], [\emptyset], [0;0] \}$$

e assim sucessivamente.

Neste caso da RdP Temporal, a sua análise é bastante difícil devido a grande quantidade de estados.

4.4.- Relação entre as Redes de Petri Temporais

Toda Rede de Petri pode ser transformada a uma RdP temporal de Merlin, sendo portanto a abrangência deste tipo de Rede muito maior que as outras classes de Redes. Assim temos que toda RdP temporizada é equivalente a uma RdP temporal do mesmo funcionamento.

4.4.1.- Relação com o Modelo de Ranchandani

Toda transição temporizada de duração "d" no modelo de Ranchandani pode sofrer uma transformação para o modelo de Merlin como mostra a figura 4.10.

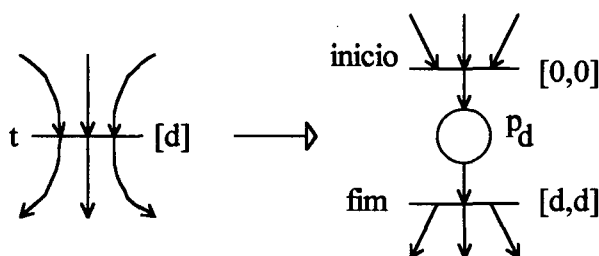


Fig. 4.10.- Transformação de uma Transição Temporizada

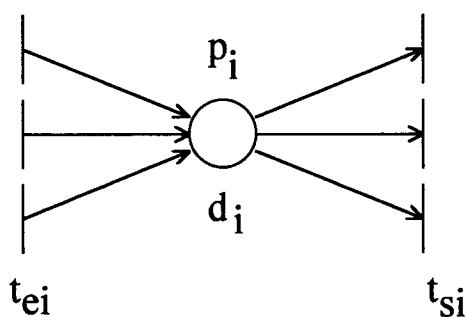
O disparo de uma transição sensibilizada no modelo de Ranchandani num instante " θ " é acompanhada da retenção de "N" fichas disponíveis em cada lugar de entrada de "t" durante o intervalo de tempo $[\theta, \theta+d]$, logo após este intervalo de tempo, fichas são colocadas em cada lugar de saída de "t".

O lugar P_d acrescentada ao modelo de Merlin simula a retenção de fichas operadas pela transição durante o intervalo $[\theta, \theta+d]$, sendo portanto o disparo das transições nesta transformação instantâneas.

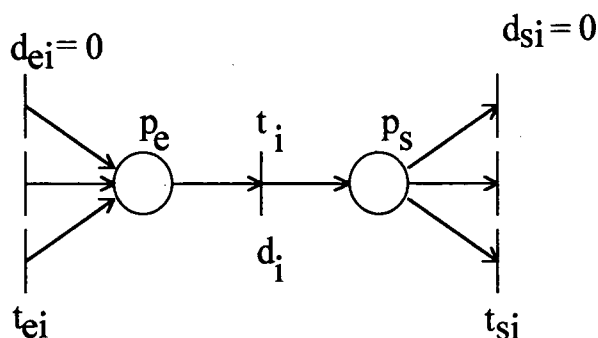
4.4.2.- Relação com o Modelo de Sifakis

Neste modelo as durações são atribuídas aos lugares. Aqui as fichas que chegam a um lugar não são disponíveis durante a duração associada a esta. Transcorrido este intervalo de tempo são elas

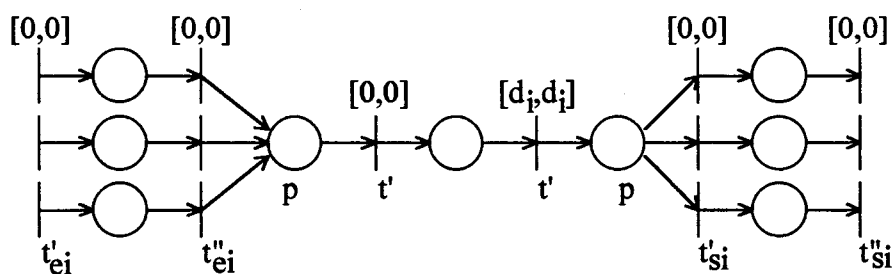
reutilizáveis para efetuar novas passagens. A figura 4.11 mostra a passagem de uma Rede P-temporizada para uma Rede T-temporizada e seguidamente utiliza a mesma transformação como no caso anterior. No caso das transições de duração nula na Rede de Ranchandani, tem-se então, um único lugar de entrada e um único lugar de saída. É então possível substituir-se diretamente nestas transições o parâmetro de duração d_k pelo par de parâmetros $[d_i, d_j]$ sem alterar a rede.



a) Rede de Petri Temporizada de Sifakis



b) Rede de Petri Temporizada de Ranchandani



c) Rede de Petri Temporal de Merlin

Fig. 4.11 - Transformação das Redes de Petri

4.5.- Conclusão

Como visto ao longo deste capítulo, o modelo básico da Rede de Petri (RdP autônoma) nos permite modelar qualquer tipo de sistema a eventos discretos, mostrando apenas como o sistema se comporta e permitindo a validação qualitativa do funcionamento do processo.

Redes de Petri não autônomas por outro lado têm um maior campo de aplicação. RdPs sincronizadas por exemplo permitem a evolução de um sistema sujeito à restrições externas para serem modeladas. Sua maior aplicação é a descrição de controladores e sistemas em tempo real.

Redes de Petri Temporizadas e estocásticas, que tomam o tempo em consideração, permitem uma análise qualitativa do funcionamento de um processo sendo utilizadas, portanto, geralmente para avaliar o desempenho (performance) de sistemas de processamento de dados, sistemas de produção, etc.

Redes de Petri Híbridas são usadas quando uma parte pode ser modelada por uma RdP Contínua, enquanto a outra parte requer uma modelagem discreta.

Redes de Petri Coloridas por outro lado facilitam a modelagem de sistemas muito grandes que possuem processos similares reduzindo assim o tamanho da rede.

Enfim, dado que existem diversidade de sistemas no mundo real, cada um destes modelos apresentados são portanto utilizados dependendo do caso a ser tratado, tendo assim, cada modelo seu próprio campo privilegiado de aplicação.

CAPÍTULO 5.- AS REDES DE PETRI E SUA RELAÇÃO COM OS SISTEMAS DE PRODUÇÃO

5.1 - Introdução

Como visto nos capítulos anteriores, as Redes de Petri são uma ferramenta matemática e gráfica de representação formal de sistemas de eventos discretos (um sistema de manufatura é um sistema discreto), sendo esta a razão de sua importância na automação industrial.

A sua facilidade para modelar situações que dificilmente são representadas por outros modelos como concorrência, paralelismo, sincronização entre vários processos, competição por um recurso, etc fazem das Redes de Petri uma ferramenta muito importante. Por exemplo, os modelos de filas são usados com muita frequência para avaliar o desempenho (performance), mas sincronizações são difíceis (impossíveis) de se representar. Neste e em outros casos, as Redes de Petri são mais apropriadas para representar formalmente paralelismo e sincronizações em ambientes de Engenharia.

As Redes de Petri, como uma ferramenta de uma ordem parcial de eventos, permite a análise e a avaliação de desempenho em vários níveis de controle de Sistemas Flexíveis de Manufatura (Rammamorthy 1980), sendo portanto adequadas aos Sistemas de Produção. Talvez um dos primeiros a utilizar as Redes de Petri para sistemas de produção foi o trabalho de Hack (1972) apresentado no MIT com o título de "Analysis of Production Schemata by Petri Net" onde trata da modelagem de um sistema de montagem, utilizando uma subclasse (de Redes de Petri) chamada "Rede de Petri livre de escolha". A partir de então as RdPs foram ganhando popularidade como uma ferramenta para modelar e analisar Sistemas Flexíveis de Manufatura. Logo, com os trabalhos de Merlin (1974), Ranchandani (1974) e Sifakis (1977), onde é incorporado a dimensão tempo às Redes de Petri, as mesmas tornaram-se mais importantes para representar e analisar problemas ligados a atividades de produção, onde o tempo é um fator importante.

A estrutura de uma RdP, vista como uma ordem parcial de transições, descreve a coordenação entre eventos concorrentes que não têm uma relação de precedência rigorosa. A alocação de recursos é um exemplo. Um outro exemplo é fazer uma escolha entre ações alternativas para solucionar conflitos entre dois eventos competindo pelo mesmo recurso.

Em Sistemas Flexíveis de Manufatura geralmente aparecem situações tais como:

- a) Processos tendo relações de precedência entre si;
- b) Processos sendo executados ao mesmo tempo (paralelismo);
- c) Processos que não podem ser realizados até que outro processo alcance um certo estado (sincronização);
- d) Processos a ser seguidos não são determinados, o que mostra uma escolha entre alternativas (não determinismo);
- e) Processos possuindo conflitos entre competição e coordenação de atividades (alocação de recursos).

Assim sendo, o uso das Redes de Petri para a modelagem, análise e controle de eventos discretos de FMS estão recebendo maior atenção nestes últimos anos por diversas razões (Favrel 1985; Rillo 1987; Al-jaar 1989; Zhou 1992 (a)):

- 1.- RdPs possuem uma forma gráfica fácil de entender na qual é possível visualizar sistemas complexos;
- 2.- RdPs podem modelar concorrência, eventos assíncronos, relações de precedência lógica e interações estruturais numa maneira simples e natural. Conflitos, bloqueios, buffers finitos, sincronizações, prioridades, e operações de montagem e desmontagem podem ser modelados fácil e eficientemente;
- 3.- RdPs representam uma ferramenta de modelagem hierárquica, com uma base matemática e prática bem desenvolvida. Análise estrutural e temporal podem ser executados usando RdPs estocásticas;

- 4.- RdPs apresentam propriedades de decomposição permitindo assim uma representação modular (Righini 1993);
- 5.- Mudanças num modelo de RdPs são feitas simplesmente pela adição de fichas, posições ou transições. Por outro lado, mudanças num modelo de cadeias de Markov usualmente requerem uma redefinição de todos os estados no modelo (Al-jaar 1990).
- 6.- RdP é um modelo abstrato, o que permite a representação de diferentes tipos de sistemas;
- 7.- RdP é um modelo que permite diversos níveis de abstração na representação de um mesmo sistema,
- 8.- RdPs permitem a descrição dos aspectos estáticos e dinâmicos do sistema a ser representado;
- 9.- Contêm o conceito de estado parcial (Rillo 1988);
- 10.- RdPs podem ser usados também para implementar sistemas em tempo real (Huang 1992) (Valette 1986);
- 11.- É um modelo de fácil aprendizado, podendo funcionar como linguagem de comunicação entre especialistas de diversas áreas;
- 12.- Possuem métodos de análise, existindo atualmente ferramentas comerciais para sua análise dos tipos mais simples de redes.

Se comparamos estas qualidades inerentes às RdPs com àquelas mostradas no item 2.12, poderemos ver claramente que as RdPs se adaptam muito bem para especificar e implementar sistemas de controle de SDED; justificando assim sua utilização em Sistemas de Manufatura.

Na literatura atual, as RdPs voltadas para Sistemas de Produção, mais especificamente em FMSs, podem ser classificados nas seguintes categorias (Al-jaar 1990):

- 1.- Modelagem e análise;
- 2.- Avaliação do desempenho de Sistemas de Manufatura;

3.- Implementação do controle em tempo real;

4.- Simulação e ferramentas de software;

5.1.1.- Modelagem e Análise

Um FMS pode ser visto como dois sistemas (Favrel 1985): um sistema físico e um sistema lógico. Do ponto de vista físico, um FMS é um conjunto de máquinas, ferramentas, matérias primas e produtos. Do ponto de vista lógico, um FMS é um conjunto de processos, assim como o estado das máquinas e materiais num job-shop. O processo de manufatura significa portanto a evolução do sistema, o qual pode ser estudado pela árvore de alcançabilidade. Ambos os aspectos, físico e lógico de um FMS, podem ser descritos portanto por uma só RdP.

Entretanto, a complexidade do modelo aumenta quando ela é muito grande (especialmente quando se deseja um modelo mais detalhado), devido à explosão combinatória de estados. Neste caso, uma abordagem modular é a maneira mais adequada para analisar uma RdP. Em RdPs, uma subrede inteira pode ser substituída por uma simples posição ou transição para a modelagem num nível de abstração maior. Por outro lado, também podemos substituir um lugar ou uma transição por uma subrede para prover uma modelagem mais detalhada, o que é feita através de blocos bem formados.

Para solucionar, portanto, o problema da explosão combinatória de estados ao efetuar a análise de uma RdP, existem algumas abordagens: uma delas é por simulação (Righini 1993), e outra por redução (Lee 1985 e 1987).

A Redução de uma RdP é um procedimento que transforma homomórficamente uma RdP para sua rede reduzida, enquanto preserva algumas propriedades desejáveis da rede original. Como consequência, a análise da rede simplificada pode prover informação suficiente para entender a rede original. Do ponto de vista lógico, este procedimento reduz os lugares de estados alcançáveis, e do ponto de vista físico, reduz a estrutura (tamanho) da rede permitindo-nos assim analisá-la passo a passo.

5.1.2.- Avaliação da Performance de Sistemas de Manufatura

A adição do tempo nas RdPs, permitiram o cálculo da avaliação da eficiência temporal dos sistemas. RdPs temporizadas e RdPs estocásticas, são usadas neste contexto. Estas últimas têm recebido muita atenção no campo de ciências da computação (Molloy 1982; Rammamorthy 1980), e ultimamente na avaliação da eficiência (desempenho) em FMSs (Hillion 1989; Al-Jaar 1990).

Modelos de RdPs estocásticas por exemplo, mostram-se superiores no cálculo da performance de sistemas sobre os modelos tais como Redes de Filas, Cadeias de Markov, Simulação, e, análise de perturbação (Viswanadham 1987; Al-Jaar 1990). RdPs são por exemplo, mais apropriadas do que Redes de Filas quando se modela um sistema com buffer finito, sincronização, prioridades e operações de montagem e desmontagem. Entretanto, quando a capacidade dos buffers são relativamente grandes, então Redes de filas e não RdPs devem ser usadas.

Utilizando o poder de modelagem das RdPs estocásticas junto com as Redes de filas obtém-se uma ferramenta híbrida de análise mais precisa e eficiente para a avaliação do desempenho de sistemas complexos (Balbo 1988).

Várias comparações têm sido feitas (Balbo 1986; Vermon 1987) para o cálculo da performance de FMSs utilizando RdPs estocásticas e Redes de filas, mostrando-se sempre que os resultados fornecidos são mais precisos quando se utilizam RdPs estocásticas e não Redes de filas de espera (Valette 1990; Al-Jaar 1990).

5.1.3.- Implementação de Controle em Tempo Real

Paralelamente à modelagem e análise, assim como à avaliação do desempenho de um FMS, a implementação de controladores baseados em RdP em tempo real têm recebido considerável atenção. Um aspecto importante das RdPs em Manufatura é seu uso nas várias etapas do projeto (desde as especificações funcionais até a implementação final) de um FMS para a implementação de um Sistema de Controle (Rillo 1988; Al-Jaar 1989). Esta implementação direta dos programas de controle a partir das especificações permite uma acentuada queda no número de erros introduzidos quando não existe

esta passagem direta entre estas duas fases. Uma aplicação de Redes de Petri e sistemas baseados em regras como metodologia para a especificação e implementação de sistemas de controle para sistemas de manufatura pode ser vista melhor em Rillo (1988), em Valette (1986) por exemplo podemos apreciar uma visão sobre o uso de RdP no desenho e implementação do controle de FMS, o mesmo que é discutido usando dois níveis de decomposição, controle local e o nível de coordenação e monitoramento. Por outro lado, Dos Santos F^o (1990) apresenta a técnica de "Mark Flow Graph" estendido, que é um derivado da RdP para a modelagem e controle de sistemas de eventos discretos.

Existem diversos trabalhos que tratam da implementação direta de uma especificação baseada em RdP. Dentre as principais formas de implementação podemos citar o jogador de fichas e as máquinas de inferência. O jogador de fichas é um programa que opera sobre o estado atual da rede, habilitando transições sensibilizadas e disparando as mesmas segundo o estado da rede e os eventos associados a ela. A máquina de inferência opera as regras de produção de um sistema baseado em regras, onde a cada ciclo de operação a partir de fatos conhecidos como verdadeiros, a máquina de inferência calcula o conjunto de regras aplicáveis para as transições e em seguida aplica uma delas.

5.1.4.- Simulação e Ferramentas de software

A medida que aumenta o número de estados de um sistema, o tamanho da rede tende a se tornar muito grande; portanto procedimentos automatizados são necessários para ajudar no projeto, análise e implementação de modelos de RdPs. É assim que muitos softwares foram desenvolvidos para ajudar nas diversas fases, análise estrutural, avaliação da performance, etc. Entre eles podemos citar o analisador/simulador de Redes de Petri desenvolvido no Laboratório de Controle e Micro Informática (LCMI) da UFSC, que é um programa para o auxílio ao projeto com Redes de Petri. Alguns outros são dedicados especificamente à área de Sistemas de Manufatura e podem ser vistos em Al-Jaar (1989), Valette (1990) e David (1994).

Linguagens em tempo real são um tipo de implementação muito usual, onde a especificação é trasladada para uma linguagem adotada. Estas linguagens utilizadas para o controle de processos devem possuir certas características que podem ser vistas melhor em Kunzle (1990), as mesmas que são

extensões das linguagens de programação científicas existentes com funções de programação em tempo real entre elas podemos citar o ADA e a linguagem C, Pascal, etc.

5.2.- Aplicação das Redes de Petri a Sistemas Flexíveis de Manufatura

Para a especificação de um sistema em ambientes de manufatura utiliza-se uma lista de recursos, operações a serem executadas, assim como as relações de precedência, descrevendo desse modo o comportamento do mesmo.

Podem ser considerados como recursos: máquinas, buffers, partes, robôs, etc. as quais por sua vez podem ser divididos em duas classes:

- a) Aqueles que são fixos como máquinas, robôs, esteiras, etc, e;
- b) aqueles que têm seu número variado como pallet, acessórios, partes a serem processadas, etc.

É sabido portanto que para a execução de operações de produção são utilizados ambos os tipos de recursos.

Como visto anteriormente, são usadas lugares para modelar operações e recursos numa RdP. Uma ficha (marcação) num lugar que representa um recurso (lugar-recurso) vai significar portanto a disponibilidade do recurso para a execução de uma operação, enquanto que uma ficha num lugar que representa uma operação (lugar-operação) vai significar que a operação naquele lugar está sendo executada (atividade em processo).

Uma operação numa RdP pode ser dividida em diversas sub-operações (refinamento sucessivo), portanto um lugar-operação pode ser substituído por uma subrede assegurando-se que o mesmo seja um lugar seguro (safe) no modelo de RdP, porque uma operação nunca pode ser executada mais do que uma vez ao mesmo tempo na mesma máquina.

A habilitação de uma transição numa RdP vai resultar portanto da satisfação de certas pré-condições (conclusão de uma operação com sucesso e/ou a disponibilidade de um recurso). Satisfeitas estas pré-condições, o disparo de uma transição significará assim o início ou fim de uma operação.

Para evitar que o sistema não entre num estado de deadlock, ou ele seja potencialmente morto (que não evolua) faz-se necessário portanto determinar o número necessário de fichas na marcação inicial dos lugares-recurso-variado.

Neste ponto, três propriedades das RdP são então importantes para implementar e operar um sistema físico:

a) Vivacidade: que vai implicar a ausência de algum possível bloqueio (deadlock). Noutras palavras esta propriedade garante que o sistema opere com sucesso;

b) Limitação (boundedness) ou seguro (safeness): que vai implicar ausência de excesso de fichas. Safeness de um lugar-recurso indica a disponibilidade de só um único recurso, e é frequentemente usado para garantir a safeness de alguns lugares-operação afins;

c) Reversibilidade: implica a execução do comportamento cíclico do sistema que realiza suas funções repetidamente, implicando também a recuperação de erros no contexto de manufatura.

Portanto para modelar um sistema de manufatura usando RdPs e garantir seu funcionamento correto é preciso que sua estrutura e a marcação inicial tornem o sistema limitado, vivo e reinicializável. Estas propriedades nas RdPs têm sido muito exploradas em sistemas de manufatura.

Um estudo preliminar do uso das RdPs temporizados no estudo do controle em tempo real e avaliação do desempenho de FMSs foi efetuado por Dubois (1983). Kamath (1986) por outra parte deu uma breve revisão do uso de RdP em FMS, enfatizando a parte da modelagem e análise. Em Favrel (1985) RdPs e suas variantes são apresentadas como ferramentas para a modelagem, análise, programação e controle de FMS; alguns novos desenvolvimentos são também apresentados tais como: a) elaboração hierárquica de RdPs para a modelagem de FMS; b) redução hierárquica e; c) um método de decomposição para reduzir a complexidade da análise dos FMS.

Redes de Petri Estocásticas também são usadas com bastante frequência para avaliar a performance de Sistemas Flexíveis de Manufatura (Al-jaar 1990).

5.3.- Ligação entre as Redes de Petri e o Grafo PERT

Redes PERT criadas em 1958 é uma teoria muito útil em situações onde os gerentes têm responsabilidade pelo planejamento, programação e controle de grandes projetos, contendo muitas atividades que são efetuadas por pessoas diferentes. Um grafo PERT é uma representação gráfica do relacionamento entre várias atividades, onde, as mesmas são operações que consomem tempo e recursos.

Os grafos PERT possuem um único nó inicial e um único nó final, o que indica que qualquer caminho que sigamos partindo do evento inicial, sempre nos conduzirá ao evento final. Assim, ao caminho de maior duração denomina-se caminho crítico o qual corresponde à seqüência de atividades em que mais se deve exercer controle, já que qualquer atraso no mesmo automaticamente determinará um atraso no projeto.

Existem duas maneiras de se representar uma rede PERT (Grislain 1979)

- a) Representação de atividades por arcos (A-on-A) ou método americano e;
- b) Representação de atividades por nós (A-on-N) ou método francês

Na figura 5.1, mostra-se um grafo PERT do tipo A-on-A, onde cada atividade é representada por um arco e cada vértice representa um evento lógico.

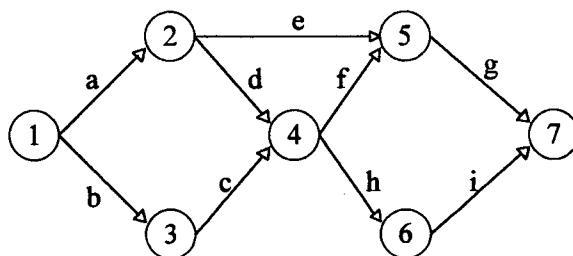


Fig. 5.1- Rede PERT A-on-A

O acontecimento do evento 5, significa que as atividades "e" e "f" foram completadas e a atividade "g" só pode ser iniciada se e somente se "e" e "f" foram completadas.

Em alguns casos é necessário utilizar atividades fantasmas para representar certas relações de restrição entre os nós.

Por outro lado a figura 5.2 representa a mesma Rede PERT da figura 5.1 pelo método A-on-N; onde as atividades são representadas por nós, e arcos representam as relações de precedência entre as atividades. Uma característica importante neste tipo de rede é que aqui não são necessárias as atividades fantasmas ou fictícias; exceto para os nós iniciais e finais.

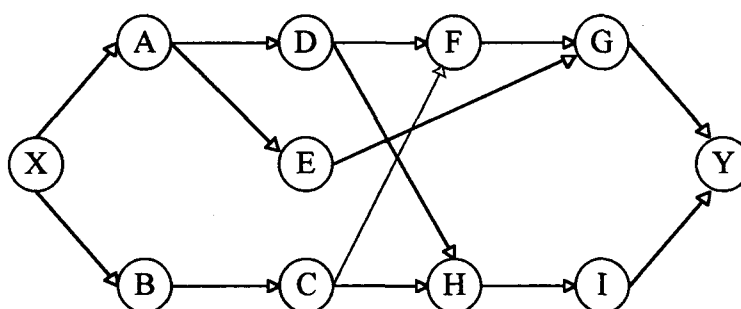


Fig. 5.2- Rede PERT A-on-N

O uso principal das redes PERT é a determinação do tempo de duração de um projeto pelo método do caminho crítico. A diferença fundamental entre o PERT e o CPM reside em que o primeiro estima a duração de cada tarefa ou operação dos projetos, baseando-se simplesmente num nível de custo; enquanto que o segundo relaciona duração com custo, do que deriva-se uma diversidade de durações para cada atividade e elegendo a duração adequada fará com que o custo total do projeto seja mínimo.

Já que um grafo PERT está composto de atividades e eventos, ela pode facilmente ser convertida numa Rede de Petri.

Assim, numa rede PERT do tipo A-on-A a cada atividade pode-se associar uma transição, e a cada evento um lugar (duplo); sendo que cada transição numa Rede de Petri tem que ser precedida por

uma posição que vai representar uma condição diferente para cada transição, como pode ser visto na figura 5.3 a qual representa a RdP equivalente a Rede da figura 5.1.

Por outro lado, para transformar uma rede PERT do tipo A-on-N numa RdP, basta substituir cada arco por uma posição e cada nó por uma transição; obtendo-se assim diretamente a RdP equivalente. A figura 5.3 representa também a Rede PERT da figura 5.2.

As Redes de Petri são portanto um excelente veículo para representar as relações de precedência e concorrência das redes PERT, sendo que a determinação do caminho crítico das operações é de particular importância nas redes PERT/CPM, este caminho apenas restringe o programa total do projeto no sentido de que as operações que fazem parte do caminho crítico não têm folga, portanto, qualquer atraso do tempo em qualquer destas operações colocará em perigo o tempo de execução normal do projeto.

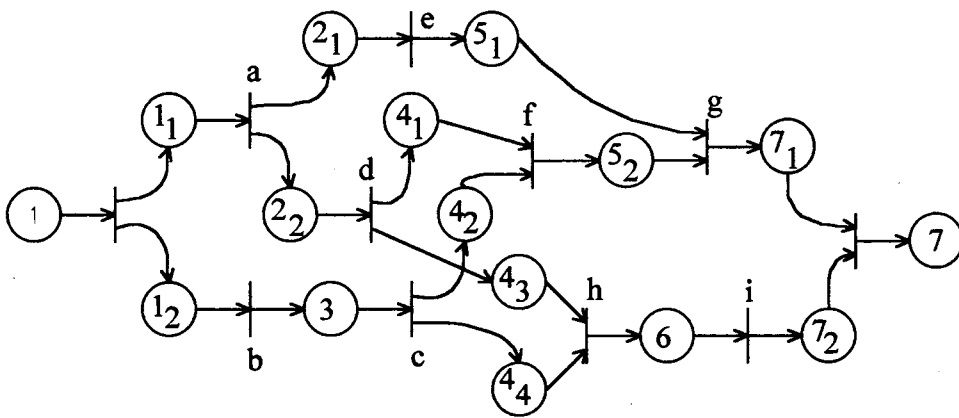


Fig. 5.3- Rede de Petri equivalente a uma rede PERT

No caso das RdP, circuitos poderão ser usados para estudar o comportamento de disparo limitante. Os circuitos com período de tempo máximo são chamados de circuitos críticos. Portanto as operações associadas com estes circuitos restringiram a execução do sistema no sentido em que se a duração de qualquer operação do circuito crítico se incrementa, a duração do tempo de ciclo do sistema também se incrementará.

Assim como no PERT/CPM o caminho crítico pode mudar quando o tempo de certas atividades muda (atividades que correspondem ao caminho crítico), na RdP os circuitos críticos também podem mudar quando as durações das operações são mudadas. Portanto, um incremento na duração de uma operação pode somente incrementar marginalmente o tempo de ciclo do sistema. Neste sentido, a noção de criticidade na RdP é também de vital importância.

Algumas outras vantagens das RdPs sobre os grafos PERT são os seguintes:

- a) O modelo PERT não pode descrever o não determinismo, tal como: escolha e repetição que são propriedades críticas de um FMS;
- b) O PERT pode representar o procedimento (programação) de cada produto no nível de produto, mas não a evolução global do FMS no nível de fábrica, porque o PERT não pode sintetizar os procedimentos de produtos manufaturados no FMS;
- c) O método do caminho crítico nos permite estudar o caminho crítico no processo de cada produto, mas é dificultoso analisar a sensibilidade de cada processo no nível de FMS com PERT/CPM;
- d) As Redes de Petri admitem ciclos repetitivos, enquanto que nas redes PERT/CPM ciclos não são admitidos;
- e) Requerimento de recursos para a execução de atividades aparece explicitamente nas Redes de Petri, na representação através de fichas.

5.4.- Alocação de Recursos em Redes de Petri

Como é sabido, recursos são os meios utilizados para a execução das tarefas podendo ser eles de dois tipos a saber: recursos consumíveis (matéria prima por exemplo) e recursos não consumíveis (maquinaria, robôs, etc).

A principal desvantagem dos grafos PERT e outros modelos gráficos é sua incapacidade para representar unidades de recursos no modelo (no caso das redes PERT é requerido um cronograma

integrado ao grafo PERT-CPM que apresenta a distribuição de recursos). As Redes de Petri, devido a sua alta versatilidade, tornam-se bem apropriadas para representar unidades de recursos físicos.

A especificação de atividades numa RdP requer que os recursos necessários para executar uma dada atividade seja representada. Assim, para descrever recursos no processo de modelagem, utilizam-se elementos distribuidores (Santos F^o 1993) de forma que o número de marcas contidas nestes elementos corresponda ao número de recursos alocados. Os arcos de saída destes elementos são conectados às transições de entrada que indicam o início das respectivas atividades, e os arcos de entrada destes distribuidores de recursos originam-se das transições de saída das atividades correspondentes. Desta forma, o início de uma atividade agrega recursos e o término de uma atividade os libera. Por exemplo, a figura 5.4 mostra uma atividade de produção utilizando um operador. As transições t'_1 e t''_1 correspondem ao início e fim da atividade de manufatura que é representada por p_5 , e t_2 é uma atividade de alocação de recurso para a atividade p_5 . As fichas em p_3 e p_4 representam operadores qualificados para realizar a atividade p_5 . Uma ficha em p_4 permitirá a habilitação de t'_1 quando p_1 estiver marcada, representando assim o disparo de t'_1 , (o operador realizando a atividade p_5); logo após o término da atividade p_5 o operador é liberado através do disparo de t''_1 , estando pronto para ser alocado para outra atividade de manufatura. Assim, fichas em p_3 irão representar portanto unidades de recursos que estão disponíveis.

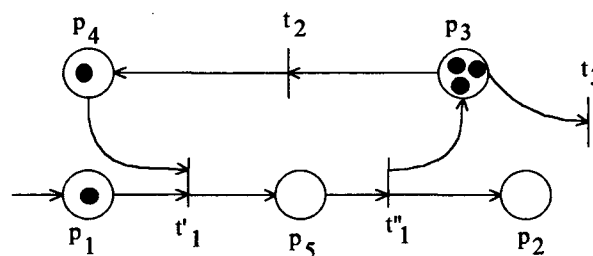


Fig. 5.4- Alocação de recursos em Redes de Petri

Os problemas de compartilhamento de recursos como máquinas e robôs levam a dois tipos de exclusão conhecidos como: exclusão mútua seqüencial e exclusão mútua paralela (Zhou 1991); os quais acontecem com muita frequência em ambientes de manufatura. Estes dois conceitos podem ser vistos através da figura 5.5, onde a figura 5.5 (a) representa uma linha de produção com duas máquinas, um

robô, e um buffer. Aqui pode-se apreciar que o robô (p_5) serve duas máquinas (p_6 e p_8). Fichas em p_1 representam o número de partes no sistema, e fichas em p_7 vão representar o número de buffer vazios. Os lugares p_6 e p_8 representam a disponibilidade das duas máquinas e os lugares p_2 e p_4 irão representar o processo de carga da máquina com uma parte e processamento, assim como a operação de descarga e p_3 que representará um buffer ocupado. Quando $m_0(p_1) > m_0(p_7)$ o sistema pode entrar num estado de deadlock com a seguinte seqüência de disparos efetuado $s = t_1 t_2 t_1 t_2 t_1$. Por outro lado a fig. 5.5b mostra um robô (p_5) servindo a duas estações paralelas (p_1 e p_3) transferindo partes para as operações (p_2 e p_4).

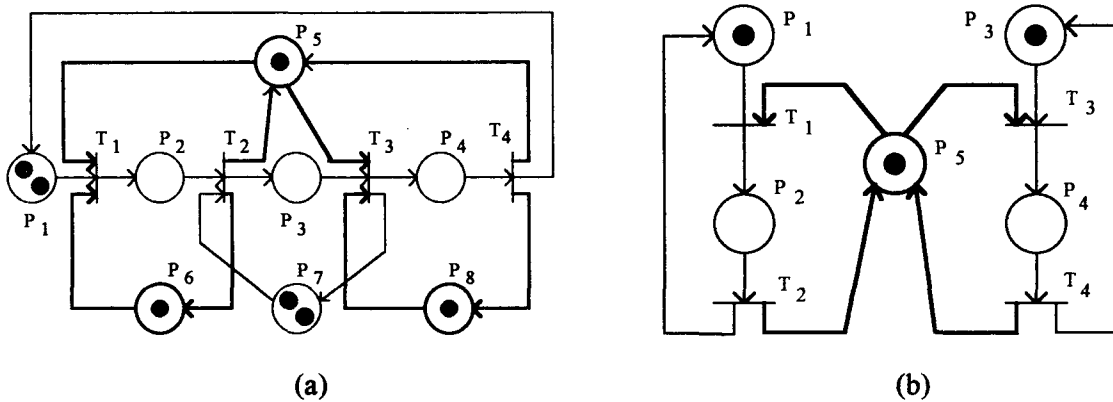


Fig. 5.5- Compartilhamento de recursos a) exclusão mútua seqüencial, b) exclusão mútua paralela.

Em geral, num processo de manufatura só recursos não consumíveis serão compartilhados por diferentes atividades numa maneira concorrente. Os recursos consumíveis, por outro lado, devem estar disponíveis em quantidades suficientes para evitar quebra no fluxo do produto.

No caso de ter recursos não consumíveis limitados, e considerando que o processo de produção é representado por uma RdP T-temporizada, onde um número real positivo é atribuído à duração de cada atividade, a ordem de alocação de recursos influenciará na duração total do sistema. Assim, deve-se verificar a ordem de execução das atividades utilizando o mesmo recurso para se obter um tempo mínimo de operação. Uma forma de se obter essa ordem é visto em Gomes (1990), onde utiliza uma árvore de estados rotulando os nós por um número real positivo que representará o intervalo de tempo

mínimo de percurso entre o nó raiz e o nó final, obtém-se assim diversas sequências que irão permitir saber a ordem de utilização do recurso único com um tempo mínimo de operação.

A figura 5.6 representa uma RdP T-temporizada utilizando um único recurso "R" requerido para realizar três atividades diferentes. Para verificar a ordem de utilização deste recurso através da árvore de estados, assim como do grafo de caminhos e temporizações associados Gomes (1990), verifica-se que para obter a operação em tempo mínimo do sistema é necessário destinar o recurso 'R' primeiro à atividade t_2 e t_5 (nessa ordem) e depois à atividade t_6 , o que corresponde a um tempo de execução de 8 unidades de tempo. Sendo que se optamos por alocar o recurso primeiro à seqüência t_2 , t_6 e t_5 o tempo de execução aumentará para 12 unidades de tempo. Por outro lado, se a seqüência fosse primeiro t_6 e logo t_2 e t_5 se obterá um tempo ainda maior, equivalente a 14 unidades de tempo.

Considerando que existissem duas unidades de recurso "R1", a duração mínima total de operação do sistema, seja qualquer que for a seqüência, será de 8 unidades de tempo. Portanto, não será necessário observar uma sincronização entre as atividades, já que se disporá de recursos suficientes para sua realização.

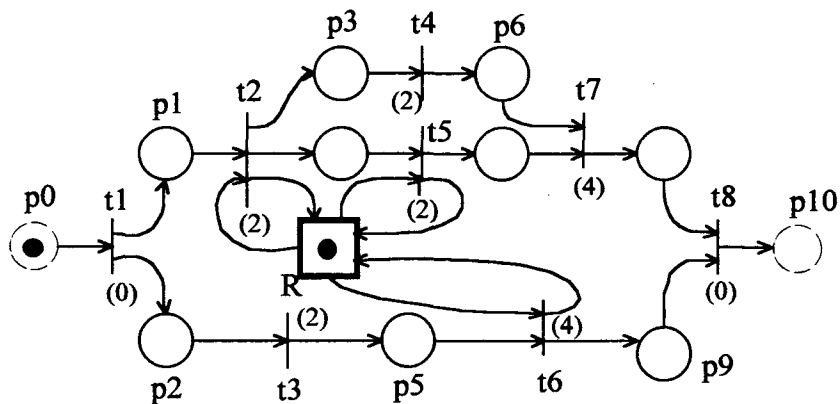


fig. 5.6- Alocação de um único recurso em RdP

Um aspecto importante em sistemas de manufatura é portanto que muitos dos recursos são compartilhados; por essa razão uma alocação inadequada dos mesmos pode levar o sistema a situações de deadlock.

5.5.- Detecção de Situações de Deadlock Utilizando Redes de Petri

Como visto no item anterior, em sistemas de produção mais especificamente em FMSs, existem diversas situações onde vários processos (atividades) competem por um único recurso o qual é compartilhado. Assim, temos por exemplo máquinas numa célula de manufatura flexível (a mesma máquina realizando atividades diferentes) ou sistemas de manuseio de material (robôs) sendo compartilhados por partes num FMS. Em sistemas em tempo real, onde recursos são compartilhados por diversos processos, uma das maiores preocupações é justamente evitar o aparecimento de situações de Deadlock. Assim, os modelos de RdP têm sido muito usados para a detecção destas situações desfavoráveis ao funcionamento normal de tais sistemas (Jensen 1981). Se durante a análise de um modelo de RdP é achado que o sistema é vivo (live), isto vai indicar que o sistema é livre de Deadlock. Por outro lado, se o modelo de RdP não é vivo, quer dizer que uma situação de Deadlock pode potencialmente existir no sistema.

A utilização de invariantes de lugar (L-invariantes) de uma RdP junto com a marcação inicial da rede facilitam a investigação da propriedade de vivacidade, podendo assim detectar possíveis situações de Deadlock (Jensen 1981; Kamath 1986). Para calcular L-invariantes são geralmente usados métodos de álgebra linear (Agervala 1979; Peterson 1981).

A figura 5.7 ilustra a ocorrência de um deadlock numa estação de trabalho composta por duas esteiras, um robô e uma máquina ferramenta. Aqui pode-se ver claramente que enquanto a máquina está processando uma peça, o robô inicia o processo de carga de uma nova peça que chegou. Desta forma, mesmo que a máquina-ferramenta conclua o processo, não há como descarregar a peça pronta concretizando-se deste modo o deadlock.

Um outro caso muito comum é quando os depósitos intermediários (buffer limitados) ficam cheios, um bloqueio (deadlock) é de fato possível, já que as máquinas não podem ser liberadas porque não há lugar nos depósitos de saída para colocar o produto que acabou de ser processado e portanto, nenhuma nova operação pode ser iniciada.

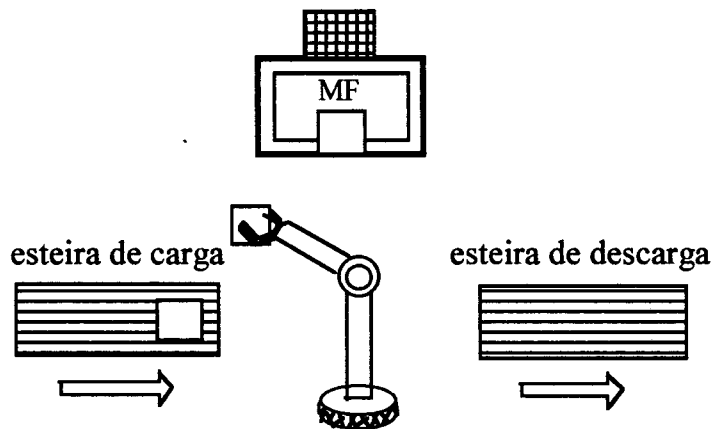


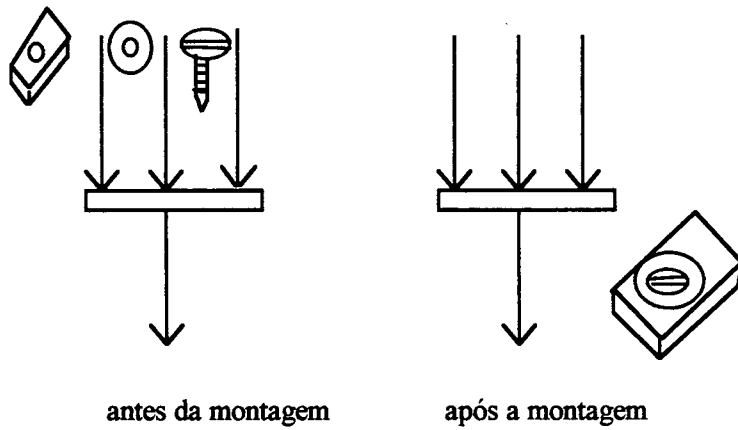
fig. 5.7 - Ocorrência de um Deadlock (Dos Santos 1993)

Assim, para evitar estas situações de deadlock é requerido o uso eficiente de uma política de alocação de recursos. Várias abordagens baseadas em RdPs foram feitas tanto para eliminar, assim como para evitar estas situações de deadlock (Viswanadham 1990; Banaszak 1990). Eliminação de deadlock refere-se a políticas de alocação de recursos estática. Por outro lado, evitar uma situação de deadlock é referido a políticas de alocação de recursos dinâmicos. No primeiro caso é utilizado geralmente o grafo de alcançabilidade das RdP para chegar a uma política de alocação de recursos estáticos, no segundo caso, Viswanadham (1990) propõe um sistema de controle e monitoramento on-line baseado em RdPs. Já por outra parte, Banaszak (1990) propõe um algoritmo chamado DAA (deadlock avoidance algorithm) sendo ambas abordagens muito eficientes na detecção de deadlock.

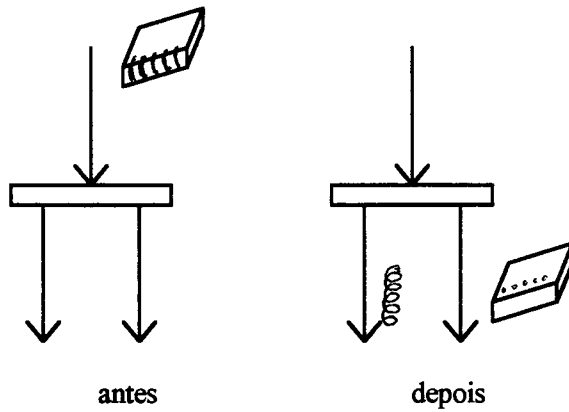
5.6.- Significado das fichas num sistema de Montagem

Numa RdP o disparo de uma transição consiste em retirar (remover) uma ficha de cada lugar de entrada da transição e colocar (após o tempo de execução da atividade), uma ficha em cada lugar de saída da transição. Esta dependência dinâmica representa um fluxo de objetos na execução de um programa de produção, descrevendo portanto operações sobre objetos.

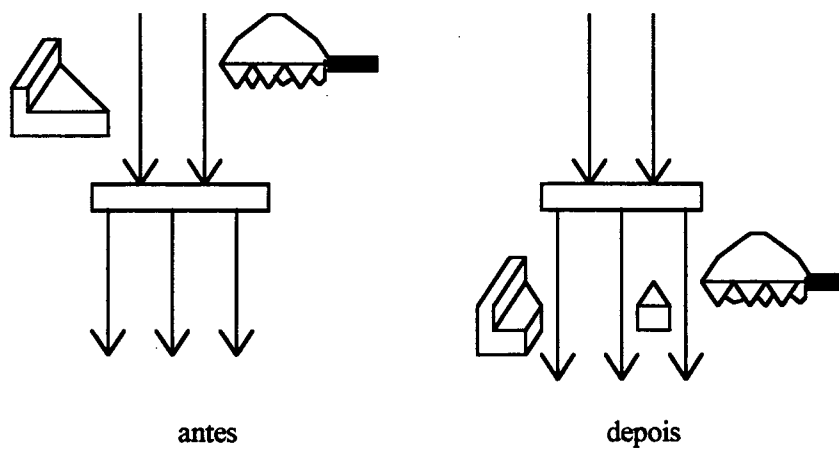
No caso de uma operação de montagem, por exemplo, são tomadas como entradas da transição todas as partes necessárias para a montagem de um determinado objeto. A operação é efetivada neste caso, só quando todas as partes necessárias estão presentes, como visto na figura seguinte:



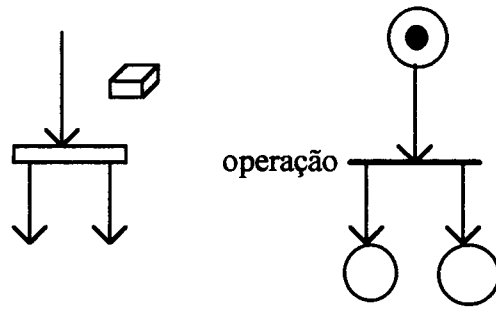
Inversamente pode-se ter uma operação de desmontagem como segue:



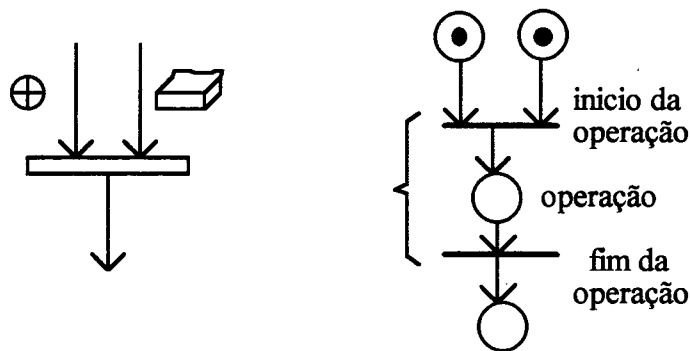
Num sentido mais geral podemos ter operações com diversas entradas e diversas saídas:



Assim , estas operações de produção pode ser representadas por Redes de Petri da seguinte maneira:



ou, como segue abaixo



5.7.- Síntese de Redes de Petri para FMS

Para problemas de síntese de Redes de Petri para um sistema de manufatura, um lugar na rede pode representar ou uma operação ou um recurso (Narahari 1985; Zhou 1989). Existem basicamente três tipos de lugares que dependem da marcação inicial de uma RdP.

Supondo que M é a marcação inicial de uma Rede de Petri "R" e "p" é um lugar em "R":

Definição 1.- Um lugar "p" é chamado de um lugar-"A" ou um lugar-operação se $M_0(p) = 0$.

Um lugar-A "p" frequentemente representa um lugar-operação. Inicialmente não existe nenhuma operação no sistema. Ela pode também representar um ou dois lugares num Buffer, e $M_0(P) = 0$ significará que o Buffer está vazio.

Definição 2.- Um lugar "p" é chamado de um lugar-"B" ou um lugar-recurso-fixo se $M_0(P)$ é uma constante.

Um lugar-'B' representa a disponibilidade de um número fixo de recursos. Por exemplo, o número de robôs ou máquinas são geralmente fixos para um certo sistema. Quando "p" representa a disponibilidade de um recurso, $M_0(p) = 1$. Quando "p" representa a disponibilidade de espaço vazio num Buffer, $M_0(p)$ é a capacidade do Buffer que pode ser maior ou igual a 1.

Definição 3.- Um lugar "p" é chamado de um lugar-"C" ou um lugar-recurso-variável se "p" é inicialmente marcada com $M_0(p) > 0$, e o número de fichas inicial em "p" é variável ($M_0(p)$ é variável).

Um lugar-"C" pode representar a disponibilidade de matéria prima, pallets, acessórios, etc. Geralmente um Sistema de Manufatura necessita ser projetado de maneira tal que o sistema tenha uma flexibilidade e eficiência muito grande, com o menor número de acessórios ou pallets. A distribuição inicial de fichas sobre todos os lugares-"C" deverão ser determinados de modo tal que o sistema seja livre de Deadlock.

5.7.1. - Metodologia para Síntese dos Modelos de Redes de Petri

A maior vantagem de utilizar uma metodologia para síntese de modelos de RdPs é que uma modelagem detalhada pode ser introduzida numa maneira incremental tal que a complexidade pode facilmente ser tratada (aliviada); e as propriedades importantes da RdP resultante podem ser garantidas de modo que o enorme esforço realizado para a análise matemática das propriedades pode ser evitado.

Este método de síntese (Zhou 1992 (b)), trata lugares-operação e lugares-recurso separadamente, sendo que no estágio inicial, pode-se construir uma RdP numa maneira top-down e numa maneira botton-up no estágio final.

Note-se que uma Exclusão Mútua Paralela (1-PME) modela um recurso compartilhado da seguinte maneira. Uma k-PME modela um recurso compartilhado por diferentes processos em paralelo. Um exemplo é dado pela figura 5.5 (a) onde $D = \{(t_1, t_2), (t_3, t_4)\}$ e $P_E = p_5$. A RdP tem as mesmas propriedades de Safeness (segura), Liveness (viva) e Reversibility (reversível) como uma rede sem p_5 . Este fato é garantido pelo seguinte teorema.

Teorema 1.- Suponha que $R' = (P'_A \cup P'_B \cup P'_C, T', I, O', M'_O)$ é uma sub-rede de $R = (P, T, I, O, M_O)$, $P = P' \cup (P'_B \cup \{p_E\}) \cup P_C$, $T = T'$, e $D = \{(t_{ai}, t_{bi}), i \in N_k\}$. se (p_E, D) é uma k -PME de R , então

- 1) R é limitada (safe) se R' é limitada ;
- 2) R é viva se R' é viva;
- 3) R é reversível se R' é reversível.

A demonstração deste teorema pode ser visto em Zhou (1991).

Por outro lado, uma SME modela um recurso compartilhado seqüencialmente em relação a grupo de processos. Segundo esta definição, $(p_E, D^1 \cup D^2)$ na figura 5.5 (a) onde $p_E = p_5$, $D^1 = \{(t_1, t_2)\}$, e $D^2 = \{(t_3, t_4)\}$ pode verificar-se facilmente ser uma SME (Zhou 1991).

Definição 4.- Dada uma RdP marcada $R = (P, T, I, O, M_O)$ a capacidade de fichas $C(t, t')$ entre as transições " t " e " t' " em " R " é o número máximo de disparos de " t " desde a marcação inicial sem disparar " t' ".

Podemos definir portanto a capacidade de fichas entre duas PME's consecutivas numa SME e denotá-lo por C^i para $1 \leq i \leq L-1$. C^i pode entender-se como a capacidade de tarefas que permanecem entre dois grupos de processos.

Para cada RdP tendo uma SME podemos então definir $\alpha^i(p)$, $p \in P$, para a i -ésima PME em sua SME e $1 \leq i \leq L-1$ como o número mínimo de fichas em p tal que cada transição na SME possa ser habilitada. Então acharemos o número máximo de disparos da transição t para uma dada $M_O(p)$, assumindo que a outra $p \in P_C$ tem suficientes fichas. Seja $\delta^i(p, t)$ proporção mínima de disparos de transições para a marcação inicial $M_O(p)$ para todos os valores de $M_O(p)$. $\beta_i(p)$ é definido como o maior $\delta^i(p, t)$ para todo $t \in T^i_a$. definições formais são achadas em Zhou (1991).

Na figura 5.5 (a) por exemplo nós temos: $L = 2$, $C = C^1(t_2, t_3) = M_O(p) = 2$, $\alpha^1(p_1) = 1$, e $\beta^1(p_1) = 1$.

Teorema 2.- Suponha que $R' = (P'_A \cup P'_B \cup P'_C, T', I', O', M'_O)$ é uma sub-rede de $R = (P, T, I, O, M_O)$, $P = P' \cup (P'_B \cup \{p_E\}) \cup P_C$, $T = T'$, e $D = D^1 \cup D^2$, $k_i = |D^i| \geq 1$, $D^i = \{(t_{aj}^i, t_{bj}^i), j \in Nk_i\}$. Seja $T_a = \{t_{aj}^i, i \in \{1,2\}\}$ e $T_b = \{t_{bj}^i, j \in Nk_i, i \in \{1,2\}\}$. Se (p_E, D) é uma SME, então:

- 1) R é limitada (safe) se R' é limitada;
- 2) R é viva se R' é viva;
- 3) R é reversível se R' é reversível.

A demonstração deste teorema é também visto em Zhou (1991).

5.7.2.- Módulos de Desenho Básico

O conceito de blocos bem formados proposto por Valette (1979) é muito importante para problemas de síntese de RdPs. A partir destes conceito, quatro modelos de desenho básico foram introduzidos por Zhou (1992) para ajudar na síntese de RdP para sistemas de manufatura que são facilmente aceitos e aplicados quando um problema prático de síntese num contexto de manufatura é achado.

Seja $S_p = (P, T, I, O, M_{s0})$, onde $P = \{p_{in}, p_{out}\} \cup P_s$, e p_{in} e p_{out} são dois lugares distintos. S_p é chamado de um lugar "PN block". $S_T = (P, T, I, O, M_{s0})$ é chamado de uma transição "PN block" se $T = \{t_{in}, t_{out}\} \cup T_s$, onde t_{in} e t_{out} são duas transições diferentes.

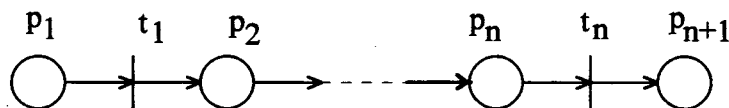


Fig. 5.8.- Definição de uma seqüência PN ($n > 0$)

Definição 5.- Um módulo seqüencial PN é definido como uma RdP (figura 5.8) se seu lugar "PN block" é formado por:

$$S_p = (\{p_1, p_2, \dots, p_{n+1}\}, \{t_1, \dots, t_n\}, I, O, 0_{n+1})$$

onde:

0_{n+1} é um vetor zero de dimensão $n+1$, $p_{in} = p_1$, $p_{out} = p_{n+1}$.

e sua transição "PN block" é formado por:

$$S_T = (\{p_2, \dots, p_n\}, \{t_1, \dots, t_n\}, I, O, 0_{n-1})$$

$$e \ t_{in} = t_1, \ t_{out} = t_n, \ n > 1$$

No contexto de sistemas de manufatura, uma seqüência PN representa uma série de operações sucessivas.

Definição 6.- A substituição de um lugar "p" em $R' = (P' \cup \{p\}, T', I', O', M_0)$ com um lugar "PN block" $S_p = (\{p_{in}, p_{out}\} \cup P_s, T_s, I_s, O_s, M_{s0})$ resulta numa RdP refinada $R = (P, T, I, O, M_0)$ onde:

$$P = \{p_{in}, p_{out}\} \cup P_s \cup P', \ e, \ T = T_s \cup T'$$

Similarmente, a substituição de uma transição t em $R' = (P', T' \cup \{t\}, I', O', M_0)$ com uma transição "PN block) $S_T = P_s, \{t_{in}, t_{out}\} \cup T_s, I_s, O_s, M_{s0})$ resulta numa RdP refinada $R = (P, T, I, O, M_0)$. Assim um lugar ou transição é dita ser substituída por um "PN block".

Teorema 3.- (refinamento): Suponha que um lugar ou uma transição em R' é substituído por um módulo de desenho básico e R é a rede refinada. Então:

- 1) R é limitada (safe) se R' é limitada;
- 2) R é viva se R' é viva, e;
- 3) R é reversível se R' é reversível.

5.7.3.- Descrição do Processo de Sínteses

passo 1.- Escolher uma RdP limitada, viva e reversível como modelo de primeiro nível de um sistema que é executável, quando a maioria dos recursos estão disponíveis, e determinar " P_A ", " P_B " e

" P_C ", onde " P_A " e " P_B " estão sujeitos a mudanças enquanto que " P_C " é fixado e seus elementos têm um número variado de fichas inicialmente.

passo 2.- decompor o sistema em diversos subsistemas expressados como lugares, e substituir estas operações por módulos de desenhos básicos mais detalhados (refinamento), até que mais divisões sejam impossíveis ou sem necessidade. " P_A " é estendido em cada passo.

passo 3.- Adicionar apropriadamente os lugares de recursos não compartilhados em cada estágio, quando uma ou mais operações requeiram os recursos. Os arcos são ligados para os lugares-recursos assim que as condições para uma exclusão mútua paralela sejam satisfeitas. Em cada adição, somente " P_B " é acrescentada.

passo 4.- Adicionar os lugares que modelam os buffer (espaço do buffer livre) completando assim a existência dos mesmos no modelo. Cada lugar do buffer terá inicialmente fichas cujo número é igual à capacidade do buffer.

passo 5.- Adicionar os lugares-recursos compartilhados " P_B " que formam uma exclusão mútua paralela segundo o teorema 1. Os lugares-recursos são lugares-"B".

passo 6.- Adicionar lugares-recursos compartilhados que formam uma exclusão mútua seqüencial e calcular a capacidade de fichas entre suas exclusões mútuas paralelas determinando também o número inicial de fichas em p , $\forall p \in P_C$, segundo o teorema 2. Os lugares-recursos são adicionados para " P_B ".

Finalmente um modelo de RdP limitado, vivo e reinicializável é sintetizado.

5.7.4.- Desenho de Modelos de RdP de Primeiro Nível

Para efetuar o procedimento do item anterior, o primeiro passo torna-se com frequência muito crítico, podendo portanto, se usar uma das duas maneiras básicas de modelos de redes comumente usados para desenhar modelos de RdP de primeiro nível.

a) Muitos sistemas de manufatura produzem, em geral, diferentes tipos de partes, de diferentes tipos de material e/ou de diferentes entradas. As partes produzidas podem ser usadas, por exemplo, para a montagem em outros sistemas. Neste caso, o desenho da RdP para tal sistema pode iniciar com uma RdP simples que contenha um lugar e transição únicos como mostrado na figura 5.9

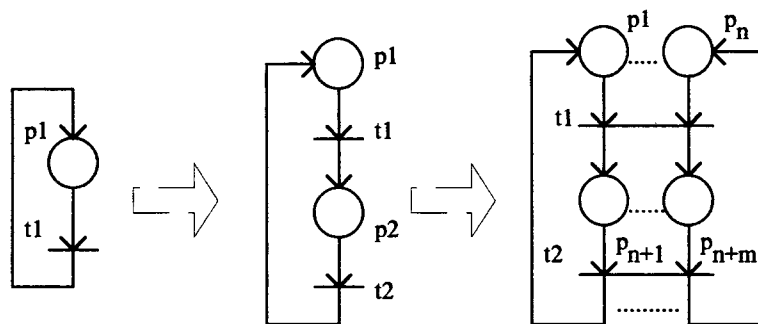


Figura 5.9 - Dois modelos de RdPs de primeiro nível. a) Rede com uma única posição e transição. b) Rede com um único recurso e operação. c) Caso geral

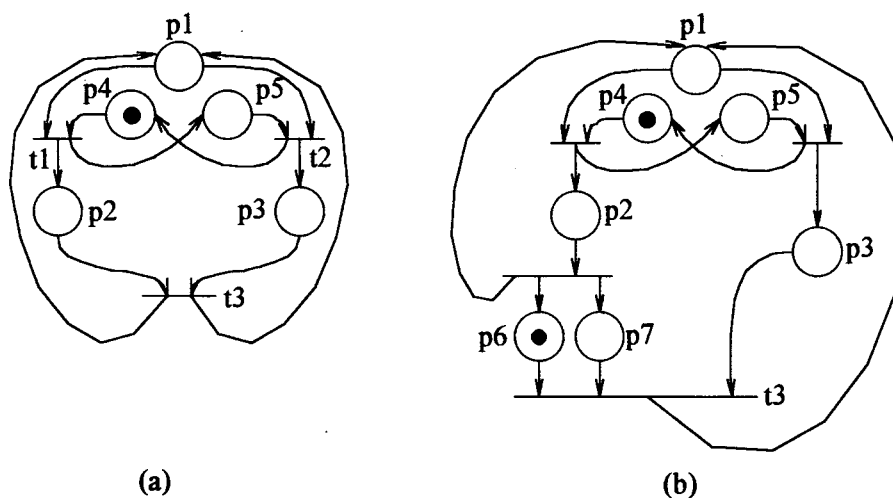
A figura 5.9 (a) pode ser primeiro acrescentada para uma rede como na figura 5.9 (b), a qual modela um sistema com uma única operação e um único recurso. Quando diversos recursos são necessários para iniciar um sistema com diversas operações concorrentes, a extensão da rede anterior pode ser representada pela rede da figura 5.9 (c).

Pode mostrar-se facilmente que esta rede é limitada, viva e reinicializável com respeito à marcação inicial $M_0(p_i) > 0$ para $1 < i < n$. Sendo assim, a rede pode ser interpretada como um sistema onde "m" operações paralelas iniciarão quando "n" recursos estiverem disponíveis, e após a sua finalização, o sistema retorna à condição inicial.

b) Por outro lado, quando um sistema envolve tarefas de montagem e as partes separadas são produzidas com o mesmo tipo de material e na mesma entrada, então a RdP de primeiro nível deverá conter uma estrutura *escolha-sincronização*. O modelo de RdP, portanto, pode ser desenhado como mostra a figura 5.10 (a) onde é de fato uma RdP generalizada desde que Pós $(t_3, p_1) = 2$. Portanto, ela pode ser facilmente convertida a uma RdP ordinária inserindo um lugar e uma transição extra entre t_3 e p_1 . Deve-se observar aqui que dois lugares de controle, p_4 e p_5 , foram introduzidos para garantir que a transição t_1 seja primeiro habilitada, e que t_1 e t_2 disparem alternativamente quando a marcação inicial

for $(m_0(p_1), 0, 0, 1, 0)^T$ e $m_0(p_1) \geq 2$. Esta rede pode ser mostrada como sendo limitada, viva e reversível com a marcação inicial anterior. Sem os lugares de controle a rede conteria um deadlock.

Para alguns sistemas de manufatura, as partes acabadas têm que seguir uma certa ordem para entrar numa estação de montagem, assim o modelo da rede pode ser desenhado como na figura 5.10 (b), onde a primeira parte processada necessita entrar primeiro à estação de montagem. p_6 pode ser interpretada como a disponibilidade de uma estação de montagem e p_7 como a disponibilidade de uma parte na estação. Essa RdP pode ser facilmente modificada para satisfazer diferentes especificações. A RdP mostrada na figura 5.10 (b) pode também ser mostrada como sendo limitada, viva e reversível quando $m_0 = (m_0(p_1), 0, 0, 1, 0)^T$ com $m_0(p_1) \geq 2$.



5.10.- RdP de primeiro nível para sistemas com uma estrutura sincronização-escolha

Os dois tipos de estruturas básicas mostradas neste item podem portanto ser utilizados juntos para desenhar uma RdP de primeiro nível e para modelar um sistema complicado.

5.8.- Análise Estrutural dos Grafos de Eventos de Sistemas de Produção

Como visto em (4.1), um grafo de eventos temporizado é uma classe especial de RdP. Este tipo de grafo é geralmente usado para a modelagem e análise de sistemas job-shop. A modelagem nos permite avaliar o desempenho em estado constante de um sistema sob um processo de produção cíclico

e determinístico. Em vista de que os tempos de processamento de cada atividade são fixados previamente, a produtividade do sistema pode facilmente ser determinado desde o estado inicial. Mostra-se também que, seja qual for a seqüência de tarefas sobre as máquinas, é possível utilizar totalmente alguma máquina em regime permanente com um número finito de tarefas em processo. Isto significa que a taxa de produção máxima é obtida quando essas máquinas (chamadas máquinas gargalo) são utilizadas totalmente (uma condição que sempre pode ser satisfeita).

Este tipo de rede é por natureza fortemente conexa e livre de conflitos, podendo ter sincronizações, e sendo portanto, o número de lugares maior ou igual ao número de transições.

Num sistema job-shop onde máquinas processam "n" produtos diferentes, podem distinguir-se três tipos de circuitos na rede. a) *Circuitos de comando* (cc) que representam o sequenciamento das tarefas sobre as correspondentes máquinas. Uma ficha neste tipo de circuito representa a disponibilidade da máquina para processar uma tarefa específica. Além do mais, nestes circuitos, só pode existir uma ficha, já que uma máquina só pode estar habilitada para processar uma tarefa de cada vez. b) *Circuitos de processo* (cp) que representam o fluxo do processamento de cada produto. Fichas circulando neste tipo de circuito representarão, portanto, produtos em processo ou "pallets".

Além destes circuitos básicos, existem outros tipos de *circuitos elementares* na rede que incluem nós de ambos os circuitos anteriormente citados. Tais circuitos são chamados de circuitos mistos.

5.8.1.- Avaliação do Desempenho

Quanto à análise do desempenho (Freedman 1991; Hillion 1989) Ranchandani mostrou que o número total de fichas em qualquer circuito elementar é invariante, sendo que um circuito elementar é um caminho que vai de um nó (lugar ou transição), e após "n" disparos volta ao mesmo nó, sem que nenhum outro nó seja repetido.

Sendo assim, para qualquer circuito elementar "c", o tempo de ciclo será definido por:

$$TC(c) = u(c)/M(c) \quad (5.1)$$

onde:

- $u(c)$, denota a soma dos tempos de disparo das transições do circuito;

- $M(c)$, é o número de fichas circulando no circuito.

Sendo portanto o tempo de ciclo máximo igual a:

$$T = \text{máx} (TC(c_j)) \quad (5.2)$$

É importante neste ponto observar que $TC(c)$ pode ser calculado diretamente da distribuição inicial de fichas M_0 , desde que a equação 5.1 implique que $M(c) = M_0(c)$

Aqueles circuitos "c" para os quais o tempo de ciclo é máximo são chamados de circuitos críticos. Tais circuitos são, portanto, aqueles que irão determinar a produtividade do sistema em regime. Mais precisamente para um grafo a eventos temporizado fortemente conectado, onde transições disparam assim que elas sejam habilitadas, o índice de disparo de todas as transições em estado constante (ou regime permanente) é dado por:

$$\lambda = 1 / T = \text{min.}(1 / TC) \quad (5.3)$$

Tal índice (taxa de produtividade) é alcançado após uma quantidade finita de tempo, onde o sistema atinge um comportamento periódico; sendo a mesma imposta pelos circuitos críticos. Portanto, para determinar o índice de produtividade usando a equação 5.3 é necessário determinar todos os circuitos elementares da rede. Neste caso as soluções de $X^T C = 0$ conduzem a um conjunto de circuitos elementares da rede. O número destes circuitos é $n_p - n_t + 1$. A enumeração de todos os circuitos da rede será também útil para minimizar o número total de fichas na rede.

Caso $T = TC(cc)$, sendo "cc" um circuito de comando; então a máquina gargalo estará plenamente utilizada. Assim, o limite superior da produtividade será determinado pelo tempo de ciclo da máquina gargalo.

5.8.2.- Funcionamento em Velocidade Máxima

Como visto no parágrafo anterior, o limite superior da produtividade será determinado pelo tempo de ciclo da máquina gargalo (T_0), o qual garante a máxima produtividade do sistema.

Seja então $\lambda_0 = 1 / T_0$ o correspondente índice, se o sistema estiver funcionando neste índice, então a máquina gargalo será totalmente utilizada. Em termos de Redes de Petri, isto significa que o correspondente circuito de comando (isto é, aquele com o máximo TC) é crítico. O problema é agora determinar se este índice é alcançável, e se é, sob que condições.

Este problema, foi mostrado em Cohem (1985) que para um sistema flow-shop ali existe sempre uma distribuição finita de recursos (isto é, uma marcação inicial dos lugares recurso-variado), tal que a máquina gargalo seja totalmente utilizada em regime permanente. Permitindo tarefas estarem em processo no estado inicial, pode-se generalizar este resultado para um sistema job-shop (Hillion 1989).

Sendo assim, nós podemos escrever a equação 5.2 na forma seguinte:

$$T_0 = \max (\max (C(cc)), \max (C(\gamma))) \quad (5.4)$$

onde "cc" representa um circuito de comando e $\gamma \neq cc$ representa qualquer outro tipo de circuito. Por definição temos que:

$$T_0 = \max (C(cc)) \quad (5.5)$$

é o índice no qual a máquina gargalo é utilizada plenamente.

Ora, um circuito γ o qual não é um circuito de comando contém no mínimo um lugar o qual é um Buffer ou um lugar recurso-variado. Se nós colocarmos bastantes fichas neste lugar (numa quantidade finita), nós podemos fazer sempre o tempo de ciclo $TC(\gamma)$ menor que T_0 , garantindo assim a plena utilização da máquina gargalo.

Já que o número de fichas em qualquer circuito de comando é 1, nós temos que para $i = 1, 2, \dots, k$

$$u(\gamma_i) \leq k \cdot T_0 \quad (5.6)$$

além do mais, supondo que cada Buffer e lugar recurso-fixo contenha exatamente uma ficha,

$$M(\gamma) \geq k \quad (5.7)$$

tendo portanto, a partir das equações 5.6 e 5.7, para qualquer circuito $\gamma \neq cc$,

$$TC(\gamma) = (u(\gamma) / M(\gamma)) \leq T_0 \quad (5.8)$$

este resultado é importante por duas razões: primeiro, ele provê diretamente o estado inicial que sempre leva à total utilização da máquina gargalo em regime permanente (seja qual for o processo de manufatura, assim como a seqüência de tarefas sobre as máquinas). Segundo, deduz-se deste resultado um limite superior para o número de recursos (pallett) necessários para assegurar a máxima produtividade do sistema.

Considerando $M_{\min}(\gamma)$ como o número mínimo de fichas no circuito " γ " para qualquer circuito elementar, tal que $TC(\gamma) \leq T_0$, isto é:

$$M_{\min}(\gamma) - 1 < u(\gamma) / T_0 \leq M_{\min}(\gamma) \quad (5.9)$$

onde T_0 representa o tempo de ciclo da máquina gargalo (equação 5.5), a seguinte condição é necessária e suficiente para assegurar a total utilização da máquina gargalo em regime permanente.

Para qualquer circuito elementar " γ "

$$M_0(\gamma) \geq M_{\min}(\gamma) \quad (5.10)$$

onde M_0 é a marcação inicial, a qual determina unicamente o número total de fichas num circuito em qualquer estado do sistema.

O seguinte exemplo foi tomado de Cury (1990) para ilustrar os conceitos emitidos, sendo que consideraremos também o caso quando o número de peças no buffer para o produto B seja igual a 1.

Ex: Consideremos o grafo a eventos da figura 5.11, onde quatro máquinas produzem dois tipos de produtos A e B, definidos pelos seguintes processos.

A: (m1/2, m2/3)

B: (m4/3, m1/4, m3/4)

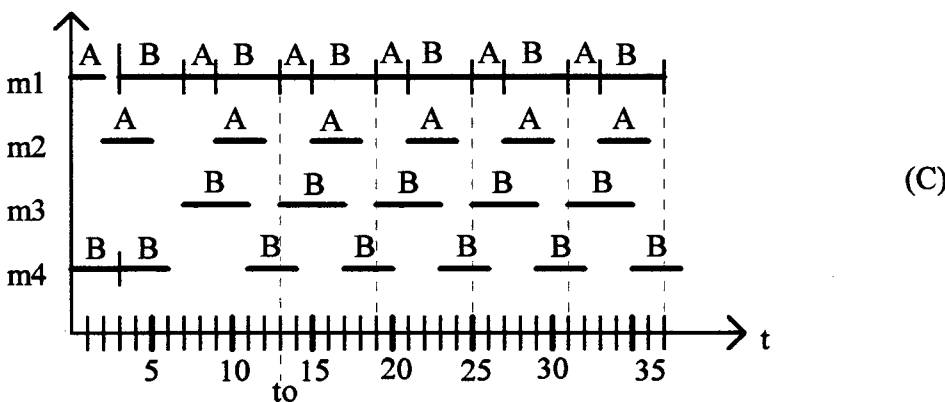
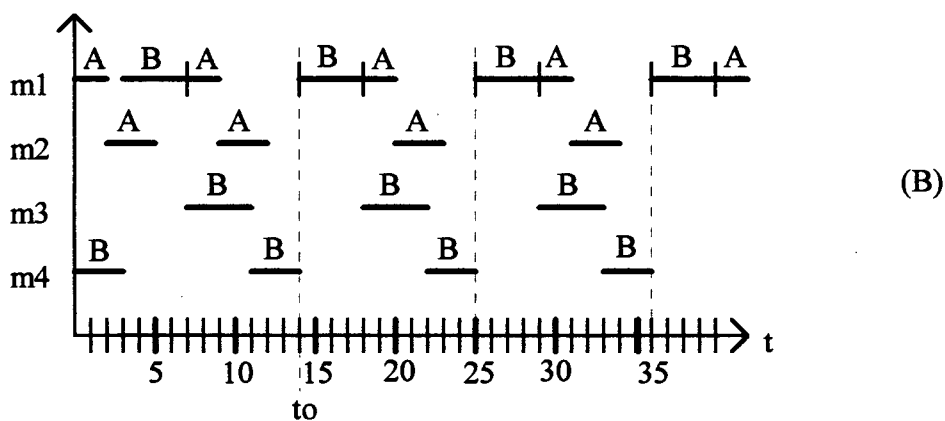
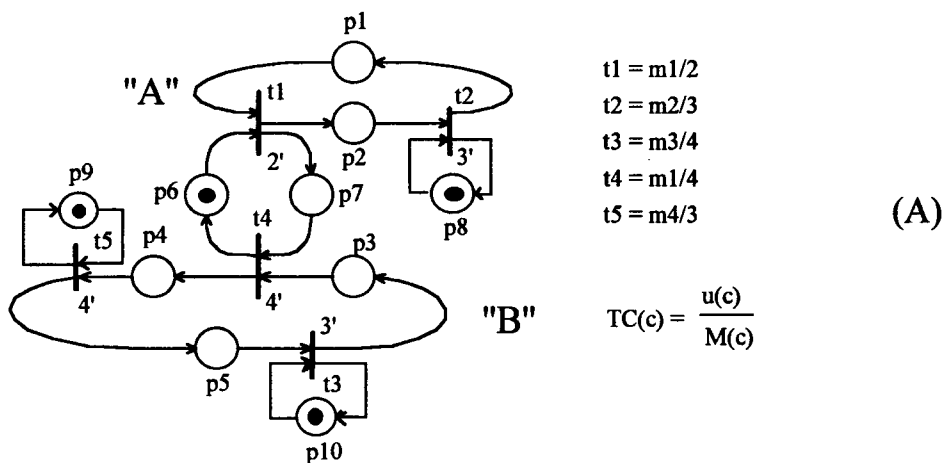


Fig. 5.11 - Grafo a eventos do sistema, e a evolução da sua execução a partir do estado inicial.

Os circuitos elementares obtidos no grafo são três que correspondem às tarefas A e B (circuitos de processo), e o circuito que corresponde ao circuito de comando da máquina m1. Sendo os tempos gastos em cada circuito os seguintes:

$$c_1 = 2+3 = 5, c_2 = 3+4+4 = 11, c_3 = 2+4 = 6$$

a) Consideremos primeiro que a marcação inicial seja $M_0 = (1, 0, 0, 0, 1)$, neste caso o grafo (b) da figura anterior mostra a evolução do grafo a partir do estado inicial considerado, tendo portanto que o tempo de ciclo máximo do sistema é igual a:

$$T = \max. TC(c_j) = \max(5, 11, 6) = 11 \text{ unidades de tempo}$$

Portanto o circuito "c₂" é um circuito crítico. Assim o sistema atingirá um comportamento periódico em regime permanente com período $T = 11$. Sendo sua taxa de produtividade $\lambda = 1 / 11$, ou seja, uma peça de cada tipo é completada a cada 11 unidades de tempo

Note-se que neste caso a máquina gargalo não esta sendo plenamente utilizada.

b) Agora, consideremos uma marcação inicial $M_0 = (1, 0, 0, 0, 2)$. O grafo (c) da figura anterior mostra sua evolução a partir de seu estado inicial. Neste caso observa-se que o sistema atinge um comportamento periódico em regime permanente com $T = 6$ e máxima taxa de produtividade (pleno uso da máquina m1). Sendo neste caso a taxa de produtividade de $\lambda = 1 / 6$, ou seja, uma peça de cada tipo é completada a cada 6 unidades de tempo.

Neste caso $c_2 = 11 / 2 = 5.5$, devido a que existem duas fichas no circuito.

Em ambos os casos, tanto na figura (b) e (c), t_0 mostra o início do período.

CAPÍTULO 6.- APLICAÇÃO DAS REDES DE PETRI: MODELAGEM E ANÁLISE DE UM SISTEMA DE MONTAGEM COM REDES DE PETRI TEMPORIZADA

6.1.- Introdução

Uma linha de montagem como foi visto anteriormente é um sistema discreto pode portanto ser facilmente modelado por uma RdP. Linhas de montagem são Sistemas de Manufatura em que diversas máquinas e/ou robôs (com ou sem ajuda do homem) realizam operações de montagem. Por exemplo na figura 6.1 mostra-se uma linha de montagem onde três tipos de máquinas podem ser distinguidos: a) Máquinas de entrada tais como M1, M2, M4, M5, b) Máquinas que realizam operações de montagem, tais como M3, M8, M9, e c) Máquinas de saída que são representadas por M6, M11, M12, M13.

Uma característica básica das linhas de montagem é que as partes fluem num único sentido, das máquinas de entrada para as máquinas de saída; sendo que as mesmas são separadas por buffers (B_{ij}). Para que uma máquina de montagem possa realizar uma operação, tem que ter no mínimo uma parte em cada um dos buffer de entrada realizando assim desse modo uma operação dada de montagem. Após a finalização da operação de montagem, então, o conjunto será colocado em seu buffer de saída correspondente.

Quando se analisa uma linha de montagem o principal interesse é geralmente com a medida do desempenho, isto é, determinar o seu índice de produção. Noutras palavras, determinar a produtividade da máquina de saída além de calcular o tamanho médio dos buffers.

Na figura 6.1 podemos observar que cada máquina está conectada no mínimo com um buffer de capacidade finita e que cada buffer está conectado por exatamente duas máquinas, uma máquina que alimenta o buffer (máquina upstream) e outra que absorve as partes do mesmo (máquina downstream).

Numa linha de montagem (DiMascolo 1991) uma máquina é dita morta (starved) se no mínimo um dos seus buffers de entrada (upstream) está vazio. De modo semelhante, uma máquina é dita

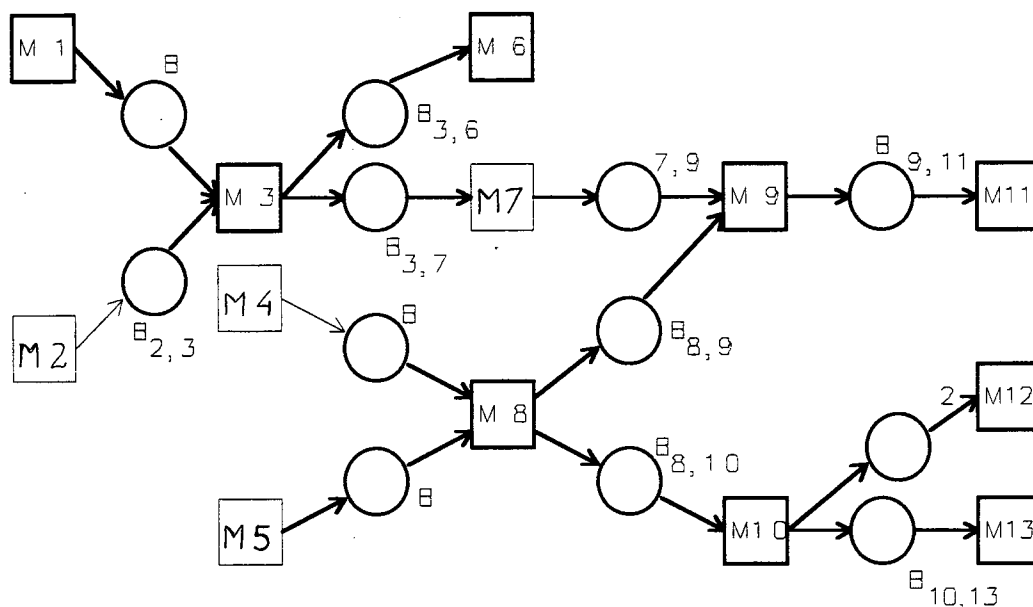


Fig. 6.1- Linha de Montagem com 13 Máquinas (DiMascolo 1991)

bloqueada (blocked) se após a finalização de uma operação, no mínimo um dos seus buffers de saída (downstream) estiver cheio. Portanto, se uma máquina está morta e/ou bloqueada estará impedida de trabalhar. Assim, quando uma máquina M_i alcança um estado que não lhe permite seguir trabalhando (por quebra ou avaria da máquina por exemplo) o número de partes de cada buffer upstream tende a incrementar, enquanto que o número de partes no(s) buffer(s)downstream tende a diminuir. Neste sentido um buffer pode tornar-se cheio ou vazio respectivamente.

Neste capítulo nós consideramos que as máquinas de entrada nunca estão mortas e as máquinas de saída nunca estão bloqueadas, e todos os buffers intermediários são de capacidade finita.

Numa linha de montagem (Moreira 1993), o conteúdo de trabalho (Q) em uma unidade de produto é medido pela soma das tarefas, o mesmo que representa o tempo que gastaríamos para fazer uma unidade de produto se houvesse um só posto de trabalho. No caso da figura 6.2, o conteúdo de trabalho será, portanto, igual à soma dos tempos consumidos pelas máquinas M1, M2, M3, M4 e M5. O tempo disponível em cada posto de trabalho é chamado de tempo de ciclo, o mesmo que é imposto pela tarefa que contenha o maior tempo de duração. Este tempo de ciclo é, portanto, quem irá determinar a eficiência do sistema, e é dado por:

$$\text{Eficiência} = \frac{\text{Conteúdo de Trabalho (Q)}}{N \times C} \quad (6.1)$$

onde "N" é igual ao número mínimo de postos de trabalho que compõem a linha de montagem, e "C", é o tempo de ciclo. Em termos de Redes de Petri a eficiência de cada circuito será dada por:

$$E_c = \frac{\text{Contúdo de Trabalho do circuito } u(\gamma)}{N_c \times T} \quad (6.2)$$

enquanto que a eficiência do sistema será dado por:

$$E_T = \frac{\text{Somatoria do conteúdo de trabalho de cada circuito } cp}{N_{cp} \times \text{tempo disponível em cada circuito}} \quad (6.3)$$

Tomando em consideração o esplanado anteriormente, procederemos então a modelar e analisar uma linha de montagem como mostrado na figura 6.2 utilizando uma RdP T-temporizada, determinando o tempo de ciclo mínimo para o sistema com a máxima produtividade, e o mínimo inventario em processo.

6.2.- Aplicação

a) Descrição do Sistema

A figura 6.2 representa um sistema de manufatura automatizado, cujo layout é mostrado na figura 6.3.

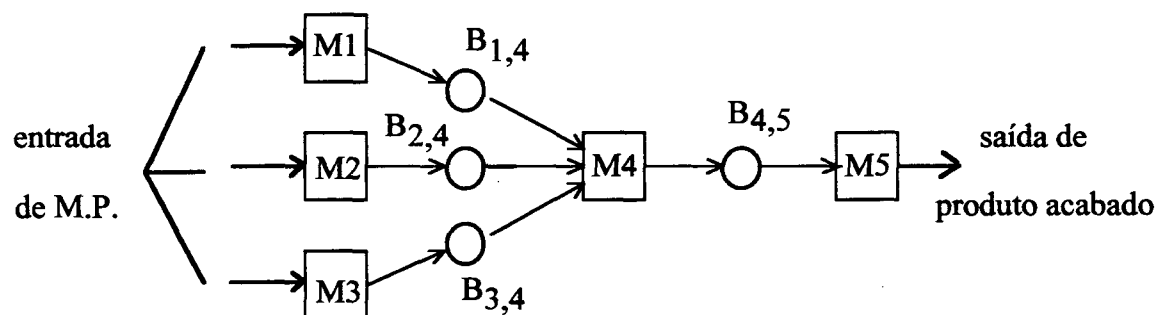


Fig. 6.2.- Linha de Montagem com 5 Maquinas

O sistema consiste de quatro máquinas (M1, M2, M3, M5), dois robôs (R1, R2), quatro Buffers (B1, B2, B3, B4) de capacidade b1, b2, b3 e b4 respectivamente, e uma célula de montagem (M4). O sistema primeiro processa três tipos de partes chamadas "A", "B" e "C" respectivamente que logo serão montadas na célula M4. Uma parte "A" primeiro é processada pela máquina M1, e então a mesma é colocada no Buffer B1. As operações de carga e descarga são executadas automaticamente da entrada de matéria prima para a máquina M1 e da máquina M1 para o Buffer B1. A transferência de uma parte "A" do Buffer B1 para a célula de montagem M4, é feita com a ajuda do robô R1. Um procedimento similar é usado para as partes "B" e "C" como mostra a figura 6.3, isto é, máquina M2, Buffer B2 e robô R1 para as partes "B"; máquina M3, Buffer B3 e robô R2 para as partes "C".

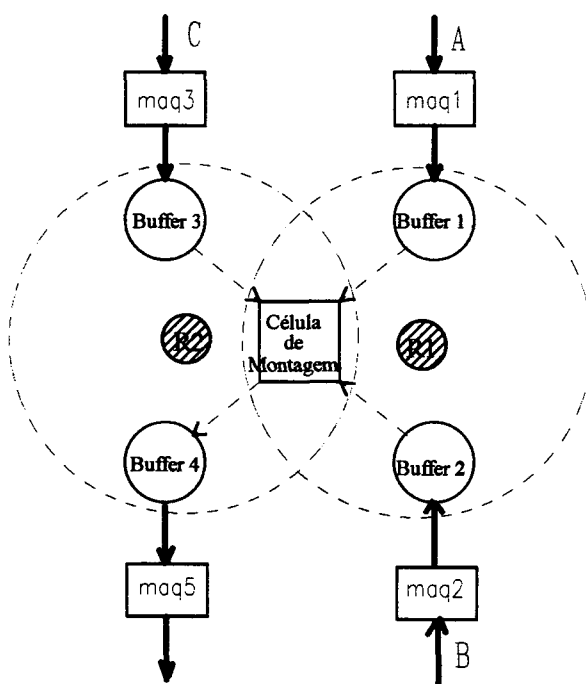


Fig. 6.3.- Layout do Sistema de Manufatura Automatizado da figura 6.2

O processo de montagem (B sobre C sobre A) é realizado na célula M4 pelos robôs R1 e R2 ambos trabalhando juntos. Primeiro, é requerido que uma parte "A" entre na célula de montagem e em seguida uma parte "C", é então montada sobre uma parte "A", para finalmente uma parte "B" ser montada sobre o conjunto "CA". R1 é responsável pelas partes "A" e "B", enquanto que R2 é

responsável pelas partes do tipo "C" e da liberação do conjunto "BCA" da célula de montagem para o Buffer B4.

Dada a especificação anterior, é então requerido desenhar um modelo em Rdp de modo tal que um bom comportamento do sistema seja garantido e que satisfaça aos requerimentos citados no final do item 6.1.

b) Processo de Modelagem

Para construir nosso modelo em Rdp seguiremos a metodologia de síntese apresentada no capítulo anterior para chegar ao nosso modelo final com uma pequena variante no passo 5.

passo 1.- Escolhemos uma Rdp viva, limitada e reversível como modelo de primeiro nível (Figura 6.4), onde $P_A = \{p_{04}, p_{05}, p_{06}, p_{11}, p_{15}, p_{18}\}$, $P_B = \{p_{20}\}$, $P_C = \{p_1, p_2, p_3\}$ com $M(p_{20}) = 1$. A rede é viva, limitada e reversível sempre que $M_0(p_1) = M_0(p_2) = M_0(p_3) \geq 1$. A explicação de todos os lugares e transições é mostrada na tabela 6.1.

passo 2.- Seis componentes nesta rede podem ser decompostos e substituídos por módulos seqüenciais "PN". Segundo o teorema 3 dado no capítulo anterior, os lugares p_{04} e p_{06} são substituídos separadamente por módulos seqüenciais "PN" consistindo de três lugares e duas transições cada um, e de dois lugares e uma transição para p_{05} ; t_{01} e t_{03} são refinados por módulos seqüenciais de duas transições e um lugar, e três transições e dois lugares para t_{02} respectivamente. A rede resultante é mostrada na figura 6.5, e a descrição detalhada a respeito das operações do sistema é mostrado na tabela 6.2.

passo 3.- Adicionando os lugares que correspondem aos recursos fixos não compartilhados p_{20} , p_{21} , p_{22} e p_{23} com $M_0(p_{20}) = M_0(p_{21}) = M_0(p_{22}) = M_0(p_{23}) = 1$, respectivamente, pode-se verificar facilmente ser uma 1-PME como mostra a figura 6.6.

passo 4.- É provado facilmente que a adição de p_{24} , p_{25} , p_{26} e p_{27} conduz à formação de módulos de buffer como mostra a figura 6.6, os quais se encontram encerrados entre linhas pontilhadas. Pode-se

mostrar portanto que a construção de módulos de buffer como visto na figura 6.6, formam blocos bem formados e preservam as propriedades de limitação, vivacidade e reversibilidade (Zhou 1992).

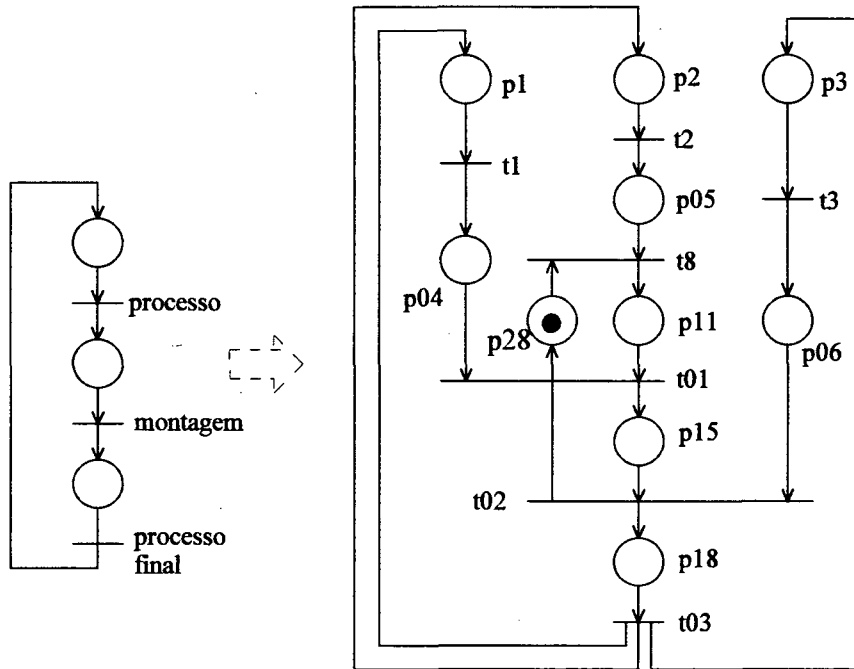


Fig. 6.4.- Rede de Petri de Primeiro Nível para o Sistema de Montagem

Lugares / transições	Significado
p1	disponibilidade de matéria prima p/C
p2	disponibilidade de matéria prima p/A
p3	disponibilidade de matéria prima p/B
p04	máquina M3 trabalhando
p05	máquina M1 trabalhando
p06	máquina M2 trabalhando
p11	Colocar uma parte A na célula de montagem M4
p28	disponibilidade de célula de montagem M4 livre
p15	disponibilidade de célula de montagem com conjunto CA
p18	disponibilidade do Buffer B4 com um conjunto BCA
t01	montagem de parte C sobre A
t02	montagem de parte B sobre CA
t03	maquinado final e liberação do produto
As demais transições representam o início e fim das operações	

Tab.6.1.- Descrição dos Lugares e Transições da RdP da figura 6.4.

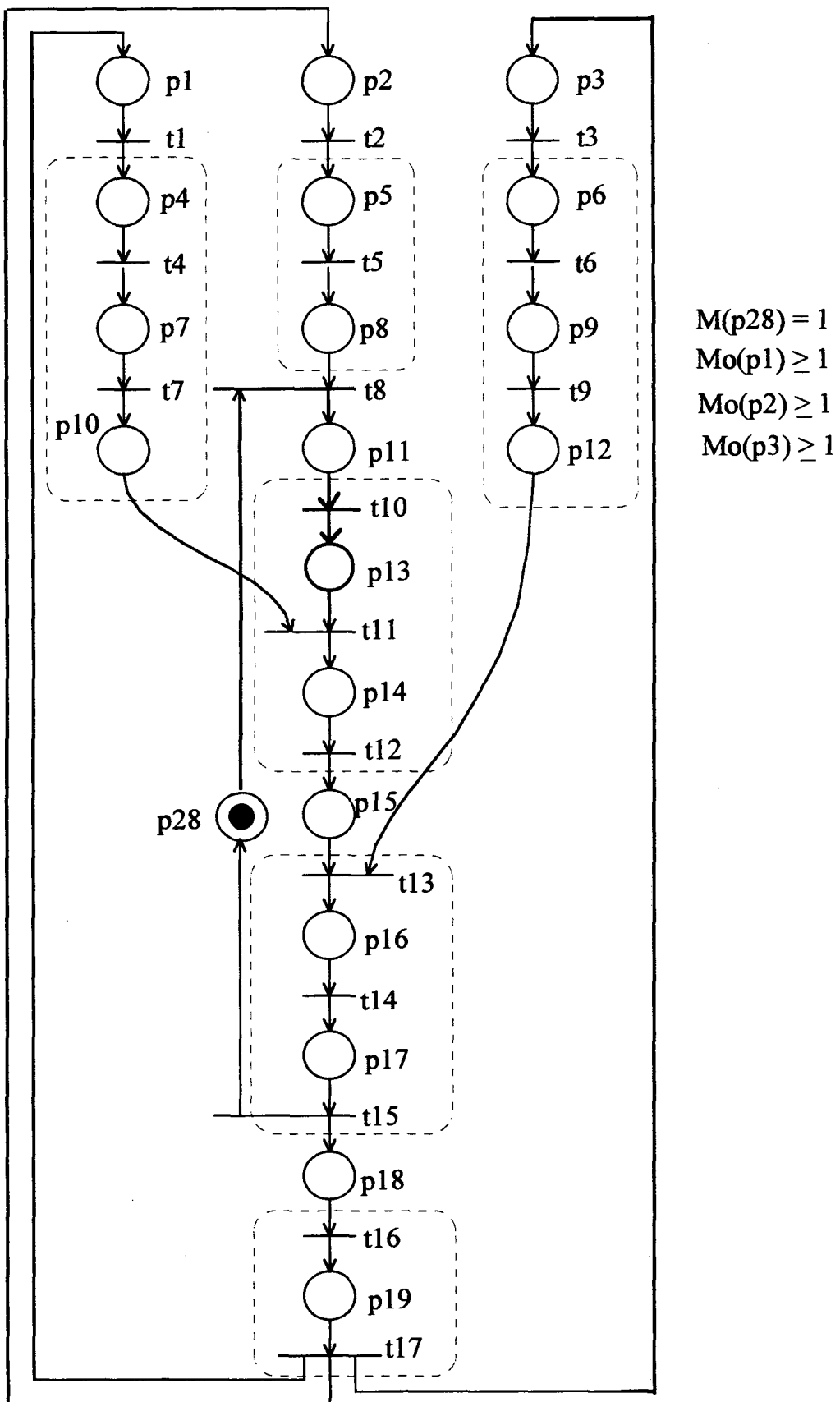


Fig. 6.5.- Rede de Petri após o Refinamento de 3 Lugares e 3 Transições

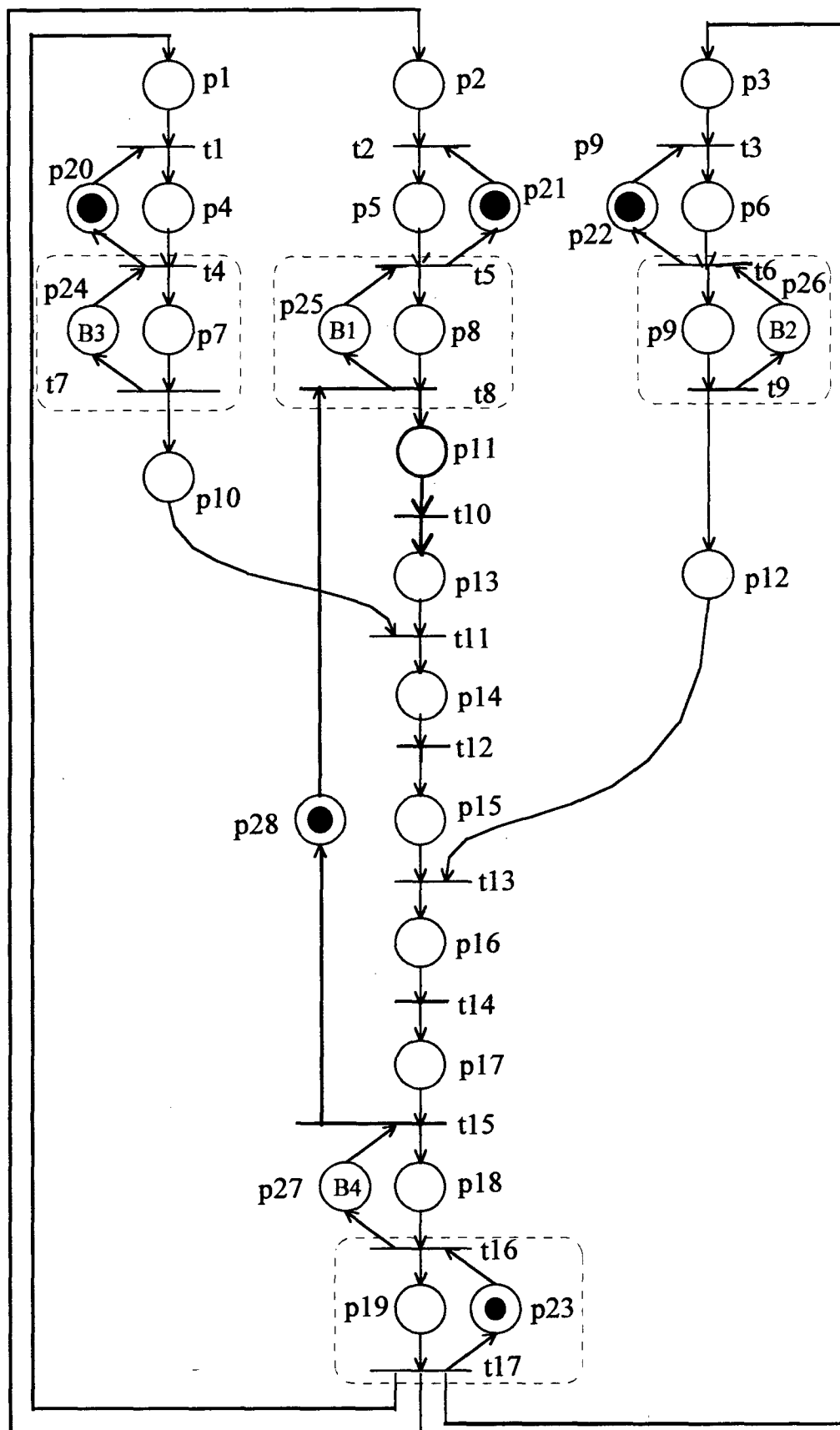


Fig. 6.6.- Rede de Petri após a Adição de Recursos não Compartilhados

passo 5.- Adicionando os recursos não compartilhados p_{28} e p_{29} (como mostra a figura 6.7) que modelam uma 1-PME; e uma 2-PME respectivamente, com $D^1 = [(t_7, t_{14}) \text{ e } P_E = p_{28}]$, e $D^2 = [(t_8, t_{11}) \cup (t_9, t_{13}) \text{ e } P_E = p_{29}]$; observa-se que duas possíveis situações de deadlock poderiam potencialmente acontecer. Primeiro: dado que existe uma ordem de entrada das partes "A", "B", e "C" na célula de montagem, isto é, que é requerido primeiro uma parte "A" na célula, para que depois uma parte "C" seja montada sobre a parte "A"; não poderia acontecer no sentido inverso (a menos que não seja requerida uma ordem de entrada), portanto, se t_7 disparar primeiro que t_8 , um deadlock seria evidente. Segundo: já que uma parte "B" tem que entrar na célula de montagem só após ter sido realizada a operação de submontagem de "C" sobre "A", pode acontecer que t_9 dispare primeiro que t_8 ; se isso acontecer, um outro estado de deadlock seria evidente no sistema.

Em ambos os casos, para evitar estas duas possíveis situações desagradáveis no sistema, precisaremos portanto controlar a ordem de entrada das partes na célula. Uma maneira possível de solucionar este problema seria utilizando arcos inibidores (Peterson 1981) para estabelecer a prioridade; mas este tipo de arcos não se adaptam bem a estas situações de prioridade, além de perder poder de análise com a introdução deste tipo de arco. Neste ponto, nós introduzimos um tipo de lugar que controle esse fato, aumentando assim a poder de modelagem sem perder poder de análise. Este tipo de lugar será chamado de lugar fantasma, já que no contexto de manufatura não tem significado nenhum, e só irá servir para estabelecer uma ordem de execução das atividades. A representação gráfica desses lugares pode ser visto na figura 6.7 e são dados pelos lugares p_{31} e p_{32} , observando agora claramente que t_7 somente irá disparar após uma parte "A" estiver disponível na célula de montagem; do mesmo modo, t_9 somente irá disparar se antes foi realizada a submontagem da parte "C" sobre a parte "A" e o robô 1 estiver livre.

Note-se que no modelo final da RdP para o sistema modelado, podem ser apreciados com facilidade operações que se realizam seqüencialmente, tal como as operações de montagem (primeiro C sobre A, e então B sobre CA), mas existem também aquelas que podem ser realizadas em paralelo, tal como p_4 , p_5 ou p_6 que realizam operações de processamento de partes diferentes, ou, o caso de mover uma peça "A" e/ou "C" para a célula E, enquanto se realiza a operação de processamento final

modelada por p18. Existem também casos de sincronização, tal como o fato de, para se fazer uma operação de montagem em p13, se requer que uma parte de "A" e "C" se encontrem disponíveis nas posições p7 e p8 respectivamente.

A tabela 6.2 dada a continuação mostra portanto uma descrição mais detalhada respeito dos lugares e transições da Rede de Petri final.

Lugar	Descrição	Tempo de execução
p1	Disponibilidade de matéria prima para partes C	
p2	Disponibilidade de matéria prima para partes A	
p3	Disponibilidade de matéria prima para partes B	
p4	Processamento de uma parte C	3
p5	Processamento de uma parte A	5
p6	Processamento de uma parte B	4
p7	Disponibilidade de uma parte C no buffer B3	
p8	Disponibilidade de uma parte A no buffer B1	
p9	Disponibilidade de uma parte B no buffer B2	
p10	R2 coloca uma parte C na célula de montagem	
p11	R1 coloca uma parte A na célula de montagem	
p12	R2 coloca uma parte B na célula de montagem	
p13	Disponibilidade de célula M4 com uma parte A	
p14	R1 e R2 montando uma parte C sobre A	3
p15	Disponibilidade da célula M4 com conjunto CA	
p16	R1 e R2 montando uma parte B sobre CA	3
p17	Mover conjunto montado BCA para buffer B4	
p18	Disponibilidade de B4 com conjunto BCA	
p19	M5 realizando processamento final sobre BCA	4
p20	Disponibilidade de máquina 3 (M3)	
p21	Disponibilidade de máquina 1 (M1)	
p22	Disponibilidade de máquina 2 (M2)	
p23	Disponibilidade de máquina 5 (M5)	
p24	Disponibilidade de buffer B3 vazio	
p25	Disponibilidade de buffer B1 vazio	
p26	Disponibilidade de buffer B2 vazio	
p27	Disponibilidade de buffer B4 vazio	
p28	Disponibilidade de célula de montagem M4 vazia	
p29	Disponibilidade de robô 2 (R2)	
p30	Disponibilidade de robô 1 (R1)	

Tab.6.2.- Significado dos Lugares na Rede de Petri final

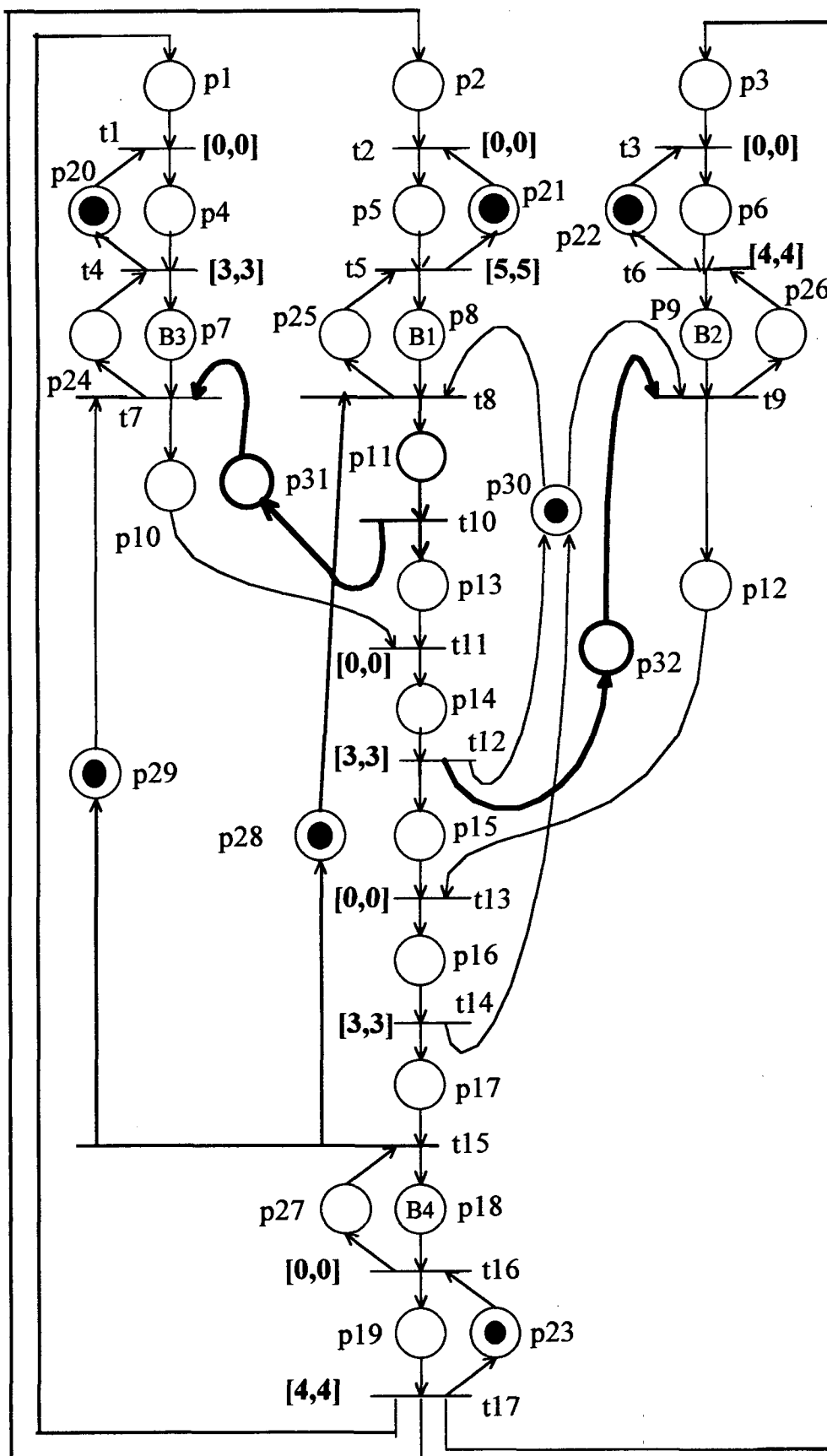


Fig. 6.7.- Modelo de Rede de Petri final para o Sistema de Montagem

b) Análise

Sendo que a Rede final da figura 6.7, sem as posições p_{29} e p_{30} , tem as mesmas propriedades de limitação, vivacidade e reversibilidade que a rede da figura 6.6, então a análise será feita a partir desta última; a qual pertence a um grafo de eventos. Este fato é garantido pelo teorema 1.

Portanto o número de L-invariantes segundo a equação 3.23, será igual a $N_L - N_T + 1$ que é igual a 12; os mesmos que são listados na tabela 6.3.

L-invariante	Lugares	Período
X1	p10,p13,p14,p15,p16,p17,p4,p7,p18,p1	13
X2	p11,p13,p14,p15,p16,p19,	6
X3	p11,p13,p14,p15,p16,p17,p5,p8,p18,p2	15*
X4	p4,p20	3
X5	p5,p21	5
X6	p6,p22	4
X7	p18,p23	4
X8	p7,p24	0
X9	p8,p25	0
X10	p9,p26	0
X11	p17,p27	0
X12	p12,p15,p16,p17,p6,p9,p18,p3	11

Tab. 6.3.- L-Invariantes da Rede de Petri para a Rede da figura 6.6

Segundo Ranchandani, um total de 12 circuitos elementares podem ser identificados; os mesmos que correspondem ao número de L-invariantes. Se consideramos um estado inicial para os lugares (pallet) que inicializam o funcionamento do sistema com $M_0(p_1) = M_0(p_2) = M_0(p_3) = 1$, o período de tempo (tempo de ciclo) para cada circuito utilizando a equação 5.1 é mostrado na tabela 6.3. Observando que o período máximo de tempo corresponde ao circuito c_3 (circuito de processamento) com 15 unidades de tempo, este circuito será, portanto, um circuito crítico, representando o tempo de ciclo do sistema.

Devido ao fato que em um "grafo a eventos", o limite superior da produtividade é determinado pela condição de plena utilização da *máquina gargalo*, ou seja, da máquina com maior tempo de processamento das tarefas que compõem sua seqüência no circuito crítico (neste caso o circuito c_3

consta de três operações com tempos de 5, 6 e 4 unidades de tempo respectivamente), então temos que o tempo de ciclo anterior de 15 unidades de tempo para a marcação inicial dada não é o ótimo. Esse tempo pode ser diminuído para 6 unidades que corresponde à operação de montagem, tarefa que tem o maior tempo de processamento no circuito crítico e que corresponde a um circuito de comando (equação 5.5). Portanto, para aumentar a produtividade do sistema teremos que diminuir o tempo de ciclo de 15 para 6 unidades, o que é feito incrementando o número de fichas em cada circuito de processamento, utilizando a equação 5.1 até se obter um valor para TC igual ou menor à operação crítica (operação de montagem), tornando-a assim uma operação gargalo. Para evitar essa penosa simulação por tentativas, podemos proceder então para o cálculo do número mínimo de fichas requerido em cada circuito de processamento (tendo como objetivo, minimizar o número de "pallets" com plena utilização da máquina gargalo), utilizar diretamente a equação 5.10. Dado que o tempo de ciclo crítico nos é conhecido (atividade com o maior tempo de processamento), o mesmo que nos permitirá determinar o número mínimo requerido de fichas em cada circuito cp_i , e nos garante, ao mesmo tempo, obter a máxima produtividade do sistema com plena utilização da máquina (ou operação) gargalo. Da equação 5.10 nós temos:

$$M_0(\gamma) = NF(cp_i) = \frac{\text{Periodo de tempo do circuito } u(\gamma)}{\text{tempo da operação gargalo do circuito crítico } T_0} \quad (6.4)$$

Os resultados são mostrados na terceira coluna da tabela 6.4, onde podemos apreciar que a marcação inicial obtida (ótima) é igual a 3 fichas para p_1 e p_2 , e 2 fichas para p_3 ; sendo, portanto, esta marcação aquela que nos permite obter a máxima produtividade com plena utilização da máquina gargalo e o mínimo de inventário em processo, já que se nós mudarmos esta marcação para menos, em qualquer um dos três lugares, estaremos diminuindo a taxa de produtividade do sistema; e se mudarmos este número de fichas iniciais para mais, estaremos incrementando o tamanho dos Buffers intermediários na proporção direta ao número de fichas, conforme pode ser visto na tabela 6.4. Estes valores são obtidos dividindo os valores da segunda coluna, que correspondem a um circuito de processamento (c_1 , c_3 e c_{12}), entre 6, que é o tempo que corresponde à operação gargalo do circuito de processamento c_3 (circuito crítico).

Voltando a utilizar a equação 5.1, obtemos então o novo tempo de ciclo para cada circuito elementar como mostrado na quarta fila da tabela 6.5, onde o tempo de ciclo máximo neste caso segundo a equação 5.4 corresponderá ao circuito c_2 , que é um circuito de comando (cc), o que significará portanto que a máquina gargalo será utilizada plenamente (Cury 1990).

Assim, a produtividade máxima do sistema em regime permanente será:

$$f = \frac{1}{T} = \frac{1}{6}$$

isto é, que uma unidade de produto será produzida a cada 6 unidades de tempo, sendo a eficiência para cada circuito (equação 6.2), assim como para o sistema total (equação 6.3) como mostrado na tabela 6.4.

No caso de um sistema job-shop, o denominador da equação 6.3 será igual ao circuito de comando que possua o maior tempo, garantindo assim a plena utilização da máquina gargalo. No exemplo mostrado no final do capítulo 6, teremos então, que o número mínimo requerido de fichas no circuito de processamento de P2 será de $11/6 = 1,83 \cong 2$, conforme visto em Cury (1990).

Marcação inicial	T	f	Eficiência				tamanho do buffer
			cp1	cp2	cp3	sistema	
p1, p2, p3	15	1/15	0,29	0,33	0,24	0,29	2B1, 1B2, 1B3
2*p1, p2, p3	15	1/15	0,29	0,33	0,24	0,29	2B1, 1B2, 1B3
p1, 2*p2, p3	13	1/13	0,33	0,38	0,28	0,33	2B1, 1B2, 1B3
2*p1, 2*p2, p3	11	1/11	0,39	0,45	0,33	0,39	2B1, 1B2, 1B3
2*p1, 2*p2, 2*p3	7,5	1/7,5	0,58	0,67	0,49	0,58	2B1, 1B2, 1B3
2*p1, 3*p2, 2*p3	6,5	1/6,5	0,66	0,77	0,56	0,66	2B1, 1B2, 1B3
3*p1, 3*p2, 2*p3	6	1/6	0,72	0,83	0,61	0,72	2B1, 1B2, 1B3
3*p1, 2*p2, 2*p3	6,5	1/7,5	0,58	0,67	0,49	0,58	2B1, 1B2, 1B3
3*p1, 3*p2, p1	11	1/11	0,39	0,45	0,33	0,39	2B1, 1B2, 1B3
4*p1, 3*p2, 2*p3	6	1/6	0,72	0,83	0,61	0,72	2B1, 1B2, 2*B3
3*p1, 4*p2, 2*p3	6	1/6	0,72	0,83	0,61	0,72	2*B1, 1B2, 1B3
4*p1, 3*p2, 3*p3	6	1/6	0,72	0,83	0,61	0,72	2B1, 2*B2, 2B3

Tab.6.4.- Tempo de Ciclo para cada Circuito da Rede da figura 6.6

Circuito	Período	NF(c _i)	TC(c _i)
c1	13	3*	4.33
c2	6	1	6*
c3	15	3*	5
c4	3	1	3
c5	5	1	5
c6	4	1	4
c7	4	1	4
c8	0	0	0
c9	0	0	0
c10	0	0	0
c11	0	0	0
c12	11	2*	5.5

Tab. 6.5.- Tempo de Ciclo para cada Circuito elementar

A determinação das propriedades: básicas é dada como segue a continuação:

Vivacidade.- Devido a que cada um dos lugares da rede pertence a pelo menos um invariante de transição, então a rede é viva, portanto livre de deadlock.

Limitação.- Para um valor de $T = 6$, a rede é K-limitada para os lugares p_1, p_2 e p_3 , com três fichas para p_1 e p_2 , e duas fichas para p_3 respectivamente, sendo que se nós considerarmos como marcação inicial aquela marcação obtida após 15 unidades de tempo por exemplo, ela será 1-limitada, ou seja, salva ou segura.

Reversibilidade.- Já que só existe um T-invariante na rede, que contém todas as transições, isto indica que o sistema tem um comportamento cíclico realizando assim suas funções repetidamente

Sendo portanto a rede viva, limitada e reversível, o funcionamento normal do sistema é garantido.

b.1) Análise da Rede Final Utilizando o Analisador ARP

Utilizando o analisador de Redes de Petri (ARP) nós obtemos os resultados mostrados a seguir, os quais são obtidos a partir da Rede final (figura 6.7).

a) Cálculo dos Invariantes

Através do módulo de análise estrutural do ARP podemos verificar que o modelo do sistema possui somente um invariante de transição, contendo todas as transições da rede, portanto a rede é viva.

Na análise são obtidas 23 invariantes de lugar. Todos os lugares da rede pertencem a algum invariante, indicando assim que a rede é limitada.

A lista dos invariantes de lugar é dada a seguir, classificados de acordo com sua interpretação física.

Invariantes de comando das máquinas (circuitos de comando):

X1 : {p20, p4}	máquina 2
X2 : {p21, p5}	máquina 1
X3 : {p22, p6}	máquina 3
X4 : {p23, p19}	máquina 5
X5 : {p30, p11, p13, p14, p12, p16}	máquina 4 (funcionamento do Robô 1)
X6 : {p29, p10, p14, p15, p16, p17}	máquina 4 (funcionamento do Robô 2)
X7 : {p11, p12, p13, p14, p15, p16, p17, p28}	máquina 4 (ocupação da célula)

Invariantes de ocupação dos Buffer:

X8 : {p24, p7}	,	X10 : {p26, p9}
X9 : {p25, p8}	,	X11 : {p27, p18}

Invariantes do fluxo das peças (circuitos de processo):

X12 : {p2, p5, p8, p11, p13, p14, p15, p16, p17, p18, p19}

X13 : {p1, p4, p7, p10, p14, p15, p16, p17, p18, p19}

X14: {p3, p6, p9, p12, p16, p17, p18, p19}

Invariantes mistos (circuitos mistos)

X15 : {p2, p5, p8, p11, p12, p13, p14, p16, p17, p18, p19, p32}

X16 : {p10, p11, p12, p14, p16, p17, p28, p31, p32}

X17 : {p2, p5, p8, p10, p11, p12, p14, p16, p17, p18, p19, p31, p32}

X18 : {p1, p4, p7, p10, p12, p14, p16, p17, p18, p19, p32}

X19 : {p2, p5, p8, p10, p11, p14, p15, p16, p17, p18, p19, p31}

X20 : {p11, p12, p13, p14, p16, p17, p28, p32}

X21 : {p10, p11, p14, p15, p16, p17, p28, p31}

X22 : {p10, p12, p14, p16, p17, p29, p32}

X23 ; {p10, p11, p12, p14, p16, p30, p31}

b) Avaliação do Desempenho

O processo de otimização utilizado pelo ARP e o proposto em Hillion (1989), baseia-se no estudo das durações (tempos de ciclo) associados aos circuitos da rede que modela o sistema. Essas durações são função da marcação do circuito e das durações que o compõem (Maziero 1990).

O circuito de comando com o máximo tempo de ciclo é denominado de "circuito crítico", e deve corresponder ao da máquina gargalo (M4) para a máxima taxa de produtividade em regime permanente.

Assim temos, via simulação que para a marcação inicial em p_1 , p_2 e p_3 (que correspondem aos circuitos de processo), igual a 3, 3 e 2 fichas respectivamente, o tempo de ciclo mínimo é de 6 unidades; sendo portanto esta marcação a quantidade mínima de fichas que tornam os tempos de ciclo dos demais circuitos inferiores ao tempo de ciclo da máquina gargalo. Por outro lado o tempo de ciclo máximo é aquele que corresponde para a marcação de $p_1 = p_2 = p_3 = 1$, que é de 15 unidades, conforme visto nos resultados obtidos com o ARP, os quais são mostrados no apêndice. Outros tempos de ciclo para diversas marcações são mostrados na tabela 6.4.

c) Verificação das Propriedades

Para a marcação inicial:

$$M_0 = (3 \cdot p_1, 3 \cdot p_2, 2 \cdot p_3, p_{20}, p_{21}, p_{22}, p_{23}, p_{24}, p_{25}, p_{26}, p_{27}, p_{28}, p_{29}, p_{30}),$$

a rede em análise é viva, limitada e reversível, conforme pode ser visto nos resultados obtidos (ver apêndice). A respeito da limitação, o lugar p_1 e p_2 são 3-limitados, o lugar p_3 é 2-limitado e o resto dos lugares são binários. Por outro lado, se nós considerarmos como marcação inicial uma de suas sucessoras, teríamos que todos os seus lugares serão binários. Por exemplo a marcação

$$M'_0 = (p_5, p_7, p_9, p_{15}, p_{19}, p_{20}, p_{22}, p_{25}, p_{27}, p_{30}, p_{32}).$$

6.3.- Conclusões

No presente capítulo podemos observar a facilidade das Redes de Petri para representar, analisar e avaliar a eficiência (desempenho) de sistemas discretos de manufatura, como parte de um processo de otimização do funcionamento do mesmo. Situações como paralelismo, concorrência, sequenciamento, (o qual foi solucionado com o uso dos lugares fantasmas), conflito e sincronização aparecem claramente neste tipo de redes. Apresenta-se também a utilização do analisador ARP na determinação dos circuitos elementares do sistema de montagem.

Conforme visto no processo de modelagem a metodologia proposta por Zhou (1992) é de grande utilidade, já que nos poupa do cálculo das propriedades.



Embora este exemplo esteja voltado para um assunto específico, deve-se ressaltar a importância de se utilizar as Redes de Petri para modelar sistemas complexos como é o caso dos FMS, e outros sistemas complexos em geral.

CAPÍTULO 7 - CONCLUSÕES E RECOMENDAÇÕES

7.1.- Conclusões

Neste trabalho foram apresentadas as Redes de Petri e algumas de suas extensões como uma ferramenta para a representação, análise e controle de sistemas automatizados de Manufatura. Foi abordada também uma metodologia de síntese desenvolvida por Zhou (1992.a) para modelar sistemas de Manufatura automatizados, que se aplicada corretamente, pode evitar o cálculo das propriedades básicas como vivacidade, limitação e reversibilidade já que estas propriedades tornam-se inerentes à Rede final obtida. Além disso, a complexidade própria deste tipo de sistemas pode ser manuseada com muita facilidade.

A aplicação feita no capítulo 6 a uma linha de montagem utilizando uma extensão de RdP (chamada RdP Temporal), demonstra a sua facilidade para modelar e analisar Sistemas de Manufatura. O tempo de ciclo que é um fator importante no funcionamento de todo Sistema Produtivo, é achado de uma maneira muito fácil, através da decomposição da Rede em circuitos elementares. Assim, a obtenção do tempo mínimo de ciclo (que corresponde a um circuito de comando) nos garante que o sistema opera com a máxima produtividade e o mínimo de inventário em processo.

Podemos portanto concluir que, a facilidade das Redes de Petri para representar situações complexas, associadas com a grande flexibilidade que este tipo de Rede apresenta, torna-a uma ferramenta muito útil para modelar, analisar e controlar sistemas complexos de manufatura.

Se comparada com outros modelos gráficos de comportamento dinâmico, tais como as máquinas de estados finitas ou Redes Pert, as Redes de Petri oferecem muita facilidade para expressar o comportamento de sistemas que são assíncronos e distribuídos. além do mais, as restrições de precedência entre operações, ligações livres associadas com recursos compartilhados

podem também ser expressado diretamente assim como a repetição de certas operações (ou seqüência de operações) que formam ciclos e que são próprios de um FMS.

Como visto no capítulo 4, existe uma grande variedade de abordagens, todas baseadas em Redes de Petri, mas com diferenças nas maneiras de associar informações às fichas e de representar o tempo; entre elas podemos citar as Redes de Petri Coloridas (Jensen 1981), Redes de Petri Predicado/Transição (Genrich 1981, 1986), Redes de Petri estendida (Valavanis 1990), Redes de Petri estado-Transição (Chen 1992), Mark Flow Graph estendido (dos Santos 1993), Redes de Petri de alto Nível (Genrich 1981), (Argenta 1987), etc. Cada uma apresentando as suas vantagens e desvantagens, e a escolha portanto, para sua utilização em Sistemas de Manufatura torna-se muito difícil. Esta falta de padronização das Redes de Petri é portanto um obstáculo para um desenvolvimento mais rápido das RdPs na Indústria (Valette 1990).

Redes de Petri são também muito usadas para o desenho e detecção de conflitos de Veículos guiados automaticamente (AGV) (Raju 1993), (Zeng 1991).

As Redes de Petri em geral, como todos os demais modelos existentes apresentam uma série de vantagens e desvantagens que podem ser resumidas a seguir:

a) Vantagens

- i) Modelos de Redes de Petri representam uma ferramenta de modelagem hierárquica com uma matemática bem definida e fundamento prático;
- ii) Deadlock, conflitos e tamanhos de Buffers podem ser modelados de uma maneira fácil e eficiente;
- iii) Redes de Petri nos permitem representar sistemas, seguindo uma abordagem top-down, assim como bottom-up em diferentes níveis de abstração e facilitando a sua decomposição em subsistemas funcionais, com uma clara interrelação entre estes subsistemas;

- iv) Redes de Petri permitem métodos de análise bem definidos e poderosos (no que se refere à quantidade de informações fornecidas) para suas extensões mais simples;
- v) Na fase de escalonamento da produção (alocação de operações a cada máquina), as Redes de Petri adaptam-se muito bem quando se estudam regimes estacionários com alimentação periódica (Cury 1990);
- vi) Redes de Petri satisfazem as necessidades de estado parcial, decomposição de sistemas, simplicidade, sincronização, concorrência entre tarefas, assim como facilitam a sua compreensão;
- vii) Pode-se descrever uma ordem parcial entre eventos, o que permite levar em conta a flexibilidade;
- viii) A grande versatilidade das RdPs as torna bem apropriadas para representar unidades de recursos físicos se comparadas com os grafos PERT e GANT.

b) Desvantagens

- i) O maior problema que apresentam as Redes de Petri é o fato de que quando é requerido uma modelagem mais detalhada, o modelo tende a ser muito grande (explosão combinatória de estados) portanto, a sua análise torná-se complicada. Embora muitas informações possam ser obtidas através da árvore de alcançabilidade, se torna impraticável para sistemas complexos, como é o caso dos FMSs, uma vez que o número de marcações alcançáveis tende a ser muito grande, (ver apêndice).
- ii) Redes de Petri são uma ferramenta passiva porém não ativa, isto é, elas servem para detectar situações contrárias ao bom funcionamento de um sistema, mas não podem gerar uma solução.

7.2.- Recomendações para futuros trabalhos

Neste trabalho, tratou-se da utilização das Redes de Petri para modelar, analisar e controlar sistemas discretos de manufatura, podendo observar que tais Redes são bem apropriadas

para representar este tipo de sistemas. Seria, portanto, interessante fazer um estudo da utilização deste tipo de redes para representar sistemas contínuos (Indústrias de Processos), tendo como referências os trabalhos de (Tazza 1987) e (Alla 1987).

A falta de padronização das Redes de Petri para representar sistemas Flexíveis de Manufatura é, na atualidade, um obstáculo para sua utilização na indústria (Valette 1990). Neste sentido propomos a criação de um único modelo de Redes de Petri que aproveite bem as vantagens que apresentam os diferentes modelos até agora existentes e estabelecer assim um modelo padrão capaz de representar de forma eficiente os Sistemas de Manufatura.

É recomendável estudar as Redes de Petri assim como outros modelos usados para determinar a performance de sistemas e ver onde, e em que situações certo modelo é mais adequado que outro, visando também a utilização conjunta de Redes de Petri com outros modelos para obter um modelo mais preciso; tendo como base os trabalhos de Balbo (1988), Cao (1990), Viswanadham (1987) e Miyagi (1989).

Uma outra sugestão seria utilizar as redes de Petri para solucionar problemas de sequenciamento. Por exemplo, no caso quando vários produtos são processados por diferentes máquinas (Cury 1990), ou quando um operário atende diversas máquinas simultaneamente (Chen 1992); enfim, qualquer problema de sequenciamento pode ser solucionado eficientemente.

Referências Bibliográficas

Al-JAAR, Robert Y. And DESROCHES, Alan A., A survey of Petri Nets in automated Manufacturing System. Proc. 12th IMACS World Congress on Scientific Computation, vol.2, pp. 503-510, (1988).

Al-JAAR, Robert Y. And DESROCHES, Alan A., Performance Evaluation of Automated Manufacturing Systems Using Generalized Stochastic Petri Nets. IEEE Transaction on Robotics and Automation, vol. 6, n. 6, pp. 621-639, december (1990).

AGERVALA, Tilak, Putting Petri Nets to Work. IEEE Transaction on Computer, vol. 12, n. 12, pp. 85-94, Dec. (1979).

ALLA, Hassane and DAVID, René, Modelling of Production Systems by Continuous Petri Nets. 3rd Int. Con. on CAD/CAM Robotics and Factories of the Future, Sothfield (USA), pp. 344348, (1988).

ARGENTA, Ilson Luiz, Um Método de Modelagem de sistemas Discretos de Manufatura com Redes de Petri de alto Nivel. XX Congresso Nacional de Informatica, SUCESU- São Paulo, pp. 615-623, (1987).

BALBO, Gianfranco; BRUELL, Steven, C. and GHANTA, Subbarao, Combining Queueing Ntworks and Generalized Stochastic Petri Nets for the Solution of Complex Models of System Behavior. IEEE Transaction on Computer, vol. 37, n. 10, pp. 1251-1268, october (1988).

BANASZAK, Zbignew, A. and KROGH, Bruce, H., Deadlock Avoidance in Flexible Manufacturing Systems with Concurrently Competing Process Flows. IEEE Transactions on Robotics and Automation, vol. 6, n. 6, december (1990)

BARAD, Miryam And SIPPER, Daniel, Flexibility in Manufacturing System: Definitions and Petri Net Modelling. International Journal of Production Research, vol. 26, n. 2, pp. 237-248, (1988).

BERTHELOT, Gérard and TERRAT, Richard, Petri Nets Theory for the Correctness of protocols. IEEE Transacción on Comunicatións, vol. COM-30, n. 12, dec. (1982).

BRUNO, Giorgio and MARCHETO, Giuseppe, Process-Translatable Petri Nets for the Rapid Prototyping of Process Control System. IEEE Transaction on Software Engineering, vol. SE-12, n. 2, Feb. (1986).

CAO, Xi-Ren and HO, Yu-Chi, Models of Discrete Event Dynamic Systems. IEEE Control System Magazine, pp. 69-76, june (1990).

CARDOSO, Janette; FARINES, Jean-Marie e CURY, Jose, Eduardo, Controle de Sistemas de Manufatura. Apostilas do Curso de Pós-Graduação em Automação Industrial, a nível de especialização, Universidade Federal de Santa Catarina & Universidade de Caxias do Sul, setembro (1994).

CASSANDRAS, Christos, G. and RAMADGE, Peter, J., Toward a Control Theory for Discrete Event System. IEEE Control Systems Magazine, pp. 66-68, june (1990).

CENIP, Administración de la Producción. Copias del X curso de Especialistas en Productividad, arequipa-Perú, (1988).

COHEN, Guy e colegas, Algebraic Tools for the Performance Evaluation of Discrete Event Systems. Proceedings of the IEEE, vol. 77, n. 1, pp. 39-57, january (1989).

CURY, Jose, Eduardo, R., Lançamento e Escalonamento de Tarefas em Sistemas de Manufatura com Alimentação Periódica. Anais do 8º Congresso Brasileiro de Automação, Belém-Pa, pp.1165-1171, (1990).

CHEN, Houn-Gee, A Petri-Net-Based State-Transition model for an optimal operator cyclic walking pattern development in GT cells. International Journal of Production Research, vol.30, n. 10, pp. 2381-2399, (1992).

CHIAVENATO, Idalberto, Introdução a Teoria Geral da Administração. 3ra Edição, São Paulo, McGraw-Hill do Brasil, (1983).

CHIAVENATO, Idalberto, Iniciação à Administração da Produção. Editora MacGraw-Hill Ltda, São Paulo (1991).

DAVID, René and ALLA, Hassane, Petri Nets for Modelling of Dynamic System- A Survey. Automática, vol. 30, n. 2, pp. 175-202, (1994).

Di MASCOLO, Maria; DAVID, Rene; DALLERY, Yves, Modelling and Analysis of Assembly System with Unreable Machines and Finite Buffers. IIE Transactions, vol. 23, n. 4, pp. 315-330, dec. (1991).

Dos SANTOS FILHO, Diolino, Jose, Proposta do Mark Flow Graph Estendido para a Modelagem e Controle de Sistemas Integrados de Manufatura. Dissertação de Mestrado apresentada à Escola Politécnica da USP, São Paulo, (1993).

DUBOIS, Didier and STECKE, Katryn E., Using Petri Nets to Represent Production Processes. In Proceeding of the 22nd, IEEE Conference on decision and Control, pp. 1062-1067, San Francisco-Texas, (1983).

ELMAGHRABY, Salah, E., The Design of Production Systems. Reinhold Publishing Corporation, New York, (1966).

FAVREL, Joel and LEE, Kwang, H., Modelling, Analyzing, Scheduling and Control of Flexible Manufacturing System by Petri Nets. in IFIP Conf. Production System, Copenhagen, Amsterdam: North-Holland, pp. 223- 243, aug. (1984).

FREEDMAN, Paul, Time, Petri Nets, and Robotics, IEEE Transaction on Robotics and Automation, vol. 7, n. 4, pp. 417-433, august (1991).

GENRICH, Harmann J., Predicate/Transition Nets. Lecture in Computer Science, n. 254, pp. 207-247, (1986).

GENRICH, H. J. And LAUNTEBACH, K., System Modelling with High-Level Petri Nets. Theoretical Computer Science, vol. 13, pp. 109-136, (1981).

GERSHWIN, S., B., Hierarchical Flow Control: a Framework for Scheduling and Planning Discrete Events in Manufacturing systems. IEEE Proceedings Special Issue on Discrete Event Systems, vol. 77, n. 1, pp. 195-209, january (1989)

GOMES de Alcantara, Carlos Augusto, Contribuição à Modelagem e Análise de Sistemas de Produção Utilizando Redes de Petri. Tese de Doutorado apresentada à UFRJ, Rio de Janeiro, março (1990).

GREENE, James, Control de la Producción y Decisiones. Editora Diana, Mexico, (1968).

GRISLAIN, Jacques A. and PUN, Lucas, Graphical methods for Production Control. Int. J. Prod. Res., vol. 17, n. 6, pp. 643-659, (1979).

GUARDA, Alvaro, Ferramenta para Apoio a Modelagem De Sistemas com Redes de Petri. Dissertação de Mestrado apresentada à Universidade Federal do Rio Grande do Sul, Porto Alegre, novembro (1989).

HACK, M., H., T., Analysis of Production Schemata by Petri Nets. Master of Science Dissertation Thesis, MIT, february (1972).

HATONO, Itsuo e colegas, Modeling and On-Line Scheduling of Flexible Manufacturing Systems Using Stochastic Petri Nets. IEEE Transaction on Software Engineering, vol. 17, n. 2, pp. 126-132, february (1991).

HILLION, Hervé P. And PROTH, Jean-Marie, Performance Evaluation of Job-shop System Using Timed Event-Graphs. IEEE Transaction on Automatic Control, vol. 34, n. 1, pp. 3-9, (1989).

HO, Y., C., Scanning the Issue: Dynamics of Discrete Event Systems. Proceedings of the IEEE, vol. 77, n. 1, pp. 3-6, january (1989).

HUANG, Han-Pang and CHANG, Po-Chu, Specification, modelling and control of a flexible manufacturing cell. Int. J. Prod. Res., vol. 30, n. 11, pp. 2515-2543, (1992).

JENSEN, Kurt, Coloured Petri Nets and the invariant-Method. Theoretical Computer Science, vol. 14, pp. 317-336, (1981).

JENSEN, Kurt, Coloured Petri Net. Lecture notes in Computer Science, n. 254, pp. 248-299, (1986).

KAMATH, Manjunath and VISWANADHAM, N., Applications of Petri Net Based Model in the Modelling and Analysis of Flexible Manufacturing System. In Proceeding of the IEEE International Conference on Robotic and Automation, San Francisco, California, pp. 312-317, (1986).

KUNZLE, L., A., Controle de Sistemas Flexíveis de Manufatura- Especificação dos Níveis Equipamento Estação de Trabalho. Dissertação de Mestrado-CEFET, Curitiba, (1990).

LEE, Kwang-Hyung and FAVREL, Joel, Hierarchical Reduction Method for Analysis and Decomposition of Petri Nets. IEEE Transaction on System, Man, and Cybernetics, vol. SMC-15, n. 2, pp. 272-280, march/april (1985).

LEE, Kwang-Hyung; FAVREL, Joel and BAPTISTE, Pierre, Generalized Petri Net Reduction Method. IEEE Transaction on System, Man, and Cybernetics, vol. SMC-17, n. 2, pp. 297-303, march/april (1987).

LEUNG, Ying-Tat and SURI, Rajan, Performance Evaluation of Discrete Manufacturing System. IEEE Control System Magazine, pp. 77-86, june (1990).

LORINI, Flávio, José, Tecnologia de Grupo e Organização da Manufatura. Editora da UFSC (1993).

MAZIERO, Carlos Alberto, Um Ambiente para a Análise e simulação de Sistemas Modelados por Redes de Petri. Dissertação de Mestrado, DEEL-UFSC, junho (1990).

MIYAGI, P., E.; FURUKAWA, C., M.; MARUYAMA, N. and TSUZUKI, F., S., G., Sistemas de Manufatura: Tendencias em Modelagem, Análise e Projeto. X Congresso Brasileiro de engenharia Mecânica, (10° COBEM), Rio de Janeiro, RJ, pp. 491-494, dezembro (1989).

MOLLOY, Michael, K., Performance Analysis Using Stochastic Petri Nets. IEEE Transaction on Computers, vol. C-31, n. 9, september (1982).

MOLLOY, Michael, K., Discrete Time Stochastic Petri Nets. IEEE Transactions on Software Engineering, vol. SE-11, n. 4, april (1985).

MOREIRA, Daniel A., Administração da Produção e Operações. São Paulo: Pioneira, (1993).

MURATA, Tadao, Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, vol. 77, n. 4, pp. 541-580, april (1989).

PETERSON, J. L., Petri Net Theory and the modelling of system. Prentice-Hall Editions, (1981).

RAMAMOORTHY, C. V. and HO, Gary S., Performance Evaluation of Asynchronous Concurrent System Using Petri Nets. IEEE Transaction on Software Engineering, vol. SE-6, n. 5, pp. 440-449, (1980).

RAJU, K. and CHETTY, O. V. Krishnaiah, Design and evaluation of automated guided vehicle system for flexible manufacturing system: and extended Timed Petri Net-based approach. Int. J. Prod. Res., vol. 31, n. 5, pp. 1069-1096, (1993).

RIGHINI, G., Modular Petri Nets for Simulation of Flexible Production System. Int. J. Prod. Res., vol.31, n. 10, pp. 2463-2477, (1993)

RIGGS, James L., Sistemas de producción. Editorial LIMUSA, S. A. de C. V., Mexico D. F., décima impresión (1993).

RILLO, Marcio, Aplicações de Redes de Petri em sistemas de Manufatura. Tese de Doutorado apresentada à Escola Politécnica da USP, São Paulo, (1988).

SILVA, Manuel and VALETTE, Robert, Petri Nets and Flexible Manufacturing, Lecture Notes in Computer Science, pp. 374-417, (1989).

STAFFORD, Joseph, Systems Analysis for Engineers and Managers. McGraw-Hill, New York, (1971).

TAZZA, Maurizio; ARGENTA, Ilson, L.; TAVARES, Antonio, C., Modelagem e análise de Industrias de Processo. XX Congresso Nacional de Informatica, SUCESU-São Paulo, pp. 563-572, (1987).

VALAVANIS, Kimon, P., On the Hierarchical Modelling Analysis and Simulation of Flexible Manufacturing System with Extended Petri Nets. IEEE Transactions on System, man, and Cybernetics, vol. 20, n. 1, pp. 94-110, january/february (1990).

VALETTE, Robert, Nets in production System. Lecture Notes in Computer Science 255, Springer Verlag, pp. 191-217, (1986).

VALETTE, Robert and SILVA, Manuel, A Rede de Petri: Uma Ferramente para a Automação Fabril. Anais do 4º Congresso Nacional de Automação Industrial, São Paulo, (1990).

VISWANADHAM, N. and NARAHARI, Y., Coloured Petri Net Model for Automated Manufacturing System. Proceeding of the IEEE International Conference on Robotic and Automation, Raleigh, North Carolina, march 30th- april 3rd, pp. 1985-1990, (1987).

VISWANADHAM, N.; NARAHARI, Y.; and JHONSON, Timothy, L., Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing System Using Petri Net Model. IEEE Transaction on Robotics and Automation, vol. 6, n. 6, pp. 713-723, december (1990).

ZENG, Laiguang; WANG, Hsu-Pin(Ben) and JIN Song, Conflict detection of automated guided vehicles: A Petri Net approach. Int. J. Prod. Res., vol. 29, n. 5, pp. 865-879, (1991).

ZHOU, Meng, Chu and DiCESARE, Frank, Adaptative Design of Petri Net Controllers for Error Recovery in Automated Manufacturing Systems. IEEE Transactions on Systems, Man, and Cybernetics , vol. 19, n. 5, pp. 963-973, september/october (1989).

ZHOU, Meng-Chu and DICESARE, Frank, Parallel and Sequential Mutual Exclusions for Petri Net Modelling of Manufacturing System with Shared Resources. IEEE Transactions on Robotics and Automation, vol. 7, n. 4, pp. 515-527, august (1991).

ZHOU, Meng-Chu, DiCESARE, Frank and DESROCHERS, Alan A., A Hybrid Methodology for Syntesis of Petri Net Models for Manufacturing Systems. IEEE Transaction on Robotics and Automation, vol. 8. n. 3, pp. 350-361, june (1992).

ZHOU, Meng-Chu; DiCESARE, Frank and RUDOLPH, Daryl L., Design and Implementation of a Petri Net based Supervisor for a Flexible Manufacturing System. Automatica, vol. 28, n. 6, pp. 1191-1208, (1992).

BIBLIOGRAFIA

BACCELLI, François and LIU, Zhen, Comparison Properties of Stochastic Decision Free Petri Nets. IEEE Transaction on Automatic Control, vol. 37, n. 12, pp. 1905-1920, december (1992).

BARBALHO, David, S.; Dos SANTOS, Mauro, R.; BARROS, Tomas, de Carvalho, Uma Técnica estruturada de modelagem por Redes de Petri: Função coordenação de sistemas de Transporte, Anais do 8° CBA-UFPA, Belém-Pa, pp. 1158-1164, (1990).

BRAND, Klaus-Peter and KOPAINSKY, Jurgen, Principles and Engineering of Process Control with Petri Nets. IEEE Transactions on Automatic Control, vol. 33, n. 2, pp. 138-149, february (1988).

CARRIE, A. S.; ADHAM, E.; STEPHENS, A. And MURDOCH, I. C., Introducing a Flexible Manufacturing System. Int. J. Prod. Res., vol. 22, n. 6, pp. 907-916, (1984).

CHAN, Chu-Chai and WANG, Hsu-Pin, Design and development of a Stochastic high-level Petri Net system for FMS performance evaluation. International Journal of Production Research, vol. 31, n. 10, pp. 2415-2439, (1993).

CHIOLA, Giovanni; MARSAN, Marco, A.; BALBO, Gianfranco and CONTE, Gianni, Generalized Stochastic Petri Nets: A Definition at the Net Level and Its Implication. IEEE Transaction on Software Engineering, vol. 19, n. 2, February (1993).

HEUSER, Carlos, Alberto, Análise Estruturada de Sistemas com Redes de Petri. XX Congresso Nacional de Informatica, SUCESU-São Paulo, pp. 668-675, (1987).

KROGH, Bruce H. And HOLLOWAY, Lawrence E., Synthesis of Feedback Control Logic for Discrete Manufacturing System. International Federation of Automatic Control. Automatica, vol. , n. 4, pp. 641-651, (1991).

LAUTENBACH, K. and THIAGARAJAN, P. S., Analysis of a Resource Allocation problem Using Petri Net. 1ST European Conference on parallel and Distributed processing, Toulouse-France February 14,16 (1979).

LEVIS, Alexandre H.; MORAY, Nevile and HU, Baosheng, Task decomposition and Allocation Problems and Discrete Event Systems. Automatica, vol.30, n.2, pp. 302-216, (1994).

LIN, Chuang and MARINESCU, Dan, Cristian, Stochastic High-Level Petri Nets and Applications. IEEE Transaction on Computers, vol. 37, n. 7 pp. 815-825, july (1988).

MARSAN, M., Ajmone and CHIOLA, G., Modelling Discrete Event Systems with Stochastic Petri Nets. Proceeding of ISCAS, pp. 739-742, (1985).

MERLIN, Philip M., A Methodology for the Design and Implantation of communications of a theoretical Study. IEEE Transactions on Communications, vol. COM-24, n. 9, pp. 1036-1043, (1976).

MERLIN, Philip M. and FARBER, David, J., Recoverability of Communication Protocols- Implications of a Theoretical Study. IEEE Transactions on Communications, pp. 1036-1042, september (1976).

MIYAGI, Paulo, E., Proposta de uma linguagem de Programação para sistemas de Manufatura. Anais do 8º CBA-UFPa, Belém-PA

PROCK, J., A New Technique for fault Detection Using Petri Nets. Automatica, vol. 27, n. 2, pp. 231-245, (1991).

PROTH, Jean-Marie and XIE, Xiao-Lan, Cycle Time of Stochastic Event Graphs: Evaluation and Marking Optimization. IEEE Transaction on Automatic Control, vol. 39, n. 7, pp. 1482-1486, july (1994).

RAMASWAMY, S.; VALAVANIS, K. and LANDRY, S., P., Modelling, Analysis and Simulation of a Materials Handling System with Extended Petri Nets. Proceeding of the 31st Conference on Decision and Control, Tucson-Arizona, pp. 1665-1672, december (1992).

REDDY, C. Eswara; CHETTY, O. V. Krishnaiah and CHAUDHURI Dipak, A Petri Net based approach for analysing tool management issues in FMS. Int. J. Prod. Res., vol. 30, n. 6, pp. 1427-1446, (1992).

RILLO, Marcio, Controle de Sistemas de Manufatura por Redes de Petri e Regras de Produção. XX Congresso Nacional de Informatica, SUCESU-São Paulo, pp. 677-681, (1989).

RILLO, Márcio, Controlador Programável Utilizando Grafos de Comando etapa-Transição. Dissertação de Mestrado apresentada à escola politécnica da USP, São Paulo, (1983).

SAHRAOUI, A. e colegas, Joining Petri Nets and Knowledge Based Systems for Monitoring Purposes. IEEE International Conference on Robotics and Automation, Raleigh, NC, pp. 1160-1165, (1987).

SIFAKIS, Joseph, Structural properties of Petri Nets, Lecture Notes in Computer Science, vol. 64, Berling, pp. 474-483, (1978).

SILVA, M. and VELILLA, S., Error Detection and Correction on Petri Net Models of Discrete Events Control Systems. Proceeding of ISCAS, pp. 921-924, (1985)

Al-JAAR, Robert Y. And DESROCHES, Alan A., A survey of Petri Nets in automated Manufacturing System. Proc. 12th IMACS World Congress on Scientific Computation, vol.2, pp. 503-510, (1988).

Al-JAAR, Robert Y. And DESROCHES, Alan A., Performance Evaluation of Automated Manufacturing Systems Using Generalized Stochastic Petri Nets. IEEE Transaction on Robotics and Automation, vol. 6, n. 6, pp. 621-639, december (1990).

BARAD, Miryam And SIPPER, Daniel, Flexibility in Manufacturing System: Definitions and Petri Net Modelling. International Journal of Production Research, vol. 26, n. 2, pp. 237-248, (1988).

BERTHELOT, Gérard and TERRAT, Richard, Petri Nets Theory for the Correctness of protocols. IEEE Transacción on Communications, vol. COM-30, n. 12, dec. (1982).

CAO, Xi-Ren and HO, Yu-Chi, Models of Discrete Event Dynamic Systems. IEEE Control System Magazine, pp. 69-76, june (1990).

CARDOSO, Janette; FARINES, Jean-Marie e CURY, Jose, Eduardo, Controle de Sistemas de Manufatura. Apostilas do Curso de Pós-Graduação em Automação Industrial, a nível de especialização, Universidade Federal de Santa Catarina & Universidade de Caxias do Sul, setembro (1994).

CASSANDRAS, Christos, G. and RAMADGE, Peter, J., Toward a Control Theory for Discrete Event System. IEEE Control Systems Magazine, pp. 66-68, june (1990).

COHEN, Guy e colegas, Algebraic Tools for the Performance Evaluatiuon of Discrete Event Systems. Proceedings of the IEEE, vol. 77, n. 1, pp. 39-57, january (1989).

DAVID, René and ALLA, Hassane, Petri Nets for Modelling of Dynamic System- A Survey. Automática, vol. 30, n. 2, pp. 175-202, (1994).

Dos SANTOS FILHO, Diolino, Jose, Proposta do Mark Flow Graph Estendido para a Modelagem e Controle de Sistemas Integrados de Manufatura. Dissertação de Mestrado apresentada à Escola Politécnica da USP, São Paulo, (1993).

FAVREL, Joel and LEE, Kwang, H., Modelling, Analyzing, Scheduling and Control of Flexible Manufacturing System by Petri Nets. in IFIP Conf. Production System, Copenhagen, Amsterdam: North-Holland, pp. 223- 243, aug. (1984).

FREEDMAN, Paul, Time, Petri Nets, and Robotics, IEEE Transaction on Robotics and Automation, vol. 7, n. 4, pp. 417-433, august (1991).

GENRICH, Harmann J., Predicate/Transition Nets. Lecture in Computer Science, n. 254, pp. 207-247, (1986).

GOMES de Alcantara, Carlos Augusto, Contribuição à Modelagem e Análise de Sistemas de Produção Utilizando Redes de Petri. Tese de Doutorado apresentada à UFRJ, Rio de Janeiro, março (1990).

HACK, M., H., T., Analysis of Production Schemata by Petri Nets. Master of Science Dissertation Thesis, MIT, february (1972).

HILLION, Hervé P. And PROTH, Jean-Marie, Performance Evaluation of Job-shop System Using Timed Event-Graphs. IEEE Transaction on Automatic Control, vol. 34, n. 1, pp. 3-9, (1989).

HO, Y., C., Scanning the Issue: Dynamics of Discrete Event Systems. Proceedings of the IEEE, vol. 77, n. 1, pp. 3-6, january (1989).

HUANG, Han-Pang and CHANG, Po-Chu, Specification, modelling and control of a flexible manufacturing cell. Int. J. Prod. Res., vol. 30, n. 11, pp. 2515-2543, (1992).

JENSEN, Kurt, Coloured Petri Nets and the invariant-Method. Theoretical Computer Science, vol. 14, pp. 317-336, (1981).

JENSEN, Kurt, Coloured Petri Net. Lecture notes in Computer Science, n. 254, pp. 248-299, (1986).

LORINI, Flávio, José, Tecnologia de Grupo e Organização da Manufatura. Editora da UFSC (1993).

MIYAGI, P., E.; FURUKAWA, C., M.; MARUYAMA, N. and TSUZUKI, F., S., G., Sistemas de Manufatura: Tendências em Modelagem, Análise e Projeto. X Congresso Brasileiro de engenharia Mecânica, (10° COBEM), Rio de Janeiro, RJ, pp. 491-494, dezembro (1989).

MOLLOY, Michael, K., Discrete Time Stochastic Petri Nets. IEEE Transactions on Software Engineering, vol. SE-11, n. 4, abril (1985).

MURATA, Tadao, Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, vol. 77, n. 4, pp. 541-580, abril (1989).

PETERSON, J. L., Petri Net Theory and the modelling of system. Prentice-Hall Editions, (1981).

RIGHINI, G., Modular Petri Nets for Simulation of Flexible Production System. Int. J. Prod. Res., vol.31, n. 10, pp. 2463-2477, (1993)

RILLO, Marcio, Aplicações de Redes de Petri em sistemas de Manufatura. Tese de Doutorado apresentada à Escola Politécnica da USP, São Paulo, (1988).

SILVA, Manuel and VALETTE, Robert, Petri Nets and Flexible Manufacturing. Lecture Notes in Computer Science, pp. 374-417, (1989).

VALAVANIS, Kimon, P., On the Hierarchical Modelling Analysis and Simulation of Flexible Manufacturing System with Extended Petri Nets. IEEE Transactions on System, man, and Cybernetics, vol. 20, n. 1, pp. 94-110, january/february (1990).

VISWANADHAM, N. and NARAHARI, Y., Coloured Petri Net Model for Automated Manufacturing System. Proceeding of the IEEE International Conference on Robotic and Automation, Raleigh, North Carolina, march 30th- april 3rd, pp. 1985-1990, (1987).

ZHOU, Meng-Chu and DICESARE, Frank, Parallel and Sequential Mutual Exclusions for Petri Net Modelling of Manufacturing System with Shared Resources. IEEE Transactions on Robotics and Automation, vol. 7, n. 4, pp. 515-527, august (1991).

ZHOU, Meng-Chu, DICESARE, Frank and DESROCHERS, Alan A., A Hybrid Methodology for Syntesis of Petri Net Models for Manufacturing Systems. IEEE Transaction on Robotics and Automation, vol. 8. n. 3, pp. 350-361, june (1992).

APÊNDICE

Resultados do Analisador / Simulador de Redes de Petri para a célula de montagem
apresentada no capítulo 6.

NET celula_de_montagem ;

nodes

```
p20,p21,p22,p23,p24,p25,p26,p27,p28,p29,p30      : place (1);
p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14   : place;
p15,p16,p17,p18,p19,p31,p32                      : place;
t1,t2,t3,t7,t8,t9,t10,t11,t13,t15,t16            : transition [0,0];
t4,t12,t14                                         : transition [3,3];
t5                                                  : transition [5,5];
t6,t17                                              : transition [4,4];
```

Structure

```
t1 : (p1,p20) , (p4);
t2 : (p2,p21) , (p5);
t3 : (p3,p22) , (p6);
t4 : (p4,p24) , (p7,p20);
t5 : (p5,p25) , (p8,p21);
t6 : (p6,p26) , (p9,p22);
t7 : (p7,p29,p31) , (p10,p24);
t8 : (p8,p28,p30) , (p11,p25);
t9 : (p9,p30,p32) , (p12,p26);
t10: (p11) , (p13,p31);
t11: (p10,p13) , (p14);
t12: (p14) , (p15,p30,p32);
t13: (p12,p15) , (p16);
t14: (p16) , (p17,p30);
t15: (p17,p27) , (p18,p28,p29);
t16: (p18,p23) , (p19,p27);
t17: (p19) , (p1,p2,p3,p23);
```

EndNet.

Performance Evaluation Oriented to EVENTS of Net celula_de_montagem.

Inicial Marking : {p20, p21, p22, p23, p24, p25, p26, p27, p28, p29, p30,
3* p1, 3* p2, 2* p3}

Desired Precision : 0.10 %

Max. of Fires : 170

Num. of Reaching : 1370

Improdutive Interac.: 0 (0.00 %)

*-----

Average number of fires from cycle transitions :

(t1: 1.00) (t10: 1.00) (t11: 1.00) (t13: 1.00) (t15: 1.00) (t16: 1.00)
(t2: 1.00) (t3: 0.48) (t7: 1.00) (t8: 1.00) (t9: 1.00) (t12: 1.00)
(t14: 1.00) (t4: 1.00) (t5: 1.00) (t17: 1.00) (t6: 1.00)

Average time of fire :

(t12: 3.00) (t14: 3.00) (t4: 3.00) (t5: 5.00) (t17: 4.00) (t6: 4.00)

Average marking in places :

(p20: 0.17) (p22: 0.33) (p23: 0.33) (p24: 0.50) (p25: 0.83)
(p26: 0.83) (p27: 1.00) (p14: 0.50) (p2: 0.17) (p4: 0.71) (p5: 1.00)
(p6: 0.75) (p7: 0.57) (p8: 0.10) (p9: 0.14) (p16: 0.57) (p19: 0.67)

*-----

Inicial Event: t1

Ed 1: t17

Average Time: 6.01 Deviation: 0.24 Probab: 100.00 % Reach: 1
Minimum time of reaching : 6.00 Maximum time of reaching : 15.00

*-----

Invariants Analysis for the net celula_de_montagem.

*-----

Inhibitions used in this analysis:

Obligatory Transitions : {}

Prohibited Transitions : {}

*-----

Transition invariant basis:

BI1 :{t1, t10, t11, t13, t15, t16, t2, t3, t7, t8, t9, t12, t14, t4, t5,
t17, t6}

*-----

Minimum positive transition invariants for this net:

IT1 :{t1, t10, t11, t13, t15, t16, t2, t3, t7, t8, t9, t12, t14, t4, t5,
t17, t6}

*-----

Transitions at none invariant:

T = {}

*-----

Sub-Net existency test (events graph):

None of found invariants is events graph.

*-----

Invariants Analysis for the net celula_de_montagem.

*-----

Inhibitions used in this analysis:

Obligatory Places : {}

Prohibited Places : {}

*-----

Place invariant basis:

- BI1 :{p20, p23, p24, p27, p29, -p1}
- BI2 :{p21, p23, p25, p27, p28, -p2}
- BI3 :{p28, -p29, -p10, p11, p13}
- BI4 :{p20, p4}
- BI5 :{p21, p5}
- BI6 :{p22, p6}
- BI7 :{p20, p23, p27, p29, -p1, -p7}
- BI8 :{p25, p8}
- BI9 :{p26, p9}
- BI10 :{p22, p23, p26, p27, p29, p10, -p12, p14, -p3, p15}
- BI11 :{p28, -p29, -p30, -p10, -p12, -p14, -p16}
- BI12 :{p22, p23, p26, p27, -p28, p29, p30, p10, p14, -p3, -p17}
- BI13 :{p27, p18}
- BI14 :{p23, p19}
- BI15 :{p28, -p29, p11, p31}
- BI16 :{p22, p23, p26, p27, p29, p10, p14, -p3, p32}

*-----

Minimum positive place invariants for this net:

- IL1 :{p20, p4}
- IL2 :{p21, p5}
- IL3 :{p22, p6}
- IL4 :{p23, p19}
- IL5 :{p24, p7}
- IL6 :{p25, p8}
- IL7 :{p26, p9}
- IL8 :{p27, p18}
- IL9 :{p11, p12, p13, p14, p2, p5, p8, p16, p17, p18, p19, p32}
- IL10 :{p28, p10, p11, p12, p14, p16, p17, p31, p32}
- IL11 :{p10, p11, p12, p14, p2, p5, p8, p16, p17, p18, p19, p31, p32}
- IL12 :{p1, p10, p12, p14, p4, p7, p16, p17, p18, p19, p32}
- IL13 :{p10, p11, p14, p2, p5, p8, p15, p16, p17, p18, p19, p31}
- IL14 :{p11, p13, p14, p2, p5, p8, p15, p16, p17, p18, p19}
- IL15 :{p28, p11, p12, p13, p14, p16, p17, p32}
- IL16 :{p12, p3, p6, p9, p16, p17, p18, p19}
- IL17 :{p29, p10, p14, p15, p16, p17}
- IL18 :{p1, p10, p14, p4, p7, p15, p16, p17, p18, p19}
- IL19 :{p28, p10, p11, p14, p15, p16, p17, p31}
- IL20 :{p28, p11, p13, p14, p15, p16, p17}
- IL21 :{p30, p11, p12, p13, p14, p16}
- IL22 :{p29, p10, p12, p14, p16, p17, p32}
- IL23 :{p30, p10, p11, p12, p14, p16, p31}

*-----

Place in all invariants:

P = {}

Places at none invariant:

P = {}

*-----

Sub-Net existency test (state machine):

All found invariants are state machine.

*-----

State Enumeration : net celula_de_montagem (72 reachable states).

Verified properties:

```
*-----  
Net under analysis is limited.  
Null places (M = 0): {}  
Binary places      : {p20, p21, p22, p23, p24, p25, p26, p27, p28, p29,  
                      p30, p10, p11, p12, p13, p14, p4, p5, p6, p7, p8,  
                      p9, p15, p16, p17, p18, p19, p31, p32}  
k-Bounded places   : {3* p1, 3* p2, 2* p3}  
Unbounded places   : {}
```

Net under analysis is not strictly conservative.

Multi-enabled Tr.: {}

Net under analysis is live.

```
Live Tr.           : {all}  
"Almost-live" Tr.: {all}  
Non-fired Tr.      : {}
```

Net never can go back to M0.

No live-locks detected.

No deadlocks detected.

```
*-----
```

State Enumeration : net celula_de_montagem.
Omitted intervals mean [0,0].

Reachable states for this net :

```
*-----  
C0   :{p20, p21, p22, p23, p24, p25, p26, p27, p28, p29, p30, 3* p1, 3* p  
      2* p3}  
      D:{t1, t2, t3}  
C1   :{p21, p22, p23, p24, p25, p26, p27, p28, p29, p30, 2* p1, 3* p2,  
      2* p3, p4}  
      D:{t2, t3, t4[3]}  
C2   :{p22, p23, p24, p25, p26, p27, p28, p29, p30, 2* p1, 2* p2, 2* p3, :  
      p5}  
      D:{t3, t4[3], t5[5]}  
C3   :{p23, p24, p25, p26, p27, p28, p29, p30, 2* p1, 2* p2, p3, p4, p5, :  
      }  
      D:{t4[3], t5[5], t6[4]}  
C4   :{p20, p23, p25, p26, p27, p28, p29, p30, 2* p1, 2* p2, p3, p5, p6, :  
      }  
      D:{t1, t5[2], t6[1]}  
C5   :{p23, p25, p26, p27, p28, p29, p30, p1, 2* p2, p3, p4, p5, p6, p7}  
      D:{t5[2], t6[1]}  
C6   :{p22, p23, p25, p27, p28, p29, p30, p1, 2* p2, p3, p4, p5, p7, p9}  
      D:{t3, t5[1]}  
C7   :{p23, p25, p27, p28, p29, p30, p1, 2* p2, p4, p5, p6, p7, p9}  
      D:{t5[1]}  
C8   :{p21, p23, p27, p28, p29, p30, p1, 2* p2, p4, p6, p7, p8, p9}  
      D:{t2, t8}  
C9   :{p23, p27, p28, p29, p30, p1, p2, p4, p5, p6, p7, p8, p9}  
      D:{t8}  
C10  :{p23, p25, p27, p29, p1, p11, p2, p4, p5, p6, p7, p9}  
      D:{t10, t5[5]}
```

C11 :{p23, p25, p27, p29, p1, p13, p2, p4, p5, p6, p7, p9, p31}
D:{t7, t5[5]}

C12 :{p23, p24, p25, p27, p1, p10, p13, p2, p4, p5, p6, p9}
D:{t11, t4[3], t5[5]}

C13 :{p23, p24, p25, p27, p1, p14, p2, p4, p5, p6, p9}
D:{t12[3], t4[3], t5[5]}

C14 :{p23, p24, p25, p27, p30, p1, p2, p4, p5, p6, p9, p15, p32}
D:{t9, t4, t5[2]}

C15 :{p23, p24, p25, p26, p27, p1, p12, p2, p4, p5, p6, p15}
D:{t13, t4, t5[2], t6[4]}

C16 :{p23, p24, p25, p26, p27, p1, p2, p4, p5, p6, p16}
D:{t14[3], t4, t5[2], t6[4]}

C17 :{p20, p23, p25, p26, p27, p1, p2, p5, p6, p7, p16}
D:{t1, t14[3], t5[2], t6[4]}

C18 :{p23, p25, p26, p27, p2, p4, p5, p6, p7, p16}
D:{t14[3], t5[2], t6[4]}

C19 :{p21, p23, p26, p27, p2, p4, p6, p7, p8, p16}
D:{t2, t14[1], t6[2]}

C20 :{p23, p26, p27, p4, p5, p6, p7, p8, p16}
D:{t14[1], t6[2]}

C21 :{p23, p26, p27, p30, p4, p5, p6, p7, p8, p17}
D:{t15, t6[1]}

C22 :{p23, p26, p28, p29, p30, p4, p5, p6, p7, p8, p18}
D:{t16, t8, t6[1]}

C23 :{p26, p27, p28, p29, p30, p4, p5, p6, p7, p8, p19}
D:{t8, t17[4], t6[1]}

C24 :{p25, p26, p27, p29, p11, p4, p5, p6, p7, p19}
D:{t10, t5[5], t17[4], t6[1]}

C25 :{p25, p26, p27, p29, p13, p4, p5, p6, p7, p19, p31}
D:{t7, t5[5], t17[4], t6[1]}

C26 :{p24, p25, p26, p27, p10, p13, p4, p5, p6, p19}
D:{t11, t4[3], t5[5], t17[4], t6[1]}

C27 :{p24, p25, p26, p27, p14, p4, p5, p6, p19}
D:{t12[3], t4[3], t5[5], t17[4], t6[1]}

C28 :{p22, p24, p25, p27, p14, p4, p5, p9, p19}
D:{t12[2], t4[2], t5[4], t17[3]}

C29 :{p22, p24, p25, p27, p30, p4, p5, p9, p15, p19, p32}
D:{t9, t4, t5[2], t17[1]}

C30 :{p22, p24, p25, p26, p27, p12, p4, p5, p15, p19}
D:{t13, t4, t5[2], t17[1]}

C31 :{p22, p24, p25, p26, p27, p4, p5, p16, p19}
D:{t14[3], t4, t5[2], t17[1]}

C32 :{p20, p22, p25, p26, p27, p5, p7, p16, p19}
D:{t14[3], t5[2], t17[1]}

C33 :{p20, p22, p23, p25, p26, p27, p1, p2, p3, p5, p7, p16}
D:{t1, t3, t14[2], t5[1]}

C34 :{p22, p23, p25, p26, p27, p2, p3, p4, p5, p7, p16}
D:{t3, t14[2], t5[1]}

C35 :{p23, p25, p26, p27, p2, p4, p5, p6, p7, p16}
D:{t14[2], t5[1], t6[4]}

C36 :{p21, p23, p26, p27, p2, p4, p6, p7, p8, p16}
D:{t2, t14[1], t6[3]}

C37 :{p23, p26, p27, p4, p5, p6, p7, p8, p16}
D:{t14[1], t6[3]}

C38 :{p23, p26, p27, p30, p4, p5, p6, p7, p8, p17}
D:{t15, t6[2]}

C39 :{p23, p26, p28, p29, p30, p4, p5, p6, p7, p8, p18}
D:{t16, t8, t6[2]}

C40 :{p26, p27, p28, p29, p30, p4, p5, p6, p7, p8, p19}
D:{t8, t17[4], t6[2]}

C41 :{p25, p26, p27, p29, p11, p4, p5, p6, p7, p19}
D:{t10, t5[5], t17[4], t6[2]}

C42 :{p25, p26, p27, p29, p13, p4, p5, p6, p7, p19, p31}

D:{t7, t5[5], t17[4], t6[2]}
 C43 :{p24, p25, p26, p27, p10, p13, p4, p5, p6, p19}
 D:{t11, t4[3], t5[5], t17[4], t6[2]}
 C44 :{p24, p25, p26, p27, p14, p4, p5, p6, p19}
 D:{t12[3], t4[3], t5[5], t17[4], t6[2]}
 C45 :{p22, p24, p25, p27, p14, p4, p5, p9, p19}
 D:{t12[1], t4[1], t5[3], t17[2]}
 C46 :{p20, p22, p25, p27, p14, p5, p7, p9, p19}
 D:{t12, t5[2], t17[1]}
 C47 :{p20, p22, p25, p27, p30, p5, p7, p9, p15, p19, p32}
 D:{t9, t5[2], t17[1]}
 C48 :{p20, p22, p25, p26, p27, p12, p5, p7, p15, p19}
 D:{t13, t5[2], t17[1]}
 C49 :{p23, p25, p26, p29, p11, p4, p5, p6, p7, p18}
 D:{t10, t16, t5[5], t6[2]}
 C50 :{p23, p25, p26, p29, p13, p4, p5, p6, p7, p18, p31}
 D:{t16, t7, t5[5], t6[2]}
 C51 :{p23, p24, p25, p26, p10, p13, p4, p5, p6, p18}
 D:{t11, t16, t4[3], t5[5], t6[2]}
 C52 :{p23, p24, p25, p26, p14, p4, p5, p6, p18}
 D:{t16, t12[3], t4[3], t5[5], t6[2]}
 C53 :{p20, p23, p25, p26, p27, p1, p2, p5, p6, p7, p16}
 D:{t1, t14[2], t5[1], t6[4]}
 C54 :{p23, p25, p26, p29, p11, p4, p5, p6, p7, p18}
 D:{t10, t16, t5[5], t6[1]}
 C55 :{p23, p25, p26, p29, p13, p4, p5, p6, p7, p18, p31}
 D:{t16, t7, t5[5], t6[1]}
 C56 :{p23, p24, p25, p26, p10, p13, p4, p5, p6, p18}
 D:{t11, t16, t4[3], t5[5], t6[1]}
 C57 :{p23, p24, p25, p26, p14, p4, p5, p6, p18}
 D:{t16, t12[3], t4[3], t5[5], t6[1]}
 C58 :{p20, p23, p25, p26, p27, p1, p12, p2, p5, p6, p7, p15}
 D:{t1, t13, t5[2], t6[4]}
 C59 :{p23, p25, p26, p27, p12, p2, p4, p5, p6, p7, p15}
 D:{t13, t5[2], t6[4]}
 C60 :{p20, p23, p25, p27, p30, p1, p2, p5, p6, p7, p9, p15, p32}
 D:{t1, t9, t5[2]}
 C61 :{p23, p25, p27, p30, p2, p4, p5, p6, p7, p9, p15, p32}
 D:{t9, t5[2]}
 C62 :{p20, p23, p25, p27, p1, p14, p2, p5, p6, p7, p9}
 D:{t1, t12, t5[2]}
 C63 :{p23, p25, p27, p14, p2, p4, p5, p6, p7, p9}
 D:{t12, t5[2]}
 C64 :{p21, p23, p25, p27, p29, p1, p11, 2* p2, p4, p6, p7, p9}
 D:{t10, t2}
 C65 :{p21, p23, p25, p27, p29, p1, p13, 2* p2, p4, p6, p7, p9, p31}
 D:{t2, t7}
 C66 :{p21, p23, p24, p25, p27, p1, p10, p13, 2* p2, p4, p6, p9}
 D:{t11, t2, t4[3]}
 C67 :{p21, p23, p24, p25, p27, p1, p14, 2* p2, p4, p6, p9}
 D:{t2, t12[3], t4[3]}
 C68 :{p21, p23, p24, p25, p26, p27, p28, p29, p30, 2* p1, 3* p2, p3, p4, p6}
 D:{t2, t4[3], t6[4]}
 C69 :{p20, p22, p23, p24, p25, p26, p27, p28, p29, p30, 3* p1, 2* p2, 2* p3, p5}
 D:{t1, t3, t5[5]}
 C70 :{p20, p23, p24, p25, p26, p27, p28, p29, p30, 3* p1, 2* p2, p3, p5, p6}
 D:{t1, t5[5], t6[4]}
 C71 :{p20, p21, p23, p24, p25, p26, p27, p28, p29, p30, 3* p1, 3* p2, p3, p6}
 D:{t1, t2, t6[4]}