

UNIVERSIDADE FEDERAL DO AMAZONAS
FACULDADE DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
ELÉTRICA
MESTRADO EM ENGENHARIA ELÉTRICA

MODELAGEM E ANÁLISE DE SISTEMAS DE
MANUFATURA FLEXÍVEL POR MEIO DE REDES DE
PETRI COLORIDA, HIERÁRQUICA E TEMPORIZADA.

MARCELO VIEIRA DE AGUIAR

MANAUS

JANEIRO/2008

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

UNIVERSIDADE FEDERAL DO AMAZONAS
FACULDADE DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓSGRADUAÇÃO EM ENGENHARIA ELÉTRICA

MARCELO VIEIRA DE AGUIAR

MODELAGEM E ANÁLISE DE SISTEMAS DE
MANUFATURA FLEXÍVEL POR MEIO DE REDES DE
PETRI HIERÁRQUICA E COLORIDA

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Amazonas, como requisito parcial para a obtenção do título de Mestre em Engenharia Elétrica, área de concentração de Controle e Automação de Sistemas.

Orientadores: Prof. Dr. João Edgar Chaves Filho

Co-orientador: Prof. Raimundo Barreto

MANAUS

2008

Ficha Catalográfica
(Catalogação na fonte realizada pela Biblioteca Central – UFAM)

Aguiar, Marcelo Vieira de

A282m Modelagem e análise de sistemas de manufatura flexível por meio de redes de Petri hierárquica e colorida / Marcelo Vieira de Aguiar . - Manaus: UFAM, 2007.

106 f.; il.

Dissertação (Mestrado em Engenharia Elétrica) — Universidade Federal do Amazonas, 2007.

Orientador: Prof. Dr. João Edgar Chaves Filho

Co-orientador: Prof. Raimundo Barreto

1. Manufatura 2. Sistemas industriais 3. Administração da produção 4. Gargalos de produção I. Chaves Filho, João Edgar II. Barreto, Raimundo III. Universidade Federal do Amazonas IV. Título

CDU 658.62(811.3)(043.3)

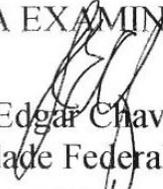
MARCELO VIEIRA DE AGUIAR

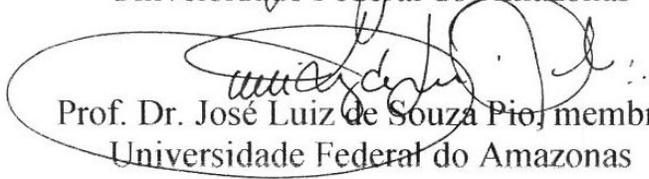
MODELAGEM E ANÁLISE DE SISTEMAS DE
MANUFATURA FLEXÍVEL POR MEIO DE REDES DE
PETRI HIERÁRQUICA E COLORIDA

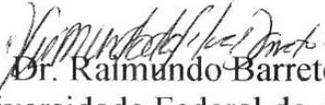
Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Amazonas, como requisito parcial para a obtenção do título de Mestre em Engenharia Elétrica, área de concentração de Controle e Automação de Sistemas.

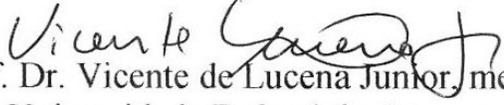
Aprovado em 03 de janeiro de 2008.

BANCA EXAMINADORA


Prof. Dr. João Edgar Chaves Filho, presidente.
Universidade Federal do Amazonas


Prof. Dr. José Luiz de Souza Pio, membro.
Universidade Federal do Amazonas


Prof. Dr. Raimundo Barreto, membro.
Universidade Federal do Amazonas


Prof. Dr. Vicente de Lucena Junior, membro.
Universidade Federal do Amazonas

Agradeço a Deus pela vida, aos meus pais pela educação e o apoio que me deram, aos meus irmãos e amigos pelo companheirismo, e em especial, à minha companheira Dulcilene Alves, pelo incentivo dado para realização deste trabalho.

Agradecimentos

A realização de todo grande projeto de vida exige grande esforço, determinação, humildade, e principalmente a participação efetivas das pessoas importantes que fazem parte do seu ciclo de convivência. Dessa forma, não poderia deixar de agradecer a todos que me apoiaram na realização deste trabalho.

Gostaria de agradecer ao Prof. João Edgar por ter acreditado nesse projeto desde o início, mesmo quando parecia uma meta tão difícil de ser alcançada. Agradeço também pelas orientações e incentivo no decorrer do desenvolvimento do mesmo.

Agradeço ao Professor Raimundo Barreto pelas orientações, pelo encorajamento e por se dispor tão prestativamente a compartilhado seu vasto conhecimento e experiência em redes de Petri e desenvolvimento de trabalhos científicos.

Ambos participaram além da função de orientador e foram de fundamental importância para o desenvolvimento desse trabalho sendo, portanto, merecedores de grande parte do mérito de sua conclusão.

Gostaria de agradecer a todo o corpo docente e coordenadores que se esforçaram na nobre tarefa de transmitir o conhecimento e tiveram coragem para se lançar na implantação do primeiro programa de Mestrado em Engenharia Elétrica da Universidade Federal do Amazonas.

Quero agradecer de forma muito especial ao meu pai José Fernandes Aguiar e minha mãe Elza Vieira de Aguiar por terem se esforçado tanto na minha educação e me ensinado muito do que sou hoje, enfatizando sempre a importância de ser humilde, perseverante e determinado na realização dos objetivos.

Agradecer fortemente à minha companheira Dulcilene Alves Ferreira por ter estado ao meu lado, me apoiando e me incentivado durante todo o desenvolvimento desses trabalhos. Agradeço principalmente pela paciência de me ouvir e me aconselhar nos momentos em que a tarefa parecia muito difícil.

Agradeço aos meus colegas de trabalho do SENAI em especial ao Profs. Lean Cláudio, José Nabir e Prof. Lúcio Santos que sempre me incentivaram através de palavras de apoio e sugestões criativas; ao gerente do CETAS e meu amigo Rodson Barros a ao Sr. Itiu Maruoka por ter me apoiado, e muitas vezes até flexibilizado meu horário de trabalho para que pudesse dedicar mais tempo ao projeto. Além disso, também agradeço pelas indicações na participação de importantes eventos nacionais e internacionais nas áreas relacionadas à automação industrial que me possibilitaram um alargamento da visão e grande crescimento profissional.

Resumo

Esta dissertação descreve o uso de Redes de Petri Coloridas Hierárquicas e Temporizadas (CPN) no desenvolvimento, simulação e análise de Sistemas Industriais Flexíveis de Manufatura, baseado em técnica modular, numa perspectiva de Sistema de Evento Discreto. O objetivo geral é propor uma metodologia *top-down* sistematizada, com instanciamento de sub-redes, utilizando técnicas de composição para manutenção de certas propriedades de interesse, visando subsidiar engenheiros no processo de construção e análise de modelos que representem Sistemas de Manufatura Flexíveis (FMS) reais, considerando suas relações de precedência, concorrência e sincronização. O *lay-out* do sistema real é considerado como ponto de partida a base para o maior nível de abstração do modelo em CPN, de tal modo que o refinamento pode ser executado pela inclusão de sub-modelos pré-definidos nas transições de substituição. Como resultado, o método proposto pretende conduzir a melhorias na implementação, visualização gráfica estática e dinâmica e análise de funcionamento, possibilitando o estudo do comportamento e os ajustes necessários nos passos anteriores à construção do sistema real. O método proposto inclui uma sintática de representação de elementos do sistema no maior nível de abstração usando CPN; uma seqüência sistematizada de projeto para construção de modelo; uma biblioteca de sub-redes válidas para sistemas industriais em CPN; uma combinação das teorias de *Dobragem Baseada em Marcas*, *Exclusão Mútua Paralela (PME)* e *Exclusão Mútua Serial(SME)* para modelos de FMS usando CPN. exploração sistemática de espaço de estado por simulação para identificação de propriedades de validação do sistema, como *limitação*, *vivacidade e reversibilidade*; e finalmente uma análise de desempenho por simulação para identificação de propriedades como taxa de produção, *gargalos de produção* e relações de causa e efeito na *parametrização* da produção.

Abstract

This dissertation describes the use of the Timed Hierarchical and Coloured Petri Nets (CPNs) in the development, simulation and analysis of Flexible Manufacturing Systems, through a Discrete Event System perspective and using modular techniques. The aim is to develop a general top-down systematic way to help engineers to construct models that represent a real FMS system, considering precedence relations, concurrency and performance. The physical layout of the real system is the start point for the highest abstraction level specification on the CPN model, in such a way that details can be carried out through the hierarchical substitution transitions. Therefore, the proposed method leads to an improvement on the implementation, visualization and performance analysis, allowing the correctness and adjustments at the prior steps before the construction of real system. The proposed method includes a syntactic representation for models elements on the highest level of the system; a systematic design sequence to construct a CPN model; a library of valid CPNs model templates for manufacturing systems; a combination of *Folding Based in Marks*, an extension of *PME* and *SME* theory and a *textual structured programming language* applied to CPN to models FMS system. A systematic exploration of state space simulation is performed to confirm the logic validate of the system through important properties like *boundedness*, *liveness* and *fairness*. Finally is executed a systematic performance analysis to identify the set-up for a better *throughput* and absence of *bottleneck*.

CONTEÚDO

1. INTRODUÇÃO	1
1.1 MOTIVAÇÃO.....	1
1.2 OBJETIVOS.....	3
1.3. ORGANIZAÇÃO DO TEXTO.....	4
2. TRABALHOS CORRELATOS.....	6
2.1. MODELAGEM DE SISTEMAS DE MANUFATURA BASEADO EM REDE DE PETRI.....	6
2.2. COMPARTILHAMENTO DE RECURSOS E CONVERSÃO DE REDES....	9
3. PROCESSOS INDUSTRIAIS DE MANUFATURA.....	12
3.1. SISTEMAS DE MANUFATURA AUTOMATIZADOS	12
3.2. TIPOS DE PROCESSOS INDUSTRIAIS DE MANUFATURA	13
3.2.1. Flow-shop.....	13
3.2.2. Job-shop.....	13
3.2.3. Células de Manufatura	14
3.2.4. Sistema Integrado de Manufatura	14
3.3 FLEXIBILIDADE EM PROCESSOS DE MANUFATURA.....	15
3.4 SISTEMAS FLEXÍVEIS DE MANUFATURA.....	16
4. REDES DE PETRI	19
4.1 INTRODUÇÃO.....	19
4.2 DEFINIÇÃO DE REDE DE PETRI BASICA	20
4.2.1 Definição formal da estrutura da rede de Petri.....	20
4.2.2 Dinâmica da Rede de Petri	22
4.3 PROPRIEDADES FUNDAMENTAIS DAS RPs	23
4.3.1 Propriedades comportamentais (Procedurais)	23
4.3.2 Propriedades estruturais	24
4.3.3 Características de Desempenho	25
4.4 DEFINIÇÃO FORMAL DAS PROPRIEDADES DAS RPs	26
4.4.1 Alcançabilidade	26
4.4.2 Limitação e Segurança:	26
4.4.3 Vivacidade.....	27
4.4.4 Reversibilidade/Reinicialização	27
4.5. TÉCNICAS DE ANÁLISE DE REDE DE PETRI	28
4.5.1. Análises pela Árvore de Alcançabilidade e de Cobertura.....	28
4.5.2. Análise por Simulação e Composição	31
4.5.3 Importância do <i>Set-up</i> na Análise por simulação	33
4.5.4 Análise por Redução	33
4.6 REDE DE PETRI COLORIDA	34
4.6.1. Definição formal da estrutura da CPN.....	35
4.6.2. Descrição formal do comportamento.....	35
4.6.3. Combinação de CPN com outras extensões.....	39
4.7. REDES DE PETRI TEMPORIZADAS.....	39
4.8. HIERARQUIA EM REDE DE PETRI.....	41
4.9. RELAÇÃO ENTRE REDE CPN E FMS	42
4.9.1 CPN como Ferramenta de Modelagem.....	42
4.9.2 Interpretação dos Elementos de uma CPN com relação a um FMS	43
5. ESTRUTURAÇÃO E CONVERSÃO DE REDES DE PETRI	44

5.1. MECANISMOS DE ESTRUTURAÇÃO E COMPOSIÇÃO	44
5.2. FUSÃO (COMPOSIÇÃO EXTERNA).....	46
5.3 DOBRAGEM (COMPOSIÇÃO INTERNA).....	49
5.4 TÉCNICAS DE REFINAMENTO E ABSTRAÇÃO	51
5.4.1 Refinamento de Transição.....	51
5.4.2 Estruturação e Modificação de Modelo	52
5.5. COMPOSIÇÃO HIERÁRQUICA	53
5.6. TEOREMA DE EXTENSÕES VALIDAS DE REDE.....	55
5.7. MODELO DE BUFFERS VÁLIDOS	56
5.8. DEFINIÇÃO DE EXCLUSÃO MÚTUA.....	58
5.8.1 Exclusão Mútua Paralela.....	58
5.8.2 Exclusão Mútua Serial	59
5.8.3 Extensão SME e PME para utilização em CPN	60
6. CONJUNDO DE SUB-REDE VÁLIDOS PARA FMS	61
6.1. CONTRUÇÃO E ANÁLISE DE SUB-MODELOS USANDO TEORIA PME/SME.....	61
6.2. BIBLIOTECA DE SUB-MODELOS USADOS FMS.....	64
7. ABORDAGEM PROPÓSTA DA METODOLOGIA.....	75
7.1. NÍVEIS DE MODELAMENTO	75
7.2. REPRESENTAÇÃO GRÁFICA DAS CPN.....	77
7.3. ESTRUTURAÇÃO E INTERPRETAÇÃO	78
7.4. SÍNTESE DO MODELAMENTO	81
8. APLICAÇÃO DA METODOLOGIA	85
8.1. SITUAÇÃO PROBLEMA.....	85
8.2. PASSOS DO MODELAMENTO	88
9. CONCLUSÃO	101
9.1. CONSIDERAÇÕES FINAIS	101
9.2. CONTRIBUIÇÃO.....	102
9.3. TRABALHOS FUTUROS.....	102

1. INTRODUÇÃO

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, este capítulo mostra o contexto do desenvolvimento desse trabalho com base nas tendências dos processos produtivos, apresentando as motivações e os objetivos a serem desenvolvidos, descrevendo o problema a ser resolvido e sua proposta de solução, resume as contribuições e mostra os trabalhos correlatos do estado da arte.

1.1 MOTIVAÇÃO

Desde o advento da revolução industrial no século 19 os processos industriais de manufatura têm passado por intensas transformações em suas estruturas produtivas e teorias administrativas. Isso acarretou no surgimento de novos modelos de produção que passaram a ganhar espaço e importância, principalmente em função do atendimento das novas necessidades de mercado. Dentre essas formas emergentes de organização produtiva encontra-se uma classe chamada Sistemas Flexíveis de Manufatura.

Os Sistemas Flexíveis de Manufatura surgiram para o atendimento de uma rápida transformação do mercado consumidor que passou a ser composto por clientes cada vez mais exigentes e com necessidades variadas, o que incentivou a migração de um sistema fabril composto basicamente de linhas de produção do tipo *flow-shop* (produção em massa) nos anos setenta para um sistema de manufatura altamente automatizado e flexível. Tais sistemas caracterizam-se pela capacidade de admitir rápidas alterações no processo para fabricar uma diversidade de produtos em pequenos e médios volumes, com alta produtividade, baixo custo e sem perder a qualidade do mesmo. Entretanto, essa mudança só foi possível em função da evolução dos computadores a partir dos anos setenta, já que essa nova concepção de produção exigia máquinas controladas por computadores e integradas por redes de comunicação

de dados com capacidade de gerar a sincronização entre os vários dispositivos do sistema.

Por outro lado, esse tipo de processo produtivo, também trouxe como conseqüência um considerável aumento do grau de complexidade dos sistemas de manufatura, nos quais o número de atividades que ocorrem de forma paralela ou concorrente é muito grande e, conseqüentemente, difícil de ser implementado. Um investimento na implantação de uma planta produtiva inadequada que venha funcionar de forma aquém das expectativas, ou que precise de ajustes não previstos, pode vir a encarecer ou até mesmo inviabilizar economicamente o projeto. Não obstante, surgiu uma demanda pela utilização de novas técnicas para a implementação de projetos de controle, modelagem e simulação que atendessem a esse novo paradigma.

“Nesse sentido, a composição modular hierárquica e estruturação de modelos baseados em redes Petri coloridas (CPN) são claramente da máxima importância prática, pois permitem uma representação gráfica com elevado grau de abstração e versatilidade, mantendo ainda uma boa visualização estática e dinâmica do sistema modelado” (Barros, 2006).

As redes de Petri Colorida são especialmente adequadas para a modelagem de sistemas caracterizados por concorrência, conflitos e sincronização. Além disso, inclui tanto as informações estruturais quanto as informações de rota de produção do sistema a ser modelado. Além disso, pela introdução da sua extensão temporizada e de modelos probabilísticos é possível estudar o desempenho do sistema através das relações de características como taxa de produção, resposta a situações de falhas, dentre com relação à organização física e configuração dos equipamentos.

Apesar de não podermos afirmar que haja algum ganho teórico na utilização da CPN, na prática constitui uma forma mais compacta e muito mais conveniente de linguagem de modelamento gráfico do que as PT-nets (*Place-Transitions Nets*), similarmente como uma linguagem de programação de alto nível é muito mais adequada para a confecção de programas práticos do que linguagem de baixo nível, como o assembler, utilizado para a montagem de programas em nível de máquina. Além disso, também é possível a conversão entre PT-nets e CPN através de procedimentos relativamente simples baseados em teorias bem fundamentadas, onde a complexidade da rede de

baixo nível, contida na representação gráfica, passa ser absorvida pelos tokens e pela notação da Rede de Petri Colorida.

A estruturação hierárquica também pode melhorar de forma extrema a construção e legibilidade de um modelo pela conexão de estruturas menores para formar um sistema mais complexo, ou seja, nós podemos construir grandes redes pela interligação de pequenas CPN de uma forma bem definida e baseada em regras que validem essas conexões. A existência de CPN hierárquicas possibilita o modelamento de grandes sistemas, reaproveitamento de estruturas e o gerenciamento do mesmo de uma forma modular.

1.2 OBJETIVOS

O objetivo desse trabalho é propor uma metodologia sistemática para modelagem, simulação e análise de sistemas de manufatura flexíveis, através de uma perspectiva de Sistemas a Eventos Discretos (SED), utilizando técnicas modulares com refinamento *top-down* e fusão de lugares, baseado em Rede de Petri Colorida, Hierárquicas e Temporizadas. A proposta inclui a utilização de ferramentas de simulação associadas à procedimentos sistematizados de análise por síntese de espaço de estado e árvore de alcançabilidade, com o propósito de testar e garantir boas características funcionabilidade, desempenho, flexibilidade e fácil visualização de sistemas complexos antes de sua implantação no chão de fábrica.

Baseado nesse contexto as metas consideradas são:

- Organizar os principais mecanismos e técnicas para estruturação de rede de Petri modulares, colorida hierárquica no estado da arte e classificá-las de acordo com sua aplicação nos sistemas de manufatura.
- Propor uma extensão para a teoria de *Serial Mutual Exclusion* (SME) e *Parallel Mutual Exclusion* (PME) aplicada a Redes de Petri Coloridas.
- Desenvolver uma metodologia modular, funcional e sistematizada para modelagem e análise de sistemas flexíveis de manufatura baseada em redes de Petri coloridas, hierárquicas e temporizadas que garanta a manutenção das propriedades e fundamentada em mecanismos de composição encontrados no estado da arte.

- Desenvolver uma biblioteca de sub-modelos padrões de CPN aplicado à equipamentos de um Sistema Flexíveis de Manufatura que possam ser instanciados na elaboração do modelo completo.
- Desenvolver estudos de casos de uma célula SMS real composta por robô, centro de usinagem, esteira transportadora, sensores, instrumentos de aquisição de medida e aferição de qualidade.
- Comprovar através de análise de desempenho que a abordagem proposta é adequada para a modelagem e análise de para sistemas flexíveis de manufatura, considerando sua visualização, flexibilidade e eficácia na construção do modelo.

1.3. ORGANIZAÇÃO DO TEXTO

No Capítulo 2 é apresentado os principais trabalhos relacionados com modelamento de sistema de manufatura baseado em redes de Petri. Os trabalhos enfatizam a utilização de redes de Petri coloridas hierárquica e as técnicas para manutenção das propriedades de interesse do modelo durante a construção de modelos complexos a partir de submodelos.

No Capítulo 3 é mostrada a descrição das diversas possibilidades de processos de manufatura encontrados no setor industrial e caracterizados de acordo com a abordagem de diversos pesquisadores enfocando suas características, equipamentos, vantagens e desvantagens. A flexibilidade do processo produtivo é enfatizada em função de que o objetivo do trabalho é lidar com Sistemas de Manufatura Flexíveis.

No Capítulo 4 é introduzido o conceito de rede de Petri através sua definição formal e suas características de modelamento. São apresentadas as principais extensões de redes de Petri, com destaque para as redes de Petri colorida (CPN) e redes temporizadas.

No Capítulo 5 é realizada uma síntese das principais técnicas de composição e de modelos de redes de Petri, filosofias de modelagem *top-down* e estruturação hierárquica. Também são abordados os teoremas de extensão de rede de Petri com manutenção de propriedades.

No Capítulo 6 é apresentado um conjunto de sub-redes válidas em CPN representativas de equipamentos facilmente encontrados em Sistemas de Manufatura Flexível. A partir desses sub-modelos, posteriormente será construído o modelo completo de um sistema de manufatura da composição dos mesmos.

No Capítulo 7 é proposta uma síntese das etapas da estruturação e análise de um modelo de Sistemas de Manufatura Flexíveis utilizando Rede de Petri Colorida, Hierárquica e temporizada a partir das técnicas de estruturação e composição apresentadas no capítulo 5.

No Capítulo 8 é realizado o modelamento de uma célula de manufatura composta por um robô, centro de usinagem, correia transportadora e vários sensores de presença de medição utilizando as técnicas apresentadas nos capítulos anteriores. Também é realizada uma análise de desempenho, comparando o resultado do modelo sob diferentes condições para encontrar a melhor configuração da planta.

Finalmente, o Capítulo 9 conclui essa dissertação e apresenta sugestões propostas de futuros trabalhos.

2. TRABALHOS CORRELATOS

Este capítulo mostra um resumo dos trabalhos relevantes que foram desenvolvidos no contexto de sistemas de manufatura flexíveis e modelamento de processos produtivos utilizando rede de Petri. O capítulo está dividido em duas seções principais: Sistemas de Manufatura Automatizados e Modelos de Manufatura baseado em redes de Petri Colorida.

2.1. MODELAGEM DE SISTEMAS DE MANUFATURA BASEADO EM REDE DE PETRI

A rede de Petri é uma metodologia que pode ser usada para projetar sistema a eventos discreto, graficamente e matematicamente. Uma rede de Petri básica (PN) é definida como uma *tríade* gráfica que consiste de *lugares*, *transições*, e *ficha (tokens)*. Na maioria da literatura que aborda rede de Petri para modelamento de sistemas de manufatura flexíveis, esses elementos gráficos denotam características específicas ligadas ao processo produtivo que obedecem a certo padrão semântico. Em geral os lugares são usados para representar estados de operação do sistema, e são classificados pelo tipo de funções. As transições são usadas para representar execução de operações que modificam o estado do sistema e designam o consumo e entrega de recursos, tais como: transformação de produtos, transporte de peças e transferência de dados. As fichas representam recursos fixos e variáveis que afetam o sistema e podem ser modificados durante a evolução da rede, incluindo matérias-primas, produtos manufaturados, equipamento, dados, e informações. Além disso, um PN pode ter um conjunto associado de regras de habilitação e disparo para determinar sob que condições uma transição está habilitada e pode disparar. Entretanto, se mais de um produto tem que ser processado por uma máquina, a falta de distinção entre os *tokens* resulta na replicação do modelo da máquina, acarretando uma explosão da representação.

Nas redes de Petri colorido (CPN), tipos diferentes de fichas são distinguidos pela adição de propriedades associadas com uma ficha. Este tipo de abordagem permite uma representação mais compacta de situações reais. Nesse sentido, vários trabalhos

têm sido desenvolvidos para garantir um modelamento que reflita a estrutura real do sistema de uma forma compacta e de fácil de entendimento.

Kasturia et al. (1988) e Viswanadham e Narahari (1990) usaram as redes de Petri colorida como base para modelamento de sistemas de manufatura automático. Contextualizado nessas propostas, situações de *deadlocks* podem ser encontrados através da técnica de análise de invariantes, possibilitando que a propriedade de vivacidade do modelo possa então ser avaliada. Contudo, apesar da proposta de análise por invariante ter um cunho matemático bastante conciso, para a proposta do presente trabalho, a sua utilização só é possível para sub-modelos não muito complexo, impossibilitando sua abrangência para o modelo completo pela grande quantidade de nós envolvidos.

Martinez et al. (1987), mostrou como a coordenação de subsistemas em sistema de manufatura flexível pode ser descrita por rede de Petri Colorida. O modelo obtido é direcionado para controle local de subsistemas com produção programada, onde a terminologia baseada em rede de Petri colorida é usada para interação entre sub-modelo. Nessa proposta, a análise é capaz de detectar *deadlocks*, problemas de decisão e possibilitar a realização de medidas de desempenho que dependem das variações do sistema modelado.

Villaroel et al. (1989) propuseram modelar separadamente as rotas de produção e a estrutura do sistema de manufatura. Usando sua ferramenta de software GRAMAN, um sistema de manufatura é modelado por uma descrição da planta e uma descrição do plano de trabalho. A estrutura do sistema é modelada a partir de blocos pré-definidos que são instanciados e parametrizados com suas relações estruturais. Adicionalmente, submodelos são conectados por fusão de transição que representam atividades sincronizadas. Contudo, com essa proposta não é possível especificar propriedades de máquinas que dependam da tarefa a ser processada.

A idéia de separação dos modelos também aparece nos trabalhos de Zimmermann e Hommel (1999). Nesta proposta é apresentada uma técnica dedicada de modelamento aplicado sistemas de manufatura que limita os tipos de tokens utilizados, ocasionando uma simplificação gráfica do modelo e na declaração de variáveis. Essa técnica usa refinamento hierárquico com instanciamento de sub-redes pré-definidas. Em adição

aos tempos de transição imediata e exponencial, também foram introduzidos os tempos de disparo não exponencial.

Qiao et al. (2002) apresenta uma metodologia para representar sistemas industriais usando rede de Petri Colorida válida. Seu modelo pode conduzir a uma solução para problemas como replanejamento dinâmico, reconfiguração de compras, processos de retrabalho de partes de mecanismos e recuperação de falhas em máquina. Além disso, *Massa Customization Manufacturing Systems* devem permitir modificações de projeto de produto mesmo em fases finais de produção, e ainda deve responder e ajustar-se rapidamente para estas mudanças, sem adiar o tempo de entrega.

De acordo Qiao et al. (2002), para que uma rede de Petri seja válida ela tem que ter três propriedades fundamentais: limitação, vivacidade, e reversibilidade. A análise dessas características é geralmente difícil e demorada. Para melhorar essa condição, Qiao et al. (2002) propuseram uma metodologia para modelagem de CPN válidas. Este abordagem define alguns modelos de PN básico cuja validade é fácil provar por abordagem matemática ou simulação. Nessa proposta, o sistema industrial para ser modelado é decomposto em células de acordo com função industrial e o tipo de processo que a célula irá realizar. Cada uma destas células é representada com uma PN apropriada e válida. Estes modelos são sub-redes do sistema completo. De acordo com o *Teorema de Extensões Válidas* de PN, as sub-redes podem ser conectadas usando encadeamentos normais ou modelos de *buffers* válido. Como resultado, garante-se que o modelo completo do sistema industrial fica válido.

As idéias de desenvolvimento modular através de criação e instanciamento de sub-redes apresentados nesses trabalhos são fundamentos essenciais para a construção top-down e reutilização de modelos. Essas técnicas constituem a base para o desenvolvimento da metodologia apresentada nesse trabalho. Na proposta, os modelos são desenvolvidos em RPC a partir das características das máquinas e equipamentos formando uma biblioteca de padrões válidos. Esses padrões são instanciados para dentro de blocos denominados de transições de substituição e conectados através de *buffers* válidos o que garante a validade do sistema completo de acordo com a definição de sistema válido apresentado por Qiao et al. (2002). A confirmação da validade e o desempenho são avaliados através de simulação como uma ferramenta chamada CNP Tool.

Diferentemente dos trabalhos de Villaroel et al. (1989) e Zimmermann e Hommel (1999), na proposta desse trabalho as rotas de produção e a estrutura do sistema de manufatura são modelados juntos. Essa forma de abordagem evita a utilização de aplicativos de softwares para a realização da *linkagem* entre os dois modelos permitindo ainda uma boa flexibilidade com relação a parametrização de máquinas e alocação de recursos fixos e variáveis.

2.2. COMPARTILHAMENTO DE RECURSOS E CONVERSÃO DE REDES.

Contudo, os modelos desenvolvidos de acordo com a metodologia apresentada na Seção 2.1, ainda não resolvem o problema de como validamente inserir recursos compartilhado ao modelo. Na maioria dos casos em Sistemas Flexíveis de Manufatura, recursos compartilhados é a razão principal para paralisações completas.

No sentido de lidar com recursos compartilhados, trabalhos anteriores de Zhou and DiCesere (1991) propõe duas abordagens que estende o alcance das redes válidas para a utilização de recursos compartilhado com base na apresentação das seguintes definições: *Paraller Mutual Exclusion (PME)*; *Sequencial Mutual Exclusion (SME)*. Se os recursos compartilhados são adicionados de acordo com estes métodos, o modelo resultante permanecerá limitado, vivo e reversível, isto é, sua validade é conservada.

Segundo Zhou and DiCesere (1991), pela utilização das teorias de PME e SME nós podemos representar diferentes operações válidas com recursos compartilhados paralelos ou seqüenciais executadas pela a mesma máquina ou pelo mesmo por um conjunto de máquinas, desde que algumas restrições sejam obedecidas.

Apesar de essas abordagens representarem um considerável avanço nesse tipo de modelamento, ainda carece de definições que permitam, de forma segura, a transformação das redes de petri convencionais em CPN equivalentes. Nesse sentido, a proposta da teoria de dobragem baseada em marcas de Jensen e Rozenberg (1991) apresenta soluções bastante confiáveis. Nessa abordagem, a complexidade da representação gráfica transfere-se para as marcas, possibilitando que estas possam conter qualquer tipo de dados. Conseqüentemente, essa técnica implica em um aumento de complexidade também nas anotações associadas aos arcos e transições.

Na *Dobragem Baseada em Marcas*, cada transição passa a ter associada uma expressão *booleana*, denominada guarda, que pode devolver um valor verdadeiro ou falso. Os arcos passam também a ter expressões algébricas associadas capazes de efetuar seleção e transformações de dados. Dessa forma, além de permitir a redução das dimensões gráficas do modelo, a modelagem em CPN também possibilita o a manipulação de dados, incluindo suas transformações.

Como forma de atender essa necessidade de manipulação estruturada de dados, Falkman (2005) propôs uma linguagem de modelamento chamada *process algebra Petri Nets (PPN)* para modelar sistemas de alocação de recursos. A PPN é uma combinação de processo algébrico e rede de Petri. A linguagem de alto nível combina a capacidade de compactação da álgebra de processo com as características gráficas da rede de Petri. Dessa forma, a linguagem é capaz de desenvolver especificações concisas e de fácil leitura para sistemas complexos.

Adicionalmente, outras técnicas têm sido propostas e estudadas com o propósito de incrementar certo grau de inteligência e autonomia nas operações dos sistemas. Dentre estes trabalhos, Zha e Du (2001) sugerem que a lógica *fuzzy* pode ser usada para adicionar ao modelo a capacidade para lidar com conhecimento incompleto, impreciso e incerto. Esse modelo pode incorporar alguns dados dos produtos modelados baseado em conhecimento de características básicas, paramétricas e restrições através de aproximação de modelos.

A utilização das técnicas apresentadas por esses autores em conjunto com a capacidade de representação gráfica da rede de Petri constituem a base para a estruturação da técnica de modelamento em CPN desse trabalho. A validação teórica dessa proposta é garantida pelos trabalhos de Zhou and DiCesere (1991) que especificam as condições suficientes para a garantia da validade para sistemas que utilizam recursos compartilhados.

O software de simulação CPN Tool é usado como ferramenta operacional para a construção do modelo em CPN. Essa ferramenta foi desenvolvida por um grupo de pesquisa do Departamento de Ciências e Computação da Universidade de Aarhus, e é capaz de dar suporte à construção, simulação, análise funcional e de desempenho para

modelos usando CPN. Esse simulador trabalha em conjunto com uma linguagem de texto estruturada chamada CPN ML que é utilizada para as declarações de variáveis e inscrições de rede. Através dessa linguagem pode-se facilmente realizar condicionamento de ações, transformações de *tokens* e de tarefas em CPN.

Cada um dos trabalhos apresentados pelos autores aborda aspectos relevantes do modelamento de Sistemas Flexíveis de Manufatura a partir de redes de Petri. Contudo, existem poucas abordagens sistematizadas que consideram a implementação dos principais elementos dessas propostas do ponto de vista operacional. A proposta desse trabalho visa unificar as principais vantagens das técnicas de modelamento dos trabalhos desses autores para definir a construção de modelos a partir de uma metodologia sistematizada, utilizando, como ferramenta, simuladores de rede de Petri colorida hierárquica e temporizada.

3. PROCESSOS INDUSTRIAIS DE MANUFATURA

Com o desenvolvimento tecnológico, os processos de manufatura passaram a incorporar equipamentos e máquinas com controles cada vez mais complexos e autônomos conduzindo-nos aos sistemas de manufatura automatizados. Este capítulo mostra um resumo das principais técnicas de manufatura considerando seus níveis de produtividade e flexibilidade.

3.1. SISTEMAS DE MANUFATURA AUTOMATIZADOS

Segundo PALOMINO, (1995, p.40). Um sistema de manufatura (sistema produtivo) automatizado 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)

As principais características desse tipo de sistema são: a segmentação, o compartilhamento de recursos e o sequenciamento controlado da produção. Isso significa que o processo é subdividido em várias etapas de operação e subsistemas que precisam ser sincronizadas. Essas etapas são executadas através do fluxo pré-determinado de diversos itens como materiais, peças, recursos, informações etc, que passam seqüencialmente por estes subsistemas, onde são realizadas transformações apropriadas até obter o produto final desejado de acordo com especificações técnicas, geometria e de qualidade requerida pelos requisitos de produção.

Dessa forma, no processo produtivo como todo, o sistema é caracterizado pelo inter-relacionamento complexo de suas partes, operações e condições. Da teoria de sistemas dinâmicos a eventos discretos (Cassandra, 1990) observa-se que as interações entre seus elementos podem ser caracterizadas como de ocorrência discreta, síncrona e seqüencial dentro da perspectiva da operação normal, mas adquire características assíncronas se considerarmos a ocorrência de falhas.

Dada à natureza discreta das atividades envolvidas nos sistemas de manufatura automatizados, é conveniente utilizar modelos dinâmicos a eventos discretos para o estudo de suas propriedades de funcionamento e desempenho.

3.2. TIPOS DE PROCESSOS INDUSTRIAIS DE MANUFATURA

Segundo algumas características do processo produtivo, os sistemas de manufatura automatizados são comumente classificados em três grupos: sistemas do tipo *flow-shop*, *job-shop* e as células de manufatura.

3.2.1. Flow-shop

Também 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 (Fig. 3.1). O grau de flexibilidade neste tipo de sistema é dado a partir da configuração das máquinas ferramentas que compõem a linha.

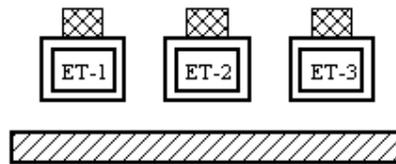


Fig. 3.1 – linha de produção do tipo flow-shop

3.2.2. Job-shop

é um tipo de produção repetitiva constituído por uma central de usinagem e um conjunto de ordens de serviço. O *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 que este sistema apresenta (já que para cada produto uma nova ordem de produção tem que ser feita), seu índice de produtividade é baixo se comparado com uma linha de produção do tipo *flow-shop*.

3.2.3. 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. 1993). As células de manufatura podem ser definidas como sendo uma configuração de estações de trabalho e um sistema de manuseio e transporte de materiais culminando em um sistema como o mostrado na Fig. 3.2 que será controlado por um computador para produzir eficientemente mais de um tipo de produto em pequenas e médias quantidades. As estações de trabalho podem ser robôs industriais, máquinas de comando numérico, sistemas de transporte e armazenamento, dispositivos de medição e dispositivos especiais para operações de montagem, etc., ou combinações destes equipamentos.

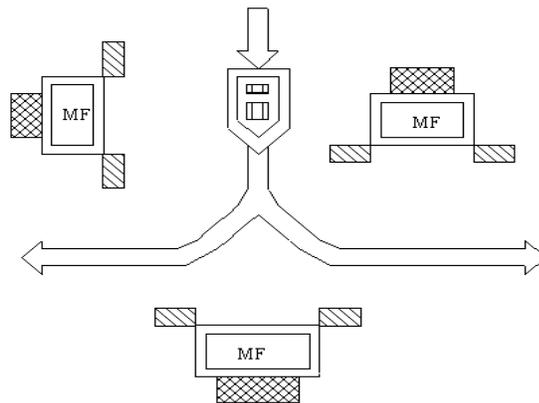


Fig.3.2 - Representação de uma célula de manufatura.

3.2.4. Sistema Integrado de Manufatura

Cada um dos subsistemas comentados nas anteriormente constitui elementos básicos de organização para a realização de processos de manufatura. Pela união desses subsistemas podemos 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 e de dados ligados ao sistema.

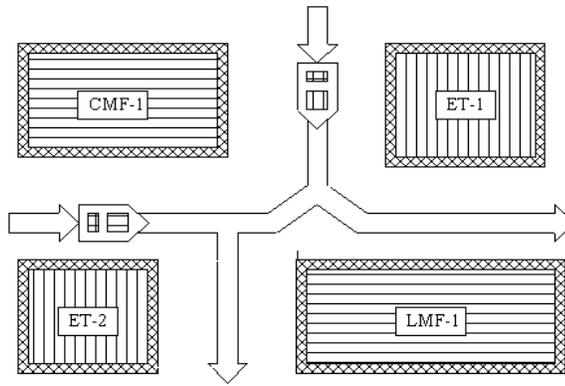


Fig.3.3 - Representação de um sistema integrado de manufatura

3.3 FLEXIBILIDADE EM PROCESSOS DE MANUFATURA

Os sistemas industriais de produção também podem ser classificados de acordo com a sua flexibilidade, ou seja, a facilidade que possui para migrar de um produto para outro com o mínimo de ajustes e como o menor tempo de *set-up*. Partindo dessa premissa, um sistema de produção automatizado pode ser classificado em três tipos básicos: automação fixa, automação programável e automação flexível.

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, porem há uma considerável perda de tempo nesse processo.

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 surgem o termo de manufatura automatizada, que engloba uma série de sistemas produtivos citados anteriormente, encontrando-se entre eles os Sistemas Flexíveis de Manufatura (FMS).

3.4 SISTEMAS FLEXÍVEIS DE MANUFATURA

Existem várias definições para Sistemas de Manufatura Flexíveis às quais guardam várias características essenciais em comum, mas que geralmente diferenciam-se um pouco no tipo de abordagem utilizada. Algumas consideram fundamentalmente as características produtivas do processo de uma forma mais generalizada com foco na quantidade e flexibilidade. Outras chegam a citar os subsistemas e máquinas que deveriam compor o processo, e algumas outras se concentram na questão dos agrupamentos hierárquicos entre nível produtivos.

Para Palomino (1995), os Sistemas de Manufatura Flexíveis (FMS) é classe de sistema 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. Nesse sentido, tanto as células de manufatura, como os sistemas integrados de manufatura poderiam ser incluídos nessa classificação.

Essa definição destaca às idéias de flexibilidade e níveis de produção que estão relacionados a esse tipo de processo de fabricação. Ou seja, a idéia principal de um Sistema de Manufatura Flexível é o de manusear diferentes produtos e suas partes, em pequenos volumes, em qualquer ordem e no tempo que sejam requeridos pela produção e com o mínimo de tempo de *set-up* de um produto para o outro. Dessa forma, o conceito de flexibilidade se estende além da possibilidade de processar mais de um tipo de produto ou peça e engloba também à capacidade de mudar rapidamente o tipo de produto a ser fabricado.

Para Rembold (1994) um sistema flexível de manufatura (FMS) é formado por máquinas-ferramentas com comandos numéricos computadorizados (CNC), interconectadas para transferência de informações (por um sistema de rede de

comunicação) e transferência de materiais (por um sistema de manipulação, troca e, transporte de peças e ferramentas com: esteiras, veículos transportadores, etc.).

Essa abordagem detalha o tipo de equipamentos, o tipo de comunicação de dados e estabelece dois fluxos fundamentais para a composição de um sistema de manufatura: fluxo de materiais e fluxo de informações.

Outros trabalhos como o de Chintamaneni (1988) e Hasegawa (1996) definem FMS como sistemas distribuídos, formados por vários subsistemas e, geralmente, organizados segundo uma estrutura de controle hierárquico. Um exemplo de hierarquia de níveis de controle baseada nos trabalhos de (Chintamaneni, 1988; Hasegawa, 1996) é apresentado a seguir:

- Nível de fábrica: composto por linhas de produção ou células de manufatura.
- Nível de célula de manufatura: composto por diversos equipamentos.
- Nível de equipamento: composto por diversos dispositivos.
- Nível de operação: que envolve as funções de cada um dos dispositivos.

Nesse sentido, as características que se deseja analisar num Sistema de Manufatura Flexível dependem do nível a ser focado. Em geral, as operações em níveis inferiores são mais críticas com relação às restrições de velocidade de processamento, sincronização temporal e precisão das informações já que nesse nível o controle geralmente é feito em tempo real. Já nas operações em nível superiores não há tanta exigência com relação às restrições de tempo real, pois a atualização dos dados não precisa ser tão rigorosa e o controle é, geralmente, do tipo qualitativo envolvendo grande quantidade de informações. Nos níveis mais altos a incorporação de procedimentos de gestão e otimização assume papel muito importante. Segundo Viswanadham (1990) “A alta produtividade neste tipo de sistema, pode ainda ser melhorada pela incorporação eficiente de princípios de tecnologia de grupo, controle total de qualidade e estratégias de produção, tais como MRP II, JIT”.

Dessa forma, pode-se afirmar que, de acordo com a literatura, para atender as características produtivas, de processo e de organização de um Sistema Flexível de

Manufatura, são necessários seis tipos de requisitos: flexibilidade de produção (capacidade de produzir diversidade de produto), flexibilidade do equipamento (máquinas com características adaptativas) e flexibilidade do processo (flexibilidade no sistema de controle e similaridade dos processos) e flexibilidade na comunicação (possibilidade de estabelecimento de comunicação entre os diversos arranjos do sistema), uma estrutura hierárquica de controle e incorporação de estratégias adequadas de produção.

Um Sistema Flexível de Manufatura seguro, também precisa ser capaz de se adaptar a distúrbios não previstos que possam ocorrer no processo tais como quebra de máquinas, trocas de ferramentas ou operação indevida.

Todas essas características podem ser conseguidas através da utilização de computadores, softwares de controle de SMS e sistemas de comunicação adequados. Esses softwares podem controlar todo o sistema, organizado a seqüência de operações de produção, respondendo adequadamente e em tempo real às atividades de produção planejadas, analisando critérios produtivos para a tomada de decisão quanto à antecipação ou retardo de necessidade recursos, reagindo em tempo real aos eventos não previstos.

Nessa dissertação utilizaremos uma situação problema para exemplificar e desenvolver uma metodologia modular de modelamento de um SMF utilizando os princípios e propriedade da CPN Hierárquica e temporizada, juntamente com ferramentas de simulação gráfica.

4. REDES DE PETRI

4.1 INTRODUÇÃO

As redes de Petri foram introduzidas pela primeira vez por Carl Adams Petri em 1962 em sua dissertação de doutorado apresentada na Universidade de Darmstadt, Alemanha. A partir de sua teoria, vários outros pesquisadores subsequentes propuseram e desenvolveram uma série de adaptações e extensões, dando origem a um conjunto de linguagens específicas, aplicáveis à modelagem do comportamento de numerosos tipos de sistemas, correspondentes a novas áreas de aplicação. Atualmente existe uma enorme diversidade de redes de Petri que foram desenvolvidas baseadas na obra original de Carl Petri da qual podemos destacar os trabalhos como os de Jensen sobre redes de Petri coloridas (Jensen, 1981), Soren Christensen, introduzindo os conceitos de canais síncronos (Christensen e Hansen, 1992) e as redes de Petri modulares (Christensen e Petrucci, 1992). Somado a isso, também existem propostas de integração de conceitos da programação orientada ao objeto nas redes de Petri.

Independente da extensão da rede de Petri em questão, a maior parte de suas vantagens resultam de suas características fundamentais, das quais, podemos destacar:

- Podem modelar concorrência, eventos assíncronos, relações de precedência lógica e interações estruturais numa maneira simples e natural.
- Dualidade estado-transição que possibilita, diferentemente da maioria das outras formas de representação gráfica, a representação do estado do sistema e da ação relacionada a ele ao mesmo tempo;
- RPs possuem uma forma gráfica fácil de entender na qual é possível visualizar sistemas complexos.

Dessa forma podemos dizer que todas essas características, associadas à grande flexibilidade que este tipo de Rede apresenta para aceitar extensões como as Redes de Petri Coloridas, Hierárquicas e Temporizadas torna a rede de Petri uma ferramenta muito útil para modelar, analisar e controlar sistemas complexos de manufatura.

4.2 DEFINIÇÃO DE REDE DE PETRI BASICA

Rede de Petri, na sua classe mais conhecida, rede de Petri **lugar-transição (RP)** é um gráfico orientado que tem dois tipos de nós: transição e posição. “Os nós são interligados através de arcos orientados que partem de alguma posição para alguma transição e vice-versa. Aos arcos são associados números inteiros (pesos). Cada posição pode ter um número inteiro de marcas (tokens) que podem, sob certas condições, mover-se ao longo dos arcos respeitando sua orientação” (Barros, 2006).

4.2.1 Definição formal da estrutura da rede de Petri

A rede de Petri lugar-transição marcada é um quintuplo $(P, T, A, W, \mathbf{m}_0)$ onde P é um conjunto de lugares, T um conjunto de transições, F um conjunto de arcos, W é uma função que atribui peso aos arcos e \mathbf{m}_0 a marcação inicial dos lugares, onde:

- $P = \{p_1, p_2, \dots, p_n\}$ é um conjunto finito de lugares, $n \geq 0$;
- $T = \{t_1, t_2, \dots, t_m\}$ é um conjunto de transições, $m \geq 0$.
- $A =$ é um conjunto finito de arcos pertencente ao conjunto $(P \times T) \cup (T \times P)$, em que $(P \times T)$ representa o conjunto dos arcos orientados de p_i para t_j , também designado por (p_i, t_j) ou $\text{Pre}(p_i, t_j)$, e $(T \times P)$ representa o conjunto dos arcos orientados de t_i para p_j , designado (t_i, p_j) ou $\text{Pos}(t_i, p_j)$;
- $\mathbf{m}_0 =$ é um vetor cuja i -ésima coordenada define o número de marcas (tokens) na posição p_i , no início da evolução da rede;
- $W : A \rightarrow \mathbb{N}^+ =$ é a função que atribui um peso w (um número inteiro) a cada arco orientado e também pode ser escrito como $w(p_i, t_j)$ ou $w(t_j, p_i)$;
- Os conjuntos T e P são disjuntos, ou seja, $P \cap T = \emptyset$;
- $m_i = m(p_i) \geq 0$ equivalente ao número de fichas tokens em um lugar específico p_i ;
- A marcação de todos os lugares é representada por um vetor n -dimensional $\mathbf{m} = [m_1, m_2, \dots, m_j]^T$, onde m_j é o número de fichas do lugar p_j ;
- $n_p = |P|$ é a cardinalidade do conjunto P , ou seja, o número de posições na RP;
- $n_t = |T|$ é o número de transições da RP.

A marcação de uma forma geral representa o estado da Rede de Petri, ou mais precisamente o estado do sistema descrito pela RP num determinado instante. Essa marcação é formada pelo posicionamento de um determinado número de *tokens* em

alguns ou todos os lugares da rede e pode ser representada por um vetor de estado. Ao conjunto de todas as marcações (vetores de estado) possíveis nós denominamos espaço de estado de Rede de Petri. A Fig. 4.1 mostra um exemplo de rede de Petri lugar-transição marcada, onde todos os arcos possuem peso 1, o que é característica de uma rede particular denominada *rede ordinária*.

Esta RP é:

$$P = (\{ p1, p2, p3 \} ,$$

$$T = \{ t1 \} ,$$

$$F = \{ (p1, t1), (p2, t1), (t1, p3) \}$$

$$M_0 = \{ (p1, 1), (p2, 1), (p3, 0) \}$$
 ou

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

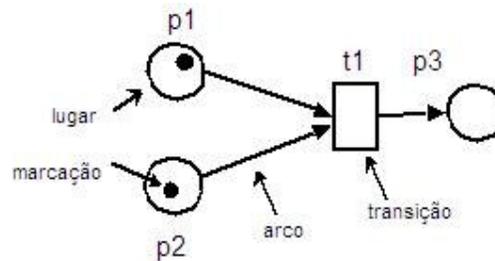


Fig.4.1 - Representação de uma RP lugar-transição ordinária marcada

Uma marcação vetor “**m**” é, portanto, uma aplicação que associa o cada lugar na RP um inteiro não negativo “ m_i ” chamado de ficha ou *token*. 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 Fig. 4.2, onde a marcação **m**₀ (marcação inicial) está dada por **m**₀ = [0, 3, 0]^T para uma RP não ordinária.

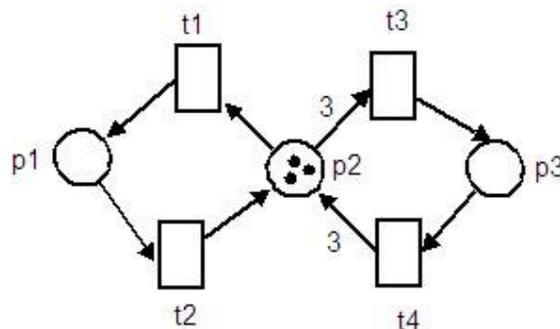


Fig.4.2 - Representação de uma RP lugar-transição marcada não ordinária .

4.2.2 Dinâmica da Rede de Petri

Quando modelamos um sistema através de Redes de Petri procuramos representar os estados antes e depois de cada evento neste evento. O elemento responsável por indicar qual o estado de uma rede de Petri é o tokens (marca). A marcação inicial faz parte do projeto do modelo e deve ser considerada para garantir as propriedades de interesse. O disparo de uma transição na PN representa a ocorrência de um evento que altera o estado do sistema, isto é altera a marcação atual (M_i) para uma nova marcação (M_{i+1}).

Habilitação da transição:

Uma marcação ($t_i \in T$) fica habilitada para disparar se e somente se todo lugar de entrada de uma transição p_j , contém o número de marcas maior ou igual ao número do peso do arco $w(p_j, t_i)$.

Formalmente:

$$\forall p \in P, M(p_i) \geq \text{Pre}(p_i, t_j).$$

Disparo de transição:

A transição dispara se está habilitada. Uma transição é disparada por meio de duas operações que correspondem à ocorrência de um evento:

- a) Remove marcas das posições do pré-set (tantas marcas quanto for o peso do arco correspondente) e
- b) Depositam em cada uma das posições do pós-set tantas marcas quanto for o peso correspondente.

Dessa forma, o disparo de t provoca mudanças na marcação da rede de \mathbf{m} para \mathbf{m}' , de acordo com a equação a seguir:

$$\forall p \in P, m'(p_j) = m(p) - \text{Pre}(p_i, t_j) + \text{Pos}(p_i, t_j).$$

A Fig. 4.3 mostra a marcação inicial e o resultado da ocorrência do disparo de da transição.

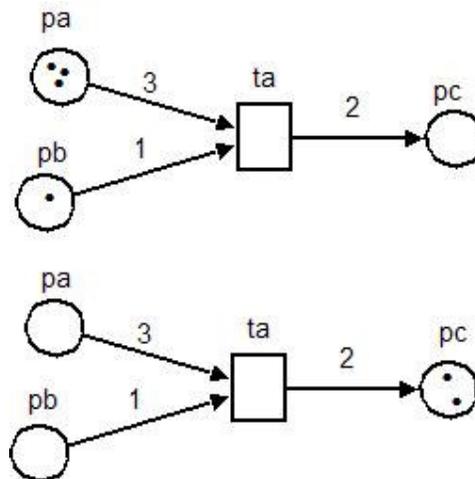


Fig.4.3 - Representação de uma RP lugar-transição marcada .

4.3 PROPRIEDADES FUNDAMENTAIS DAS RPs

As propriedades para um modelo de Redes de Petri aplicada a sistemas de manufatura podem ser classificadas em propriedades comportamentais, estruturais e características de desempenho:

4.3.1 Propriedades comportamentais (Procedurais)

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 (espaço de estado). As propriedades procedurais são fundamentais e básicas para estudar as características 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.

a) Limitação: Uma RdP pode ser dita como k-limitada. Em uma RdP 3-limitada, nenhum dos *places* pode exceder o número de *tokens* definido por k, neste caso, três.

b) Segurança: A propriedade de segurança é um complemento à propriedade limitação. Um *place* k-limitado é um *place* seguro. Se estivermos modelando uma porta lógica, a RdP dever ser 1-limitado. A RdP é segura se em todos os seus estados os *places* tiverem um ou nenhum token (1 ou 0 - binário).

c) Vivacidade: Uma RdP livre de impasses (*deadlocks*). Neste caso não existe a ocorrência de uma transição que não possa ser disparada. Uma transição morta é uma transição que não pode mais ser disparada. Uma RP morta se encontra em impasse total, ou seja, todas as transições estão mortas.

d) Cobertura: Quando uma determinada marcação pode ser obtida através de uma outra marcação. Este é o problema cobertura de uma marcação.

e) Persistência: Se para qualquer par de transições habilitadas, o disparo de uma delas não desabilita o disparo da outra.

f) Reversibilidade: Quando existe o retorno à marcação inicial ou uma outra marcação qualquer.

4.3.2 Propriedades estruturais

São aquelas que só dependem da estrutura topológica das RPs. 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".

a) Limitação: Se a RP é limitada estruturalmente para qualquer marcação inicial, o número de fichas em cada lugar não excede um número finito k para qualquer marcação alcançável desde M .

b) Conservação: Independente dos disparos das transições existentes na RP, o número de *tokens* não se altera (não há criação ou destruição).

c) Repetitividade: Uma RP é repetitiva se para uma determinada marcação e um conjunto de disparos de transições, houver o disparo ilimitado de todas as transições da RP.

d) Consistência: Uma RP é consistente se for possível voltar ao estado inicial M_0 disparando pelo menos uma vez todas as transições da RP.

Dentre as propriedades procedurais e estruturais úteis no estudo dos sistemas de manufatura destacam-se: vivacidade, reversibilidade, limitação. A vivacidade é muito importante em sistemas de manufatura com processos concorrente e

compartilhamento de recursos desde que *deadlock* pode ocasionar parada na produção (Zhou e Dicesare, 1989). Reversibilidade significa que o sistema em algum momento vai retornar para seu estado inicial de algum de seus estados, incluindo estados de falha. Em ambientes de sistema de manufatura a limitação garante que nenhum buffer vai sofrer problema de “*overflow*”, ou seja, todos os *buffers* vão ter uma capacidade limitada.

4.3.3 Características de Desempenho

Além das propriedades citadas na Seção 4.3.2, para garantir a eficiência do sistema, também são necessárias avaliações de propriedades de desempenho do sistema. Com o objetivo de analisar tais propriedades, tempos de atrasos são associados com as transições e com os arcos orientados, caracterizando um tipo especial de rede denominado *Timed Petri Net*.

Resumidamente, podemos dizer que as propriedades estruturais e comportamentais estão relacionadas com a validade do sistema, ou seja, se o mesmo possui estados que impeçam o correto funcionamento ou que tendam a levar o mesmo para situações impossíveis de serem alcançadas pelo sistema real. Já as propriedades de desempenho estão preocupadas com o desempenho do sistema dentro de determinadas condições de configuração.

4.4 DEFINIÇÃO FORMAL DAS PROPRIEDADES DAS RPs

4.4.1 Alcançabilidade

Definição - Na RP (P,T,A,W,m_0) denomina-se de alcançabilidade de uma marcação m_0 (representada por $Ac(R;m_0)$) ao multi-conjunto de todas as marcações (estados) geradas a partir de m_0 .

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 $s = t_1 t_2 t_3 \dots t_n$. Neste caso m_n é alcançável desde m_0 por "s", podendo-se escrever então: $m_0 [s \rightarrow m_n]$.

4.4.2 Limitação e Segurança:

Definição - Uma RP marcada $N = (P,T,A,W,m_0)$ é dita ser k-limitada ou simplesmente limitada se o número de fichas em cada lugar não excede um número finito k para qualquer marcação alcançável desde M_0 , isto é:

$$\exists k \in \mathbb{N} / m(p) \leq k, \forall m \in Ac(R, m_0) \quad (4.1)$$

A *segurança* é um caso especial de uma rede limitada, ou seja, quando $k = 1$ diz-se que o lugar é seguro, salvo ou binário. Por exemplo, na rede da Fig. 4.4.a, cada marcação m' , a qual pode ser alcançável desde m_0 , tem no máximo uma ficha em p. Dessa forma, rede marcada N é segura se e somente se todos os seus lugares são seguros, diz-se também *salva* ou *binária*.

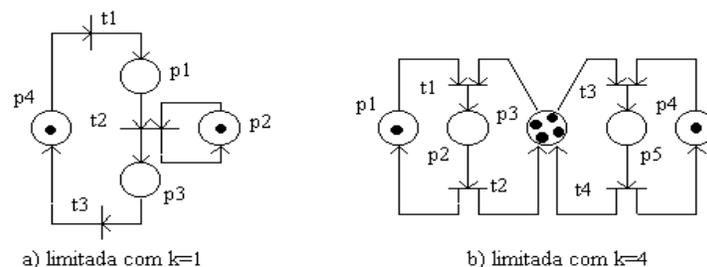


Fig.4.4 – Redes de Petri marcadas k-limitadas

4.4.3 Vivacidade

O conceito de vivacidade está relacionado com a total ausência de auto-bloqueios (*deadlock*)

Def. 01. As transições podem ser vivas se for possível sempre encontrar uma seqüência de disparos que as contém, ou quase viva se for somente alcançáveis uma vez.

Def. 02. Uma transição é viva, dado um estado inicial \mathbf{m}_0 , se e somente se ela é habilitada a partir de algum estado decorrente de \mathbf{m}_0 .

Uma RP é viva, dado um estado inicial \mathbf{m}_0 , se e somente se todas as suas transições são vivas.

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

$$\forall \mathbf{m} \in \text{Ac}(\mathbf{R}, \mathbf{M}_0), \exists \mathbf{m}' \in \text{Ac}(\mathbf{R}, \mathbf{m}) \text{ que também é habilitada por } \mathbf{m}'. \quad (4.2)$$

4.4.4 Reversibilidade/Reinicialização

Definição.- Uma RP marcada $N = (\mathbf{R}, \mathbf{m}_0)$ é reinicializável para toda marcação se e somente se seu grafo de marcações acessíveis $\text{GA}(\mathbf{R}, \mathbf{m})$ é fortemente conexo, isto é:

$$\forall \mathbf{m}' \in \text{Ac}(\mathbf{R}, \mathbf{m}_0) \exists s \text{ tal que } \mathbf{m}' \xrightarrow{s} \mathbf{m}_0 \quad (4.3)$$

Exemplo: Considerando a RP da Fig. 4.5 cujo grafo de marcações é dado pela Fig. 4.6 ela é não reinicializável, pois não existe nenhuma seqüência que permite voltar à marcação inicial $\mathbf{M}_0 = p1 \ p4$ após o disparo da transição t_1 .

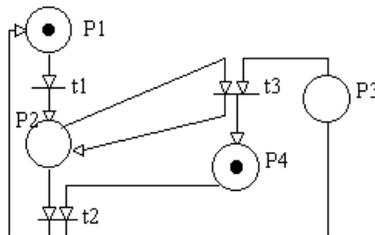


Fig.4.5 – Exemplo de uma rede de Petri não reinicializável.

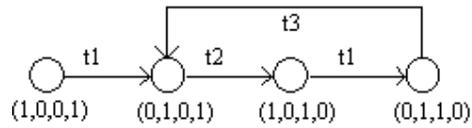


Fig.4.6 – Grafo de marcações acessíveis da rede não reinicializável.

No entanto, se considerarmos para a rede da Fig. 4.5 a marcação $\mathbf{m}_0 = p_2 p_4$, ela passa a ser ao mesmo tempo viva e reinicializável. Assim, conclui-se que a reinicialização depende da marcação inicial e da estrutura da rede.

4.5. TÉCNICAS DE ANÁLISE DE REDE DE PETRI

Nas sessões anteriores pudemos verificar as principais propriedades de Rede de Petri de interesse para sistemas de manufatura, agora se faz necessário determinar se uma RP proposta possui ou não tais propriedades. Com esse sentido, desenvolveu-se certas metodologias de verificação com base nas árvores de alcançabilidade e nas matrizes de incidência:

4.5.1. Análises pela Árvore de Alcançabilidade e de Cobertura

Este método envolve essencialmente a enumeração de todas as marcações alcançáveis feitas através de um gráfico do tipo árvore em que os nós são do tipo vetores de estado, alcançados sucessivamente e alternativamente pela rede, e os arcos são as correspondentes transições executadas. A dinâmica simula a execução da RP a partir de uma marcação inicial. Conforme dito, o grafo de marcações acessíveis pode ser ilimitado ou exponencialmente grande para redes complexas, o que gera uma dificuldade de implementação e análise. Nesse sentido, a utilização de um software simulador de boa qualidade e com recursos adequados à identificação de certas propriedades executado por computadores com boa capacidade de processamento é muito importante para superar essa questão. Além disso, muitos algoritmos para tentar resolver o problema de grafo ilimitado foram propostos, dentre estes, um que permite decidir se uma RP é k -limitada foi proposto baseado na construção de uma árvore chamada "**árvore de cobertura**".

Árvore de Cobertura

Conforme visto, o Grafo de Marcações Acessíveis de uma RP pode ser ilimitado. Para torná-lo limitado define-se um ente algébrico ω , chamada “pseudo-infinito” e constrói-se um novo grafo denominado de *Árvore de Alcançabilidade* que terá dois tipos de nós, o nó raiz (igual a \mathbf{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 que foi proposto por Murata (1989):

Algoritmo:

1. Cria-se \mathbf{m}_0 como raiz da árvore e nó fronteira;
2. Enquanto existir nó fronteira fazer
3. Selecionar uma marcação \mathbf{m} ;
4. Verifique se existir outra marcação igual a \mathbf{m} na árvore, considerá-la duplicada e finalizar seu processamento;
5. Se \mathbf{m} não habilitar nenhuma $t \in T$, considera-la terminal e finalizar seu processamento;
6. Para cada $t \in T$ habilitada por \mathbf{m} fazer:
7. Obter \mathbf{m}' disparando t ;
8. Se no caminho que liga \mathbf{m}' a \mathbf{m}_0 existir outra marcação \mathbf{m}'' , tal que \mathbf{m}' é cobertura de \mathbf{m}'' , isto é $\mathbf{m}'(p) \geq \mathbf{m}''(p)$, fazer $\mathbf{m}'(p) = \mathbf{m}''(p) + 1$, onde $\mathbf{m}'(p) > \mathbf{m}''(p)$;
9. Considerar \mathbf{m}' como fronteira.
10. Finalizar o processamento quando todos os nós forem classificados em terminal, duplicado ou interno.

Como exemplo, vamos considerar a RP mostrada na Fig. 4.7 para a marcação inicial $\mathbf{M}_0 = (1, 0, 0)$. Só a transição t_1 está habilitada.

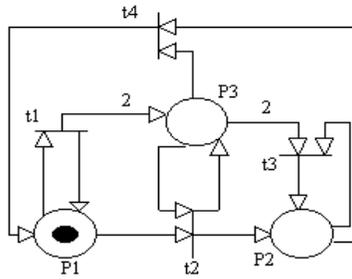


Fig.4.7 – Rede de Petri marcada com possível Deadlock

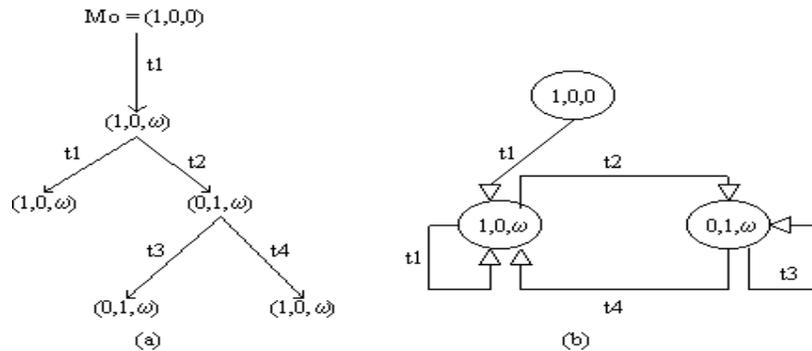


Fig.4.8 – a) Árvore Alcançabilidade e b) Grafo de Marcação acessíveis para Fig. 4.7.

Propriedades Analisadas pela Árvore de Cobertura

a) Limitação - A RP é limitada (para qualquer marcação inicial) se e somente se o " ω " não aparecer na árvore de cobertura. A aparição do símbolo " ω " como parte da árvore de cobertura significa que o número de fichas nesse lugar é potencialmente ilimitado, assim a rede é ilimitada. Se ω não aparece pode-se ainda determinar a k-limitação pelo exame de todos os nós da árvore. A maior de todos os estados fornece o k da k-limitação, que em um sistema de manufatura determina o tamanho do maior estoque intermediário.

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

c) Alcançabilidade - m é sempre alcançável, se aparecer explicitamente em todos os ramos da árvore de alcançabilidade.

d) Vivacidade - Se não existir nós duplicados é sempre morta. Essa condição é suficiente, mas não necessária.

Limitações da Árvore de cobertura e na Análise das Redes de Petri

A análise pela árvore de cobertura pode ser usada para solucionar os problemas de cobertura, limitação e conservação, mas ela é limitada para ser usada para solucionar os problemas de alcançabilidade ou vivacidade ou para definir ou determinar que seqüências de disparo sejam possíveis. Esses problemas são limitados pela existência do pseudo-infinito (ω), pois o mesmo causa uma perda de informação sobre a RP. Por exemplo, pode haver uma mesma árvore representando mais de uma.

Um problema similar acontece para o problema de vivacidade. As Figs. 4.7 e 4.9, são duas RPs cuja árvore de alcançabilidade é dada pela Fig. 4.8 (a), entretanto a Fig. 4.7 pode morrer após a seqüência $s = t_1 t_2 t_3$ por exemplo, por causa da existência de um possível nó terminal (*deadlock*), enquanto que a RP da Fig. 4.9 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 RP, para uma nova marcação m_0 , tem-se que construir uma nova árvore, pois as propriedades podem ser alteradas.

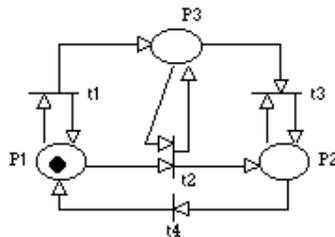


Fig.4.9 – Rede de Petri sem Deadlock

4.5.2. Análise por Simulação e Composição

Quando a complexidade de sistemas do mundo real nos leva a construção de uma rede muito grande, torna-se impossível analisar toda ela utilizando o grafo de alcançabilidade ou métodos invariantes. Esse problema se agrava ainda mais quando

utilizamos as chamadas redes de Petri de alto nível como é o caso das Redes de Petri Coloridas (CPN) em função de que as transições, arcos e lugares podem assumir um conjunto muito maior de possibilidades de habilitação e disparo. Contudo, na prática, muitas CPN em sistemas de manufatura possuem conjunto de estados de espaço finito, o que possibilita uma verificação completa do sistema. Por meio de bons simuladores e de computadores com boa capacidade de processamento, o estado de espaço de dessas PN podem ser completamente construída automaticamente, e proveniente desses espaços de estado construído é possível responder a um grande conjunto questões de verificação importante para os sistemas de manufatura. Além do mais, se o sistema não tiver sido projetado com as propriedades adequadas, o método de estado de espaço por simulação pode fornecer informações sobre nós específicos ou subconjunto do estado de espaço onde ocorre o problema. Além do mais, frequentemente é possível provar e refutar propriedades de um sistema confiando em espaços de estado parciais, ou seja, subconjuntos finitos de um espaço de estado completo A. Zimmermann et al (1996).

Junto com a simulação, a composição pode ser muito útil para superar o problema de elaboração e validação de modelos extensos. Com esse tipo de modelamento podemos construir submodelos Petri Nets básicos cuja validade é fácil de provar através das técnicas convencionais ou por simulação, e através de algum tipo de composição construir modelos maiores e mais complexos que mantenham suas propriedades.

A utilização de simulação também é importante para análise de propriedades de desempenho do sistema como *throughput*, percentagem de utilização de máquinas, tamanho de *buffers* entre outros. “Contudo, para o estudo de desempenho do sistema é necessário incluir a noção de tempo e probabilidade dentro do modelo” (Barros, 2006). Isto é usualmente feito pela associação de tempos de retardo aos “tokens” nas transições ou nos arcos direcionados, o que vem a constituir as redes temporizadas. Dessa forma, as propriedades analisadas nas redes temporizadas podem determinar a eficiência do modelo e permitir para o projetista realiza os ajustes necessários para obter a otimização do sistema.

4.5.3 Importância do *Set-up* na Análise por simulação

Como citado anteriormente, uma das principais inconveniências da análise de espaço de estado é a ocorrência de um número ilimitado de nós inviabilizando ou dificultando uma análise por simulação suficientemente apurada. Desta forma, muitas vezes, o primeiro passo em análise de espaço de estado é fazer algumas modificações para que o modelo CPN com o objetivo de torná-lo tratável por análise de espaço de estado. Nesse sentido, o significado de tratável é a “obtenção de um modelo com um número finito de estados de espaço e dentro dos limites dos recursos computacionais” (Zimmermann et al., 1996). Para isso podemos utilizar várias estratégias como definir um número limitado de *tokens* na marcação inicial, definir um número limitado de estoques intermediários, ou gerar contadores para a que em um determinado número de passos de simulação o mesmo pare.

4.5.4 Análise por Redução

Muitos pesquisadores têm desenvolvido técnicas de redução 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 RP 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 RPs Predicado/transição, Rede de Petri Coloridas (Jensen 1981) e algumas RPs estendidas, podem ser usadas também para superar o problema de complexidade.

4.6 REDE DE PETRI COLORIDA

As redes de Petri não coloridas nas suas extensões Petri Nets Estocástica e Petri Nets Estocásticas Generalizadas são duas das mais populares no modelamento de sistemas de manufatura. Todavia, se mais de um produto é processado por uma máquina, devido à falta de distinção entre os *tokens*, o modelo das máquinas tem que ser replicada tantas vezes quanto forem os diferentes *tokens*, fazendo com que o modelo fique muito confuso e extenso, além de não conseguir representar a estrutura real do sistema (Zimmermann e Hommel, 1999). Nesse sentido, as redes de Petri coloridas (CPNs) e tem têm por objetivo reduzir o tamanho do modelo, permitindo que os *tokens* sejam individualizados através de cores atribuídas a eles. Assim, diferentes processos ou recursos podem ser representados em uma mesma estrutura de rede por diferentes cores. As cores não significam apenas padrões, elas podem representar tipos de dados complexos, usando a nomenclatura de colorida apenas para referenciar a possibilidade de distinção entre os *tokens*.

Uma CPN consiste de três partes principais: Estrutura, Inscrição e Declarações.

Estrutura é um grafo dirigido com dois tipos de vértices (lugares e transições). Os lugares são representados graficamente por círculos (ou por elipses) e as transições por retângulos. Essa representação herda a propriedade das redes coloridas originais de podem armazenar em cada lugar marcas de tipos diferentes, além de poder representar valores associados a tipos de dados mais complexos.

Declarações compreendem a especificação dos conjuntos de cores, declarações de variáveis, constantes e funções.

Inscrições variam de acordo com o componente da rede. Os lugares possuem três tipos de inscrições: nomes, conjunto de cores e expressão de inicialização (marcação inicial). As transições têm dois tipos de inscrições: nomes e expressões de guarda. Já os arcos possuem apenas um tipo de inscrição dado pela expressão. Como formas para distinguir as inscrições, nomes são escritos com letras normais, cores em itálico, expressões de inicialização sublinhadas e as expressões guarda são colocadas entre colchetes. Nomes, quando associados aos lugares, não têm significado formal, apenas facilitam a identificação. As expressões de guarda associadas às transições são

expressões *booleanas* que devem ser atendidas para que seja possível o disparo das transições.

4.6.1. Definição formal da estrutura da CPN

Nessa sessão definiremos formalmente rede de Petri colorida e Hierárquica com o propósito

- Uma CPN não-hierárquica é uma tupla
($\Sigma, P, T, A, N, C, G, E, IN$):

Σ = conjunto finito, não vazio de tipos (*color sets*).

P = conjunto finito de lugares (*places*).

T = conjunto finito de transições.

A = conjunto finito de arcos tal que:

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

N = função de nó: $A \rightarrow P \times T \cup T \times P$

C = função de cor: $P \rightarrow \Sigma$.

G = função de guarda. É definido como uma função de T em uma expressão, tão que:

- $\forall t \in T: [Type(G(t)) = Boolean \wedge Type(Var(G(t))) \subseteq \Sigma]$.

E = função de expressões de arcos: $A \rightarrow$ expressões

- $\forall a \in A: [Type(E(a)) = C(p)MS \wedge Type(Var(E(a))) \subseteq \Sigma]$

Onde p é o lugar de $N(a)$.

IN = função de inicialização: $P \rightarrow$ *marcações* $0(p)$

4.6.2. Descrição formal do comportamento

- A distribuição de *tokens* é uma função $M: P \rightarrow \Sigma$ e uma marcação de uma Rede de Petri Colorida é uma distribuição de *tokens*. A marcação inicial das redes coloridas é obtida através da avaliação da expressão de inicialização para cada *place* da rede, ou seja, $M_0(p) = I(p) \Leftrightarrow \forall p \in P$.

- A distribuição de ligações é uma função $Y: T \rightarrow B$. Denomina-se elemento de Y o par (t, b) , onde b é uma ligação tal que $b \in Y(t)$.

- Variáveis são ligadas (*binding*) dinamicamente aos valores dos *tokens* nos *places* de entrada, durante a avaliação de cada transição.
 - Cada possível ligação é avaliada, uma por vez, em ordem aleatória.
 - Todas as ocorrências de uma variável devem ser ligadas a uma mesma cor.
 - Variáveis associadas a diferentes transições são totalmente independentes.
- Um passo é um multi-conjunto de elementos ligados ou uma distribuição de ligações não-vazia.
 - Um passo Y é habilitado em uma marcação M se a seguinte propriedade é satisfeita: $\forall p \in P: \sum_{(t,b) \in Y} E(p,t) \langle b \rangle \leq M(p), \forall p \in P$.
 - Quando um passo Y é habilitado em uma marca $M1$ ele pode ocorrer, mudando a marcação $M1$ para outra marcação $M2$, definida por: $\forall p \in P: M2(p) = (M1(p) - \sum_{(t,b) \in Y} E(p,t) \langle b \rangle) + \sum_{(t,b) \in Y} E(t,p) \langle b \rangle$, onde $E(p,t)$ e $E(t,p)$ correspondem às expressões dos arcos de entrada e saída da transição (em t) do passo, respectivamente.
 - Em outras palavras, um passo $Y = (t, b)$ está habilitado quando o número de marcas de cores correspondente à ligação do passo é maior ou igual à avaliação da expressão do arco que interliga os lugares de entrada à transição t .
 - A primeira soma é chamada de remoção de marcas (dos lugares de entrada) enquanto a segunda é chamada de adição de marcas (aos lugares de saída). Além disso, dizemos que $M2$ é diretamente alcançável de $M1$ pela ocorrência do passo Y , no qual denotamos: $M1 [Y > M2$.
- Uma seqüência de ocorrência é uma seqüência de marcações e passos:
 - $M1 [Y1 > M2 [Y2 > M3 \dots Mn [Yn > M_{n+1}$ tal que $Mi [Yi > M_{i+1}$ para todo $i \in 1 \dots n$ é alcançável de M .

A Fig. 4.10 apresenta uma rede colorida, possuindo a representação original, onde são realmente utilizadas cores para os *tokens*. Nessa figura, os arcos são rotulados com cores (a, b, c).

No exemplo da Fig. 4.10, utiliza-se o modo mais elementar de redes coloridas, no qual se associa ao arco uma determinada cor, assim, o *token* se destinará ao arco cuja cor for idêntica a da marca. Observando-se essa figura, pode-se perceber que os *tokens* de P0 não habilitarão a transição t0, pois o arco que liga P0 a t0 só aceita cores do tipo “a”, e o lugar P0 só possui marcas do tipo “d”. Em contrapartida, P1 possui marcas do tipo “a”, podendo habilitar a transição t1.

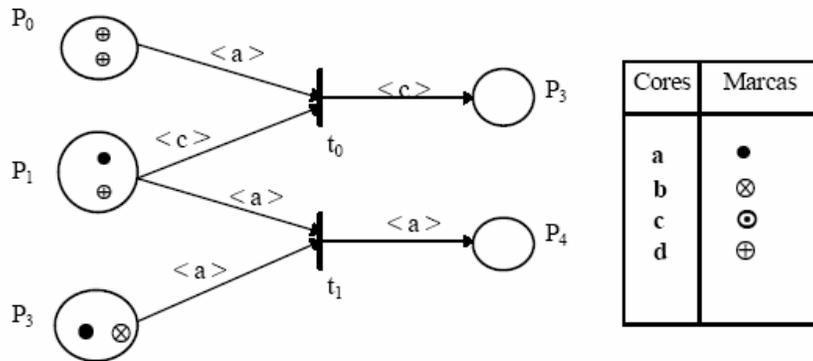


Fig.4.10 – Exemplo de rede de Petri colorida original

Ainda que um tanto quanto rudimentar, a rede colorida original provê mecanismos que possibilitam efetuar uma escolha determinística. Esse poder de escolha já significa um grande avanço em direção a uma representação mais clara de um modelo, porém modificações (acréscimos) posteriores vieram dar maior adequação às redes coloridas, com relação à representação das escolhas não-determinísticas. Uma das contribuições desse trabalho é identificar quais propriedades poderiam ser acrescentadas às CPN originais de forma que possam facilitar a construção de modelos da área de manufatura. Faz-se referência às redes coloridas com as melhorias adicionais, chamando-as somente por redes de Petri coloridas.

Para ilustrar a aplicação de Redes de Petri Coloridas, será mostrada uma situação clássica de geração de impasse: o jantar dos filósofos. Essa situação consiste de três filósofos que podem estar em três estados diferentes: comendo, pensando ou com fome. Os filósofos estão à volta de uma mesa, sendo que cada um deles tem à sua frente um garfo e um prato de comida. São, no entanto, necessários dois garfos para que um filósofo possa comer, ou seja, um filósofo precisa do seu garfo e do de seu vizinho. O impasse ocorrerá quando todos os filósofos pegarem o garfo da direita e

aguardarem a liberação do garfo da esquerda. A Fig. 4.11 apresenta o jantar dos filósofos modelado em rede de Petri colorida.

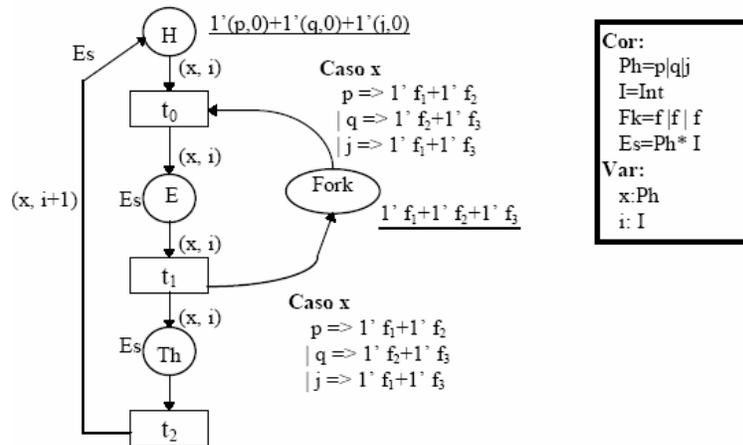


Fig.4.11 – Jantar dos filósofos modelado em rede de Petri colorida

Na Fig. 4.11, os lugares representam os estados possíveis para os filósofos (H para “com fome”, E para “comendo” e Th para “pensando”) e os recursos do sistema (no caso, os garfos representados por Fork). A variável x indica o filósofo que irá passar para o estado “comendo” (E) e a variável i indica o número de iterações que já ocorreram. Na primeira iteração, o lugar H possui três marcas (marcação inicial) que estão sublinhadas. Qualquer das marcas residentes na posição H pode ser consumida pela expressão (x, i) já que essa expressão não identifica um valor fixo e sim um conjunto cartesiano de valores e não existe nenhuma prioridade especificada.

Dependendo da atribuição dada à variável x , uma das três expressões é avaliada no arco que liga o lugar Fork à transição t_0 . Assim, se, por exemplo, $x = p$, então a expressão $1' f_1 + 1' f_2$ é avaliada e apenas a marcação $1'(p, 0)$ do lugar H irá à transição t_0 , significando que apenas o filósofo p irá passar para o estado “comendo”. Desta forma, para cada atribuição de x (p , q ou j) haverá uma situação diferente, ou seja, uma marca diferente no lugar E.

4.6.3. Combinação de CPN com outras extensões

As redes de Petri também oferecem extensões para descrição hierárquica, ou seja, nós podemos construir uma grande CPN através da interligação de CPN menores em uma maneira bem definida. Esse tipo de construção dar uma facilidade no desenvolvimento de similares à utilização de sub-rotinas. A existência de hierarquia possibilita modelar grandes sistemas de uma forma gerenciável e modular.

Além disso, as CPN podem ser estendidas com concepção de tempo, ou seja, é possível utilizar a mesma linguagem de modelamento para validação de propriedades lógicas (tais como ausência de *deadlocks*) e propriedades de desempenho (como tempo médio de espera, *throughput*). A idéia geral por trás da extensão de tempo é introduzir um relógio global e possibilitar para cada *token* uma marca adicional de tempo ao valor de dado que ele já carrega. Intuitivamente, a marca de tempo especifica o tempo em que o toquem está pronto para ser usado, ou seja, consumido pela transição.

4.7. REDES DE PETRI TEMPORIZADAS

Até o momento estudamos RP com transições que disparam a desde que os *places* de entrada possuam *tokens* suficientes para a ocorrência do disparo. No entanto, pode ser necessário associar ao disparo um retardo. A introdução de especificações de tempo é essencial se queremos usar esta classe de modelo para considerar, por exemplo, desempenho, agendamento de operações, controle em tempo real e assim por diante. Em muitas áreas como processos de manufatura, protocolo de comunicação, análise de sistemas de softwares, tempo é essencial para assegurar que o sistema está correto (Barreto, 2005). Sendo assim, a introdução de temporização também deve adicionar mecanismos para computação de métricas de desempenho. Existem diferentes maneiras para incorporar tempo em uma Petri Net. Os tempos podem ser associados com lugares, *tokens*, arcos e transições. Desde que transições representam atividades que mudam o estado (marca) da rede, parece natural associar tempo à transições.

Segundo Murata (1989) podemos extrair os seguintes significados dos retardos inseridos em uma rede de Petri:

- Tempo de aquisição de um recurso.
- Tempo de utilização de um recurso.
- Tempo em que não necessita de um recurso.

O disparo de uma transição em um modelo de Rede de Petri corresponde a um evento que muda o estado do sistema real. Segundo Barreto (2005), existe duas diferentes políticas de disparo em uma rede temporizada:

- Disparo em três fases: uma primeira fase instantânea em que a transição habilitada remove *tokens* de seus lugares de entrada, depois uma fase de temporização em que a transição está trabalhando, e uma fase final instantânea em que os *tokens* são depositados dentro dos lugares de saída. Essa informação de tempo é chamada duração;
- Disparo atômico: Os *tokens* se mantêm nos lugares de entrada durante todo o retardo de disparo; após esse período eles são consumidos dos lugares de entrada e gerados nos lugares de saída quando a transição dispara. O disparo em si não consome nenhum tempo.

Muitas extensões de Rede de Petri Temporizada têm sido propostas e adotadas pela comunidade científica, contudo, como resultado das pesquisas realizadas nas diversas literaturas especializadas no assunto a que mais se adequa a aplicações em Sistemas de manufatura flexíveis é a extensão denominada *Time Petri Net*.

Segundo Barreto (2005), *Time Petri Net* é definida como (PN, I) , onde PN é uma Rede de Petri de base e I é um intervalo de tempo expressando restrições de tempo, onde $I_i = (EFT_i, LFT_i)$ associado com cada transição t_i . EFT expressa o início do tempo de disparo e LFT expressa o final do tempo de disparo.

Um transição t_i habilitada somente poderá disparar em um intervalo $EFT_i < \delta < LFT_i$, ou seja, t_i estará continuamente habilitado quando o tempo global for pelo menos maior que EFT_i unidades de tempo. Mas o que acontece quando a transição t_i é habilitada por LFT_i ? O conceito de modo de disparo está relacionado com esta questão.

Existem dois modos de disparo: modo de disparo forte e modo de disparo fraco. Considerando que a transição t_i é habilitada no tempo θ . De acordo com o modo de disparo forte, uma transição é forçada a disparar no tempo $\theta + LFT_i$, se t_i não for disparado ou não for desabilitado pelo disparo de outra transição. O modo de disparo fraco, por outro lado, não força uma transição habilitada disparar, isto é, uma transição habilitada pode ou não disparar. O simulador CNP Tools que será empregado nesse trabalho utiliza uma extensão *Time Petri Net* com modo de disparo fraco.

O leitor deve notar que time Petri nets são equivalentes a redes de Petri padrão, se todo $EFT = 0$ e todo $LFT = \infty$. Também é importante notar que o conjunto de marcas alcançáveis de Petri nets é igual ou um subconjunto de seu modelo equivalente não colorido. Isto é verdade porque as regras de habilitação para o modelo temporizado são as mesmas que no modelo não temporizado. A única diferença é dividida às restrições de tempo impostas nas regras de disparo. Assim, as informações de tempo podem restringir o conjunto de marcas alcançáveis, mas nunca aumentá-las.

4.8. HIERARQUIA EM REDE DE PETRI

a) 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) conforme visto na Fig.5.4.

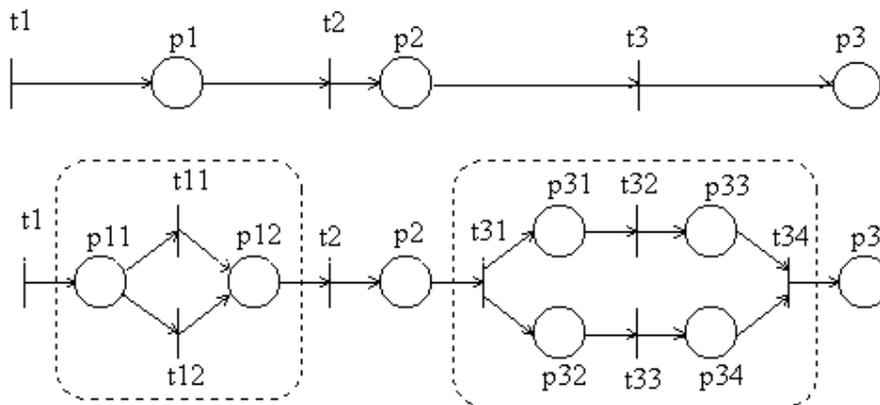


Fig.5.4 – Hierarquia numa rede de Petri

4.9. RELAÇÃO ENTRE REDE CPN E FMS

Como visto nos capítulos anteriores, a Redes de Petri Coloridas é uma ferramenta matemática e gráfica de representação formal de sistemas de eventos discretos com capacidade de modelar concorrência, eventos assíncronos, relações de precedência lógica e interações estruturais numa maneira simples e natural. Também modela conflitos, bloqueios, *buffers* finitos, sincronizações, prioridades, e operações de montagem e desmontagem de forma eficientemente. Além disso, ainda tem potencial de distinguir entre diferentes elementos, usando uma estrutura compacta e teoricamente bem fundamentada.

4.9.1 CPN como Ferramenta de Modelagem

As Redes de Petri, como uma ferramenta ordem parcial de eventos, permite a análise e a avaliação de desempenho em vários níveis de controle de Sistemas Flexíveis de Manufatura, sendo, portanto adequadas aos Sistemas de Produção (Rammamorthy, 1980). 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 RPs foram ganhando popularidade como uma ferramenta para modelar e analisar Sistemas Flexíveis de Manufatura. Logo, outros trabalhos incorporada 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.

Dentre os principais aspectos de Sistemas Flexíveis de Manufatura, geralmente aparecem situações tais como: Processos tendo relações de precedência entre si; Processos sendo executados ao mesmo tempo (paralelismo); processos que não podem ser realizados até que outro processo alcance certo estado (sincronização); processos a ser seguidos que não são determinados, o que mostra uma escolha entre alternativas (não determinismo); processos possuindo conflitos entre competição e coordenação de atividades (alocação de recursos).

Se compararmos estes requisitos com as aquelas características inerentes às Redes de Petri Coloridas, poderemos constatar claramente que as CPN se adaptam muito bem para especificar tais sistemas. 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.

4.9.2 Interpretação dos Elementos de uma CPN com relação a um FMS

Para cada tipo de sistema os elementos da rede de Petri podem representar diferentes entidades, ou seja, os elementos, os estados e as ocorrências do sistema real, dessa forma é necessário o estabelecimento de uma interpretação adequada que os relacione. Para os sistemas de manufatura que constituem o foco desse trabalho os *tokens*, os lugares, as transições e os arcos poderão assumir uma das seguintes interpretações:

Os Tokens - recursos fixos como máquinas e equipamentos; recursos variáveis como matérias prima, produtos e subprodutos; ou informações de controle e processamento “residentes” nas posições ou movidos através das transições.

Os Lugares - representam os estados que esses recursos podem assumir durante o transcorrer do processo, tais como: executando, consumindo, aguardando em manutenção, com defeitos, transmitido entre outros.

As transições - representam operações realizadas ou situações detectadas, tais como operação de uma máquina CNC, operação de um robô ou detecção da chegada ou saída de uma peça por um sensor.

Os arcos orientados - indicam as possibilidades de evolução dos *tokens* de um estado a outro quando pela ocorrência de uma transição e indicam também a quantidade de recursos que é retirado ou colocado em um lugar, ou seja, indica a direção do fluxo e o *throughput* do sistema. Dessa forma, a evolução do estado da RP corresponde a uma evolução da marcação, a qual é causada pelo disparo de transições como veremos posteriormente.

5. ESTRUTURAÇÃO E CONVERSÃO DE REDES DE PETRI

Este capítulo introduz os principais mecanismos de estruturação de redes de Petri que dão suporte à construção da metodologia de modelamento proposta nessa dissertação. Ele está dividido em sete seções: mecanismos de estruturação e composição, fusão, dobragem, técnicas de refinamento e abstração, composição hierárquica, teorema de extensão válida para Petri Net, modelo de *buffers* válidos.

5.1. MECANISMOS DE ESTRUTURAÇÃO E COMPOSIÇÃO

Os mecanismos de estruturação e composição de Redes de Petri são de extrema importância prática para o modelamento de sistemas reais com maior nível de complexidade, pois só através de algum tipo de modularização é possível utilizar redes de Petri para uma representação de suas características de funcionamento e de controle em níveis de abstração suficientemente detalhada e sem perder as propriedades de visualização e legibilidade do modelo. Mesmo as formas mais simples de modularização e estruturação hierárquica podem melhorar a legibilidade do modelo por permitir uma análise local de cada parte da estrutura.

Apesar de nas últimas duas décadas terem sido escritos vários bons textos de tutoria sobre redes de Petri, como por exemplo: (Jensen, 1981); Murata, (1989); (Christensen e Hansen, 1992) (David e Allá, 1992); (Silva, 1993); (Zurawski e Zhou, 1994); (Reisig e Rozenberg, (1998) e (Barros, 2006), poucos destes têm abordado essa questão de forma significativa.

Provavelmente, tal deve-se ao fato destas técnicas de composição não serem significativas do ponto de vista teórico, pois tipicamente, nada acrescentam ao poder de modelação das redes. No entanto, são claramente da máxima importância prática, pois só através de alguma forma de modularização se torna possível a utilização de redes de Petri para a modelagem de sistemas reais (Barros, 2006, p. 24)

Nesta seção apresenta-se o estado da arte no que diz respeito aos mecanismos e técnicas de estruturação e composição de modelos baseados em redes de Petri, de acordo com os principais trabalhos e artigos de pesquisadores e engenheiros ligados à área. Algumas dessas técnicas serão, mais tarde, utilizadas como subsídio para uma a

apresentação de uma metodologia híbrida de construção de modelos para sistemas flexíveis de manufatura.

Para começar é necessário estabelecer uma classificação que nos ajude a organizar os principais mecanismos de acordo com suas características fundamentais. Com base nas principais publicações e artigos especializados, podemos classificar os mecanismos de estruturação de redes de Petri em dois grandes grupos: composição e refinamento/abstração. Essa classificação é expandida e bem definida na tese de doutorado de João Paulo Barro, intitulada Modularidade em Rede de Petri (Barros, 2006) e será utilizada nesse trabalho para identificar as metodologias empregadas. Segundo Barros, as técnicas de estruturação podem ser subdivididas e organizadas conforme o gráfico da Fig. 5.1, considerando características específicas de cada tipo de estruturação.

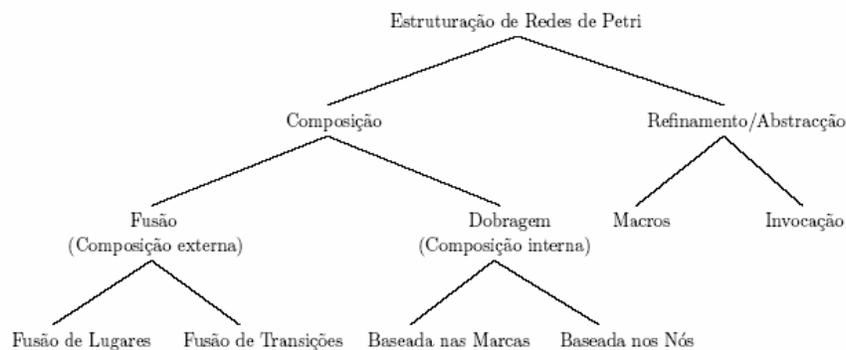


Fig.5.1 – Classificação e estruturação dos tipos de estruturação

A classificação proposta começa pelos dois tipos fundamentais de estruturação: composição e refinamento/abstração. Dentre as técnicas de composição, além da fusão de lugares e fusão transições também chamadas de composição externa, encontram-se também as técnicas de dobragem baseada em marcas e ainda um mecanismo de estruturação menos conhecido, baseado no conceito de vetores de nós, chamado dobragem baseada nos nós. Esses dois últimos são conhecidos como composição interna e constituem a base para a transformação das redes de Petri estado-transição para as redes de Petri coloridas.

No grupo de **refinamento/abstração** incluem-se as técnicas que possibilitam modificar o nível de abstração da rede. Elas são usadas como base para uma

estruturação hierárquica, a partir das filosofias ascendentes (*bottom-up*), descendentes (*top-down*) ou da utilização de uma combinação dessas duas, chamada de estruturação híbrida (Barros, 2006).

A **técnica de composição** corresponde à interligação entre vários sub-modelos. Esta interligação é tipicamente feita através da fusão entre dois ou mais lugares, ou entre duas ou mais transições. Por essa razão, os vários modelos podem ser vistos como módulos do sistema total que se pretende modelar.

Essa técnica constitui um claro suporte à construção de modelos de forma ascendente (*bottom-up*) e também permite a reutilização de módulos já existentes, dessa forma é de grande interesse para o nosso trabalho. Aqui, generaliza-se esta idéia de composição de modelos considerando dois tipos de composição: a composição externa (fusão) e a composição interna (dobragem).

A **composição externa** corresponde ao conceito de fusão entre nós de redes (o páginas) distintas, ou seja, ao caso já descrito em que duas ou mais redes são utilizadas para criar uma única rede maior e mais complexa.

Segundo Barros (2006) constitui-se na aplicação mais direta e reconhecida, dentro do âmbito dos modelos de redes de Petri, para decomposição dos sistemas em módulos desde que foi proposto proposta por (Parnas, 1972).

A **composição interna** (dobragem) permite a composição entre nós da mesma rede, transferindo a complexidade das posições para as marcas. Esse tipo de composição dá subsídio para uma redução do número de nós, mas em compensação tem como consequência o emprego de uma notação mais elaborada para os elementos básico da rede como marcas, posições e transições.

5.2. FUSÃO (COMPOSIÇÃO EXTERNA)

A fusão de nós, sob a forma de fusão de lugares ou de transições, constitui a forma mais intuitiva e graficamente explícita de compor redes de Petri a partir de sub-modelos. A fusão de nós oferece um suporte importantíssimo quando o objetivo é uma composição horizontal das redes, no qual as redes são posicionadas lado a lado e

interconectadas através dos seus lugares ou transições. Dessa forma as várias sub-redes que compõem o sistema maior podem ser vistas com parte de um quebra-cabeça em que cada peça pode executar uma função completa, mas também contém partes delas que a podem ser ligadas a outras peças. As fusões de transição e de lugares são as duas possibilidades de fusão de nós. Cada uma implementa certas características específicas que se adaptam melhor a determinados tipos de sistemas. Nesse trabalho, utilizaremos fusão de lugares para realização de composição para composição externa horizontal e para a estruturação hierárquica fundamentados em três razões principais:

a) Conforme Juan, Tsai e Murata (Juan et al., 1996, 1998) a comunicação síncrona entre processos, característico da fusão de transição, não pode ser utilizada de forma natural para análise de propriedades baseadas nos estados.

b) A fusão de lugares pode ser vista como partilha de recursos (por exemplo, buffers intermediários entre equipamentos) permitindo a comunicação e o fluxo de recursos assíncrono de recursos entre sub-processos.

c) A disponibilidade de ferramentas de simulação de alta qualidade para o desenvolvimento e testes de modelos baseados em redes de Petri hierárquicas que utilizam fusão por lugares como elemento de conexão.

O exemplo mais significativo de fusão por lugares é, provavelmente, o das próprias redes de Petri coloridas hierárquicas proposta por Jensen em sua monografia e em outros dois trabalhos subsequentes (Jensen, 1997) em que realizam a composição de sistemas pela fusão de lugares, mas não utilizam a fusão de transições.

A fusão de lugares visa atingir dois objetivos distintos: uma como conveniência gráfica, de forma a aumentar a legibilidade dos modelos (conforme referido anteriormente), e outra, como suporte à estruturação hierárquica de redes. Neste último caso, a fusão dá suporte à implementação de macro transições, também denominadas **transições de substituição** (Barros, 2006).

Um exemplo bem simples de composição por fusão de lugares é mostrado na Fig. 5.2, onde existem três postos de trabalhos com estoques unitários, ou seja, somente uma peça pode ser trabalhada por vez em cada posto. Esses postos podem ser unidos, através de fusão de lugar ou transição para formar o modelo de uma linha de produção

do tipo *flow-shop*. Esse tipo de arranjo, em ciências da computação, geralmente é denominado de arranjo do tipo FILA ou FIFO (*first-in first-out*) do inglês.

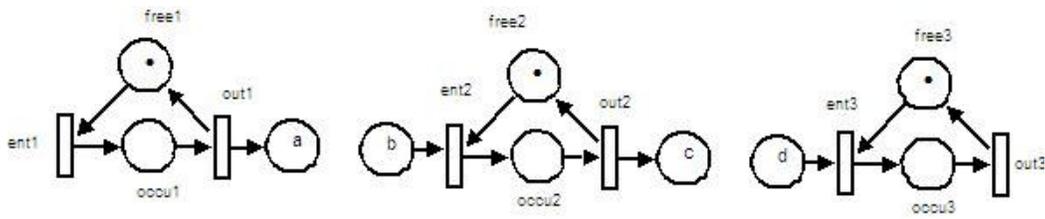


Fig.5.2 – Postos de trabalho individuais com estoque unitário

O modelo completo pode ser obtido pela fusão dos lugares a e b que passam a ser denominadas ab e dos lugares c e d que passa a ser chamados de cd. Os lugares a, b, c e d são geralmente denominados de soquete e podem ter ou não uma interpretação física. A Fig. 5.3 mostra o resultado da fusão.

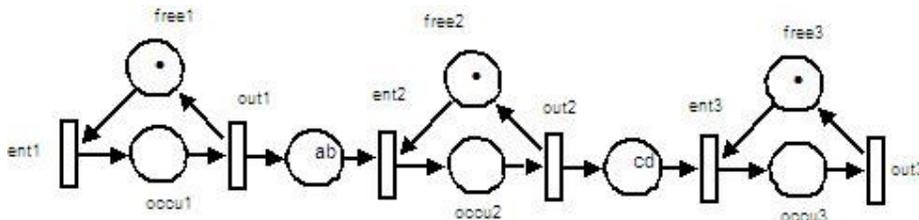


Fig.5.3 – Resultado da fusão de lugar

Se considerarmos uma não temporizada, os lugares intermediários possibilitam uma comunicação assíncrona entre os processos. Contudo, quando utilizarmos redes temporizadas, podemos conseguir comunicação sincronizada desde que haja um $t > 0$ para as operações internas dos *buffers* e um $t = 0$ para os *buffers* intermediários. Nesse caso o conjunto lugar-transição pode ser substituído por uma única transição culminando na sincronização entre os *buffers*.

Como já citamos a fusão por transição é desejável quando queremos estabelecer uma comunicação síncrona entre as partes. As Figs. 5.4 e 5.5 ilustram esse tipo de fusão.

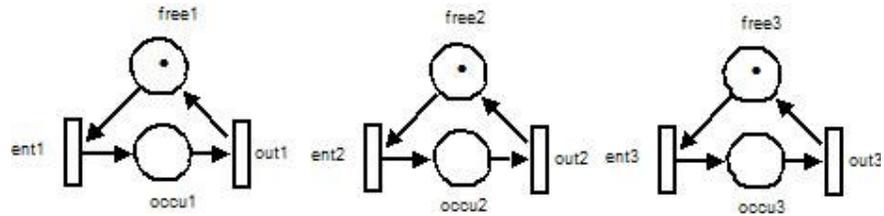


Fig.5.4 – Postos de trabalho individuais com estoque unitário

O modelo da linha de produção pode ser obtido pela fusão das transições out1 e ent2 que passaremos a denominar de mov1to2 e pela fusão das transições out2 e ent3 que passaremos a chamar de mov2to3. Assim obtemos o modelo completo mostrado na Fig. 5.5.

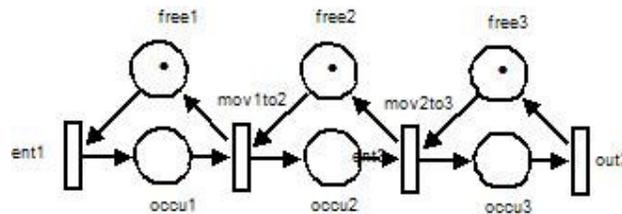


Fig.5.5 – Composição através de fusão de transição das redes da Fig. 5.4

5.3 DOBRAGEM (COMPOSIÇÃO INTERNA)

Corresponde ao conceito de dobragem. Em outras palavras, a dobragem é a composição de uma rede com ela própria: as repetições de idênticos padrões estruturais são como que dobradas sobre si próprias de forma a fundir lugares, arcos e transições. A dobragem tem como consequência uma maior complexidade de notação das marcas, lugares, arcos e transições.

Na **dobragem baseada em marcas**, a complexidade da representação gráfica transfere-se para as marcas, possibilitando que estas possam conter qualquer tipo de dados. Consequentemente, essa técnica implica em um aumento de complexidade também nas anotações associadas aos arcos e transições. A dobragem baseada em marcas é a que caracteriza as redes de Petri de alto nível (Jensen e Rozenberg, 1991).

Na dobragem baseada em marcas, cada transição passa a ter associada uma expressão booleana, denominada guarda, que pode devolver um valor verdadeiro ou falso. Os

arcos passam também a ter expressões algébricas associadas, capazes de efetuar transformações de dados, ou seja, além de permitir a redução das dimensões gráficas do modelo, também possibilita o modelamento dos dados e suas transformações.

A Fig. 5.6 mostra o resultado da aplicação da dobragem baseada em marcas no exemplo da Fig. 5.5. Nessa nova representação os lugares **occ1**, **occ2** e **occ3** fundidos em um único lugar chamado **occ**, enquanto que os lugares denominados **free1**, **free2** e **free3** são fundidos em um novo lugar chamado **free**. Da mesma forma, as transições **mov1to2** e **mov2to3**, devido a executarem funções semelhantes, também são fundidas em uma única transição chamada **mov**. As transições **ent1** e **out3** não sofrem alterações porque executam funções diferentes das demais transições.

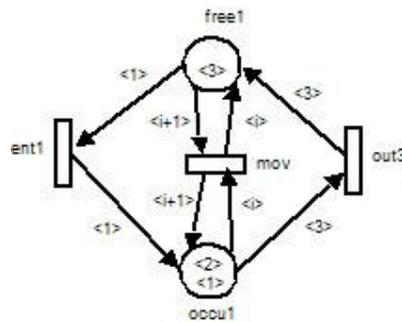


Fig.5.6 – Dobragem baseado em marcas do modelo da Fig 5.3

Na literatura é possível identificar duas classes principais de redes de Petri de alto nível. Uma baseada nos trabalhos de (Genrich, 1987; He, 1996) e a outra é a redes de Petri colorias (Jesen, 1996, 1997). Essa segunda tem especial interesse para esse trabalho, pois se adequa às características do modelamento proposto e, além disso, possui poderosas ferramentas computacionais de simulação que suportam suas característica, uma das quais, utilizaremos nesse trabalho, chamada CPN Tools.

A ferramenta CPN Tools é baseada nas redes de Petri coloridas hierárquicas, estendendo a capacidade de modelamento através da utilização de lugares especiais, denominados **lugares de fusão, sokets e portas** que são utilizados para realizar a conexão horizontal entre os módulos. Estes lugares também são usados para interligar as sub-páginas nas macro-transições (transições de substituições) às super-páginas na hierarquia superior, possibilitando, assim, a realização de estruturas com vários níveis de abstração, subsidiando as técnicas de refinamento e abstração. Além disso, a

ferramenta CPN Tools também possibilita a associação de tempo aos *tokens* (time stamps) através das transições ou das expressões de arco.

5.4 TÉCNICAS DE REFINAMENTO E ABSTRAÇÃO

Estas duas técnicas constituem a base para estruturação de modelos hierárquica, onde o refinamento utiliza uma filosofia descendente de construção, enquanto que a abstração ascendente. Tais técnicas são muito úteis por permitirem a representação do modelo em vários níveis de abstração inter-relacionados.

O **refinamento** caracteriza-se pela substituição de um nó de rede em nível hierárquico superior por uma rede de petri mais complexas que represente o detalhamento do modelo relacionado a esse nó em um nível hierárquico inferior. A rede no nível superior que contém o nó é chamada de super-rede ou super-página e a rede que é inserida no lugar do nó é chamada de sub-rede ou sub-página. Esses nós podem ser uma transição ou um lugar, dependendo dos elementos a que eles estão interligados.

A **abstração** constitui-se pela simplificação da representação de uma estrutura complexa de rede, pela substituição desta por uma estrutura mais simples, geralmente um único nó de rede em um nível hierárquico superior, ou seja, a super-rede. Tal técnica permite a visualização somente dos parâmetros de interesse, mantendo escondidos os demais.

Ambas as técnicas citadas permitem o desenvolvimento de modelos em uma perspectiva modular e sistemática e, além disso, possibilita, de uma forma simples e direta, uma rápida modificação do nível de abstração durante a observação e análise do mesmo.

5.4.1 Refinamento de Transição

Essa técnica subsidia a estruturação descendente (top-down) de modelos através da substituição de uma transição por uma rede que introduz detalhes de funcionamento para um sistema. O refinamento continua até que o nível de detalhamento satisfaça as especificações do sistema. A metodologia top-down tem a vantagem da visualização global do sistema por iniciar do final da síntese (Barreto, 2005).

Na proposta de modelo apresentada nesse trabalho, esta técnica será utilizada com três objetivos prioritários: simplificar a representação gráfica do modelo, permitir um tratamento hierárquico entre os blocos e para tratar a ocorrência de eventos prioritários em caso de processos acionados por eventos.

5.4.2 Estruturação e Modificação de Modelo

Os mecanismos de estruturação e abstração apresentadas resumidamente nas sessões anteriores foram motivados pela necessidade de expandir o poder de modelamento da rede de Petri aos níveis geralmente encontrados nas linguagens textuais de alto nível. Tais mecanismos de estruturação enfatizam a construção modelos mais complexos a partir de submodelos ou, simplesmente, permitir a visualização hierárquica de modelos não hierárquicos. No entanto para sua efetiva realização é necessária a definição de operação que possibilitem a construção desses modelos e ainda assim mantenham determinadas propriedades. Esse tipo de estruturação é tradicional nas abordagens mais formais e é conhecido na literatura como **composição com preservação de propriedade**, pois se preocupa com a manutenção de propriedades dos modelos. Outro fator a ser considerado é a disponibilidade de formas genéricas e práticas de especificar modificações em modelos. Esta última enfatiza a aplicabilidade, generalidade e legibilidade relativa a composição e modificação de modelos.

A seguir apresentam-se duas operações para a estruturação e modificação em modelos de Rede de Petri que serve tanto para composição horizontal quanto para a composição hierárquica. Tais operações subsidiam a construção de modelos de forma descendente ou ascendente, através da modificação destes tanto por via aditiva quanto subtrativa. Correspondentemente às suas características, essas operações foram denominadas **adição de redes** e **subtração de redes** respectivamente.

5.5. COMPOSIÇÃO HIERÁRQUICA

Como já referido anteriormente, a perspectiva ascendente de construção de modelos parte de modelos mais simples, aqui denominados de sub-redes para a construção de sistemas mais complexos. Para exemplificar esse tipo de construção realizaremos a estruturação de um parque de estacionamento para carros constituídos pelas sub-redes Enter, Parking Área, Leave e Passag mostrado na Fig. 5.7.

Para exemplificar a construção desse modelo utilizaremos a proposta de linguagem textual e o diagrama de adição introduzida por João Paulo Barros em sua tese de doutorado “Modularidade em Redes de Petri” (Barros, 2006). Essa proposta utiliza essas redes como geradoras de instâncias e permite uma melhor visualização das fusões de rede, além de permitir modificações transversais em modelos construídos a base de sub-redes.

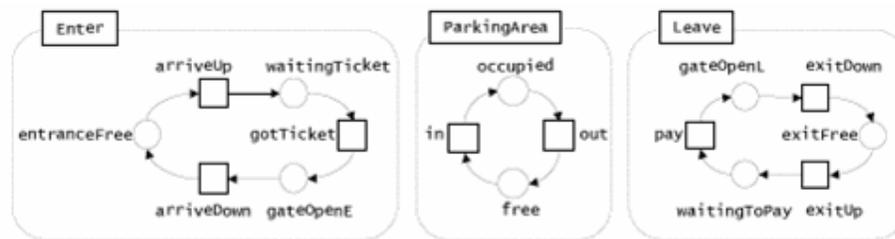


Fig. 5.7 – Sub-redes para estruturação do modelo do parque de estacionamento

A partir das redes da Fig. 5.5 podemos gerar a macro-rede para o parque de estacionamento com uma entrada, uma área de estacionamento e uma saída conforme indicado a seguir. Na Fig. 5.8 são mostrados dois tipos de descrição do modelo, um textual e um gráfico.

```
ParkA := (Enter + ParkingArea + Leave)(/gotTicket/in/ -> in,
                                           /out/pay/ -> out)
```

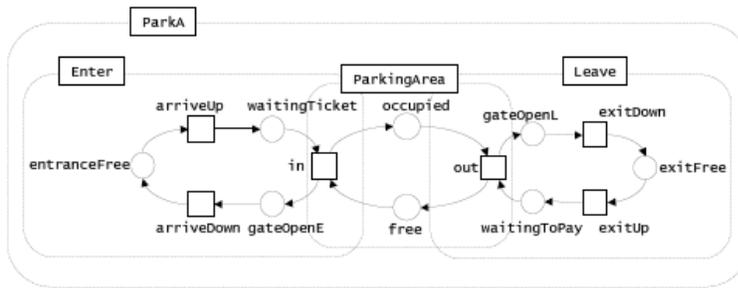


Fig.5.8 – Resultada da operação de adição entre as redes *Enter ParkingArea e Leave*

A representação textual proposta por Barros designa a operação de adição entre os submodelos Enter, ParkingArea e Leave formando a macro-rede ParkA. Essa operação é constituída por duas partes, em primeiro lugar uma união disjunta das redes envolvidas (Enter + ParkingArea + Leave) e depois uma lista de pares a serem fundidos, separados por vírgulas denominados **colapso**. Cada um desses pares tem um conjunto de nós a fundir (/sublista/), denominados **conjunto de fusão** e o nome do nó resultado. A representação o arranjo (/sublista/ → resultado) denomina-se **conjunto de fusão nominativa**. Dessa forma o colapso é uma lista de **conjuntos de fusão nominativa**. Na definição do modelo ParkA o colapso é (/gotTicket/in/ → in, /out/pay/ → out) que utiliza **identificadores reduzidos** para os nós das redes adicionais. Nesse caso isso só é possível porque não existe ambigüidade ente os elementos da rede, caso contrário seria necessário utilizar identificadores completos, comporto pela identificação das redes e das sub-redes.

Uma outra forma proposta por Barros para a representação da operação de adição seria através de um diagrama chamado **diagrama de adição**. Esse diagrama é mostrado a na Fig. 5.9.

O diagrama de adição aumenta o nível de abstração e permite a visualização tanto da união dos vários submodelos quanto da sua interligação por meio de fusão de nós, lembrando que esses podem ser lugares ou transições. No exemplo a seguir, as transições fundidas são representadas por pequenos quadrados que são denominadas transições de interface ou transições de fusão. Quando os elementos fundidos são os lugares (lugares interface), estes são representados por circunferências ou elipses.

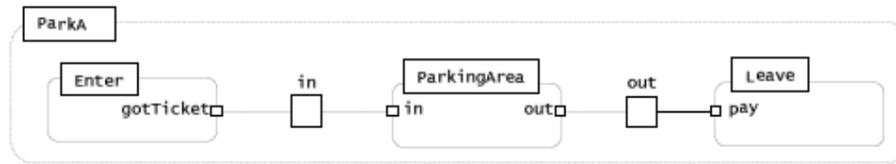


Fig.5.9 – Diagrama de adição

No diagrama de adição, a ligação entre os módulos fica explícita no nível da super página, o que força uma interpretação para cada um dos nós correspondentes. Contudo a escolha do tipo do macro nó obriga à observação de três regras para a determinação dos nós de interface: a) se o macro nó deva ser interpretado como macro lugar, todos os seus nós de interface devem ser transições; b) caso um módulos seja interpretado como macro transição, todos os seus nós de interface devem ser lugares; c) e caso um módulos deva ser interpretado como macro bloco, os seus nós de interface devem incluir pelo menos um lugar e uma transição.

5.6. TEOREMA DE EXTENSÕES VALIDAS DE REDE

Um dos grandes problemas para modelamento de redes de Petri é que a complexidade da análise cresce exponencialmente à medida que o tamanho da rede ou o número de marcas cresce linearmente, dessa forma é fundamental definir formas de conexão que garantam a manutenção das propriedades das redes maiores a partir de redes menores válidas. O teorema de extensões válidas para redes de Petri pode ser utilizado para garantir o crescimento da rede conservando as características desejadas para sistemas de manufatura, ou seja, as propriedades de vivacidade, limitação e reversibilidade (Qiao et al, 2002). Segundo esse teorema é possível estender as redes de Petri sob determinadas condições que são enunciadas a seguir:

Considerando a Fig. 5.10, assumo que $Z = \{P, T, I, O, M_o\}$ e $S = \{O_s, T_s, I_s, O_s, M'_o\}$ são duas sub-redes de $Z' = \{P', T', O', M'\}$, onde $P' = P \cup O_s$, $T' = T \cup T_s \cup \{T_s\} \cup \{T_e\}$ e $M' = \{M_o, M'_o\}$, Então:

- Z' é limitado se e somente se Z e S são limitados
- Z' é vivo se e se somente se Z e S são vivos
- Z' é reversível se e se somente se Z e S são reversíveis

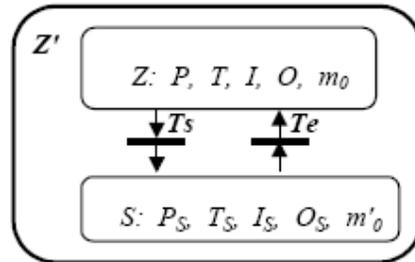


Fig.5.10 – Extensão Válida de Petri Net

Lugares conectados por T_s e T_e podem ser o mesmo lugar in Z ou diferentes lugares de um evento serial. Quando T_s e T_e são invertidos, eles são chamados “backward firing”, e o teorema continua válido. A prova desse teorema pode ser encontrada na referência (Zhou, 1989).

5.7. MODELO DE BUFFERS VÁLIDOS

De acordo com Qiao (2002), Um sistema de manufatura pode ser decomposto em células de trabalho de acordo com função ou o tipo de cada processo que a célula de manufatura vai realizar. Cada uma dessas células é representada por um modelo válido. Dessa forma, esses modelos serão sub-redes do modelo do sistema completo. Utilizando a teoria de extensões válidas para redes de Petri, conectam-se as sub-redes através de conexões padrão ou modelos de buffers modelos.

Na Fig. 5.11a, uma rede de Petri básica de buffer FIFO é mostrada. Este modelo pode representar qualquer buffer do tipo fila (primeiro a entrar, primeiro a sair) com partes indistinguíveis entre si. Esse tipo de estrutura é muito usado como buffer entre duas máquinas CNCs. Ele armazena partes enviadas pelo CNC1 e estão em espera para ser processadas pelo CNC2.

A Petri net na Fig. 5.11b apresenta um buffer do tipo LIFO (ultimo a entrar, primeiro a sair), também utilizado para partes que não podem ser distintas umas das outras, ou seja, um modelo baseado em Petri net não colorida.

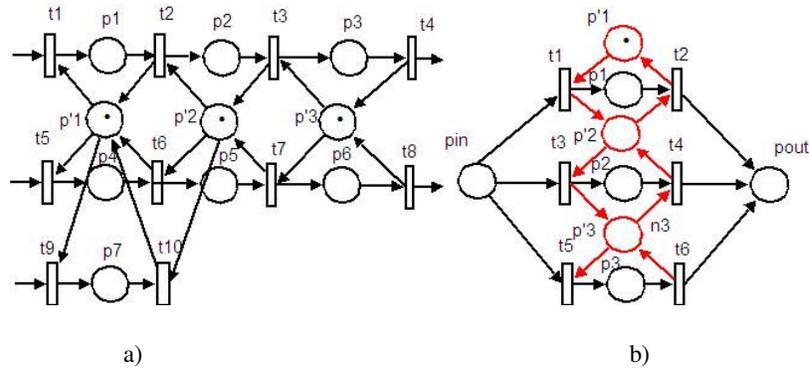


Fig.5.11 – Buffers PN – a) tipo FIFO – b) tipo LIFO

A partir da técnica da dobragem baseada em marcas e de uma linguagem estruturada de programação associada ao gráfico é possível converter os *buffers* FIFO e LIFO em rede não colorida para uma rede do tipo CPN com transferência da complexidade para as marcas e para as funções de arco sem perda de propriedades. A Fig. 5.12 foi obtida pela fusão dos lugares $p_1/p_4/p_7 \rightarrow p_s$; das transições $t_1/t_5/t_9 \rightarrow t_i$ e das transições $t_2/t_6/t_{10} \rightarrow t_o$. Os dois gráficos da Fig. 12 diferenciam-se entre si pelas funções associadas aos seus arcos de entrada e saída que organizam suas listas de *tokens* de acordo com uma estrutura FIFO ou LIFO. Dessa forma as funções $ffi(Prdi)$ e $ffo(Prdi)$ são responsáveis pela organização em estrutura FIFO na Fig. 12 a), enquanto que as funções $lfi(Prdi)$ e $lfo(Prdi)$ são responsáveis pela organização da estrutura LIFO na Fig. 12 b).

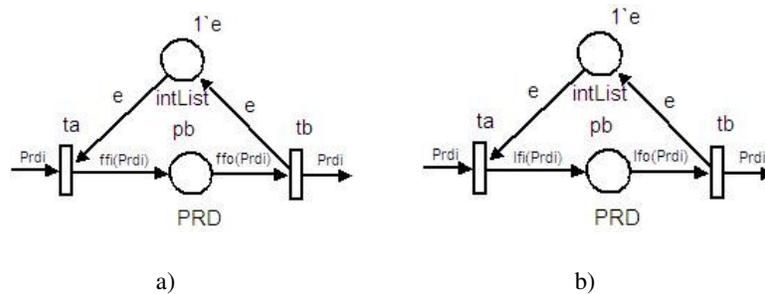


Fig.5.12: Buffers CPN tipo FIFO e LIFO

5.8. DEFINIÇÃO DE EXCLUSÃO MÚTUA

Em problemas de modelamento de Sistemas de Manufatura com rede de Petri a partir de síntese, o lugar pode representar tanto estados dos recursos como estados de operação. Na definição de *mutual exclusions* existem basicamente três tipos de lugares que dependem da marcação inicial da rede de Petri são definidas como segue:

Um lugar “ p ” é chamado um *A-place* (P_A) ou *lugar de operação* se somente se $m_0(p) = 0$. Um *A-place* geralmente representa um lugar de operação. Inicialmente não existe nenhuma operação sendo executada pelo sistema, ou seja, $m_0(p) = 0$.

Um lugar “ p ” é chamado um *B-place* (P_B) ou lugar de recurso fixo se somente se $m_0(p)$ é uma constante. *B-place* representa a disponibilidade de um número fixo de recursos. Por exemplo, o número fixo de robôs ou máquinas é geralmente fixo para um sistema dado.

Um lugar p é chamado um *C-place* (P_C) ou lugar de recursos variáveis se somente se p é inicialmente marcado com $m_0(p) > 0$ e o número de tokens iniciais em p é variável, ou seja, $m_0(p)$ é variável. Um *C-place* pode representar disponibilidade de matérias prima, ferramentas, *pallets*, etc. Os sistemas de manufatura geralmente precisam ser projetados para que o sistema tenha a maior flexibilidade e eficiência possível com um pequeno número de ferramentas ou *pallets* (Zhou et al, 1991).

5.8.1 Exclusão Mútua Paralela

O teorema a seguir revela o fato de que se um sistema contém vários processos independentes com certas propriedades qualitativas de interesse do projeto, então quando este processo tem que competir por recursos com oportunidades iguais, e depois devolve tais recursos após os mesmos tenha sido adquiridos e usados, o sistema ainda mantém tais essa propriedade.

Teorema 1: Assuma que $Z = (P_A \cup P_B \cup P_C, T, I, O, m_0)$ é uma subnet de $Z' = (P', T', \Gamma, O', m'_0)$, $P' = (P_A \cup (P_B \cup \{p_E\})) \cup P_C$, $T' = T$, e $D = \{(t_{ai}, t_{bi}), i \in N_k\}$, If (p_E, D) é um k -PME de Z' , então podemos dizer que as seguintes afirmações são verdadeiras:

1. Z' é limitado (salvo) se Z é limitado (salvo);

2. Z' é vivo se Z é vivo;
3. Z' é reversível se Z é reversível.

A prova para esse teorema pode ser encontrado em (Zhou e DiCesare, 1991)

5.8.2 Exclusão Mútua Serial

A modelagem de rede de Petri com SME apresenta um pouco mais de dificuldades com relação a PME. Pode ser mostrado que existe uma série de transições mortas em redes de Petri com um SME quando o número inicial de *tokens* dos recursos variáveis (C-places) excede alguma constante se existe limitação de capacidade de *tokens* para C^i (Zhou, 1991). O teorema a seguir aborda as condições que permitem com que uma rede de Petri contenha uma SME limitada, viva e reversível.

Teorema 2: Assuma que $Z = (P_A \cup P_B \cup P_C, T, I, O, m_0)$ é uma subnet de $Z' = (P', T', \Gamma, O', m'_0)$, $P' = P_A \cup (P_B \cup \{p_E\}) \cup P_C$, $T' = T$, e $D = D^1 \cup D^2 \cup \dots \cup D^L$, $k_i = |D^i| \geq 1$, $D^i = \{(t_{aj}^i, t_{bj}^i), j \in N_{ki}\}$. Façamos $T_a = \{t_{aj}^i, i \in N_L\}$ e $\{T_b = \{t_{bj}^i, j \in N_{ki}, i \in N_L\}$ e $\{T_b = t_{bj}^i, j \in N_{ki}, i \in N_L\}$. If (p_E, D) é um SME, então podemos dizer que as seguintes afirmações são verdadeiras:

1. Z' é limitado (salvo) se Z é limitado (salvo).

Além do mais, se:

C1. $\exists C^i, m_0, \alpha, \text{ e } \beta^i, \exists m_0(p) \geq \alpha(p) \text{ e } \beta^i(p)m_0(p) \leq C^i, p \in P_C$.

C2. $\forall m \in R(Z, m_0), m(p) \leq m_0(p)$ se $p \in P_B \cup P_C$, e

C2. se um A-path $EP(t_{bu}^i, t_{av}^{i+1})$ existe para $1 \leq i \leq L - 1, 1 \leq u \leq k_i, \text{ e } 1 \leq v \leq k_{i+1}$, e se $C(t_{bu}^i, t_{av}^{i+1})$ é finito, então $C(t_{bu}^i, t_{av}^{i+1})$ é efetivo.

Então:

2. Z' é vivo se Z é vivo;
3. Z' é reversível se Z é reversível.

A prova para esse teorema pode ser encontrada em (Zhou e DiCesare, 1991). As condições de C1 até C3 podem ser explicadas como segue:

a) C1 define a faixa da marcação inicial. Esta condição fornece informações, por exemplo, de quantas tarefas estão liberada a serem enviadas para o sistema em algum intervalo de tempo dado.

b) C2 declara que a rede de Petri sem esta SME nunca produz mais que os *tokens* inicialmente dados nos lugares de recursos em qualquer marca alcançável.

d) C3 exige que a capacidade de tokens entre qualquer PME's consecutivos seja *efetivo* em uso se eles existem como um número finito.

5.8.3 Extensão SME e PME para utilização em CPN

Nas redes de Petri coloridas, os arcos passam a ter expressões algébricas associadas, capazes de efetuar transformações de dados, e de permitindo a redução das dimensões gráficas do modelo. Através de uma combinação da técnica de dobragem baseada em marcas e utilização de uma linguagem textual para definir as expressões dos arcos é possível converter as redes básicas, PME/SME para redes de Petri coloridas equivalente.

Nas PMEs, as repetições padrões estruturais em paralelo compartilhados são como que dobradas sobre si próprias de forma a fundir lugares, arcos e transições em uma única transição. Os diferentes são representados na nova rede por diferentes *tokens*. A definição de qual tokens serão consumidos em cada disparo depende da expressão do arco de entrada ou da função de guarda.

Nos SMEs, as repetições de padrões estruturais compartilhados em série são como que dobrados sobre si próprios de forma a fundir lugares, arcos e transições. Os diferentes processos em série são conseguidos através de uma expressão de transição colocados em um arco de retorno ligando a transição de saída como o lugar de entrada. Dessa forma, cada vez que o *token* passa para um novo estado do processo ele sofre uma transformação formando um novo *token*. Esse novo *token* é que representa o novo estado do processo.

Pode-se fazer uma combinação de ambas as representações para obter um modelo compartilhado de recursos em série e paralelo ao mesmo tempo mantendo-se ainda a propriedade de vivacidade, reversibilidade e limitação. A Fig. 6.1 e 6.2 da seção 6.1 mostram essa transformação.

6. CONJUNDO DE SUB-REDE VÁLIDOS PARA FMS .

A abordagem modular consiste na construção de redes menores para descrever subsistemas simples que são posteriormente agrupados para formar modelos mais complexos. Esse agrupamento pode ser feito tanto pela filosofia *bottom-up* quando *top-down*. A filosofia *top-down* com instanciamento de sub-redes pré-definidas é uma abordagem promissora, pois permite dispor de um conjunto de sub-redes padrões adequadas para cada ramo de aplicação. Estas redes são usadas como base para construção de modelos complexos e podem ser reaproveitados facilmente, evitando reinventar modelos já existentes. Neste capítulo apresentaremos alguns submodelos escritos em CPN aplicadas à área de automação da manufatura que podem ser utilizadas para composição de modelo de sistemas produtivos maiores. Essas sub-redes serão apresentadas no formato lugar-transição convencional e no seu formato equivalente em CPN.

6.1. CONTRUÇÃO E ANÁLISE DE SUB-MODELOS USANDO TEORIA PME/SME.

Pela combinação das teorias de PME e SME nós podemos representar diferentes operações válidas com recursos compartilhados paralelos e sequenciais executadas pela a mesma máquina ou pelo mesmo conjunto de máquinas, desde que algumas restrições sejam obedecidas. A Fig.6.1 mostra a um rede com estrutura PME/SME sem limitação de buffer para duas linhas de produção, e executando duas operações sequenciais cada uma e compartilhando o mesmo recurso localizado no lugar *buf*.

Os lugares *pp00* e *pp01* são chamados *lugar de recurso variável* e representam a disponibilidade de matéria-prima, instalações, *pallets*, etc. O lugar *buf* é chamado *lugar de recurso fixo* e representa recurso fixo compartilhado necessário para a operação como máquina e ferramentas. Suas fichas também podem ser vistas como ficha elementar representando os estados de operação de máquina. Os outros lugares são chamados lugares de operação.

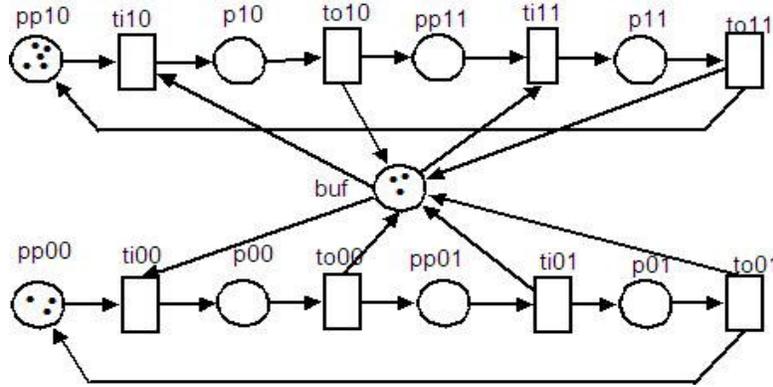
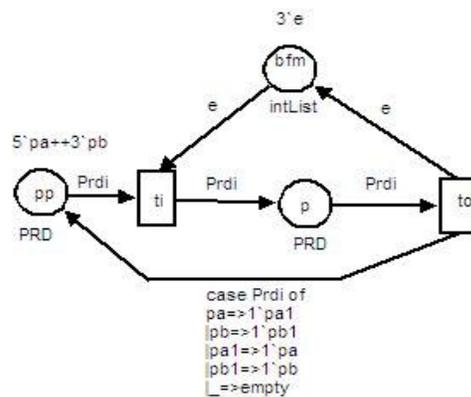


Fig.6.1: representação de duas linhas de produção em paralelo usando as PME e SME em uma rede de Petri não colorida

A transformação de Rede de Petri não colorida em uma rede de Petri colorida válida equivalente foi apoiada pela teoria de *dobradiça baseada em marca*, em que a complexidade de gráfico é transferida para a marca por um conjunto de fichas coloridas produzindo o resultado como o representado na Fig.6.2a. Nesta representação, as transições *ti00*, *ti01*, *ti10* e *ti11* são convertidas em *ti*; as transições *to00*, *to01*, *to10* and *to11* são substituídas por *to*; os lugares *p00*, *p01*, *p10* e *p11* são representados por *p*; *pp00* e *pp10* geral *pp*. O *Prdi* é uma variável com os valores *pa*, *pa1*, *pb* e *pb1*. A equação entre *to* e *pp* é uma transformação de etapa de produção do produto na modelagem em SME. O conjunto de declarações de tipos usando linguagem de CPN ML é mostrado no Fig.6.2b. A marcação inicial deve obedecer as restrições apresentadas na Seção 5.8.2.



Declarations:
colset intList= unit with e;
colset PRD= with pa|pa1|pb|pb1;
var Prdi: PRD;

a)

Fig. 6.2- a) representação PME/SME em CPN; b) conjunto de declarações em CPN ML.

A validação desta transformação é obtida usando uma ferramenta de simulação de espaço de estado chamada CNP Tool. Se compararmos os resultados das informações estatísticas, de vivacidade, limitação e reversibilidade em ambos modelos da Fig. 6.1 e Fig. 6.2, verificamos os resultados das simulações foram os mesmos, como é mostra na Tab. 6.1. Estes dados confirmaram a manutenção das características depois da transformação para no CPN.

[Home properties]			[Statistics]
-----			-----
Home markins			State Space
Initial Marking is a home			Nodes: 448
marking			Arcs: 1
			0
[Boundedness]			Status: Full

Boudeds	Upper	Lower	[Liveness]
FMS'buf 1	3	0	-----
FMS'p 1	3	0	Live transitions
FMS'pp 1	8	5	All
			Dead Marrkings
			None

Tab. 6.1: Resultados da simulação das Figs. 6.1 e 6.2

As experiências de modelamento também evidenciaram que praticamente todas as expressões de controle de fluxo e restrição de *tokens* podem ser conseguidas através de expressões condicionais do tipos *case < > of; <condição n> => <ação n>* para n condições.

Informações estatísticas. Do relatório de estatísticas na Tab. 6.1, nós podemos ver que o número de espaço de estado obtido em ambas as simulações foram de 448 nós e 1448 arcos. Nós calculamos todo o espaço de estado, e isto levou apenas alguns segundos.

Propriedades de Vivacidade da Tab. 6.1, nós podemos ver que não existe nenhuma transição morta. Quer dizer que cada transição está habilitada em pelo menos uma marcação alcançável (o qual é uma propriedade bastante fraca). Nós também verificamos que todas as transições são vivas. Uma transição viva é uma transição que pode sempre, não importa o que acontece, ficar novamente habilitado. Quando existirem *dead marks* não pode haver nenhuma transição viva. Para ser ao vivo implica que não existe nenhuma possibilidade de paralisação completa.

Reversibility. Esta propriedade está relacionada com a *home marking*, a qual é uma marcação que pode sempre ser alcançado, independentemente do que previamente aconteceu. Nós vemos que nossa rede tem uma marcação de *home* que também é uma marcação inicial, de forma que garante a reversibilidade. Esta característica é muito importante para recuperação de erro desde indica que o sistema pode retornar a seu inicial oficial de qualquer estado atual.

Limitação. A Tab. 6.1 também mostra os limites superiores e inferiores inteiros, isto é, o número máximo e mínimo de fichas que podem ser localizados nos lugares individuais no considerando o conjunto das marcações alcançáveis. Limitação indica a ausência de transbordamento no sistema modelado. Esta característica permite a especificação de um limite no número de fichas (*tokens*) que podem estar em um lugar em qualquer tempo.

Utilizando o teorema de extensão válida para redes de Petri, nós podemos conectar as sub-redes válidas usando conexões padrões ou modelos de *buffers* válidos para conservar as propriedades do sistema. A prova deste teorema pode ser encontrada na referência (Qiao et al, 1989). Alguns destes *buffers* são os FIFO e LIFO mostrados na Fig.5.13 no formato de rede não colorida e CPN respectivamente.

6.2. BIBLIOTECA DE SUB-MODELOS USADOS FMS

Compartilhamento de recurso em paralelo: A existência de recursos compartilhados é uma das principais características de uma Célula de Manufatura Flexível. Essa característica permite uma maior flexibilidade na utilização dos recursos do arranjo produtivo, contudo também gera problemas de conflitos. No caso da célula com recurso compartilhado em paralelo da Fig. 6.3a, tem-se conflito do tipo confusão. Observa-se que t_a e t_b estão habilitados, mas o disparo de uma inabilita a outra, caracterizando a indisponibilidade do recurso para o outro produto ou para a outra linha de produção. Contudo, assim que a tarefa é concluída o recurso fica novamente disponível. Nessa interpretação, o *token* disposto no *anti-place*, como recurso fixo, é o elemento a ser compartilhado.

A transformação da rede é feita utilizando composição por dobramento baseado em marca, acarretando uma notável redução no número de nós, pois a complexidade gráfica passa a ser assumida pela notação. Dessa forma passa a ser necessário a declaração de uma lista de tipos e variáveis. A declaração em CPN ML mostrada a seguir será utilizada para todos os modelos, a não ser que se diga o contrário.

Declaração de tipo

Colset PRD = with palpbpcplpdpelpf; (*declaração do tipo produto*)

Colset DT = unit with e; (*declaração do tipo elementar*)

Declaração de variáveis

Var Prdi,Prdo:PRD; (* variáveis relacionadas ao produto*)

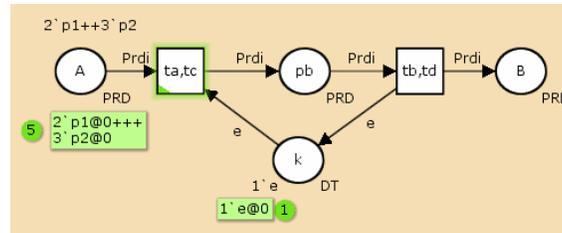
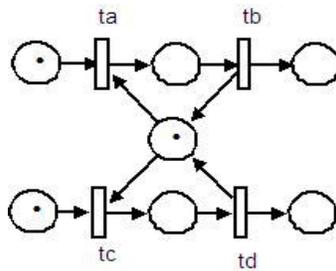


Fig.6.3: Compartilhamento de recursos em paralelo a) Modelo em RP; b) Modelo em CPN

O modelo pode ser generalizado para n células pela implementação dos seguintes passos: utilização de *tokens* diferentes no lugar *buf ta* tantos quantos forem o número de processos em paralelo; a inclusão de um arco de realimentação da transição *to* para o lugar *pp* acrescido de uma função de transformação de fase do produto e por último uma função de associação nas transições que ligam o lugar *buf* para a transição *ti* e *to* como mostra a Fig. 6.4.

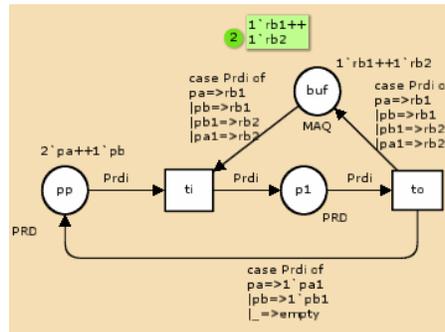
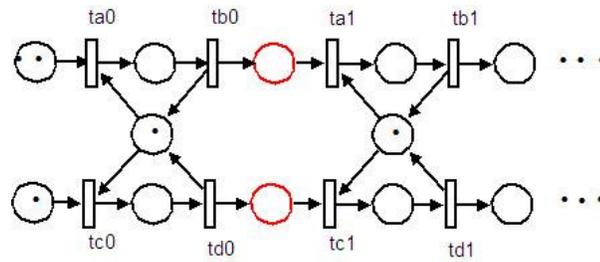


Fig.6.4: Compartilhamento de recursos em paralelo para duas células

Compartilhamento de recurso em série: É quando existem células em série compartilhando o mesmo recurso e a célula posterior depende da disponibilidade do recurso fixo compartilhado e do recurso variável gerado pela célula anterior. A Fig. 6.5 mostra um exemplo de recurso compartilhado em série. Nesse caso, só existe um recurso fixo a se compartilhado representado pelo *buffer buf*. Para que o modelo seja válido é necessário obedecer às regras apresentadas na Seção 5.8.2

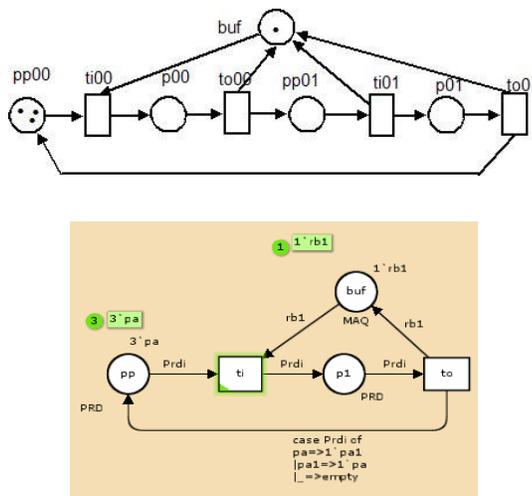


Fig.6.5: Compartilhamento de recursos em série

Concorrência o paralelismo: É quando duas ou mais partes de um produto são manufaturados em células separadas e a montagem do produto subsequente só puder ocorrer quando todas as suas partes estiverem disponíveis. Na Fig. 6.6 as marcas nos estoques p_1 e p_2 representam as partes necessárias (Moraes e Kastrucci, 2001). Pela utilização de CPN é possível reduzir a representação gráfica pela transferência da complexidade dos lugares para os *tokens* como mostra a Fig. 6.6b.

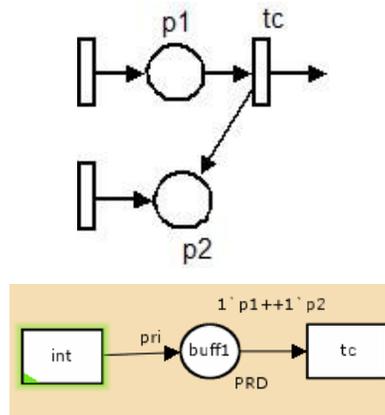


Fig.6.6 - a) Representação em RP; b) Representação em CPN

Estoque Intermediário (buffer limitado) e recursos compartilhados: Os estoques intermediários representam quantidade de peças que podem ser armazenadas entre duas máquinas. Geralmente são usados quando uma das máquinas produz mais rápido que a outra, ou quando um subsistema compartilhado precisa manipular várias peças pra obter o produto final, como é o caso da montagem de produtos (Moraes e Castrucci, 2001).

Na Fig. 5.4a as marcas do lugar A da RP representam peças disponíveis no instante inicial da operação. Cada vez que o lugar A tem uma peça pronta e disponível a RP impõe que a transição t_a só poderá disparar quando houver pelo menos uma marca na posição PV . Por outro lado, t_a interrompe a admissão de peças quando p_v fica vazio e p_b tem k marcas. Contudo, sempre que uma peça é transferida para a máquina B , uma nova marca é acrescentada em p_v . Ou seja, a transmissão de peças da máquina A para a máquina B admite um acúmulo máximo de k peças. Pela utilização de CPN é possível distinguir entre os produtos que serão armazenados e, portanto fazer

armazenagem de produtos diferentes. A estrutura para modelo é o mesmo do recurso compartilhado em paralelo mostrado na Fig. 6.3.

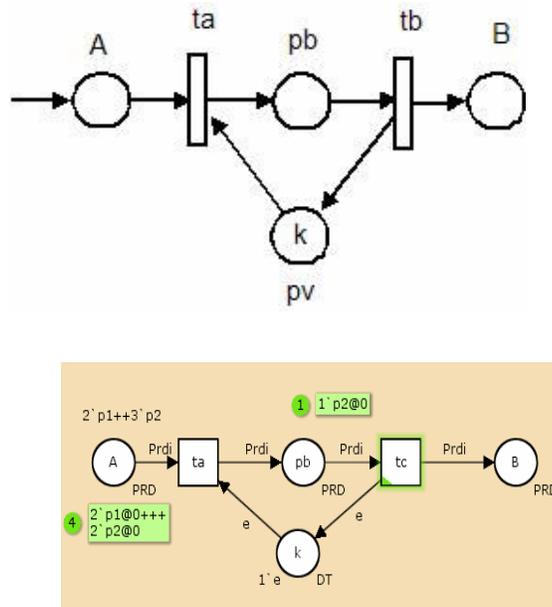


Fig.6.7 - a) Estoque limitado, representação em RP; b) Representação em CPN temporizada

Máquinas de transformação e montagem: O modelamento de máquinas deve ser capaz de representar seus principais estados internos, operações de transformação relacionadas com o produto e seus tempos de execução. A representação dos estados internos pode ser realizada através dos lugares tanto na rede estado-transição básica como na CPN. Contudo, para possibilitar a análise de desempenho é necessária à utilização de CPN temporizadas. O tipo de extensão utilizada é a rede temporizada determinística com disparo instantâneo (atômico).

Máquinas com processo de operação de transformação como tornos, centro de usinagem e frezadoras CNC, robôs de montagem, máquinas injeção de injeção plástica entre outras podem ser modeladas como na Fig. 6.8. Assim como os *buffers*, as máquinas também possuem estoque intermediário com pelo menos uma peça, correspondente à peça sendo trabalhada. Contudo no modelamento de máquina, o *anti-place* k possui semântica diferente da assumida pelo buffer de estoque limitado, pois aqui os *tokens* indicam os recursos fixos utilizados para operar a tarefa referente ao produto. No modelo temporizado da fig. 6.8b, o arco de entrada (ti, p2) associa

tempos às operações em função do *token* (produto) que é atribuído à variável **Prdi** ($p1=3s$ or $p2=6s$). O arco $(tp, p4)$ aplica a transformação no *token*, também de acordo com o valor do *token* atribuído à variável **Prdi** no processo de habilitação da transição. O modelo proposto permite trabalhar com várias máquinas em paralelo e vários produtos de entrada.

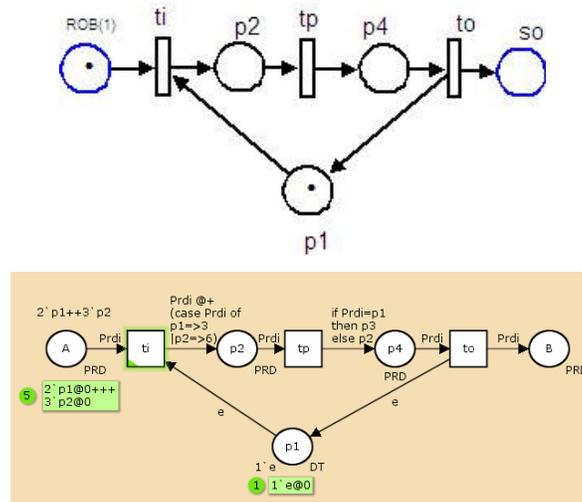


Fig.6.8 - a) Representação de máquina em RP; b) Representação em CPN temporizada.

p1 = máquina Ok e livre tp = ocorrência de peça pronta
p2 = máquina Ok e ocupada to = entrega da peça
p4 = máquina ociosa

Pode-se utilizar uma versão mais simplificada do modelamento de máquinas eliminando-se o estado de ociosidade “p4”. Conseqüentemente o modelo simplificar fica como mostra a Fig. 6.9.

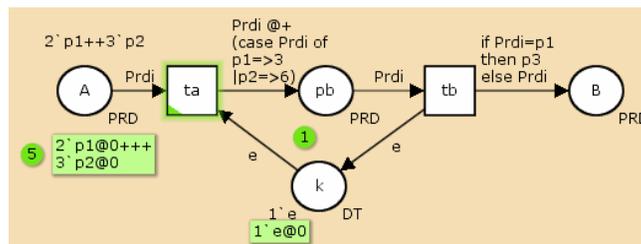


Fig.6.9 – Modelamento com CPN simplificado

Máquinas com detecção de falha: A detecção de falha em máquinas será modelada considerando o tempo médio de falha (TMDF) e a probabilidade da mesma ocorrer. A detecção de falha pode ser modelada pelas transições “tr” e “tf” e pelo lugar p3. Utilização de CPN permite a associação de diferentes falhas para um determinado estado de operação. A declaração de variáveis inclui uma nova variável de falha denominada de “Fal”. Nesse modelo acrescentamos a variável “Fal” que conterá os valores correspondentes a cada falha:

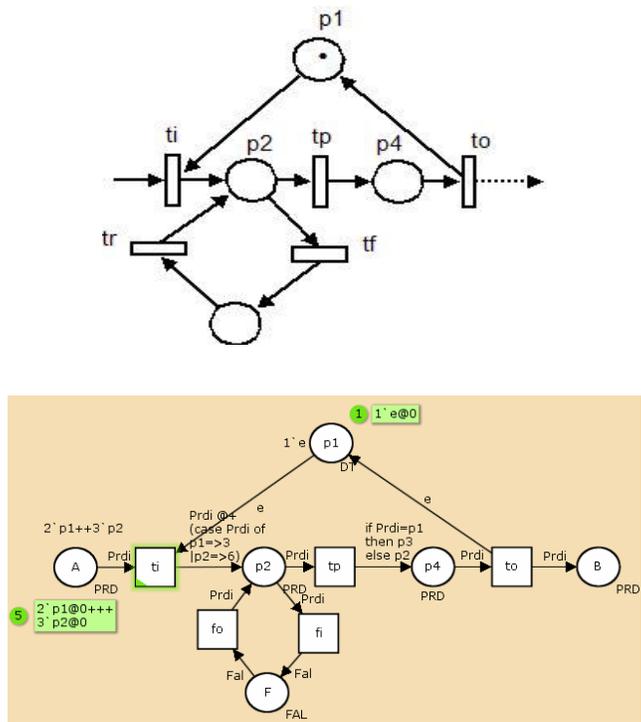


Fig.6.10 – Máquina com detecção de falha

- | | |
|---------------------------|--------------------------------|
| p1 = máquina Ok e livre | tp = ocorrência de peça pronta |
| p2 = máquina Ok e ocupada | tf = ocorrência de falha |
| p3 = máquina defeituosa | tr = ocorrência de reparo |
| p4 = máquina ociosa | to = entrega da peça |

Sistema de escolha automática e aleatória de servidor: Esse sistema é utilizado para resolver o problema de um grupo de máquinas trabalhando em paralelo. Aqui sempre que houver mais de uma máquina disponível a escolha da que será utilizada será aleatória. Todas as máquinas iniciam o processo em disponibilidade. Quando houver recursos na entrada, se qualquer uma das máquinas for liberada, esta assume

imediatamente a operação da peça. O modelamento utilizando CPN reduz bastante a representação gráfica à medida que podemos representar as máquinas através dos diferentes *tokens* correspondentes às marcações iniciais nos *anti-places* e não mais pela estrutura gráfica como ocorre na Fig. 6.11a. Para isso marcamos inicialmente os *anti-places* com as cores correspondentes às máquinas e utilizamos variáveis nos arcos do *anti-place* como mostra a Fig. 6.11b. Na representação em CPN, além da realização da escolha aleatória das máquinas, também é atribuído um tempo de operação por produto.

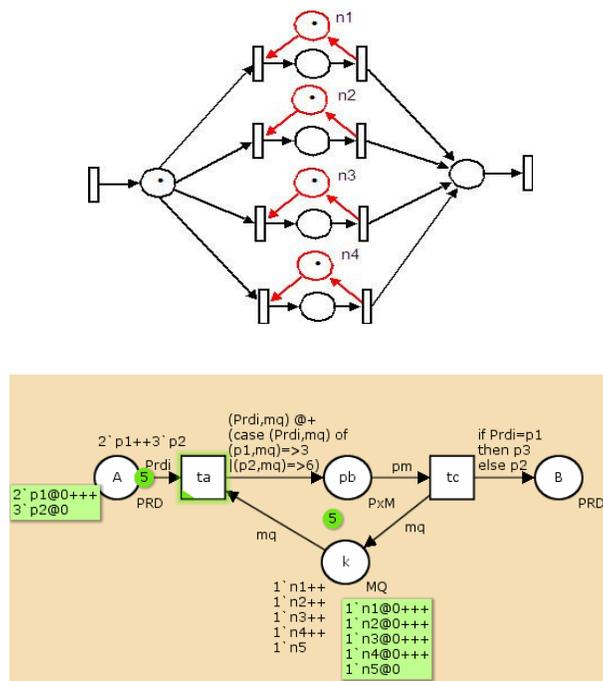


Fig.6.11 – a) Representação em CP; b) Representação em CPN

Maquinas com indicação de final de tarefa: Em muito sistema, como no caso dos sistemas de usinagem trabalhando em conjunto com robôs de transporte de peças, a máquina B precisa enviar um comando indicando o final de uma tarefa para que outra peça possa ser aceita. Dessa forma, podemos, ainda, acrescentar uma funcionalidade ao modelo, permitindo que a saída de peça do estoque para máquina B seja comandada por ela própria no momento que termina seu trabalho numa peça. Dessa forma a máquina B poderá apresentar dois estados possíveis: em trabalho (marca em B) e em ociosidade (marca em *puf*). A RP resultante é mostrada na Fig. 6.12.

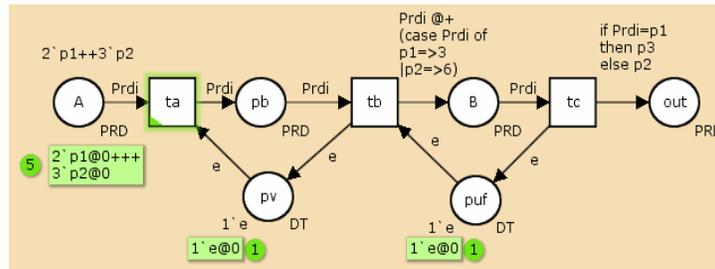
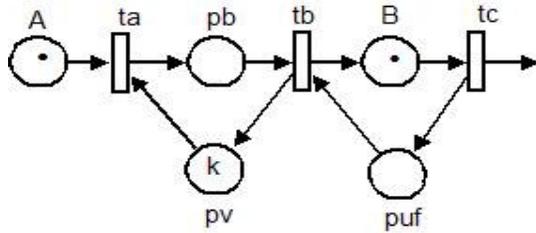


Fig.6.12 - a) Representação em RP; b) Representação em CPN temporizada

Maquinas com controle de entrada por estouro de buffer: Em muitos casos é necessário o intertravar a entrada de novas a sere, processadas ou transportadas em função de que *buffers*, em processos posteriores os buffers já estão lotados. Para resolver esse problema podemos utilizar um arco bidirecional conectando o *anti-place* do *buffer* lotado à saída do buffer que fornecerá o produto para o processo ou para o transporte. Na Fig. 6.13 a seguir vamos considerar que o lugar *_out* seja o *anti-place* do buffer *out*. Dessa forma quanto o buffer *out* estiver lotado o *anti-place* *_out* estará vazio intertravando a saída do *buffer* *pb*. Em função do arco bidirecional o disparo de *tb* não influencia na marcação de *_out*.

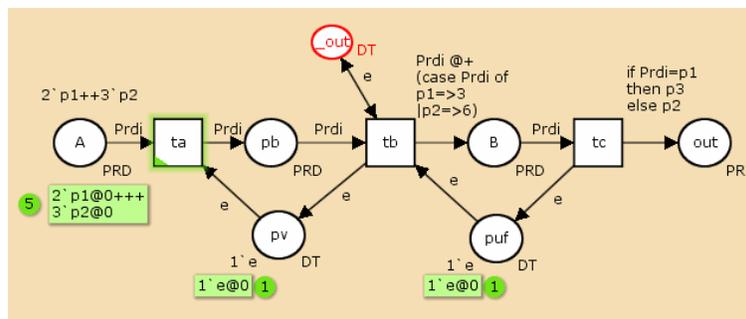


Fig.6.13 – utilização de arco bidirecional para controle de entrada

Buffer tipo Fila (FIFO): Nas redes lugar-transição esse tipo de estoque intermediário pode ser modelado de maneira simples, encadeando várias *buffers* com estoque unitário para formar uma seqüência produtiva. Nesse tipo de arranjo, o número de posições será o dobro do número que define a capacidade de estoque, o que poder tornar o modelo um pouco extenso.

Nas CPN com a utilização do conceito de lista e de instruções da linguagem CPN ML que permitem inserir elementos no final da fila de um buffer (^) e retirar elementos na ponta da fila (::) é possível modelar um buffer tipo fila com um número menor de elementos, além de permitir trabalhar com cores diferentes (cores).

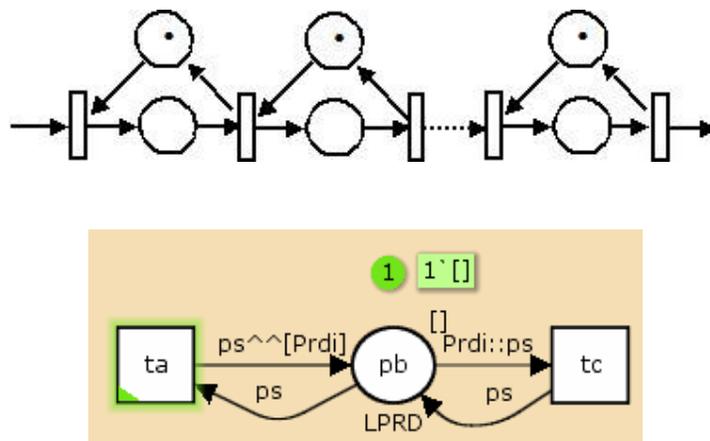


Fig.6.14 - a) Representação em RP; b) Representação em CPN com lista

Arco inibidor e buffer alimentado: Em algumas situações é necessário habilitar uma determinada transição (ação) quando um lugar estiver vazio, ou na situação mais geral quando não tiver um *token* de uma determinada cor. Uma situação corriqueira que exemplifica isso em sistema de manufatura é os *buffers* de entra que geralmente são alimentados por operadores humanos ou automáticos. Nesse tipo de *buffer* é necessário disponibilizar uma nova peça somente quando a última disponibilizada tiver sido retida. No exemplo a seguir com CPN, como não existe a função arco inibidor no CPN Tool, foi feito um arranjo que pudesse representar essa função de forma aproximada.

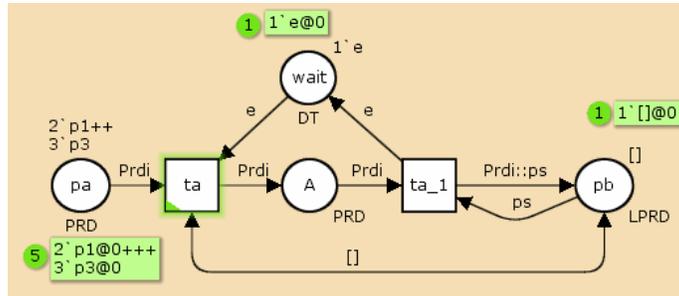
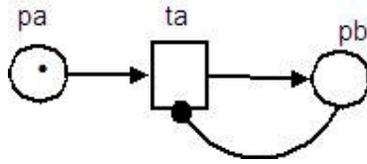


Fig.6.15 - a) Arco inibidor em RP; b) Arranjo em CPN com lista

7. ABORDAGEM PROPÓSTA DA METODOLOGIA

Neste capítulo é introduzida uma metodologia que permite o modelamento e análise de sistemas de manufatura através de uma abordagem hierárquica e modularizada baseada em redes de Petri Colorida e temporizada. Esta metodologia considera a utilização e interação de várias teorias apresentadas em trabalhos anteriores para sintetizar uma sistemática de desenvolvimento de projetos.

Como já foi dito, os Sistemas de Manufaturas Flexíveis são compostos por um conjunto de máquinas controladas por microprocessadores, linguagens específicas de programação, parametrização e configuração usadas para especificar suas tarefas e a forma pela qual se comunicam com o meio exterior. Como exemplos de máquinas que possuem sua própria linguagem, podemos citar os robôs industriais, as máquinas de usinagem a CNC e as insensoras automáticas de componentes eletrônicos.

Outros equipamentos passivos utilizam a capacidade de processadores de um controlado de uso geral, como é o caso dos CLP. Dentre esses equipamentos que necessitam de controle externo estão os alimentadores, estoques de peças, sistemas de transporte por esteira, *pallets* e elevadores dentre outros.

Sendo assim, consideramos que cada programa associado a uma máquina de um sistema de manufatura é responsável pela implementação do detalhamento de suas operações internas deixando a interação dentre os equipamentos para um maior nível hierárquico maior de controle. Dessa forma, para atender esse nível de interação entre os equipamentos, o modelo de sistema de manufatura proposto nesse trabalho fundamenta-se na representação da planta, na identificação de recurso através de *tokens*, nas transformações de produtos, nas macro-operações de equipamentos e nos tempos relativos a estas operações utilizando estratégias de refinamento. Essa metodologia considera como técnica base de modelagem as redes de Petri coloridas hierárquicas e temporizadas com auxílio de softwares de simulação.

7.1. NÍVEIS DE MODELAMENTO

Dentro do contexto da automação de sistemas de manufatura, considera-se que os elementos que o compõem devem ser interconectados formando uma “rede” onde

fluem materiais e informações. Por outro lado, do ponto de vista hierárquico, um sistema de manufatura pode ser dividido em níveis: nível de fábrica: nível de célula e nível de equipamento. Dessa forma, uma abordagem para modelagem, validação e análise de desempenho do sistema deve levar em consideração fatores como especificações *descrição da planta (modelo estrutural)*, *descrição dos processos internos* e *descrição das rotas de produção e garantias de ausência de situações de bloqueio*. Os detalhamentos das operações ou as micro-operações relacionadas ao transporte de uma peça ficam por conta dos programas específicos de operação cada máquina.

Modelamento do nível de célula: No nível de célula considera-se o estado de operação dos equipamentos e suas interações entre si. Além disso, é necessário considerar possíveis situações de bloqueio que ocorrem principalmente no compartilhamento de recursos em processos concorrentes, como na utilização de um único robô de carregamento e descarregamento para várias máquinas ou com limitação na capacidade de buffer. Nesse nível, a estratégia é representar o layout dos equipamentos como maior nível de abstração do modelo, identificando suas interligações físicas, seu posicionamento relativo, fluxo local da produção e restrições à manipulação dos produtos, estabelecendo o plano de produção da célula. Essas estruturas são modeladas através das transições de substituição que posteriormente serão substituídos pelos modelos internos e conectadas através de lugares denominados de soquetes.

Modelamento do nível de equipamento (descrição dos processos internos): Um modelo interno é gerado e avaliado através de simulação computacional ou processos matemáticos. É a descrição da planta em um nível maior de refinamento considerando seus macros estados de operação internos e sua lógica de relacionamento entre tarefas, produtos e tempo de operação. Para cada transição de substituição um submodelo predefinido é instanciado e parametrizado com as definições internas de operação e suas relações estruturais internas. Com esse tipo de modelamento é possível especificar especificações de operações das máquinas que dependam da tarefa a ser processada, ou seja, é possível definir tempo e outros parâmetros de operação de acordo com as tarefas e processos de transformação de acordo com o produto e com a etapa da produção.

Modelamento do nível de fábrica: No nível de fábrica é especificado o plano geral de manufatura através da reestruturação das rotas para movimentação de materiais entre as células que compõem a fábrica. Para modelamento nesse nível é fundamental dispor de técnicas para garantir a autonomia do sistema de movimentação de material e fluxo de informações entre os diversos arranjos do sistema. No modelo proposto nesse trabalho o plano de produção geral pode ser obtido diretamente pela interligação dos modelos das células válidas a utilizando *bufferes* válidos.

Um fator interessante nesse nível de modelamento é que a implementação de simulação da comunicação entre diferentes células não levam em consideração nenhuma característica específica do protocolo ou da tecnologia. Os dados modelados podem ser transmitidos por qualquer meio tais como I/O digital, *Hart*, *ASI*, *ProfBus* ou mesmo *Ethernet*.

A proposta dessa dissertação consiste no modelamento de células utilizando a metodologia descendente ou *up-down* com instanciamento de sub-redes, onde os equipamentos nos níveis de equipamento tais como alimentadores, robôs, máquinas de usinagem, transportadores e equipamentos de inspeção são instanciados a partir uma biblioteca de sub-redes válidas que são inseridos dentro das transições de substituição e conectados através de lugares no nível de célula, denominados de *sokets*. A manutenção das características de validação, tais como vivacidade, limitação e reversibilidade são garantidas a partir da utilização de teorias de conexões utilização teorias de extensão válidas para garantir as propriedades no modelo completo.

7.2. REPRESENTAÇÃO GRÁFICA DAS CPN

As entidades que constitui um modelo devem ser capazes de representar o processo modelado da forma mais precisa e sintética possível, nesse sentido é necessária uma interpretação para cada entidade do modelo que as relacione com seus equivalentes no sistema real. Para atender esses requisitos as CPN são bastante adequadas, pois oferecem facilidade de modelamento avançado tais como *tokens* coloridos e refinamento hierárquico de transições (Zimmermann and Hommel, 1999). O único

inconveniente é que a forma simples da representação gráfica quando utilizamos CPN é prejudica pela necessidade de definir os tipos de cores e variáveis. Contudo para restritas áreas de aplicação é possível definir uma classe de CPN dedicadas, com uma reduzida quantidade de tipos (Zimmermann and Hommel, 1999). Para o escopo desse trabalho serão definidos apenas 3 tipos de *tokens*. *Tokens de recurso fixos* que representam máquinas e equipamentos fixos utilizados na operação dos produtos, *tokens de recurso variáveis* que modelam produtos e subprodutos dentro do sistema de manufatura e consiste de um nome associado ao produto e a seu estágio de manufatura. *Tokens elementares* não tem uma cor especial e são similares às redes de Petri não coloridas.

7.3. ESTRUTURAÇÃO E INTERPRETAÇÃO

Na metodologia proposta, cada uma das transições de substituição é refinado pelas sub-página, que descreve o comportamento das máquinas e transportadores com mais detalhe. Sub-modelos provenientes de uma biblioteca de modelos padrões são instanciados e parametrizados durante o processo de refinamento do modelo. Somente produtos e sub-produtos serão considerados no nível mais alto de abstração (modelamento da planta). No nível de equipamento, além desses, também são representados os recursos fixos como máquinas e equipamentos. Cada equipamento no nível de célula é representado por uma composição mínima formado por um lugar uma transição de substituição. Os lugares denominados *sokets* são utilizados para realizar as conexões horizontais entre as transições de substituição, enquanto que lugares designados com *portas*, localizados nas sub-páginas, são usados para as conexões verticais entre os modelos da planta na super-página e os sub-modelos internos dentro das transições de substituição. Fusões entre lugares dentro das redes das sub-páginas também são utilizadas para introduzir lógicas de sincronização e permitir conexões válidas entre essas redes. A seguir interpretaremos os elementos fundamentais de uma rede de Petri da seguinte forma:

Transição

A Transição no nível de macro-transição (transição de substituição) designará um bloco geral de ações transformação ou transporte relacionado com um determinado equipamento do processo, agindo sobre um produto especificamente a ele associado. Contudo, vale a pena ressaltar que as macro-transição é apenas uma representação gráfica, em um nível hierárquico maior de uma sub-rede mais detalhada e que não podem assumir estados de habilitação ou realizar ações de disparo diretamente. Essas ações são realizadas pelas suas transições internas de sub-redes. Também são nas suas transições e arcos orientados internos que serão modeladas as restrições de entrada de produtos através das funções de guarda ou funções associadas ao arco. Contudo, ainda podemos olhar para as transições de substituição uma transição onde o disparo ocorrerá em dois passos de tempos distintos, ou seja, o início de execução consumirá os *tokens* dos *sokets* de entrada e um final de execução que gera os *tokens* para os seus respectivos *sokets* de saída.

Já a transição padrão como elemento de modelamento interno de uma sub-página representará uma ação instantânea do processo, como o acionamento de um sensor ou a conclusão de uma tarefa, ou a transferência de produtos entre estados internos de uma máquina. Nesse caso a execução da transição ocorrerá em um único passo, consumindo os *tokens* dos lugares de entrada e gerando os *tokens* para os lugares de saída ao mesmo tempo.

Além das transições padrões, o modelo proposto também considera a utilização de transições temporizadas. Nesse caso, o disparo da transição depende também de um temporizador geral da rede e de um *stamp* de tempo associado a cada *token* durante o disparo de uma transição. Essa técnica é chamado transição com temporização atomizada e serve para representar o tempo de execução de cada tarefa de máquina associada ao produto de entrada. A partir dessa extensão é possível analisar o desempenho da rede.

Lugares

Os lugares podem possibilitar três tipos de interpretações. Na super-página, ao mesmo tempo serve com elemento de fusão entre as macro-transições e como designador do estado de entrada dos equipamentos com relação aos sub-produtos, matéria prima e dados requeridos para a execução de uma operação. Nas sub-páginas os lugares representam estados internos de operação dos equipamentos ou o estado assumido pelos produtos e subprodutos em uma etapa do processo de máquina.

Os lugares de fusão são aqueles utilizados para realizar a conexão entre partes do modelo. Os lugares podem assumir nomes diferentes de acordo com seu posicionamento relativo na hierarquia do modelo. Na super-página o lugar é denominado de soquete e sub-página é denominado de porta. A fusão no maior nível hierárquico é realizada pela fusão entre soquetes, enquanto que a conexão entre níveis é realizada entre as portas e os soquetes. Para efeito de sincronização entre modelos de sub-páginas algumas vezes são feitas conexões entre portas.

Tokens

Os *tokens* também apresentam mais de uma interpretação. Na descrição em nível de planta designam produtos, subprodutos e matéria necessários para a execução de uma determinada operação de manufatura ou transporte por parte de um equipamento. Por outro lado, nos níveis internos (nível de equipamento) de modelamento, além do fluxo de produtos pode representar dados provenientes dos estados das máquinas e combinações de recursos fixos e variáveis.

Arcos

Os arcos representam o fluxo de operação. Têm papel importante no modelo. Além de servir como orientador do fluxo e elemento de conexão entre os nós da rede, podem também conter funções de transformação e função de temporização. Essas funções de transformação modificam o estado do produto ou da informação. A função de temporização pode associar *stamps* de tempo aos *tokens* assim como as transições.

Além disso, no nível de super-página, suas inscrições podem funcionar com indicador visual do fluxo de produção.

7.4. SÍNTESE DO MODELAMENTO

A metodologia inicia pela construção de um gráfico denominado gráfico de operação que representa a modelagem das operações individuais de cada um dos equipamentos através de uma representação simplificada, retirada diretamente do layout da planta. Esse gráfico identifica o fluxo das operações do processo, quem realiza essas operações e onde essas operações são realizadas, como mostra a Fig. 6.1. Esse gráfico é uma contribuição desse trabalho e serve como uma referência visual para os próximos passos da definição do modelo. O fluxo de cada operação será indicado por um arco orientado, partido das origens dos recursos, passando pela unidade executora da operação e chegando ao destino dos recursos. Se a unidade de destino também executar uma operação, esta também será representada. Dessa forma, cada operação será indicada por um círculo hachurado desenhado no interior da unidade executora. Os recursos compartilhados se caracterizam por ter mais de um fluxo de operação, e estes fluxos são entre equipamentos diferentes.

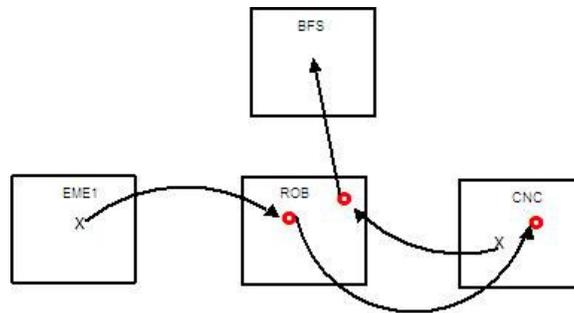


Fig.7.1: Gráfico de fluxo de operação

Na próxima etapa é desenvolvido a RP lugar/transição mínima equivalente para cada operação. Cada equipamento será representado inicialmente por uma transição e uma ou mais posições de entrada ligada do mesmo lado da transição. O fluxo da operação entre dois equipamentos indicado por um arco orientado ligando a saída da transição do dispositivo de origem com o um dos lugares do dispositivo destino.

A conexão entre as RPs básicas equivalentes das operações são realizada através da técnica de fusão de lugares (composição externa) para obter a representação completa da planta. Quando uma determinada transição executar mais de uma operação, essa será representada pelo nome da transição indexado a um número que indica o número da operação. Dessa forma se um robô ROB executar três operações diferentes, essas são indicadas por ROB(1), ROB(2) e ROB(3). Semanticamente, considera-se que cada índice diferente representa uma operação compartilhando o mesmo robô ROB. Essa interpretação evita que tenhamos que escrever estrutura de anti-places para todas as operações ligadas ao mesmo recurso fixo. Em geral as operações diferentes estão relacionadas com o produto diferentes a serem manipulado. Esse tipo de estruturação oferece um suporte imediato para uma composição horizontal das redes. As várias redes podem ser vistas como peças de um quebra-cabeça que serão unidas para a obtenção do sistema maior. O compartilhamento de recurso apóia-se nas teorias de PME e SME, dessa forma as condições de inicialização prevista nesses dois teoremas deve ser obedecidas para a manutenção da validade da rede.

Em seguida aplica-se a dobragem baseada em marcas para conversão da rede de Petri básica em CPN de alto nível (Jensen e Rozenberg, 1991). Esta consiste em seguir o fluxo do processo e sempre que uma nova transição com índice $n > 1$ for encontrada, retorna o arco orientado para a transição que foi representada anteriormente com índice $n = 1$ e identificar o número do índice ou o código do produto no arco orientado em questão ou através de função de guarda para restringir os *tokens* para aquela transição. Fazer essa operação para todas as transições até que todo o fluxo de processo seja concluído. Com isso obteremos o macro-modelo com suas restrições de produtos. Nessa etapa, os *tokens* só poderão ser do tipo objeto. A extensão das teorias de PME e SME para redes CPN é garantida como foi mostrado na Seção 6.1.

Nessa etapa, a declaração de variáveis e funções deve ser feita. O número de variáveis é simplificado pela utilização de três tipos bem adaptadas para sistemas de manufatura: variáveis de recursos, tokens usados para modelar produtos e subprodutos; recursos fixos, usados para modelar máquinas e equipamentos; tokens elementares que não têm uma cor especial, mas representam estados de operação (Zimmermann and Hommel 1999). Além disso, as funções de guarda são criadas para restringir o fluxo de produtos, enquanto que as funções de arco são usadas para

realizar transformações de produtos. Na lista de produtos da situação problema da Tabela 9.1, no Capítulo 9, todos os elementos foram declarados com o mesmo tipo “Prdi”.

Em seguida utiliza-se refinamento de transição para detalhar o funcionamento dos subsistemas. Essa etapa será realizada pelo instanciamento e inserção nas transições de substituição dos submodelos de manufatura provenientes da biblioteca de padrões pré-definidos e de acordo com as restrições impostas pelas teorias de PME e SME. Dessa substituição obteremos a rede de Petri detalhada. É importante ressaltar que a ferramenta de simulação hierárquica usada fornece facilidades na construção de modelos através da construção automática páginas hierárquicas com seus elementos de conexão.

Nessa fase, analisam-se as propriedades de limitação, vivacidade e alcançabilidade para a obtenção da validade da CPN. Para realização dessa etapa utiliza-se método de simulação através de ferramentas com o CPN Tool ou métodos analítico no caso de redes mais simples. Caso alguma propriedade não seja satisfeita, fazem-se as correções necessárias na estrutura do modelo ou na marcação inicial para obter os resultados esperados. Os ajustes de conexão entre sub-redes podem ser feito através de buffers válidos conseguidos a partir de *anti-places*, obtidos através de fusão de lugares localizados dentro de uma ou mais transições de substituição. Essa avaliação também pode ser feita através de analogia com estruturas de Petri nets válidas.

Essas sub-modelos de CPNs recebem a parametrização adequadas de modo a corresponder aos tempos de processo, tamanho de *buffers* e associações de tarefas correspondentes ao modelo real. Os sub-modelos devem possuir o nível adequado de abstração, considerando todos os sensores, atuadores os eventos externos e internos conseqüentes.

Em seguida, ajustam-se os *links* de conexão entre as transições para atender os requisitos do “Teorema de Validade de Extensão de Petri Net” (Zhou 1989). Esse requisito pode ser atingido pela utilização de link padrão ou por modelos de Buffer de Petri Net Válidos proposto por Qiao et all (2002). A utilização desses princípios garantirá que a rede manterá suas propriedades de validade.

Com o objetivo de avaliar o desempenho de um sistema de manufatura e sugerir melhorias. Tempos de retardos são associados aos disparos através dos arcos ou em alguns casos com das próprias transições. Considerando o tempo fixo das operações de processo e transporte de peças, pois é o que geralmente ocorre em sistemas de manufatura reais. Dessa forma dizemos que o retardo de tempo é determinístico. Para obter resultados mais realísticos nesse tipo de processo, método de análise incorporando disparos não exponencial distribuído tem que ser usado (Qiao et al, 2002). Simulação deve ser usada nesse estágio do modelamento em função da complexidade e do tamanho do espaço de estado gerado.

A análise do modelo é feita através de comparação do desempenho do sistema com base em suas variáveis de saída com *throughput* e *índice de falhas*. São empregados métodos estatísticos aplicados aos resultados da observação do comportamento dessas variáveis em função de ajustes dos parâmetros de entrada, considerando a média obtida e o desvio padrão. Os parâmetros de entrada considerados são os tempos de execução de tarefas, número de máquinas disponíveis em paralelo, tamanho máximo dos *bufferes* e tipo de buffer utilizado e ocorrência de falha.

Como mencionado, na fase de simulação e análise do modelo será utilizado o software de simulação CPN Tools. Essa ferramenta foi escolhida por permite a edição, simulação e análise de Redes de Petri hierárquicas e coloridas, estendendo sua capacidade de estruturação através de fusão de lugares e macro transições denominadas transição de substituição. Todas essas características convergem com o objetivo desse trabalho.

8. APLICAÇÃO DA METODOLOGIA

Este capítulo apresenta o modelamento de uma célula de manufatura, com base na s apresentada na Seção 6.4. A proposta possibilita uma metodologia modular e hierárquica através de uma estruturação *top-down* para o desenvolvimento do programa para controle de células de manufatura integradas por computador. O sistema a ser modelada é baseado em uma célula didática utilizada no laboratório de Mecatrônica do SENAI do Amazonas. Essa célula é composta por um centro de usinagem, um robô articulado com cinco graus de liberdade, um sistema de transporte por esteira e alguns *buffers* de montagem e de armazenagem. Este sistema de manufatura foi construído pela empresa *AMATROL Learning Systems for Technical Education* localizada nos Estados Unidos. Sua compra foi realizada através da empres Nova Didacta que também é responsável pela manutenção da mesma. Esse sistema foi escolhido pelo fator da grande quantidade de documentação disponível e pela experiência no manuseio desse equipamento por parte do autor.

8.1. SITUAÇÃO PROBLEMA

Descrição do produto:

O produto final é uma caixa de engrenagem composta pelas seguintes partes: 1 caixa de acondicionamento, quatro pinos de fixação, duas engrenagens e uma tampa com dois furos para encaixe das engrenagens e quatro furos para encaixe da tampa com a caixa de condicionamento através dos pinos de fixação.

Descrição geral do processo:

A estrutura foi idealizada para montagem de um produto didático (caixa de engrenagem) com quatro partes: a caixa a engrenagem com os pinos de fixação já montados, as duas engrenagem e uma tampa com seis furos.

As partes de entrada são: a caixa de acondicionamento já com os pinos fixados, a tampa lisa sem furo e as engrenagens. As caixas entrarão no processo através de um alimentador gravitacional com um sensor de presença de peças. As tampas sem furo estarão localizadas em um alimentador em pilha.

- O operador fornece a matéria prima de entrada do processo através dos alimentadores de entrada com as caixas de condicionamento, as engrenagens e a tampa sem furo quando for requisitado pelo processo.
- A as tampas são furadas nas máquinas ferramentas CNC.
- O conjunto formado por caixa condicionadora, engrenagens e tampa é montado para forma a caixa de engrenagem.
- A caixa montada é levada ao CNC novamente para sofrer um processo de final de acabamento.

Identificação dos equipamentos e das operações:

O sistema é composto por um centro de usinagem CNC, um robô articulado de 5 graus de liberdade, um uma esteira de transporte acionada pelo controlador do robô, trilho de deslocamento horizontal do robô acionado pelo controlador do robô ao, um buffer de fixação e montagem de peças, uma estação de medição das dimensões da peça, um leitor de código de barras, um alimentador gravitacional e um arranjo de armazenamento para produto final.

Operações do robô: movimenta-se horizontalmente no em um trilho horizontal e realiza as seguintes tarefas:

- a) retira a tampa lisa do alimentador e coloca-a no centro de usinagem para furação (ROB1);
- b) retira a tampa usinada do centro de usinagem e levando até o estoque de tampas furadas (ROB2);
- c) o robô monta o produto no buffer de montagem se todas as peças de entrada estiverem disponíveis (ROB3): caixa, tampa furada, engrenagens;
- d) o robô leva o produto montado para a usinagem de acabamento (ROB4);
- f) o robô leva o produto para a esteira transportadora de saída (ROB5).

Operações do centro de usinagem:

- a) realiza a furação da tampa e indica o término do trabalho;
- b) realiza a usinagem de acabamento e indica o término do trabalho;

Operações da esteira:

- a) transporta o produto para a área de saída de produtos;
- b) transporta o produto para área de rejeito de produtos conforme o resultado da medição.

Operações do controlador (CLP):

- a) controla a seqüência de operações e as condições de precedência para a realização do produto;
- b) maximizar a utilização dos equipamentos e demais recursos de produção;
- c) matém registro de quantidade de produtos produzidos e de rejeitos;
- e) trata a ocorrência dos eventos externos prioritários.

LAY-OUT

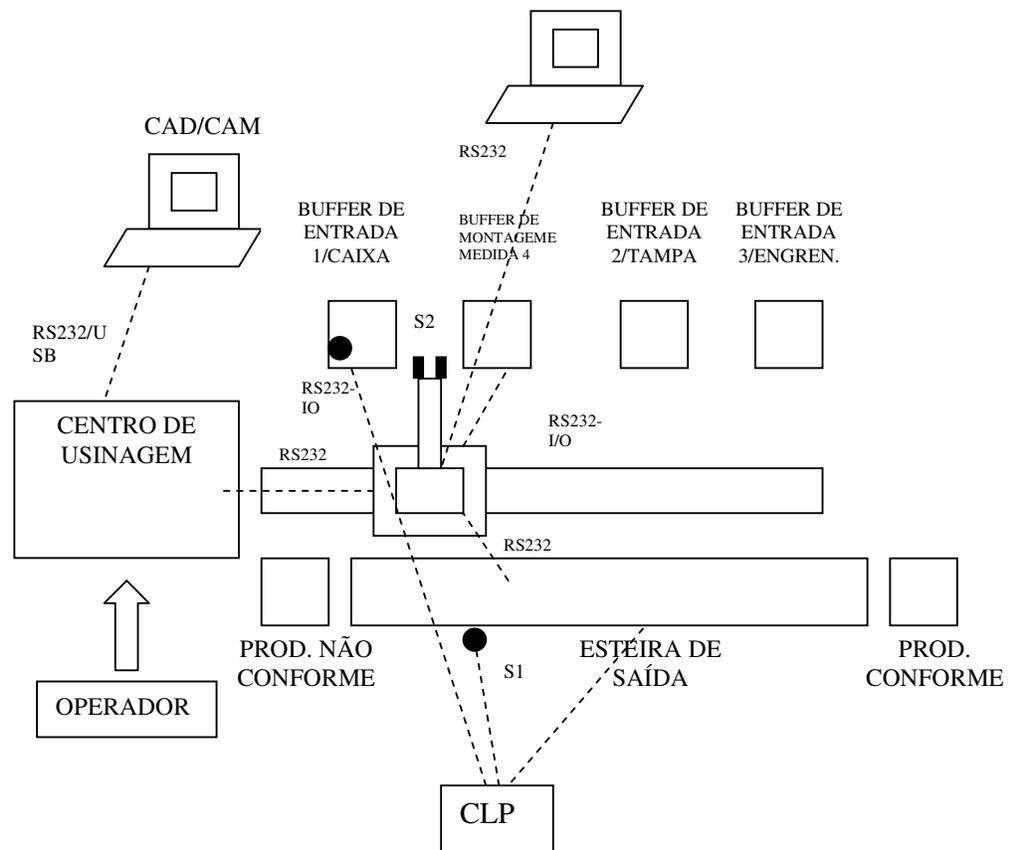


Fig.8.1: Célula de Manufatura a ser modelada.

8.2. PASSOS DO MODELAMENTO

1° PASSO: Desenhar o diagrama de operações:

A metodologia inicia pela definição do layout físico como mostra a Fig.8.1 e pelo conjunto de valores (símbolos) e variáveis relacionado com os recursos envolvidos. Considera-se que cada modificação executada em um produto gera um produto diferente. Dessa forma, cada recurso, máquina ou equipamento correspondem a um *token* diferente como mostrado na Tabela 8.1.

token	Products
pa	flat cover
pb	drilled cover
pc	box
pd	engine
pe	set of peaces
pf	Prod. assembled
pg	Prod. finished
e	Machine

Tab. 8.1: Lista de recursos e sua representação

2° PASSO: Desenhar o diagrama de operações:

O processo é formado por seis equipamentos que se relacionam de acordo com o diagrama de operação mostrado na Fig. 7.2.

Cada círculo hachurado identifica uma operação do processo e é desenhado dentro do equipamento que realiza a operação. As operações podem ser de vários tipos como transporte, montagem ou usinagem. A ponta da seta indica o destino do produto e “x” um fornecimento de peça o origem da operação.

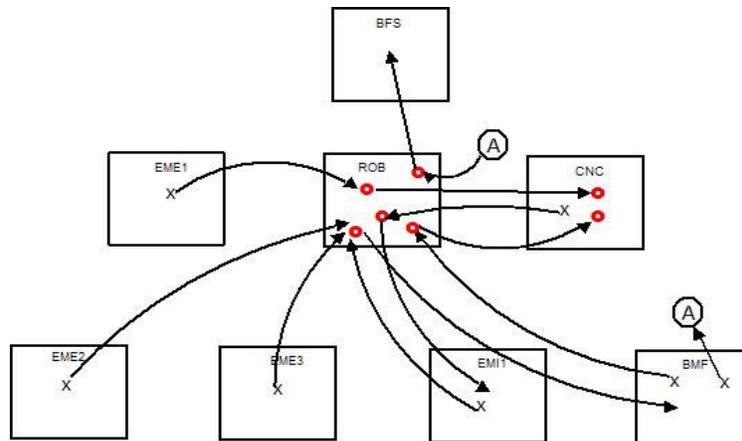


Fig.8.2 – Diagrama de operações

Legenda de equipamentos:

EMI1 – estoque intermediário de tampas furadas

EME1 – estoque de tampas sem furo

EME2 – estoque de caixa de acondicionamento

EME3 – estoque de engrenagens

BFM – buffer de montagem intermediário

BFS – buffer de saída

ROB – robô de transporte e montagem

CNC – Máquina ferramenta CNC

2º PASSO: Desenvolver o modelo de RP simplificado de cada operação:

Desenhar as operações separadamente utilizando rede de Petri padrão. As operações indicarão o fluxo de trabalho mínimo através da interconexão orientada entre os equipamentos, baseado no diagrama de operação da Fig. 8.2.

Cada posição representada na super-página representa estoques intermediários de peça. Os *tokens* devem ser entendidos como produtos e sub-produtos. As transições, nesse nível hierárquico, devem ser entendidas como a representação geral do conjunto das operações de máquina e equipamentos que serão executadas quando o *token* for consumido por essa transição.

Se um equipamento realizar mais de uma operação, cada operação será indicada pelo nome da transição que representa o equipamento e um índice que indica qual foi a operação executada no mesmo equipamento começando pelo índice 1. No representação da Fig. 8.3 o robô executa 5 operações: ROB(1), ROB(2), ROB(3), ROB(4) e ROB(5). O CNC executa duas operações: CNC(1) e CNC(2). O modelamento começa pelo equipamento com maior número de operações, que no caso da referida planta é o robô.

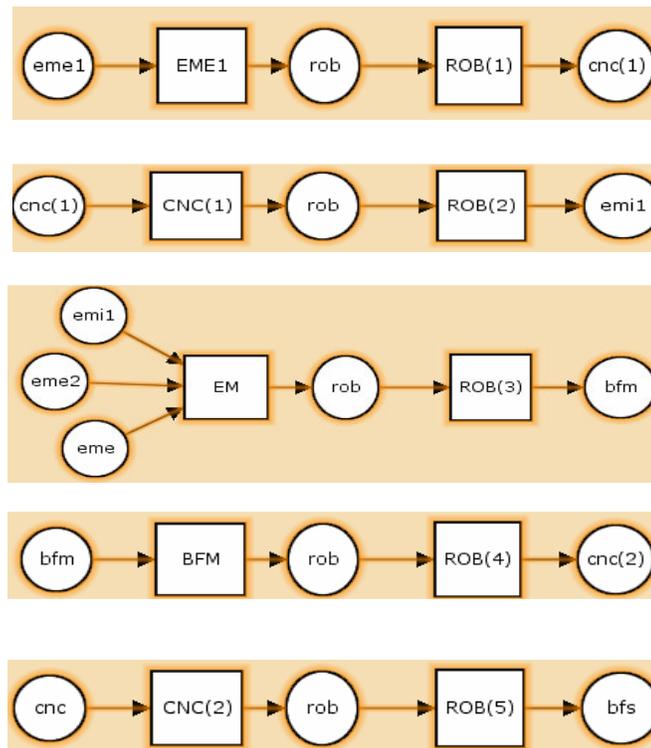


Fig.8.3 – RP não colorida das operações básica

3º PASSO: Fusão das sub-redes:

Realizar a composição por fusão externa de lugares entre as sub-Petris para obter a rede de Petri da planta simplificada (RPSP). A fusão é realizada entre os lugares que se repetem em cada operação e está indicado no diagrama pelas linhas pontilhadas.

Pela fusão entre as operações 1 e 2 através dos lugar $cnc(1)/cnc(1) \rightarrow cnc(1)$ obtemos o **sub-processo 1** que corresponde à transformação da tampa lisa em tampa furada:

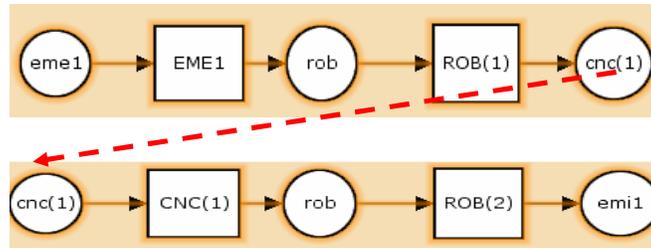


Fig.8.4 – Fusão externa de lugares

A operação da Fig. 8.5 representa a montagem da caixa, que por si só já corresponde a um sub-processo de transformação que designaremos **sub-processo 2**:

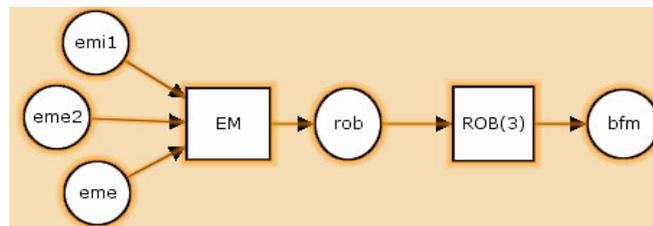


Fig.8.5 – RP não colorida da montagem da caixa

Pela conexão entre as operações entre os lugares denominados $cnc(2)/cnc \rightarrow cnc(2)$ podemos obter o **sub-processo 3** que corresponde ao acabamento final da caixa:

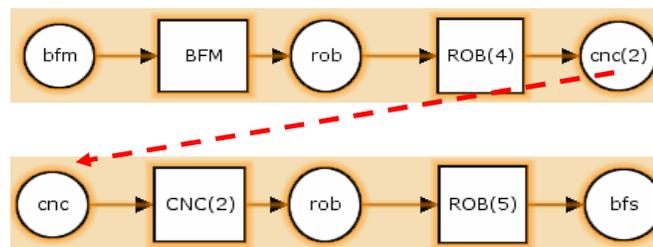


Fig.8.6 – RP não colorida do sub-processo 3

Cada sub-processo gera um sub-produto de entrada para o sub-processo subsequente de produção até que o produto final esteja. Desta forma o sub-processo 1 gera produto de entrada para o sub-processo 2 e o sub-processo 2 gera produto de entrada para o sub-

processo 3. Dessa forma podem encadear esses sub-processos através da fusão de lugares para obter a rede de petri da planta simplificada (RPSP) como mostra a Fig. 8.7.

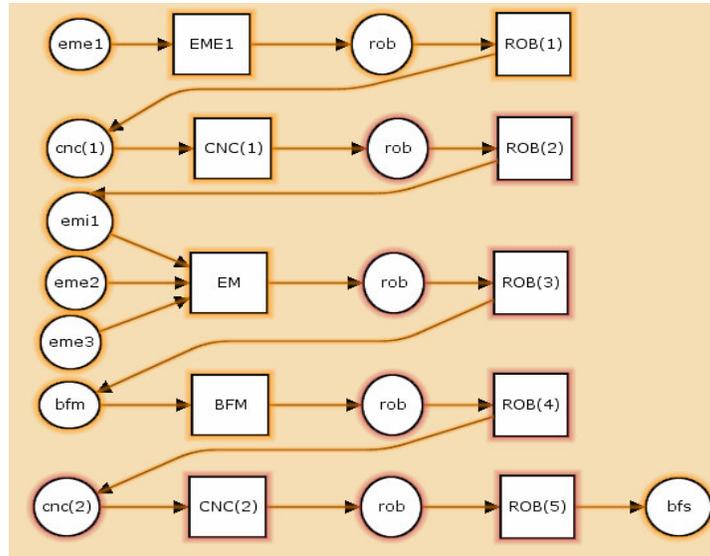


Fig.8.7 – RP não colorida do processo total

4º PASSO: Conversão da Rede de Petri Lugar-Transição para Rede de Petri Colorida:

Realizar a conversão da rede de Petri básica para CPN hierárquica através da composição interna utilizando dobragem baseada em marcas (Jensen e Rozenberg, 1991). Para uma melhor organização do modelo, resolvemos usar uma variável elementar que denominaremos “*e*” com apenas um valor de *token* para identificar o estado de máquina e uma variável “*Prd*” definida como o conjunto de produtos. A variável “*Prdi*” é do tipo PRD (*enumerado*) que é um identificador utilizado para quantidade limitada de valores. Como as variáveis que utilizaremos servirão para representar um número limitado de produtos e máquinas, o mais adequado é utilizarmos *clorset* do tipo *enumerated value*, conforme demonstrado na lista de declarações escrita em COM ML. Esse conjunto de declarações pode ser considerado para a grande maioria dos modelos aplicados a sistema de manufatura.

Modelamento do robô com distribuição de peças:

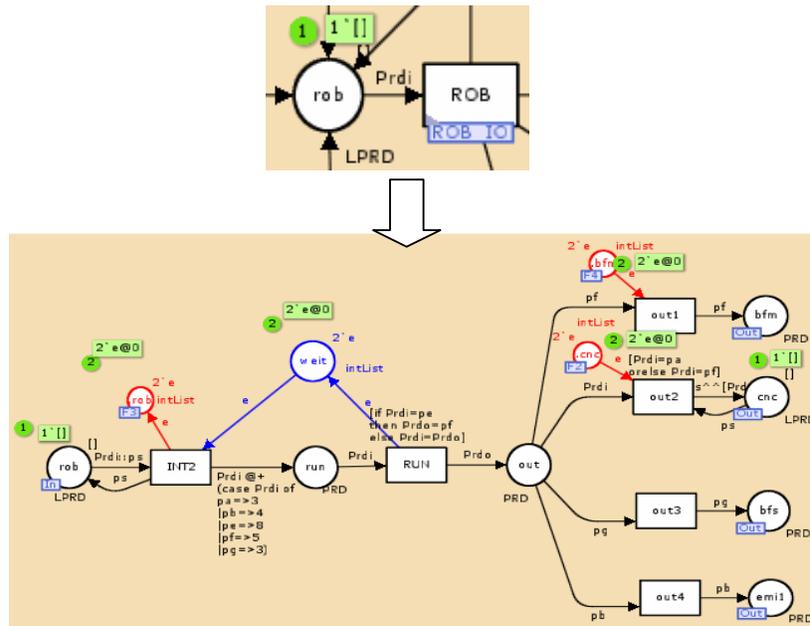


Fig.8.9: Detalhamento da transição de substituição ROB (robô com distribuição de peças)

O modelo do robô utiliza um buffer tipo fila denominado *rob* na entrada e um modelo de distribuidor de peças na saída. O lugar *.rob* em vermelho é o *anti-place* usado para formar o *bufferes válidos* de entrada na conexão com buffer *EMEI*, enquanto que os lugares *.bfm* e *.cnc* são os *anti-places* de saída que constituirão os *bufferes válidos* das ligações entre ao robô e o CNC e o robô e o buffer BFN respectivamente para a garantindo as propriedades de vivacidade, reversibilidade e limitação segundo o teorema de extensão de Petri Net válido (Zhou 1989).

Estoque limitado de entrada EME1:

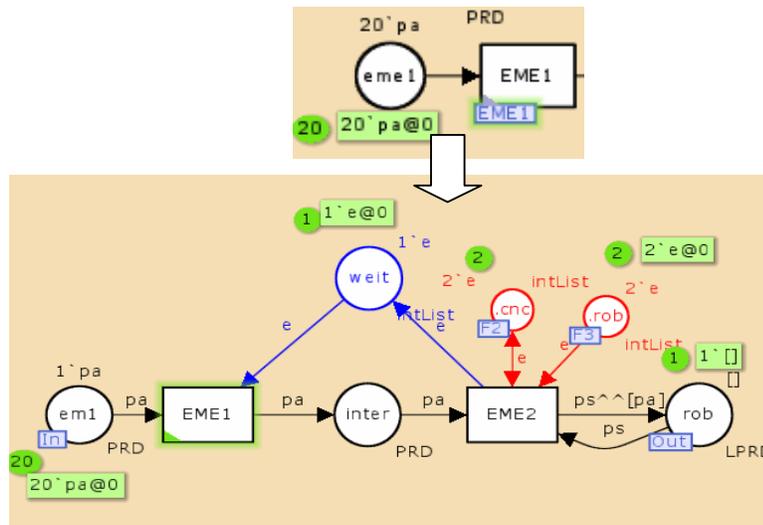


Fig.8.10: Detalhamento da transição de substituição EME1 (buffer de entrada).

O Estoque de entrada está conectado à entraves de um buffer do tipo FIFO pela utilização da instrução “ps^[pa]” que insere o elemento no final da fila. O lugar *.rob* em vermelho é o *anti-place* usado para formar o *buffers válidos* para a conexão de saída com a transição ROB, enquanto que os lugares *.cnc* é usado para limitar a saída e produtos do buffer EME1 quando o buffer *cnc* estiver cheio.

Buffer intermediário BFM:

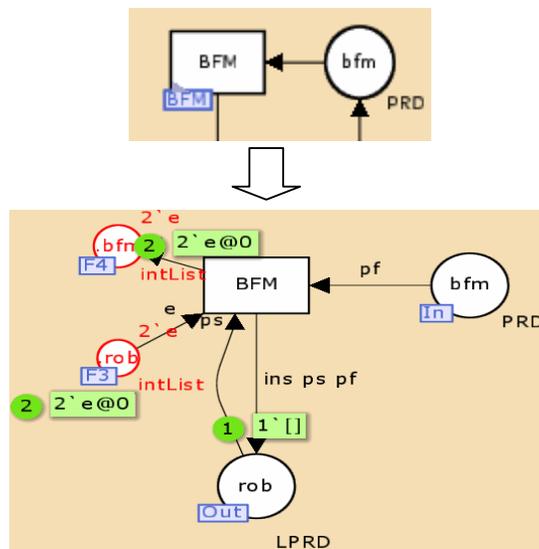


Fig. 8.11 – Detalhamento da transição de substituição BFM (buffer limitado intermediário).

O buffer intermediário está conectado através de um buffer FIFO pela utilização da instrução “*ins ps pf*” que insere o elemento no final da fila. O lugar *.bfm* em vermelho é o anti-place usado para formar o *buffers válidos* de entrada proveniente conexão com a transição ROB, enquanto que os lugares *.rob* são usado para limitar o buffer de saída para o ROB. O buffer EM1 utiliza a mesma estrutura.

Máquina CNC:

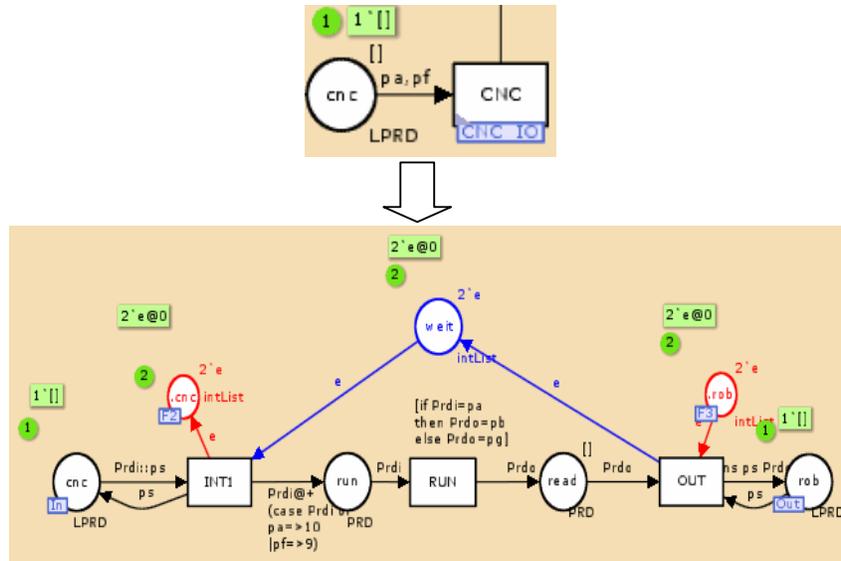


Fig.8.12 – Detalhamento da transição de substituição CNC (Máquina CNC).

O arco (INT1, run) contém a expressão de atribuição de tempo para cada operação e a transição RUN executa a operação de transformação do produto.

6º PASSO: Verificação das propriedades de validade e ajustes:

Analisar as propriedades de limitação, vivacidade e alcançabilidade para a obtenção da conformidade qualitativa da CPN. Para essa etapa utiliza-se método analítico ou ferramenta de simulação CPN Tools. Caso alguma propriedade não seja obedecida, ajusta-se a rede através da adição de novas posições ou transições. Para realização dessa verificação utiliza-se o método de enumeração com a ferramenta de simulação CNP Tools. A Tabela 8.1 mostra o resultado obtido para o modelo completo do sistema de manufatura proposto.

[Home properties]			[Statistics]
-----			-----
Home Markings:			State Space
Initial Marking is			Nodes: 18033
a home marking			Arcs: 64695
			Secs: 65
[Best Int. Roundedness]			Status: Full

Boudeds	Upper	Lower	[Liveness]
FMS'eme3 1	8	0	-----
OPR'eme3 1	3	0	Live transit.: All
EME1'eme1 1	4	0	Dead Markings: None
FSM'bfd 1	4	0	Dead Transit.: None

Tab.8.1 – RP colorida marcada da super-rede

O relatório de espaço de estado acima de foi automaticamente produzido em alguns minutos e contém muitas informações altamente úteis sobre o comportamento da rede modelada. Conseqüentemente, também obtemos uma idéia do funcionamento do sistema, uma vez que normalmente reflete erros do processo do sistema real. Os resultados estatísticos demonstram que a simulação foi completa, ou seja, contém todos os estados alcançáveis com um total de 18033 nós e 64695 arcos. Os resultados também mostram que o modelo manteve as propriedades desejadas de *vivacidade, limitação e reversibilidade*.

7° PASSO: Definição dos tempos de operação

Após a confirmação das propriedades de validade do modelo, é necessário inserir os parâmetros de tempo de operação para cada tarefa executada pelas máquinas. Isso se faz necessário para que possamos proceder a análise de desempenho do sistema. Esses parâmetros são introduzidos através de funções que definem os tempos de execução de tarefas em função dos *tokens* a serem manipulados, do equipamento e do estado de operação. Os tokens temporizados precisam ser especificados no processo de declaração de variáveis. A propriedade de tempo (*stamp* de tempo) é atribuída aos *tokens* no disparo das transições através das *funções de guarda* ou das *funções de arco*. Em seguida mostra-se a lista de tempo para as operações do sistema modelado e

sua declaração na linguagem CNP ML com escopo de temporização associado as variáveis.

TEMPO DE OPERAÇÕES:

Operações do robô

Mover produto pa (INT1) = 3s

Mover produto pb (INT1) = 4s

Mover produto pf (INT3) = 8s

Transformar pe em pf (INT4)

Operações do CNC

Transformar pa em pb = 10s

Transformar pf em pg = 9s

SINTÁTICA DA DECLARAÇÃO EM CPN ML:

Declaração de tipo

Colset PRD = with palpb|pcl|pdl|pel|pf timed;

Colset DT = unit with e timed;

Declaração de variáveis

Var Prdi,Prdo:PRD;

8° PASSO: Avaliação de desempenho

O objetivo da investigação proposta nessa fase do trabalho é obter um melhor entendimento da relação entre os detalhes do sistema de manufatura e a medida de desempenho de modo a permitir sugestões que possibilite o aumento da produtividade do sistema e das relações custo/benefício. Nesse sentido, tempos de retardos devem ser associados aos disparos das transições. Considerando que o tempo de processo dos produtos é fixado para programação das máquinas, optou-se por utilizar distribuição de tempo determinístico com disparo atomizado em detrimento de outras propostas como distribuição de tempo exponencial (tempo estocástico) sugeridas para alguns sistemas de informática. A análise dos dados do modelo foi feito através de processos estatísticos, considerando a média e o desvio padrão das amostras. Os parâmetros de entrada foram ajustados e seus efeitos comparados para identificar o melhor “*setup*” do sistema em função do “*throughput*”. Os parâmetros de entrada utilizados para realizar os ajustes são: o número de máquinas trabalhando em paralelo, tempo de operação de máquinas, tamanho dos *buffers* e tipos de *buffers* intermediários.

Análise do throughput em função do número de máquinas considerando o número e o tipo de buffer fixo:

Tempo de execução para 20 peças			
Tipo de buffer: escolha estatística baseado nos buffer			
Tamanho dos buffers: 1			
No. da amostra: 20			
No. Robot=1	No. Robot=1	No. Robot=2	No. Robot=2
No. CNC=1	No. CNC=2	No. CNC=1	No. CNC=2
Tempo média	Tempo média	Tempo média	Tempo média
492,3333	454	340,8333	208
DesvPad	DesvPad	DesvPad	DesvPad
29,221	25,2349	11,10705	4,732864

Tab.8.2 – Análise do *throughput* com variação de número de máquina e buffer fixo e limitado.

A utilização de dois robôs e dois CNCs reduz o tempo de operação em 58% do valor total como mostra a coluna quatro, o que justificaria o investimento. Já a utilização de dois CNCs aparece como o pior investimento em termos de redução de tempo pois só possibilita uma redução em cerca de 8% o tempo de operação. A utilização de dois robôs e um CNC aparece como um resultado intermediário na coluna três. Se considerarmos limitação de investimento a curto prazo a combinação de dois robôs e um CNC poderia ser considerada a melhor.

Análise do *throughput* em função do limite dos buffers intermediários, considerando número fixo de máquinas: robôs = 2, CNC = 2 unidades e o tipo de buffer fixo:

Tempo de execução para 20 peças			
Tipo de buffer: estatístico por peças			
Tamanho do buffer: limitado			
No. da amostra: 20			
No. Robot=2; No. CNC=2			
2	3	4	5
Tempo médio	Tempo médio	Tempo médio	Tempo médio
216,5	197,8333	189,5	173,5
DesvPad	DesvPad	DesvPad	DesvPad
22,24185	19,75264	16,245	34,04556

Tab.8.3 – Análise do *throughput* com buffer variável

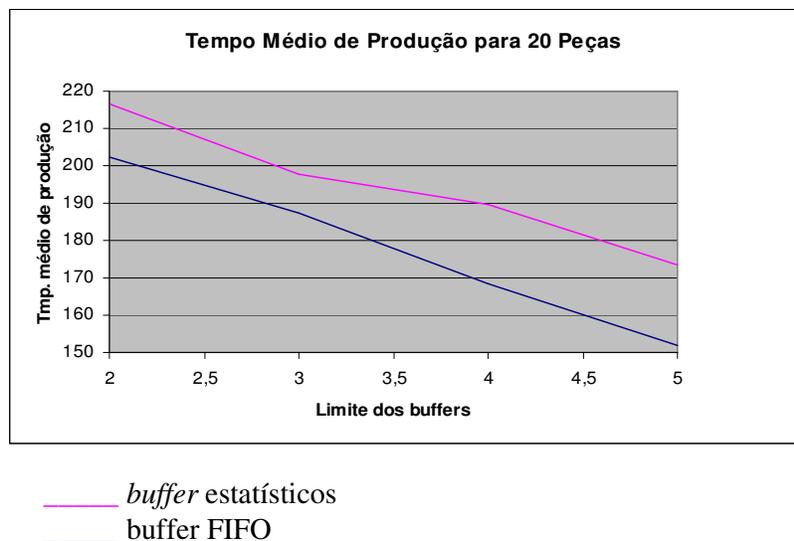
O aumento do tamanho dos *buffers* intermediários melhorou o *throughput* do sistema. Comparando o desempenho do sistema com *buffer* de 2 unidades com o *buffer* com 5 unidade, obtemos uma redução no tempo de produção para 80% do valor total.

Análise do *throughput* utilizadno buffers do tipo FIFO, considerando número fixo de máquinas e de *buffers*: 2 CNC e 2 robôs.

Tempo de execução para 20 peças			
Tipo de buffer: FIFO			
Tamanho do buffer: limitado			
No. de amostras:20			
No. Robot=2; No. CNC=2			
2	3	4	5
Tempo médio	Tempo médio	Tempo médio	Tempo médio
202,3333	187,5	168,5	151,8333
DesvPad	DesvPad	DesvPad	DesvPad
13,64795	20,19653	18,72699	20,23281

Tab.8.4 – Análise do *throughput* com buffer tipo FIFO

A utilização de *buffers* intermediários do tipo FIFO reduz em cerca de 18% o *throughput* para todos os limites de *bufferes* analisados. O Gráfico 8.1 mostra uma comparação de desempenho entre os sistemas com escolha estocástica e utilizando *buffers* do tipo FIFO. Podemos verificar claramente a melhoria com a utilização do buffer FIFO nesse sistema.



Graf. 8.4 – Tempo de produção x limites dos buffers

9. CONCLUSÃO

9.1. CONSIDERAÇÕES FINAIS

Esta dissertação apresentou uma metodologia para modelagem de sistemas de manufatura flexíveis baseado em rede de Petri colorida hierárquica e temporizada com o propósito de possibilitar um desenvolvimento de projeto mais sistematizado por partes de engenheiros que não tenham um conhecimento tão aprofundado sobre a teoria de redes de Petri. Dentre as vantagens obtidas, a utilização de CPN nos possibilitou lidar com diferentes produtos simultaneamente sem repetição de estruturas, reduzir os modelos e executar transformações de produtos que são fundamentais para representação dinâmica de máquinas em um FMS. A estrutura hierárquica possibilitou, ainda, uma melhor organização e visualização do modelo através da organização da representação em níveis adequados de abstração. A utilização de conceitos de temporização e processos estatísticos associados às transições e aos arcos mostrados na seção 8.2 possibilitaram a modelagem dos tempos de operação e a análise do desempenho do sistema.

Apoiado nessas três extensões de rede Petri Net, a metodologia propôs a construção do modelo por refinamento que consiste da criação da rede geral não colorida baseada em um diagrama denominado diagrama de operação; transformação para rede colorida através da fusão baseada em marcas; refinamento *top-down* através do instanciamento e parametrização de sub-redes; realização de conexões válidas baseadas nos teoremas de *Valid Extension of Petri Net* e *Sequential and Mutual exclusion* e finalmente, análise de desempenho por simulação com base em ajustes de parâmetros de tamanho e tipo de buffer, número de máquinas e tempo de operação.

Os resultados da simulação e da análise de espaço de estado sugerem que o método de modelamento proposto é adequado para prevenção de *deadlock* e “explosão” dos estoques intermediários. Além disso, também permite a visualização e adequação do desempenho a partir do ajuste dos tempos de operação de máquinas e tamanho de *buffers*, quando a implantação de nova linha de produção são moderadamente diferentes em suas especificações de trabalho. Adicionalmente, tais resultados, possibilitam uma efetiva aprendizagem por parte do projetista fazendo com que o processo modelamento seja mais fácil e menos propenso a erro.

9.2. CONTRIBUIÇÃO

Esse trabalho enfocou o problema da construção e análise de modelos válidos para sistemas de manufatura complexos com visualização gráfica mais simples, propondo alternativas para lidar com as etapas de desenvolvimento, conversão entre redes não coloridas para coloridas, composição de módulos com conservação de propriedades a partir das contribuições a seguir:

- Adaptação e utilização de métodos de projetos bem estabelecidos no estado da arte como dobragem baseada e marcas, teoria de Petri nets válidas, SME e PME para aplicação em Redes de Petri Colorida e temporizadas no contexto do desenvolvimento de processos produtivos flexíveis.
- Desenvolvimento de uma seqüência de passos sistematizados para modelagem e análise de sistemas flexíveis de manufatura baseada em redes de Petri coloridas, hierárquicas e temporizadas e fundamentada nos mecanismos de composição do estado da arte.
- Elaboração de um conjunto padrões de sub-modelos válidos em CPN aplicado a equipamentos de um Sistema Flexíveis de Manufatura que possam ser facilmente instanciados e parametrizados na construção *top-down* do modelo completo.
- Desenvolver estudos de casos de uma célula SMS real composta por robô, centro de usinagem, esteira transportadora, sensores, instrumentos de aquisição de medida e aferição de qualidade.

9.3. TRABALHOS FUTUROS

Como sugestão para futuros trabalhos pode-se citar a criação de um simulador com extensão de CPN dedicada a Sistemas de Manufatura Flexíveis, implementação de técnicas para diagnósticos de falha e procedimentos para transformação de CPN em linguagem *Ladder* equivalente para CLPs. Além disso, pode ser introduzidas diferentes metodologias para otimização automática do funcionamento do sistema ou

diagnóstico de falha baseadas em técnicas de inteligência artificial (AI), tais como Sistemas especialistas, lógica *fuzzy* ou redes neurais.

A criação de um simulador com extensão de CPN dedicada a Sistemas de Manufatura Flexíveis que pudessem facilitar o modelamento e a representação gráfica consiste na utilização de elementos gráficos configuráveis na forma de transição ou lugares especiais, derivados de estruturas teoricamente sedimentadas e válidas das redes de Petri de propósito geral, como é o caso dos lugares com limitação e arcos inibidores com seleção de cores.

No processo de conversão para linguagem *Ladder* o controle de processo do sistema real é representado pela evolução das marcas juntamente com a própria evolução do processo através de uma descrição lógica com CPN é desenvolvida de acordo com as especificações de controle. A parte do controle é separada da parte do processo e transferida para uma função booleana que é facilmente convertida para linguagem *Ladder*.

A utilização de Inteligência Artificial possibilitará a escolha da melhor transição a ser disparada quando da ocorrência de duas transições estiver habilitadas ao mesmo tempo. Em técnicas com capacidade de aprendizagem é o caso das redes neurais ou algoritmos genéticos o objetivo seria que o próprio sistema pudesse ajustar seus parâmetros de entrada de modo que o resultado da produção fosse otimizado.

REFERENCES

- BARROS, João P. Mestre P. Ramos. *Modularidade em Redes de Petri*. Lisboa: UNL, 2006. Dissertação de Doutorado, Faculdade de Ciências e Tecnologia, Universidade de Nova Lisboa, Fev. 2006.
- BARRETO, Raimundo da Silva. *Time Petri Net-based Methodology for Embedded Hard Real-Time Software Syntheses*. Recife: UFP, 2005. PhD Thesis in Computation Science, Universidade Federal de Pernambuco, Apr. 2005.
- CASSANDRAS, Christos, G. and RAMADGE, Peter, J. *Toward a Control theory for Discrete Event System*. IEEE Control Systems Magazine, p. 66-68, junho 1990.
- CHRISTENSEN, Soren e Hansen, N. D. *Coloured Petri Nets Extended with Channels for Synchronous Communication*. In: Lecture Notes in Computer Science, Vol. 815; Application and Theory of Petri Nets 1994, Proceedings 15th International Conference, Zaragoza, Spain, pages 1992. p. 159-178.
- CHRISTENSEN, Soren e Petrucci, Laure. *Towards a Modular Analysis of Coloured Petri Nets*. In: 13th International Conference on Application and Theory of Petri Nets 1992, Sheffield, UK: v. 616. 1992. p. 113 -133.
- Dos SANTOS, FILHO, Diolino, Jose, *Proposta do Mark Graph Estendido para a Modelagem e Controle de Sistemas de Manufatura*. Dissertação de Mestrado Apresentada à Escola Politécnica da USP-São Paulo, (1993).
- FABER , P. Merlin and D. J. *Recoverability of communication protocols: Implications of a theoretical study*. IEEE Transactions on Communications, 24(9):1036 – 1043, Sept. 1976.
- HASEGAWA, Kensuke. *Modeling, Control and Deadlock Avoidance of Flexible Manufacturing Systems*, In: Conferências Plenárias of XI CBA, São Paulo, SBA. 1996.
- HACK, M. *Analysis of Production Schemata by Petri Nets*. Master of Science Dissertation Thesis, MIT, february 1972.
- PALOMINO, Reinaldo Chile. *Uma abordagem para a Modelagem, Análise e Controle de Sistemas de Produção Utilizando Redes de Petri*. Santa Catarina: UFSC, 1995. Dissertação de Mestrado em Engenharia da Produção, Universidade Federal de Santa Catarina, Ago. 1995.
- JENSEN, Kurt. *Coloured petri nets and the invariant-method: Theoretical Computer Science*, v. 14, 317-336, 1981.
- JENSEN, Kurt. *Condensed State Spaces for Symmetrical Coloured Petri Net*:. Formal Methods in System Design, v.9, p. 7-40, 1996. Internal Note: Submitted by: kjensen@daimi.aau.dk.

JENSEN, Kurt. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Volume 3, Practical Use*. Monographs in Theoretical Computer Science. Springer-Verlag, Berlin, Germany, 1997.

JENSEN, Kurt. *An Introduction to the Practical Use of Coloured Petri Nets*. Computer Science Department, University of Aarhus. Aarhus C, Denmark, 1998.

JENSEN, Kurt. *Introduction to the Theoretical Aspects of Coloured Petri Nets*. Article in Computer Science. Computer Science Department, Aarhus University. Aarhus C, Denmark, August 28-30, 1999.

KRISTENSEN, Lars M., Christensen, Soren and Jensen, Kurt. *The Practitioner's guide to colored Petri Nets*. Article in Computer, University of Aarhus Denmark, 1998.

KASTURIA, E., DICESARE, F., DESROCHERS, A. *Real-time Control of Multi-level of Manufacturing Systems Using Colored Petri Nets*. *IEEE Int. Conf. Robotics and Automation*, Raleigh, NC,: 1114-1119, 1988.

MARTINEZ, J. P. Muro and SILVA, M. *Modelling, validation and software implementation of production systems using high level Petri nets*. Proceedings, International Conference on Robotics and Automation, Raleigh, North Carolina, pp. 1180–1185, 1987.

MURATA, Tadao. 1989. Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, 77 (4): 541-580, 1989.

MORAES, C. Cícero; Castrucci, L. Plínio. *Engenharia de Automação Industrial*. LTC – Livros Técnicos e Científicos Editora, Rio de Janeiro, 2001.

MURATA, T. *Petri nets: Properties, analysis and applications*. Proc. IEEE, 77(4):541–580, April 1989.

PENHA, Dulcinéia Oliveira et al. *Modelagem de Sistemas Computacionais usando Redes de Petri: Aplicação em projeto, análise e avaliação*. Tutoria para o Programa de Pós-graduação em Engenharia Elétrica. Laboratório de Sistemas Digitais e Computacionais. Pontifícia Universidade Católica de Minas Gerais, 2004.

QIAO, Guixiu et al. *Simulation System Modeling for customization Manufacturing*. Proceedings of the 2002 Winter Simulation Conference, pp. 2031-2036, 2002.

RAMAMOORTHY, C. V. and HO, Gary S., *Performance Evaluation of synchronous Concurrent System Using Petri Nets*. IEEE Transaction on software Engineering, vol. SE-6, n. 5, pp. 440-449, (1980).

REISIG, W. e Rozenberg, G., editores. *Lectures on Petri Nets II: Applications*. Number 1492 in Lecture Notes in Computer Science; Advances in Petri Nets. Springer-Verlag, Germany. 1998.

REMBOLD, U., NNAJI, B., Storr, A. “*Computer Integrated Manufacturing and Engineering*”, Addison-Wesley, 1994.

SAITOU K., MALPATHAK, S. e Qvam. H. *Robust Design of Flexible Manufacturing Systems using, Colored Petri Net and Genetic Algorithm*. Journal of Intelligent Manufacturing. Germany, 2002: 339-351.

SANTOS FILHO; DIOLINO, José dos, *Controle de Sistemas Antropocêntricos de Produção baseado em Redes de Petri Interpretadas*. Tese de doutorado, Escola Politécnica da USP, São Paulo, 1998.

TSAI, J.; Yang, S. and Chang, Y. H. *Timing constraint petri nets and their application to schedulability analysis of real-time system specifications*. IEEE Trans. Software Engineering, 21(1):32-49, January 1995.

VISWANADHAM N.; NARAHARI, Y. Johnson L. *Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models*. IEEE Trans. on Robotics and Automation 16(6): 713-722, 1990.

VILLAROEL, J. L.; MARTINEZ, J. and SILVA, M. *Graman: a graphic system for manufacturing system design*, S. Tzafestas (ed.), IMACS Symposium on System Modelling and Simulation, Elsevier Science, pp. 311-316, 1989.

ZURAWSKI, Richard e Zhou, MengChu. *Petri Nets and Industrial Applications - a Tutorial*. IEEE Transactions on Industrial Electronics, 41(6):567-583, 1994.

ZHOU, M. C. *Adaptative design of Petri net controllers for error recovery in automated manufacturing systems*. IEEE Trans. On Systems, Man and Cybernetics. 1989: 963-973.

ZHOU, M. C, DiCesare F. *Parallel and seqüencial mutual exclusinos for petri net mdeling of Manufacturing systems with shared resources*. IEEE Transactions on Robotics and Automation, vol. 7, n. 4, pp. 5515-527, ougust 1991.

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.

ZIMMERMANN, A. G. Hommel, *Modelling and Evaluation of Manufacturing Systems Using Dedicated Petri Nets*. The International Journal of Advanced Manufacturing Technology. Germany, 15:132-138, 1999.

ZIMMERMANN, A., S. Bode, and G. Hommel, *Performance and Dependability Evaluation of Manufacturing Systems Using Petri Nets*. In: 1st Workshop on Manufacturing Systems and Petri Nets, 17th Int. Conf. on Application and Theory of Petri Nets, Osaka, Japan, 235-250, 1996.

ZIMMERMANN, A. et al. *A Modelling and Analysis Method for Manufacturing Systems Based on Petri Nets*. The International Journal of Advanced Manufacturing Technology. Germany, 1999.

ZIMMERMANN, A. et al. *Performance and Dependability Evaluation of Manufacturing Systems Using Petri Nets*. Technisch Universitat, Berlin. 1988.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)