



Pós-Graduação em Ciência da Computação

**“Um Ambiente de Suporte à Modelagem Hierárquica por  
Redes de Petri para Sistemas de Produção”**

Por

**Wellington João da Silva**

**Dissertação de Mestrado**



Universidade Federal de Pernambuco  
posgraduacao@cin.ufpe.br  
[www.cin.ufpe.br/~posgraduacao](http://www.cin.ufpe.br/~posgraduacao)

RECIFE, FEVEREIRO/2005





UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE INFORMÁTICA

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

WELLINGTON JOÃO DA SILVA

“Um Ambiente de Suporte à Modelagem  
Hierárquica por Redes de Petri para Sistemas de  
Produção”

ESTE TRABALHO FOI APRESENTADO À PÓS-GRADUAÇÃO  
EM CIÊNCIA DA COMPUTAÇÃO DO CENTRO DE IN-  
FORMÁTICA DA UNIVERSIDADE FEDERAL DE PERNAM-  
BUCO COMO REQUISITO PARCIAL PARA OBTENÇÃO DO  
GRAU DE MESTRE EM CIÊNCIA DA COMPUTAÇÃO.

ORIENTADOR: Prof. Dr. Paulo Romero Martins Maciel

RECIFE, FEVEREIRO/2005

Silva, Wellington João da

Um ambiente de suporte à modelagem hierárquica por Redes de Petri para sistemas de produção / Wellington João da Silva. – Recife : O Autor, 2005.

xv , 161 folhas : il., fig., tab.

Dissertação (mestrado) – Universidade Federal de Pernambuco. Cin. Ciência da Computação, 2005.

Inclui bibliografia, glossário e apêndices.

1. Ciência da computação – Modelagem de sistemas. 2. Redes de Petri – Formalização de modelo – Formalização matemática e representação gráfica. 3. Sistemas de produção – Planejamento estratégico, controle e funções. 4. Modelagem de sistemas de produção – Modelos para produto, recursos e roteiro de produção – Ferramenta de modelagem – PMSTool – I. Título.

004.414.23

CDU (2.ed.)

UFPE

003.011

CDD (22.ed.)

BC2005-186

## Agradecimentos

Gostaria de registrar meus agradecimentos a todos que direta ou indiretamente contribuíram para a realização deste trabalho.

Ao Professor Paulo Maciel, Doutor, pela orientação, paciência, conselhos e amizade, pela constante disposição em tirar minhas dúvidas, o que foi essencial para a realização deste trabalho.

Aos professores Nelson Rosa, Doutor, e Ricardo Massa, PhD, pela disponibilidade em participarem da banca examinadora, contribuindo com seus comentários para melhorar a qualidade deste trabalho.

Ao amigo Mauro Silva, companheiro de mestrado, com quem trabalhei em parceria durante todo o período do mestrado, pelo grande auxílio prestado na definição do estudo de caso.

À Universidade Federal de Pernambuco e ao CNPq (Conselho Nacional de Pesquisa e Desenvolvimento) que proporcionaram os meios para a realização deste trabalho.

A Deus, pelo conforto e confiança transmitidos, que foram essenciais para superar as dificuldades encontradas.

Finalmente, gostaria de agradecer à minha família pelo apoio prestado, especialmente à minha mãe e à minha avó, que se constituem no motivo de dedicação a tantos anos de estudo.

## Resumo

A globalização da economia ocorrida nas duas últimas décadas aumentou a competitividade entre as empresas e a busca pelo aumento de produtividade através da inovação tecnológica. Novas formas de comunicação e produção foram propostas, o que levou as empresas a adotar novos sistemas de produção. Todavia, a operação destes sistemas representa um grande desafio para profissionais em engenharia, ciência da computação, matemática e gerenciamento, sendo a modelagem, o controle e a análise de desempenho destes sistemas alguns dos aspectos deste desafio.

As Redes de Petri (RdPs) são um modelo formal adequado para a representação de sistemas de produção por conseguir capturar aspectos inerentes a tais sistemas como concorrência, paralelismo e sincronização.

Existem algumas ferramentas que usam RdPs para a modelagem de sistemas de produção, contudo estas ferramentas ou são de propósito geral, sendo adaptadas para a modelagem de sistemas produtivos, ou abordam apenas um dos aspectos dos sistemas de produção, qual seja, a manufatura.

Neste trabalho foi desenvolvido um modelo em RdPs para a modelagem e análise de sistemas de produção. Esta modelagem envolve a estrutura do produto, os recursos produtivos e o roteiro de produção.

Uma ferramenta baseada nesta tecnologia foi desenvolvida, a *Production Management System Tool (PMSTool)*, a qual provê um ambiente de modelagem de alto nível, que possibilita a modelagem de sistemas sem a real necessidade de conhecimento em RdPs, o que se constitui em outro diferencial em relação a ferramentas similares.

A PMSTool utiliza ícones para representar os elementos do sistema produtivo a ser modelado (máquinas, *buffers*, etc.), além de suas características principais (recursos disponíveis, tamanho de *buffer*, tempo de falha, tempo de recuperação, etc). Estes ícones são mapeados automaticamente em RdPs.

**Palavras-Chave:** Sistemas de Produção, Redes de Petri, Modelagem, Ferramenta de Modelagem.

## Abstract

The globalization of the economy in the last two decades has increased the competitiveness amongst companies and renewed interest in the issues of increasing productivity through technologic innovations. New ways of production and communication were proposed making companies to adopt new production systems. However, the operation of these systems represents a big challenge for professionals in engineering, computer science, mathematics and management. Modelling, control and performance analyses of these systems is one aspect of this challenge.

Petri nets is a formal model suitable for the representation of production systems because it can capture various aspects of these system such as concurrency parallelism and synchronization.

There are Petri net based tools for modelling production systems. However, these tools are either of general purpose, being adapted to model production systems, or model only one of its aspects, that is, the manufacturing.

In this work Petri nets are used for modelling and analysis of production systems. This model covers the product structure, productions resources and workplan.

A modelling tool based on this technology was developed. The PMSTool is a tool that provides an environment for modelling production system without the need of Petri net knowledge, hence being a differential to other similar tools.

PMSTool uses icons to represent the production system elements (e.g., machines, buffers, etc.) and its main characteristics (available resources, buffer size, time to fail, time to repair, etc.). These icons are mapped automatically onto Petri nets.

**Keywords:** Production Systems, Petri Net, Modelling, Modelling Tool.



# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação . . . . .	1
1.2	Evolução dos Sistemas de Produção . . . . .	2
1.3	Redes de Petri . . . . .	2
1.4	Objetivos do Trabalho . . . . .	5
1.5	Relevância . . . . .	5
1.6	Trabalhos Relacionados . . . . .	6
1.7	Estruturação do Trabalho . . . . .	8
<b>2</b>	<b>Planejamento e Controle da Produção</b>	<b>11</b>
2.1	Sistema . . . . .	11
2.2	Sistemas de Produção . . . . .	12
2.3	Classificação dos Sistemas de Produção . . . . .	14
2.4	Funções dos Sistemas de Produção . . . . .	15
2.5	Planejamento e Controle da Produção . . . . .	17
2.6	Planejamento Estratégico da Produção . . . . .	19
2.7	O Papel dos SAP . . . . .	19
2.7.1	Just in Time (JIT) . . . . .	20
2.7.2	Material Requirement Planning (MRP) . . . . .	22
2.7.3	Manufacturing Resource Planning (MRP2) . . . . .	25
2.7.4	Optimized Production Technology (OPT) . . . . .	27
2.7.5	Enterprise Resource Planning (ERP) . . . . .	29
2.8	Considerações Finais . . . . .	32
<b>3</b>	<b>Redes de Petri</b>	<b>33</b>
3.1	Introdução . . . . .	33
3.2	Elementos Básicos . . . . .	34
3.3	Estrutura de uma RdP . . . . .	36
3.4	RdP Marcada . . . . .	38
3.5	Propriedades das RdPs . . . . .	39
3.6	Métodos de Análise e Validação de uma RdP . . . . .	40

3.6.1	Árvore de Cobertura . . . . .	41
3.6.2	Matriz de Incidência e Equação de Estado . . . . .	42
3.6.3	Redução . . . . .	42
3.7	Classes de RdPs . . . . .	44
3.7.1	RdPs Ordinárias . . . . .	45
3.7.2	Extensões de RdPs . . . . .	46
3.8	Considerações Finais . . . . .	52
<b>4</b>	<b>Modelagem de Sistemas de Produção</b>	<b>55</b>
4.1	Modelo de um Sistema . . . . .	55
4.2	Modelagem Usando RdPs . . . . .	56
4.3	Elementos de Modelagem . . . . .	57
4.4	Estrutura do Produto (EP) . . . . .	59
4.4.1	Modelo Intermediário da EP . . . . .	61
4.4.2	Substituição de Atributos na EP . . . . .	64
4.4.3	Refinamento de Lugares da EP . . . . .	66
4.4.4	Modelo de RdP Final para a EP . . . . .	67
4.5	Recurso Produtivo (RP) . . . . .	69
4.5.1	Modelo Intermediário do RP . . . . .	71
4.5.2	Substituição de Atributos no RP . . . . .	72
4.5.3	Refinamento de Lugares do RP . . . . .	73
4.5.4	Modelo de RdP Final para o RP . . . . .	75
4.6	Roteiro de Produção (ROT) . . . . .	79
4.6.1	Grafo do ROT . . . . .	80
4.6.2	Modelo Intermediário do ROT . . . . .	83
4.6.3	Refinamento de Lugares do ROT . . . . .	84
4.6.4	Modelo de RdP Final para o ROT . . . . .	85
4.7	Definição de Sistema Produtivo . . . . .	87
4.8	Considerações Finais . . . . .	93
<b>5</b>	<b>Ambiente de Modelagem</b>	<b>95</b>
5.1	Introdução . . . . .	95
5.1.1	Petri Net Markup Language (PNML) . . . . .	96
5.1.2	Petri Net Kernel (PNK) . . . . .	97
5.2	PMSTool . . . . .	98
5.2.1	Características . . . . .	99
5.2.2	Visão Geral da Ferramenta . . . . .	100
5.2.3	Estrutura do Código . . . . .	102
5.2.4	Interface Gráfica . . . . .	104
5.2.5	Etapas da Modelagem Usando a PMSTool . . . . .	106
5.2.6	Integração com PNK . . . . .	110

5.3	Considerações Finais . . . . .	119
<b>6</b>	<b>Estudo de Caso</b>	<b>121</b>
6.1	Introdução . . . . .	121
6.2	LAFEPE . . . . .	122
6.3	AAS . . . . .	122
6.4	Escopo do Estudo de Caso . . . . .	125
6.5	Modelagem da EP para o AAS . . . . .	125
6.6	Modelagem dos RPs para o AAS . . . . .	132
6.7	Geração do ROT para o AAS . . . . .	134
6.8	Análise dos Modelos . . . . .	142
6.9	Considerações Finais . . . . .	143
<b>7</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>145</b>
<b>A</b>	<b>Teoria Básica</b>	<b>149</b>
<b>B</b>	<b>Glossário</b>	<b>153</b>



# Lista de Figuras

2.1	Elementos do Sistema de Produção . . . . .	13
2.2	Estrutura Operacional . . . . .	16
2.3	Visão Geral das Atividades do PCP . . . . .	18
2.4	Visão Geral do Sistema MRP . . . . .	23
3.1	Exemplo de Rede . . . . .	36
3.2	Arcos Multivalorados . . . . .	37
3.3	Seis Transformações Preservando Vivacidade, Segurança, e Limitação. . . . .	43
3.4	Espaço dos Formalismos. . . . .	45
3.5	Rede com Arco Inibidor . . . . .	47
3.6	Exemplo de Hierarquia em Redes de Petri . . . . .	51
3.7	Extensões de RdPs . . . . .	52
4.1	Etapas do Processo de Modelagem . . . . .	58
4.2	Modelo de Alto Nível da Estrutura do Produto . . . . .	60
4.3	Modelo Intermediário do Produto . . . . .	63
4.4	Atribuição de Valores aos Rótulos da Rede. . . . .	66
4.5	Exemplo de Refinamento de um Lugar . . . . .	67
4.6	Modelo de RdP Final para a Estrutura do Produto . . . . .	68
4.7	Modelo de Alto Nível de um RP . . . . .	70
4.8	Recurso Produtivo Usando RdPs . . . . .	71
4.9	Exemplo de Refinamento de um Lugar . . . . .	74
4.10	Fusão de Lugares . . . . .	75
4.11	Modelo de uma Operação Usando RdPs . . . . .	77
4.12	Modelo de Recurso com duas Operações Possíveis . . . . .	78
4.13	Modelo Final do Recurso . . . . .	79
4.14	Associando Produto com Operações . . . . .	80
4.15	Modelo Intermediário para o ROT . . . . .	84
4.16	Modelo de RdP Final para o ROT . . . . .	86
4.17	Visão Geral de Sistema de Produção . . . . .	88

4.18	Modelo de Entrada . . . . .	89
4.19	Modelo de Saída . . . . .	90
4.20	Modelo de Fechamento . . . . .	90
4.21	Modelo Final do Sistema . . . . .	92
5.1	Camadas do Petri Net Kernel . . . . .	98
5.2	Arquitetura da Ferramenta . . . . .	101
5.3	Área de Menu . . . . .	104
5.4	Área de Botões . . . . .	106
5.5	Interface Gráfica da PMSTool . . . . .	107
5.6	Atributos de um Item de Produto . . . . .	108
5.7	Atributos de uma Operação . . . . .	109
5.8	Mapeando Atividades nos Produtos . . . . .	109
5.9	Escolha do Tipo de RdP . . . . .	114
5.10	Iniciando a PMSTool . . . . .	114
5.11	Geração do Modelo Refinado . . . . .	115
5.12	Editor do PNK . . . . .	116
5.13	Definindo a RdP a ser Utilizada no Refinamento . . . . .	117
5.14	Mapeando Recurso e Operação . . . . .	117
5.15	Rede Pertencente à Biblioteca . . . . .	118
6.1	Fluxo de Produção do LAFEPE . . . . .	123
6.2	Fluxograma de Produção do AAS . . . . .	124
6.3	Modelo do Produto para AAS . . . . .	126
6.4	Modelo Intermediário para o AAS . . . . .	128
6.5	Modelo de Produto Simples . . . . .	129
6.6	Modelo de Estoque . . . . .	129
6.7	Modelo de RdP para AAS . . . . .	130
6.8	Modelo de Sistema de Produção AAS . . . . .	131
6.9	Modelo de Alto Nível para os RPs do AAS . . . . .	133
6.10	Modelo de Alto Nível para o ROT do AAS. . . . .	137
6.11	Modelo Intermediário para o ROT do AAS . . . . .	138
6.12	Modelo de RdP Final para o ROT do AAS . . . . .	139
6.13	Modelo do Sistema de Produção do AAS. . . . .	141

# Lista de Tabelas

2.1	Classificação dos Sistemas de Produção . . . . .	14
4.1	Atributos de uma Operação . . . . .	77
4.2	Mapeamento de Recursos na Estrutura do Produto . . . . .	81
5.1	Elementos da Tag Toolspecific . . . . .	112
6.1	Matéria-prima Utilizada na Produção do AAS . . . . .	127
6.2	Informações de Estoque do AAS . . . . .	127
6.3	Mapeamento de Recursos para Produção do AAS . . . . .	135



# Lista de Abreviaturas e Siglas

<b>AAS</b>	Ácido Acetil-Salicílico
<b>ALAP</b>	As Late As Possible
<b>API</b>	Application Program Interface
<b>APICS</b>	The Education Society for Resource Management
<b>ASAP</b>	As Soon As Possible
<b>BOM</b>	Bill Of Materials
<b>CAD</b>	Computer Aided Design
<b>CM</b>	Closure Model
<b>CPN</b>	Coloured Petri Net
<b>CRP</b>	Capacity Resource Planning
<b>DOM</b>	Document Object Model
<b>EP</b>	Estrutura do Produto
<b>ERP</b>	Enterprise Resource Planning
<b>FMS</b>	Flexible Manufacturing Systems
<b>GSPN</b>	Generalized Stochastic Petri Net
<b>GUI</b>	Graphical User Interface
<b>ICS</b>	Item Componente Simples
<b>ICE</b>	Item Componente Estoque
<b>IFM</b>	Input Flow Model

<b>JIT</b>	Just-in-Time
<b>LAFEPE</b>	Laboratório Farmacêutico de Pernambuco
<b>MPS</b>	Master Production Schedule
<b>MRP</b>	Material Requirement Planning
<b>MRP2</b>	Manufacturing Resource Planning
<b>MVC</b>	Model-View-Controller
<b>OFM</b>	Output Flow Model
<b>OPT</b>	Optimized Production Technology
<b>PCP</b>	Planejamento e Controle da Produção
<b>PMP</b>	Planejamento Mestre da Produção
<b>PMS</b>	Production Management System
<b>PMSTool</b>	Production Management System Tool
<b>PNK</b>	Petri Net Kernel
<b>PNML</b>	Petri Net Markup Language
<b>PNTD</b>	Petri Net Type Definition
<b>PP</b>	Plano de Produção
<b>PrT</b>	Predicado/Transição
<b>RdP</b>	Rede de Petri
<b>ROT</b>	Roteiro de Produção
<b>RP</b>	Recurso Produtivo
<b>RRC</b>	Recurso Restritivo Crítico
<b>SAP</b>	Sistemas de Administração da Produção
<b>SFC</b>	Shop Floor Control
<b>SPN</b>	Stochastic Petri Net
<b>TPN</b>	Temporized Petri Net

**TOC**      Theory of Constraints

**XML**      Extensible Markup Language



# Capítulo 1

## Introdução

*Neste capítulo é apresentada uma introdução aos Sistemas de Produção com uma breve descrição de sua evolução no tempo. Em seguida é apresentada uma introdução às Redes de Petri. Na seqüência, são apresentados os objetivos deste trabalho e a sua relevância. Em seguida, são apresentados alguns trabalhos relacionados. Finalmente, é apresentada a organização da dissertação.*

### 1.1 Motivação

A globalização da economia ocorrida nos últimos tempos levou a profundas mudanças no processo produtivo, fazendo surgir novas formas de fabricação e comunicação. Os sistemas de produção modernos requerem grandes investimentos e são muito complexos. Eles são usualmente compostos de produtos, máquinas, sistemas de transporte, etc. O adequado projeto de tais sistemas tem um grande impacto no desempenho, e em última análise no seu sucesso econômico.

Devido à sua complexidade, é frequentemente difícil prever o comportamento de tais sistemas com relativa precisão sem o auxílio de técnicas de modelagem e avaliação quantitativa. Este é o caso, por exemplo, quando recursos são sujeitos a falhas, portanto diminuindo a produção [ZH98].

Redes de Petri (RdPs) têm sido utilizadas com sucesso na modelagem de sistemas de produção devido à sua capacidade de descrever e analisar características inerentes a tais sistemas, como sincronização, compartilhamento de recursos e atividades concorrentes [ZFH01].

Muito trabalho tem sido realizado na área de modelagem de sistemas de produção usando RdPs, porém com ênfase na manufatura. Este trabalho concentra-se na área do Planejamento e Controle da Produção.

## 1.2 Evolução dos Sistemas de Produção

A manufatura foi praticada e aperfeiçoada em todo o mundo, acompanhando a evolução da civilização. Várias invenções como o moinho de vento (século X), o moinho de água (século XIII), a fornalha (século X), a máquina a vapor (século XVIII), entre outras, contribuíram para o estado corrente da produção.

Em meados de 1800, uma fábrica era geralmente composta de um sistema de máquinas produtivas. Um grande marco para a integração da manufatura ocorreu em 1913 quando Henry Ford desenvolveu uma linha de montagem na qual um motor era progressivamente montado em 84 estágios, o que diminuiu o trabalho e aumentou a produtividade. A linha de montagem de Henry Ford é baseada nos princípios de planejamento, escalonamento e controle, e também no movimento contínuo do produto através da fábrica.

A transição da linha de montagem para os sistemas de produção modernos deu-se através do desenvolvimento de novas tecnologias tais como máquinas de controle numérico (máquinas NC), robôs, sistemas de manipulação de material e sistemas controlados por computador.

Computadores *mainframe* passaram a ser utilizados para planejamento, escalonamento e controle de produção em lote nos anos 1950s e 1960s. Várias companhias integraram os sistemas de organização e manufatura. Um grande avanço ocorreu quando computadores passaram a controlar equipamentos de chão de fábrica tais como máquinas de controle numérico e elementos de manipulação de material. Os desenvolvimentos citados anteriormente culminaram na evolução dos Sistemas Flexíveis de Manufatura (FMS), que teve início nos anos 1960s.

Vários desenvolvimentos ocorridos nos anos 1980s levaram a grandes avanços na área de manufatura. O aparecimento de novos materiais, novas tecnologias e novos produtos fizeram surgir vários métodos de planejamento e controle da produção concorrentemente com as tecnologias de automação flexíveis. Dentre estas tecnologias destacam-se o *Material Requirement Planning (MRP)*, o *Manufacturing Resource Planning (MRP2)*, o *Just-in-Time (JIT)* e o *Optimized Production Technology (OPT)*. Estes desenvolvimentos têm estimulado grandes mudanças no setor de manufatura.

## 1.3 Redes de Petri

A teoria inicial das RdPs surgiu da tese *Kommunikation mit automaten* (Comunicação com autômatos), defendida por Carl Adam Petri em 1962,

na Alemanha.

RdPs são uma ferramenta gráfica e matemática de modelagem que pode ser aplicada para representar diversos tipos de sistemas, fornecendo um bom nível de abstração em relação a outros modelos gráficos [Pet81]. Trata-se de um modelo do tipo estado-evento, onde cada evento possui pré-condições que possibilitam a sua ocorrência e pós-condições decorrentes desta, as quais são por sua vez pré-condições de outros eventos posteriores.

Uma RdP também pode ser vista como um grafo orientado que permite modelar propriedades estáticas de um sistema de eventos discretos, constituído de dois tipos de nós: *transições* (correspondem a eventos que caracterizam mudanças de estados do sistema), e *lugares* (correspondem a condições que devem ser satisfeitas) interligados por arcos direcionados ponderados. Marcas (em manufatura, normalmente representam recursos disponíveis) podem estar associadas aos lugares da rede [Pet81].

Existem várias classificações para RdPs na literatura. A seguinte classificação é encontrada em [MLC96]:

- RdPs de baixo nível.
  - RdPs Binárias ou Condição-Evento.
  - RdPs Lugar/Transição.
- RdPs de alto nível.
  - RdPs Predicado-Transição.
  - RdPs Coloridas.
  - RdPs Hierárquicas.

A classe *Binária (ou Condição/Evento)* representa as redes mais simples entre todas as classes. Nessa classe, os lugares podem conter no máximo uma marca e todos os arcos têm peso unitário. Nas redes *Lugar/Transição*, os lugares podem acumular marcas e os arcos podem ser valorados. As Redes Predicado/Transição (Redes PrT) [GL81] e seu melhoramento, as RdP Coloridas (CPN) [Jen90, Jen93] foram introduzidas com o objetivo de construir modelos mais compactos que os obtidos com as RdPs de baixo nível. Nas redes PrT, um predicado chamado *guarda* pode ser adicionado às transições do modelo. As CPN associam cores às marcas da rede.

As principais diferenças entre as RdPs de alto nível e as de baixo nível estão no fato de que (1) nas RdPs de alto nível, as marcas têm um tipo e podem representar objetos estruturados, e (2) expressões formais (também

chamadas *inscrições*) podem ser adicionadas aos arcos do grafo [DHP<sup>+</sup>93, p. 194]. Nas RdPs de baixo nível, as marcas são tipos de dados não estruturados (ex: um inteiro não negativo);

Todas as classes de rede citadas anteriormente têm o mesmo poder computacional, todavia algumas destas classes possibilitam um maior ou menor nível de abstração dos modelos. Algumas *extensões* às RdPs aumentam o seu poder computacional, fornecendo meios para a especificação de outras propriedades dos sistemas computacionais tais como prioridade e tempo (ex: RdPs determinísticas, estocásticas e com prioridade). Infelizmente, essas extensões reduzem o poder de decisão das RdPs [MLC96, GV03]. por exemplo, em muitos casos o novo modelo tem o mesmo poder de expressão das máquinas de Turing e então o problema da alcançabilidade – Uma marcação é atingível a partir de uma outra? – se torna não-decisível. Em decorrência, são realizados alguns estudos com sub-classes de RdP que possibilitem aumentar o poder de decisão sem diminuir em demasia seu poder de modelagem.

RdPs têm sido amplamente utilizadas para a modelagem, análise, simulação e controle de sistemas de manufatura [DAJ95, Mur89], pois:

- A representação gráfica ajuda na visualização de sistemas complexos.
- Conflitos e *buffers* podem ser modelados de forma fácil e eficiente.
- Travamentos do sistema (*deadlocks*) podem ser detectados.
- Várias extensões de RdPs permitem análises qualitativas e quantitativas da utilização de recursos, falhas, etc.
- Os modelos de RdPs representam uma ferramenta de modelagem hierárquica com bases matemáticas bem desenvolvidas.
- Permite a modelagem através de vários níveis de abstração.
- Possibilita a modelagem de sistemas paralelos e concorrentes.
- Sincronização de eventos.
- Verificação metódica de propriedades do sistema.

As RdPs serão apresentadas com mais detalhes no Capítulo 3.

## 1.4 Objetivos do Trabalho

O objetivo desta dissertação é apresentar um processo e ferramenta para a modelagem de sistemas de produção. A modelagem tem as RdPs como formalismo base, devido à sua capacidade de modelar e analisar sistemas dinâmicos, além da capacidade de avaliar o desempenho de tais sistemas. São definidos modelos para a estrutura do produto, os recursos produtivos e o roteiro de produção.

O processo de modelagem inicia com os modelos de alto nível que representam a estrutura do produto ou um recurso produtivo. Estes modelos são denominados modelos de alto nível por abstrair a utilização de RdPs. A partir do modelo de alto nível é gerado um modelo de RdP intermediário, o qual é posteriormente refinado gerando-se um modelo de RdP final. Através dos modelos definidos para a estrutura do produto e para os recursos produtivos, é gerado de forma automática um modelo para o roteiro de produção.

Uma ferramenta baseada nesta tecnologia foi desenvolvida, a PMSTool, a qual provê um ambiente de modelagem que enfatiza não apenas a modelagem da manufatura, mas também o planejamento e controle da produção.

O conhecimento prévio de RdPs não é um requisito para utilização da ferramenta. Os modeladores sem conhecimento prévio de RdPs podem utilizar os modelos de alto nível definidos para a estrutura do produto, recurso ou roteiro de produção. Os usuários com conhecimentos em RdPs, por sua vez, podem utilizar os modelos de RdP intermediário ou final, dependendo do problema que se está querendo atacar.

## 1.5 Relevância

Existem algumas ferramentas baseadas em RdPs que são utilizadas para a modelagem de sistemas de produção. Contudo, a maioria ou é de propósito geral, sendo adaptada para a modelagem de sistemas de produção, o que torna sua utilização relativamente complexa, ou dá ênfase apenas à modelagem do chão-de-fábrica.

A PMSTool é uma ferramenta para a modelagem de sistemas de produção que permite o planejamento e o controle da produção. Neste sentido, a modelagem inicia com a estrutura do produto em vez do chão-de-fábrica, o que vem a ser um diferencial em relação à ferramentas existentes.

A simplicidade na modelagem é outro aspecto importante. A modelagem é baseada em ícones que abstraem a utilização de RdPs. Desta forma, o usuário final não precisa ser um conhecedor de RdPs.

Em síntese, o que se procura é uma ferramenta de modelagem de sistemas de produção, de fácil utilização, e que auxilie a tomada de decisão.

## 1.6 Trabalhos Relacionados

Existem algumas ferramentas baseadas em RdPs que podem ser utilizadas para a modelagem e análise de sistemas de manufatura [DAJ95, MLC96], contudo, são na maioria das vezes ferramentas de RdP de propósito geral, sendo adaptadas para a modelagem de sistemas de manufatura, o que torna difícil a modelagem e análise de tais sistemas. Outro aspecto importante é que mesmo as ferramentas que apresentam módulos específicos para sistemas de manufatura ou são muito difíceis de utilizar, ou não oferecem fácil suporte a integração com outras ferramentas baseadas em RdPs. Outro problema comum, relacionado com a implementação de tais ferramentas, é a dependência de plataforma, o que limita a sua utilização. Finalmente, um grande problema com as ferramentas existentes é o fato de as mesmas adotarem, na maioria das vezes, um formato de arquivo específico para armazenar seus modelos de RdP. Isto dificulta ou mesmo inviabiliza a utilização de mais de uma ferramenta de RdP, em conjunto, para a modelagem e análise de sistemas.

*TimeNET (Timed Net Evaluation Tool)* [Tec99] é uma ferramenta de avaliação de desempenho e análise que utiliza RdPs estocásticas, não-markovianas, sem cores, para a modelagem e análise de sistemas discretos [GKZ95].

Os desenvolvedores da *TimeNET* a definem como um *toolkit* (conjunto de ferramentas de *software*) gráfico e interativo para a modelagem com RdPs estocásticas. Seus algoritmos de avaliação incluem análise, aproximação e simulação. Análise significa uma avaliação de desempenho numérica direta e exata, com uma exploração completa do grafo de alcançabilidade. Algoritmos de aproximação também são técnicas numéricas diretas, mas procuram evitar o cômputo de algumas das partes de avaliação mais custosa, o que introduz alguma imprecisão nos resultados. Algoritmos de simulação não computam o grafo de alcançabilidade, em vez disto, procuram aplicar alguma heurística para obtenção dos resultados bem como alguns refinamentos [Tec99].

Ela também apresenta um componente especialmente projetado para sistemas de manufatura, que inclui:

- Modelagem com RdPs estocásticas coloridas e hierárquicas.
- Modelos independentes para estrutura e plano de trabalho.
- Análise estacionária e simulação

Este componente de manufatura em particular, necessita ser mais bem documentado. Por exemplo, na versão para o idioma inglês do *TimeNET* versão 3.0, alguns dos campos presentes nas telas do componente de manufatura ainda estavam no idioma alemão.

*TimeNET<sub>MS</sub>* (*TimeNET for Manufacturing Systems*) é uma extensão da ferramenta *TimeNET* que utiliza RdPs estocásticas coloridas, sendo voltada especificamente para a modelagem de sistemas de manufatura [ZF98].

A classe das RdPs estocásticas coloridas utilizadas pelo *TimeNet<sub>MS</sub>* foram introduzidas especialmente para a modelagem de sistemas de manufatura. Dois tipos de cores são definidos: marcas do tipo *Objeto* modelam itens dentro do sistema de manufatura e consistem de um nome e o estado atual. Marcas do tipo *Elementar* são equivalentes as marcas de RdPs não coloridas. Lugares podem conter apenas marcas de um mesmo tipo.

Estrutura e plano de trabalho são modelados de forma independente. O modelo estrutural descreve os recursos do sistema de manufatura tais como máquinas, *buffers*, etc. Modelos do roteiro de produção especificam características dependentes do plano de trabalho do sistema de manufatura. Cada roteiro corresponde a um caminho através do sistema de manufatura. Estes dois modelos são posteriormente unidos resultando em um modelo completo, que contém desde as restrições de recursos até a sincronização dos passos de produção.

A corretude do sistema modelado pode ser verificada através de simulação interativa (*token game*). Propriedades quantitativas são derivadas através de análise numérica e simulação.

*XPN-FMS* é uma ferramenta para a modelagem, análise estrutural e simulação interativa de sistemas de manufatura [CW95]. Trata-se de uma ferramenta CAD para FMSs baseada no método de síntese de RdPs denominado *técnica knitting* [Cha99]. A ferramenta integra as funções de desenho, análise, redução [MK80], síntese e simulação de FMSs.

Usando a interface gráfica *X Window* e animação, *XPN-FMS* torna a modelagem e análise de um FMS visualizável e fácil de entender e manipular. Ela é útil para a especificação de FMSs, validação e exploração de diferentes alternativas de projeto, monitoramento de *status* e controle. Todavia, está limitada às RdPs Escolha-Livre (*Free-Choice*) que não têm marcas diferenciadas, o que torna muito difícil sua aplicação na modelagem de sistemas

de manufatura reais.

*GRAMAN* [VMS89] permite a modelagem em separado da estrutura do sistema de manufatura e dos roteiros de produção. RdPs coloridas são utilizadas para a modelagem do plano de trabalho, enquanto a estrutura do sistema é modelada com blocos pré-definidos. Um modelo interno é gerado a partir destas duas descrições. A conexão dos sub-modelos que são criados a partir dos blocos pré-definidos é realizada pela fusão das transições que representam atividades sincronizadas.

Esta ferramenta descreve uma planta de manufatura através de um conjunto de *blocos de construção* (do inglês, *building blocks*) e suas inter-relações. Um modelo sub-CPN é atribuído a cada bloco, que é parametrizado através das relações estruturais que afetam o bloco.

*GRAMAN* não foi projetado para atingir um nível elevado de detalhes, porque ele leva em conta apenas informações que são necessárias para modelar funcionalidades gerais (robôs, *buffers*, etc.) mas não as tarefas específicas de processamento em uma dada planta [BCM99]. Antes de utilizar a ferramenta, o modelador necessita aprender duas linguagens de descrição. Além disso, não é possível especificar propriedades de máquina que dependam de uma tarefa em execução [ZF98].

## 1.7 Estruturação do Trabalho

Este trabalho está organizado como segue:

No Capítulo 2 são apresentados os sistemas de produção e suas características principais.

No Capítulo 3 são apresentadas as Redes de Petri, bem como aspectos relacionados à sua modelagem e técnicas de análise.

No Capítulo 4 são apresentados os modelos desenvolvidos para o suporte à análise dos sistemas de gerenciamento da produção.

O Capítulo 5 apresenta o ambiente de modelagem desenvolvido de acordo com a metodologia proposta.

O Capítulo 6 apresenta um estudo de caso para validar o trabalho desenvolvido.

Finalmente o Capítulo 7 traz um resumo do trabalho desenvolvido, mos-

trando como os objetivos propostos foram atingidos, as dificuldades encontradas, conclusões e recomendações para trabalhos futuros.



# Capítulo 2

## Planejamento e Controle da Produção

*Este capítulo apresenta as diferentes funções exercidas pelo Planejamento e Controle da Produção (PCP). Primeiramente é feita uma abordagem sobre a função da produção. Em seguida, definem-se e classificam-se os sistemas de produção. Finalmente, são apresentadas as funções e princípios do PCP e as suas atividades: previsão de demanda, planejamento mestre da produção, administração de estoques, seqüenciamento, emissão e liberação de ordens.*

### 2.1 Sistema

Um sistema pode ser conceitualmente definido como um conjunto de partes inter-relacionadas que existem para atingir um determinado objetivo. As inter-relações entre as partes integrantes do sistema são as comunicações ou as interdependências.

Um sistema pode ser decomposto em uma coleção de subsistemas, um conjunto de elementos organizados para a realização de um propósito específico e descrito por um conjunto de modelos, provavelmente sob diferentes pontos de vista.

Uma empresa, por exemplo, pode ser considerada um sistema composto de vários departamentos (subsistemas) e fazendo parte de um sistema maior, que é a própria sociedade.

Um sistema é geralmente composto dos seguintes componentes:

- Fronteira – Representa o limite do sistema;

- Subsistema – Agrupamento de elementos que compõem o sistema;
- Entrada (*input*) – Representa os insumos ou variáveis independentes do sistema;
- Saída (*outputs*) – Representa os produtos ou variáveis dependentes do sistema;
- Processamento – Engloba as atividades desenvolvidas pelos subsistemas que interagem entre si para converter as entradas em saídas;
- Realimentação (*feedback*) – É a influência que as saídas do sistema exercem sobre as suas entradas no sentido de ajustá-las ou regulá-las ao funcionamento do sistema.

## 2.2 Sistemas de Produção

Produção<sup>1</sup> é o processo de transformação pelo qual insumos (material, energia, trabalho, equipamentos, etc.) são transformados em produtos ou serviços. Naturalmente, os produtos ou serviços resultantes têm um valor econômico maior que o dos elementos utilizados como entrada. A produção, segundo sua natureza, pode ser classificada em três tipos: Produção primária (ou extrativa), secundária (ou de transformação) e terciária (ou prestadoras de serviço) [Chi91].

Um Sistema de Produção é um conjunto de atividades e operações inter-relacionadas envolvidas na produção de bens (caso da indústria) ou serviços [Mor02].

Um sistema de produção é composto por dois subsistemas principais [GV03]:

- O subsistema físico – Composto dos recursos físicos tais como correia transportadoras, robôs, *buffers*, estações de trabalho, etc.
- O subsistema de controle – Determina como utilizar o subsistema físico para organizar e otimizar o processo produtivo.

Em um sistema de produção, alguns elementos fundamentais podem ser identificados. São eles os insumos, o processo de criação ou conversão, os produtos ou serviços, e o subsistema de controle (veja a Figura 2.1).

---

<sup>1</sup>Neste trabalho os termos *produção* e *manufatura* serão utilizados de forma indistinta.

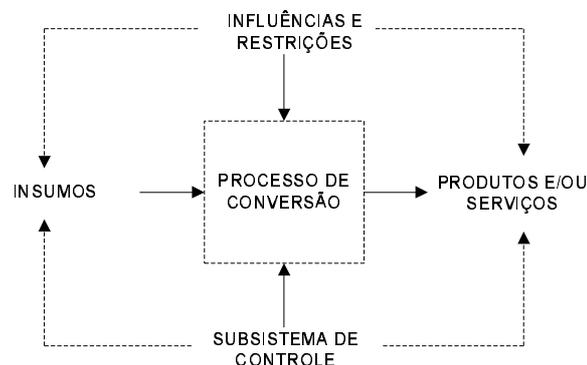


Figura 2.1: Elementos do Sistema de Produção

Os *insumos* são os elementos que entram no processo de produção de mercadorias ou serviços, tais como matéria-prima, mão-de-obra, capital, máquinas, equipamentos, instalações, conhecimento técnico, etc.

O *processo de conversão*, no caso da manufatura, modifica a forma da matéria-prima ou muda a composição dos recursos. Em serviços não há propriamente transformação: o serviço é criado. O processo de transformação geralmente envolve uma seqüência de passos chamados *operações de produção*. Cada *operação de produção* é um processo de transformação das entradas em saídas, com uma adição de valor ao produto final. Intercaladas à estas operações que adicionam valor ao produto final existem operações tais como transporte, armazenagem e inspeção que não adicionam valor ao produto e devem, portanto, ser minimizadas ou mesmo eliminadas.

O *sistema de controle* monitora os demais elementos do sistema de produção. Trata-se de um conjunto de atividades que visam assegurar que o que foi programado seja cumprido, que padrões sejam obedecidos, que a qualidade desejada seja obtida e que os recursos sejam utilizados de forma eficaz.

Finalmente, é importante citar as influências tanto internas quanto externas as quais o sistema de produção está sujeito, e que afetam o seu desempenho. No caso do ambiente interno, o sistema de produção está sujeito a outras áreas funcionais da empresa tais como marketing, finanças e recursos humanos. Como influências externas podem ser citadas as condições econômicas do país, as políticas e regulamentações governamentais, a competição e a tecnologia.

Tabela 2.1: Classificação dos Sistemas de Produção

Critério	Classificação	Características
Pelo grau de padronização dos produtos	Sistemas que produzem produtos padronizados	Alto grau de uniformidade
		Produção em grande escala
		Produtos facilmente encontrados no mercado
	Sistemas que produzem produtos sob medida	Cliente define o produto
		Lotes normalmente unitários
Pelo tipo das operações	Processos contínuos	Produtos não podem ser individualmente identificados
	Processos discretos	Os produtos podem ser identificados individualmente
Pela natureza do produto	Manufatura de bens	Produtos que são tangíveis (e.x. rádio, carro)
	Prestador de serviços	Produtos que são intangíveis (e.x. consulta médica)

## 2.3 Classificação dos Sistemas de Produção

Existem várias classificações para os sistemas de produção. Na Tabela 2.1 são apresentadas três das formas mais conhecidas de classificação: pelo grau de padronização dos produtos, pelo tipo de operação a que são submetidos os produtos e pela natureza do produto [Tub00].

Produtos padronizados são bens ou serviços com alto grau de uniformidade e produção em grande escala, com o objetivo de tornarem-se amplamente disponíveis para os clientes. Dentro deste grupo estão incluídos a fabricação de bens como automóveis, eletrodomésticos, roupas, etc., ou a prestação de serviços como planos de saúde, serviços bancários, entre outras.

Produtos sob medida visam atender um cliente em particular. Por depender dos clientes para a definição dos produtos, estes normalmente não são produzidos para estoque e os lotes são normalmente unitários. Em decorrência da baixa quantidade produzida, a automação dos processos é menos aplicável, por não justificar os investimentos. Sistemas que se enquadram nesta categoria normalmente possuem grande capacidade ociosa e dificuldade de padronizar os métodos de trabalho e recursos produtivos.

vos, gerando produtos mais caros que os padronizados. Como exemplo de produção sob medida pode-se citar a construção civil, estaleiros, etc., e a prestação de serviços tais como táxis, restaurantes, clínicas médicas, entre outros. Os processos contínuos são empregados quando existe alta uniformidade na produção de bens e prestação de serviços e os produtos e processos produtivos são totalmente independentes, favorecendo a automação. Não há flexibilidade no sistema.

Os processos discretos de produção podem ser subdivididos em:

- Processos Repetitivos em Massa – produção em grande escala de produtos altamente padronizados;
- Processos Repetitivos em Lote – volume médio de produção, onde cada lote segue uma série de operações que necessitam ser programadas a medida que as operações anteriores forem realizadas; e
- Processos por Projeto – têm como objetivo atender uma necessidade específica do cliente, com todas as suas atividades voltadas para esta meta. O produto tem uma data específica para ser concluído e uma vez concluído o sistema produtivo se volta para um novo projeto. Exemplos de processos por projeto estão na fabricação de bens como navios, aviões, hidroelétricas, etc., e na prestação de serviços como escritórios de advocacia, agências de propagandas, etc.

Um sistema de produção pode ainda ser classificado em relação à natureza do produto. Quando o produto fabricado é algo tangível, tem-se uma *manufatura de bens*; quando o produto é intangível, tem-se uma *prestação de serviços*. Embora ambos os conceitos sejam similares do sentido de transformar insumos em produtos com valor agregado, uma diferença básica reside no fato de a manufatura de bens ser orientada ao produto enquanto a prestação de serviços é orientada à ação.

## 2.4 Funções dos Sistemas de Produção

Um sistema de produção deve exercer uma série de funções operacionais a fim de atingir seus objetivos. De um modo geral, estas funções podem ser sumarizadas em três: produção, *marketing* e finanças. Se existe uma produção, ela deve ser vendida (comercializada), mas para produzir e vender é necessário recurso financeiro. A produção não pode planejar um aumento na sua capacidade produtiva sem o aval do setor financeiro. Da mesma forma, o setor de *marketing* não pode fazer um plano de vendas que

a produção não consiga executar. Percebe-se, portanto, que uma função é dependente da outra.

O sucesso de um sistema produtivo depende da forma como estas funções se relacionam. A Figura 2.2 ilustra como *marketing*, produção e finanças devem operar em conjunto na definição, projeto e análise de novos produtos que atendam a necessidade dos clientes.

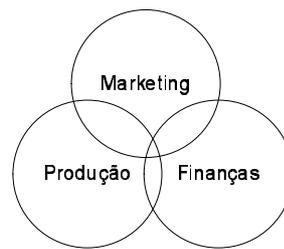


Figura 2.2: Estrutura Operacional

Produção consiste em todas as atividades que estão relacionadas com o a produção de bens e serviços. A essência da função de produção consiste em agregar valor aos bens ou serviços durante o processo de transformação.

A função de *marketing* liga a empresa (ou prestador de serviços) aos seus clientes e fornecedores. Ela está encarregada de vender e promover os bens e serviços produzidos por uma empresa.

A função de finanças está encarregada de administrar os recursos financeiros de uma empresa e alocá-los onde forem necessários.

Tubino [Tub00] destaca outras funções de suporte ou apoio que estão diretamente relacionadas ao PCP:

- Engenharia – responsável pelo projeto de produtos e pelos processos de fabricação e montagem dos bens ou serviços.
- Compra/Suprimento – responsável por suprir o sistema produtivo com as matérias-primas, componentes e equipamentos necessários para a produção de bens ou serviços.
- Manutenção – responsável por manter os equipamentos e instalações em perfeito estado de funcionamento.
- Recursos Humanos – responsável pelo recrutamento, treinamento e relações trabalhistas com os funcionários.

## 2.5 Planejamento e Controle da Produção

O PCP é uma função administrativa que tem por objetivo fazer os planos que orientarão a produção e servirão de guia para o seu controle, que também é feito pelo PCP. Em termos simples, o PCP determina *o que* vai ser produzido, *como* vai ser produzido, *onde* vai ser produzido, *quem* vai produzir e *quando* vai ser produzido.

Para que possa atingir os objetivos, o PCP deve ser permanentemente suprido de informações das áreas mais diretamente ligadas ao sistema produtivo, tais como: a engenharia do produto, engenharia do processo, *marketing*, manutenção, compra/suprimentos, entre outras. Essas áreas também recebem informações do PCP, para que possam melhor desempenhar suas atividades [Tub00].

As atividades do PCP são exercidas nos três níveis hierárquicos do planejamento das atividades produtivas. No nível estratégico, são definidas as políticas estratégicas de longo prazo da empresa. No nível tático, são estabelecidas as políticas de médio prazo. No nível operacional, são preparados os programas de produção de curto prazo e o acompanhamento dos mesmos.

A seguir será apresentada uma descrição das atividades desenvolvidas pelo PCP:

- Planejamento Estratégico da Produção – Consiste em estabelecer um PP de longo prazo que é definido em função do plano de vendas e disponibilidade de recursos da empresa. O PP deverá estar compatível com os planejamentos estratégicos de *marketing* e finanças. Um PP leva em conta famílias de produtos.
- Planejamento-Mestre da Produção – Consiste em estabelecer, a partir do PP, um PMP de produtos finais, detalhado a médio prazo, período a período. Enquanto o PP considera famílias de produtos, o Planejamento Mestre da Produção (PMP) leva em conta produtos finais que fazem parte destas famílias.
- Programação da produção – Com base no PMP e nas informações de estoque, estabelece, a curto prazo, quando e quanto comprar, fabricar ou montar cada item que compõe um produto final. Na programação da produção são emitidas as ordens de compra, de fabricação e montagem. Também é realizada a administração de estoques;
- Acompanhamento e Controle da Produção – Consiste na verificação se a execução está de acordo com o planejado.

Uma visão geral do inter-relacionamento das atividades do PCP pode ser vista na Figura 2.3.

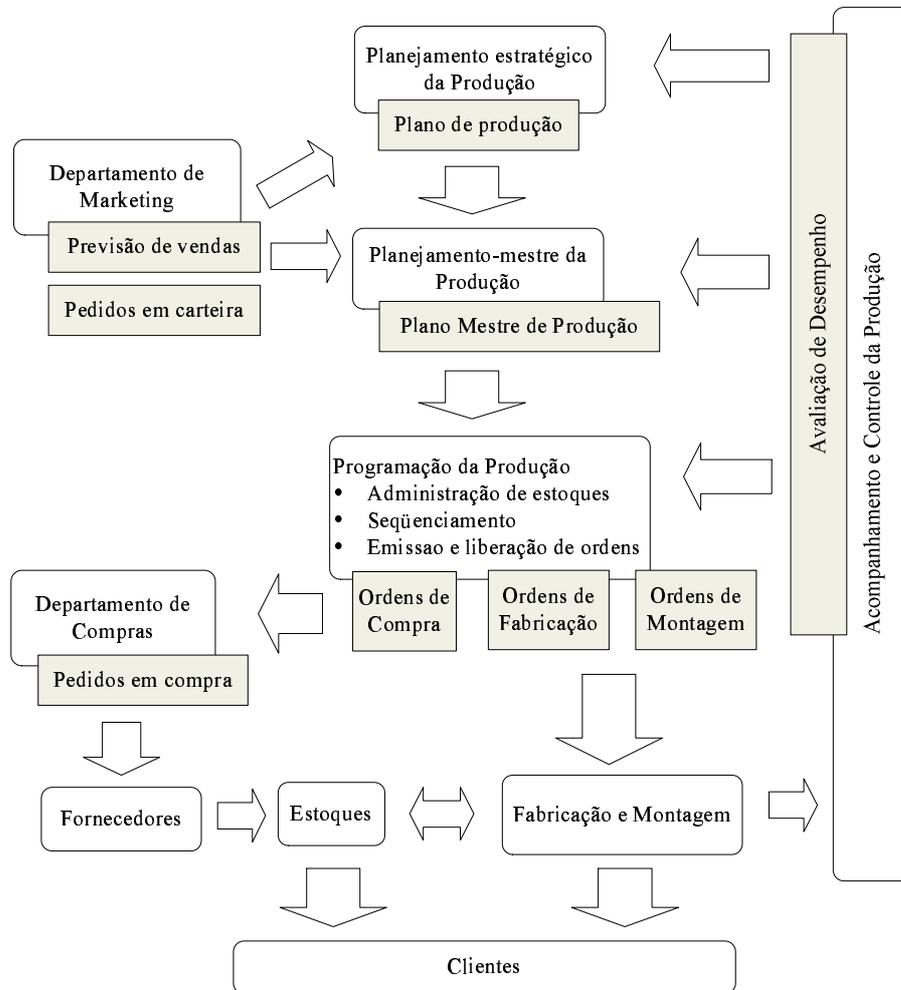


Figura 2.3: Visão Geral das Atividades do PCP

As informações dentro dos três níveis hierárquicos do planejamento das atividades (estratégico, tático e operacional) devem estar consolidadas, ou seja, o *Plano-Mestre de Produção* deve estar compatível com as decisões tomadas a longo prazo, previstas no *Planejamento Estratégico da Produção*. Da mesma forma, a programação da produção de um dado componente será eficiente se a capacidade produtiva do sistema tiver sido bem equacionada pelo *Planejamento-Mestre da Produção*.

## 2.6 Planejamento Estratégico da Produção

O principal objetivo da estratégia de produção é fazer com que uma empresa consiga atingir uma vantagem competitiva de longo prazo. Esta vantagem competitiva é conseguida através do adequado gerenciamento dos recursos de produção.

Segundo [CG96], há cinco prioridades competitivas principais baseadas nas quais a produção pode contribuir para a competitividade da empresa.

- Fazer os produtos *gastando menos* que os concorrentes, obtendo vantagens em custos.
- Fazer produtos *melhores* que os concorrentes, obtendo vantagem em qualidade.
- Fazer os produtos *mais rápido* que os concorrentes, obtendo vantagens em velocidade de entrega.
- Fazer os produtos *no prazo* prometido, obtendo vantagens em confiabilidade de entrega.
- Ser capaz de *mudar muito e rápido* o que se está fazendo, obtendo vantagens em flexibilidade.

A importância relativa de cada um dos objetivos citados varia conforme o mercado particular em que a empresa está competindo.

## 2.7 O Papel dos SAP

Os Sistemas de Administração da Produção (SAP) são o coração dos processos produtivos. Eles têm o objetivo de planejar e controlar o processo de produção em todos os seus níveis, incluindo materiais, equipamentos, pessoas, fornecedores e distribuidores.

Corrêa [CG96] define SAPs como sistemas que provêem informações que dão suporte ao gerenciamento eficaz do fluxo de materiais, da utilização de mão-de-obra e dos equipamentos, a coordenação das atividades internas com as atividades dos fornecedores e distribuidores e a comunicação/interface com os clientes no que se refere às suas necessidades operacionais. Ainda segundo [CG96], o ponto chave nesta definição é a necessidade gerencial de usar as informações para tomar decisões inteligentes.

É importante frisar que os SAPs não tomam decisões ou gerenciam sistemas. Sua função é dar suporte aos administradores para que possam executar sua função de forma adequada [VBW92].

### 2.7.1 Just in Time (JIT)

O JIT surgiu no Japão nos meados de 1970, sendo sua idéia básica e seu desenvolvimento creditado à *Toyota Motor Company* que buscava um sistema de administração que pudesse coordenar a produção com a demanda específica de diferentes modelos e cores de veículos com o mínimo atraso.

O sistema JIT pode ser definido como um sistema de produção cujo objetivo é otimizar os processos e procedimentos através da redução contínua dos desperdícios. As metas estabelecidas pelo JIT em relação aos problemas variados da produção são:

- Quantidade zero de defeitos.
- Tempo de preparação (*setup*) igual a zero.
- Estoque zero.
- *Lead time* igual a zero.
- Lote unitário.

O sistema JIT tem como objetivo fundamental a melhoria continua do processo produtivo através da redução dos estoques, os quais tendem a camuflar problemas.

As principais limitações do JIT estão ligadas à flexibilidade do sistema produtivo no que se refere à variedade de produtos oferecidos ao mercado e às variações de demanda de curto prazo. O sistema JIT requer que a demanda seja estável para que se consiga um balanceamento adequado dos recursos, possibilitando uma produção contínua.

O planejamento da produção no sistema JIT deve garantir uma carga de trabalho estável, que possibilite o estabelecimento de um fluxo contínuo de material. O sistema de programação e controle da produção está baseado no uso de cartões para a transmissão de informações entre os centros produtivos. Este sistema é denominado de sistema *Kanban* e segue a lógica de “puxar” a produção, produzindo somente o necessário, na quantidade e no momento necessários, de forma a atender a demanda dos centros consumidores [CG96].

### **Kanban**

O *Kanban* é a técnica de controle do JIT, fornecendo as condições operacionais para suportá-lo [SCJ97]. *Kanban* é um termo japonês que significa *cartão* ou *cartaz*, e é utilizado para controlar a ordem dos trabalhos em um processo seqüencial. Em outras palavras, o *Kanban* é o meio de sinalizar para a estação de trabalho antecedente que a estação de trabalho seguinte está preparada para que a estação de trabalho anterior produza outro lote de componentes.

No sistema *Kanban* de puxar a produção, não se produz nada até que o cliente (interno ou externo) de seu processo solicite a produção de determinado item. Shingo [Shi96] apresenta as principais características do sistema *Kanban*:

- Melhoria total e contínua dos sistemas de produção.
- Regulagem do fluxo de itens globais com controle visual a fim de executar essas funções com precisão.
- Simplificação do trabalho administrativo dando autonomia ao chão de fábrica.
- Informação transmitida de forma organizada e rápida.

O sistema mais difundido atualmente é o sistema de dois cartões, utilizado inicialmente na fábrica da *Toyota Motors Cia* no Japão. Este sistema utiliza dois cartões, denominados *Kanban* de produção e *Kanban* de transporte. Basicamente estes cartões substituem a maioria das outras formas em papel de controle da produção no chão de fábrica. O *Kanban* de produção dispara a produção de um pequeno lote de peças de determinado tipo, em um determinado centro de produção da fábrica. O *Kanban* de transporte autoriza a movimentação do material pela fábrica, do centro que produz determinado componente para o centro que consome este componente.

Dentre os benefícios decorrentes do sistema *Kanban* citados na literatura, destacam-se [Fil99]:

- Redução dos desperdícios, fora e dentro do chão de fábrica.
- Melhoria dos níveis de controle da fábrica, pela descentralização e simplificação dos processos operacionais.
- Redução do tempo de duração do processo (*lead-time*).

- Aumento da capacidade reativa da empresa (resposta aos clientes).
- Elevação do nível de participação e engajamento das pessoas, através da descentralização do processo decisório.
- Ajustamento dos estoques à flutuação regular da demanda.
- Redução dos estoques de produtos em processo.
- Diminuição dos lotes em produção.
- Eliminação dos estoques intermediários e de segurança.
- Sistematização e aperfeiçoamento do fluxo de informações, assim como dos mecanismos de comunicação entre o pessoal de produção.
- Integração do controle de produção nos demais mecanismos de flexibilidade da empresa.
- Maior facilidade na programação da produção.

O *Kanban* se aplica a empresas que têm produção repetitiva. Se esta produção é irregular no tempo e em quantidade, a obtenção de bons resultados não é garantida. De acordo com Monden [Mon83], o *Kanban* não pode ser utilizado, por exemplo, em ambientes com ordens de trabalho com curto tempo de manufatura (vida curta), tempos de *setup* significativos, muito refugo, ou grandes e imprevisíveis flutuações de demanda.

### 2.7.2 Material Requirement Planning (MRP)

O MRP é um sistema computadorizado que controla os níveis de estoque, planos de produção, auxilia no gerenciamento dos pedidos e contribui com o sistema de controle da manufatura com respeito a produção das partes montadas.

O MRP foi introduzido a partir de 1970, sendo grande parte do crédito de sua introdução dado a *Joseph Orlicky*, *George Plossl*, e *Oliver Wight*. A introdução do MRP também é creditada à APICS.

O MRP permite que as empresas calculem as quantidades de materiais de determinado tipo que são necessários para fabricar os produtos finais em uma determinada data [SCJ97].

### Objetivos do MRP

Os objetivos do MRP são os seguintes [GF01]:

- Melhorar o serviço ao cliente.
- Reduzir investimentos em estoques.
- Melhorar a eficiência operacional da fábrica.

Melhorar os serviços ao cliente significa não apenas ter os produtos disponíveis quando requisitados pelos clientes, mas também cumprir as promessas de entrega e abreviar tempos de entrega.

No MRP, os investimentos em estoque têm de ser reduzidos porque os pedidos de matéria-prima são programados para chegar aproximadamente no tempo em que o item final da matéria prima vier aparecer no programa mestre de produção (MPS). Em outras palavras, o MRP determina quando produzir mercadorias ou comprar materiais de forma a manter o estoque mínimo.

Todos estes benefícios resultam da *filosofia dos sistemas MRP*. Os sistemas MRP se baseiam na filosofia segundo a qual cada matéria-prima, peça e montagem necessária à produção devem chegar simultaneamente na hora certa para produzir os itens finais no MPS.

### Elementos do MRP

A Figura 2.4 apresenta uma visão geral do funcionamento do MRP [RI00].

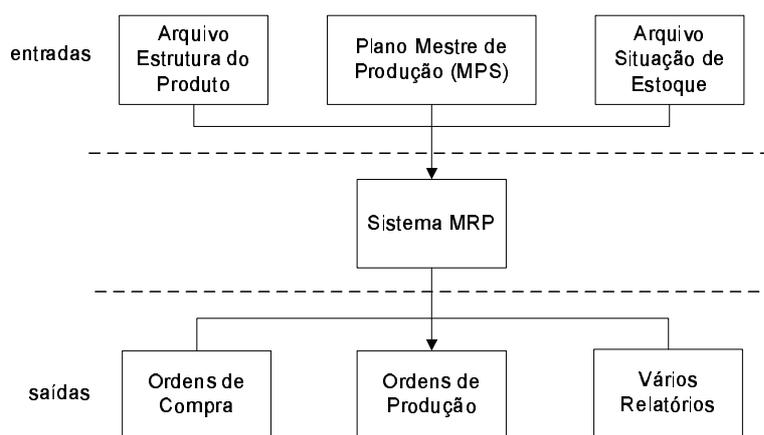


Figura 2.4: Visão Geral do Sistema MRP

As três entradas principais do MRP são os arquivos de estrutura do produto, o programa mestre de produção, e os arquivos-mestres de estoque.

**Programa Mestre de Produção** – O MPS guia o sistema MRP especificando quais produtos finais devem ser produzidos, quantos produtos são necessários, e quando eles são necessários. Os números que estão no MPS representam previsões, e não demanda.

Um MPS é desenvolvido para renovar estoques de itens acabados ou para atender requisições de clientes. Inicia como um programa experimental a ser testado quanto à sua viabilidade através do MRP e do CRP. O MPS não considera restrições de capacidade produtiva do sistema.

O MRP “explode” o MPS em necessidades de materiais. Caso estas necessidades não possam ser atendidas pelos materiais disponíveis no estoque ou pelos materiais pedidos, o MPS precisará ser recalculado [GF01].

**Arquivo de Estrutura do Produto** – O arquivo de estrutura do produto contém a *lista de materiais* (BOM) para cada item produzido. A BOM é uma lista dos materiais e suas quantidades necessárias para produzir uma unidade de um produto ou item final. Em outras palavras, este arquivo contém todas as partes componentes de um item final. Por exemplo, na produção de um carro, as partes componentes poderiam ser portas pneus, bancos, etc. A estrutura do produto não contém apenas as partes componentes, ela também fornece informações sobre a ordem em que estas partes devem ser montadas.

**Arquivo Situação de Estoque** – O arquivo mestre de estoque é um arquivo computadorizado com um registro completo de cada material mantido em estoque. Para que o MRP funcione adequadamente é necessário que o arquivo de situação de estoque seja o mais preciso possível.

**Relatórios de Saída do MRP** – As saídas do MRP fornecem dinamicamente o programa de materiais para o futuro - quantidade de cada material necessária em cada período de tempo para sustentar o MPS. As duas saídas principais são:

- Programação de pedidos – programa da quantidade de cada material a ser pedida em cada período de tempo.
- Mudança nos pedidos programados – modificação dos pedidos programados anteriormente.

Outras saídas secundárias do MRP são:

- Relatórios de exceções – destacam os itens que necessitam da atenção da administração a fim de fornecer a quantidade certa de materiais em cada período de tempo. Exceções típicas são os relatos de erros, pedidos atrasados, produção de refugo, etc.
- Relatórios de desempenho – indicam quão bem o sistema está operando.
- Relatórios de planejamento – utilizados em atividades futuras de planejamento de estoques, tais como previsão de estoque, planejamento de necessidade materiais de longo prazo, entre outras.

O sistema O MRP evoluiu em complexidade e assim passou a MRP2 e por último, ao agregar tanto os recursos financeiros e comerciais quanto a rede de fornecedores, evoluiu para o sistema ERP.

### 2.7.3 Manufacturing Resource Planning (MRP2)

O MRP2 surgiu por volta de 1980, a partir da evolução do MRP e da necessidade de estender a gestão para outras áreas da empresa. A definição genérica do MRP2 elaborada pela APICS: “É um método efetivo de gestão de todos os recursos de uma empresa. Transforma o planejamento operacional em unidades, planejamento financeiro em dinheiro e tem capacidade de efetuar simulações baseadas em perguntas do tipo “*what-if*”. É feito a partir de uma variedade de funções que se interligam: planejamento de negócios, planejamento de produção, tabelas de tempos de produção, planejamento de material e requisições, planejamento de capacidades e o funcionamento do sistema para capacidades e prioridades. O resultado deste sistema deve ser integrado com outros relatórios financeiros: balanços, encomendas, compras, estoques, produção, etc. ”.

O MRP2 é um sistema hierárquico de administração da produção, em que os planos de longo prazo de produção são sucessivamente detalhados até se chegar ao nível do planejamento dos componentes e máquinas específicas [CG96]. O MRP2 agregou no MRP, além da capacidade de cálculo de necessidade de materiais, a capacidade de cálculo de necessidade de outros recursos de produção, tais como necessidade de mão-de-obra e equipamentos.

O objetivo principal dos sistemas de cálculo de necessidade, como o MRP2, é o cumprimento dos prazos de entrega dos pedidos com formação reduzida de estoques, planejando as compras e a produção de elementos apenas para quando estes realmente forem necessários.

### **Módulos MRP2**

O MRP2 apresenta cinco módulos principais, que desempenham diferentes funções mas que apresentam relacionamento e dependência entre si: Planejamento da Produção (PP), Planejamento Mestre da Produção (MPS), Cálculo de Necessidade de Materiais (MRP), Cálculo de Necessidade de Capacidade (CRP) e Controle de Chão de Fábrica (SFC).

A saída de cada módulo é entrada para o módulo seguinte, na ordem indicada. Realimentações entre os módulos garantem que o resultado final seja o mais próximo possível do esperado.

### **Planejamento da Produção (PP)**

O PP objetiva auxiliar a decisão dos planejadores quanto aos níveis agregados de estoques e produção período-a-período, baseando-se também em previsões de demanda agregada (níveis de demanda do conjunto de produtos).

Este módulo atua em um nível mais agregado de planejamento de produção e o planejamento de produção é de mais longo prazo, podendo chegar a alguns anos. Como se trata de dados agregados sobre a produção de produtos diferentes, a unidade usada no planejamento é muitas vezes monetária. As decisões são referentes a volumes vendidos ou níveis de estoques a serem mantidos, em unidade monetária.

### **Planejamento Mestre da Produção (MPS)**

O MPS é responsável pela elaboração do plano de produção de produtos finais, período a período, item a item, que é o dado de entrada para o MRP. Este módulo não leva em conta, por exemplo, a estrutura do produto final. Este nível de detalhamento é alcançado no módulo MRP.

Dentre as principais informações necessárias para a tomada de decisões no MPS, destacam-se: a posição dos estoques, a previsão de vendas detalhada produto a produto, e a carteira de pedidos já aceitos.

### **Cálculo de Necessidade de Materiais (MRP)**

O módulo MRP ajuda a produzir e comprar apenas o necessário e apenas no momento necessário, visando eliminar estoques. Ele gera um plano de produção para o módulo CRP.

O MRP não considera limitações de capacidade produtiva do sistema, o que é feito posteriormente, no módulo CRP.

### **Cálculo de Necessidade de Capacidade (CRP)**

O CRP é o módulo responsável pelo cálculo da necessidade de capacidade do sistema produtivo. É o CRP que verifica se um dado plano de produção é ou não viável, com base em informações variadas como máquinas disponíveis, quantidade de funcionários, número de horas trabalhadas, eficiência de máquinas/trabalhadores, roteiros de produção, etc.

Os módulos acima do CRP ou trabalham com uma visão mais agregada da produção, ou levam em consideração apenas a necessidade de materiais, sem verificar se há de fato capacidade produtiva para manufaturar o produto. Por exemplo, o MRP, que é o módulo cuja saída é entrada para o CRP, calcula a necessidade de materiais com base apenas na estrutura do produto e dos *lead-times* associados com este produto e seus componentes. O *lead-time* corresponde ao tempo decorrido desde a liberação de uma ordem de compra/produção até a execução da mesma.

Através do CRP é possível identificar excesso ou falta de capacidade produtiva do sistema, o que pode auxiliar na tomada de decisões como a renegociação de prazos, aquisição de novas máquinas, efetuação de turnos extras de trabalho para atender a um pedido, etc. O cálculo do CRP é um processo complexo, que varia de empresa para empresa e que requer uma grande acuidade no que se refere às informações fornecidas ao mesmo, sob pena de se obterem como resposta deste módulo resultados que não correspondem à realidade.

Identificar excessos/insuficiência de capacidade é importante pois a falta de capacidade produtiva pode deteriorar o desempenho de uma empresa em termos de cumprimentos de prazos. Por sua vez, o excesso de capacidade produtiva pode representar um custo desnecessário, o qual pode muitas vezes ser reduzido, caso esta capacidade em excesso seja tão logo identificada.

### **Controle de Fábrica (SFC)**

O SFC é responsável pelo seqüenciamento das ordens por centro de produção, dentro de um período de planejamento e pelo controle da produção no nível da fábrica. Ele busca garantir que o que foi planejado será executado da forma mais fiel possível aos planos de produção.

### **2.7.4 Optimized Production Technology (OPT)**

OPT é uma técnica de gestão da produção que usa as idéias da *Teoria das Restrições (TOC)*, desenvolvida com ênfase nas restrições de capacidade ou

gargalos de uma operação. Localizando as restrições e trabalhando para removê-las e em seqüência procurando a próxima restrição, uma operação está sempre focada na parte crítica que determina a taxa de saída.

OPT é uma técnica baseada em *software* e ferramenta que auxilia a programação dos sistemas produtivos na taxa ditada pelos seus recursos mais carregados, ou seja, os gargalos do sistema. Se a taxa de produção em uma parte qualquer excede a do recurso gargalo, então estão sendo produzidos itens que não podem ser consumidos. Se a taxa de produção é inferior a taxa do recurso gargalo, então o sistema está sendo sub-utilizado.

Os seguintes princípios do OPT demonstram o seu foco nos recursos gargalo:

- Balancear o fluxo e não a capacidade.
- A utilização de um recurso não-gargalo é determinado por alguma outra restrição do sistema, e não pela sua própria capacidade.
- Utilização e ativação de um recurso não são sinônimos.
- Uma hora ganha em um recurso gargalo é uma hora ganha em todo o sistema.
- Uma hora ganha em um recurso não-gargalo não é nada, é só uma miragem.
- O lote de transferência pode não ser e, freqüentemente não deveria ser, igual ao lote de processamento.
- O lote de processamento deve ser variável e não fixo.
- Os gargalos não só determinam o fluxo do sistema, mas também definem seus estoques.
- O escalonamento das atividades e a capacidade produtiva devem ser considerados simultaneamente e não seqüencialmente. Os *lead-times* são um resultado do escalonamento e não podem ser assumidos *a priori*.

OPT é uma técnica baseada em *software* a exemplo do MRP2. Entretanto, seus princípios diferem bastante dos princípios do MRP2. Enquanto o MRP, como um conceito, não determina *lead-times* ou tamanhos de lotes fixos, muitas operações o executam com estes valores fixos por simplicidade. Entretanto, como a demanda, fornecimento e o processo produtivo apresentam variações não planejadas, os gargalos são, portanto, dinâmicos,

mudando sua localização e severidade. Por esta razão, os gargalos raramente são constantes no tempo. De forma similar, como os gargalos determinam o escalonamento da produção, o tamanho dos lotes pode mudar através da planta em função de o centro de trabalho ser ou não um recurso.

Um dos pontos considerados mais fortes do sistema OPT refere-se à maneira com que programa as atividades. O OPT usa a terminologia “*drum, buffer, rope*” numa referência ao trio de elementos que são chaves para o métodos: “tambor, estoque protetor, corda”. O tambor, que representa os gargalos e recursos restritivos críticos (RRC), determina o ritmo e o volume da produção do sistema. O estoque protetor garante que o RRC não pare por falta de material. Finalmente, a corda representa a sincronização entre as necessidades de chegada de materiais no estoque protetor e a admissão de matéria-prima no sistema.

### 2.7.5 Enterprise Resource Planning (ERP)

ERP é um termo da indústria para o amplo conjunto de atividades que dinamicamente balanceiam e otimizam os recursos da empresa.

ERP é uma solução que integra todos os subsistemas existentes em uma empresa, tais como engenharia, finanças, recursos humanos, serviços, etc. Ao integrarem todos os subsistemas, os ERP conseguem resultados melhores que o total dos subsistemas em separado. Os sistemas de aplicação tradicionais tratam todas as transações isoladamente. Elas são feitas e usadas para dar resposta a funções específicas para as quais foram destinadas. Os ERP deixaram de olhar para uma transação como sendo um processo isolado, mas parte de todo um conjunto de processos interligados que perfazem toda a existência de uma empresa [SSJS00]. Os sistemas ERP são também denominados *Sistemas Integrados de Gestão*.

#### Características dos ERP

Os sistemas ERP não servem apenas para integrar os vários organismos de uma empresa. Segundo Simões [SSJS00], para verdadeiramente ser considerado ERP, o sistema deve possuir algumas das seguintes características:

- Flexibilidade – um sistema ERP é flexível de forma a responder às constantes transformações das empresas.
- Modularidade – o sistema ERP é um sistema de arquitetura aberta, isto é, pode usar um módulo livremente sem que este afete os restantes.

- Compreensivo – o sistema está apto a suportar diferentes estruturas.
- Conectividade – o sistema não se deve confinar ao espaço físico da empresa, mas permitir a ligação com outras entidades pertencentes ao mesmo grupo empresarial.
- Seleção de diferentes formas de negócio – deve conter uma seleção das melhores práticas negociais em todo o planeta.
- Simulação da Realidade – deve permitir a simulação da realidade da empresa em computador.

Devido à sua complexidade, os sistemas ERP são compostos por módulos, parametrizáveis em função de um cliente específico, tendo todos os fornecedores um conjunto de módulos/aplicações base:

- Vendas, distribuição e marketing.
- Planejamento de capacidades.
- Gestão financeira.
- Gestão de produção.
- Contabilidade.
- Recursos humanos.
- *E-Business*.

Sistemas MRP2 primitivos eram direcionados apenas para as indústrias. Os sistemas ERP atuais podem ser utilizados nas empresas independentemente da sua estrutura orgânica, tamanho, distribuição física, ramo de negócios, etc. Dentre os diferentes setores de negócio onde os ERPs encontram aplicabilidade, podem ser citados:

- Indústria metalomecânica e minas.
- Administração pública, saúde e educação.
- Telecomunicações.
- Petróleo, energia e derivados.
- Indústria automobilística.

O ERP automatiza os processos de uma empresa, com a meta de integrar as informações através da organização, eliminando interfaces complexas e caras entre sistemas heterogêneos. Por integrarem todos os organismos de uma empresa, os ERPs são normalmente sistemas grandes e complexos. Para adequar estes sistemas às suas necessidades particulares, as empresas recorrem à customização. Este processo de customização torna a atualização do sistema cara e difícil.

Para evitar a armadilha da customização, muitos ERPs tornaram-se parametrizáveis. Entretanto, quanto mais parametrizável, mais complexo o sistema se torna. Como nunca é possível tornar todos os casos totalmente parametrizáveis, a necessidade de customização continua a existir.

### **Fornecedores de sistemas ERP**

Existem vários fornecedores de sistemas ERP atualmente, os quais fornecem todo um conjunto de produtos e soluções. Os cinco fornecedores de sistemas ERP dominantes são:

- SAP
- Baan
- PeopleSoft
- JDEdwards
- Oracle

Cada fornecedor, por razões históricas, está especializado em uma área particular tal como *Baan* em manufatura, *PeopleSoft* em recursos humanos, *SAP* em logística e *Oracle* em finanças. Juntos, os cinco maiores controlam mais de 60% do mercado global [Nah02].

*SAP* foi fundada em 1972 na Alemanha por cinco engenheiros da *IBM*, sendo hoje em dia a maior empresa do seu ramo. O seu sistema *R/3* foi otimizado para gerir os processos de produção e gestão, logística e recursos humanos.

*PeopleSoft* é o segundo maior fornecedor mundial, sendo a sua imagem de marca os módulos de gestão de recursos humanos. A *Peoplesoft* está atualmente direcionando seus produtos para as áreas dos serviços, com produtos de auxílio de controle de custos.

A *Oracle* produz e vende aplicações ERP desde 1987, sendo a maioria dos seus clientes empresas ligadas à produção e consumo de produtos,

sendo assim um adversário direto da *SAP*. Curiosamente em cerca de 80% dos casos, o *software* da *SAP* opera sobre uma base de dados da *Oracle*.

*Baan* é uma empresa holandesa e uma forte concorrência à *SAP*. Recentemente, tal como outros fornecedores, tem dedicado especial atenção ao mercado de pequenas e médias empresas, fato que tem resultado numa enorme variedade de produtos que oferece bem como um rápido retorno financeiro dessa aposta.

O último dos grandes fornecedores é a *JDEdwards*. Apesar de vender *software* já a vários anos, só se tornaram conhecidos mundialmente a poucos anos. Desde que lançaram o *OneWorld*, *software* ERP, conseguiram uma importante cota dentro do mercado mundial de ERP.

Uma descrição mais detalhada destas empresas e seus principais produtos pode ser encontrada em [SSJS00] e [Nah02].

## 2.8 Considerações Finais

A competitividade imposta pelas transformações econômicas ocorridas nas últimas décadas tem estimulado a indústria de produção a tornar seus processos produtivos mais eficientes. O Brasil tem experimentado profundas mudanças no seu setor produtivo referentes à modernização de seus processos de produção, melhoria da qualidade de seus produtos e racionalização administrativa.

Na área específica do Planejamento da Produção tem havido um aumento do uso de sistemas computacionais de apoio à decisão que buscam englobar vários aspectos que interferem nos processos de produção os quais são, pela sua própria natureza, difíceis de serem analisados de forma racional por planejadores.

Tais sistemas permitem que se escolham as melhores alternativas de produção com respeito aos custos envolvidos e às restrições inerentes ao processo. Dentre esses sistemas, podem ser citados OPT, JIT, MRP, MRP2, ERP, entre outros.

Tanto os problemas de planejamento como os de controle podem ser formulados como problemas de otimização combinatória que é uma área deficiente de métodos de análise gerais e eficientes. Uma dificuldade adicional é a dimensão de problemas práticos, que os torna intratáveis através de métodos de otimização exatos. Neste sentido, RdPs constituem-se em formalismo poderoso para a modelagem de tais sistemas.

# Capítulo 3

## Redes de Petri

*Neste capítulo, são apresentados os conceitos de RdPs, o formalismo utilizado no desenvolvimento deste trabalho. Primeiro, são apresentados os componentes do modelo e sua notação gráfica. Em seguida, são mostradas as propriedades que podem ser capturadas a partir do modelo, bem como os métodos adotados. Finalmente, são apresentadas algumas classes de RdPs.*

### 3.1 Introdução

Redes de Petri (RdPs) é um termo genérico que se refere a uma família de técnicas para especificação formal de sistemas que possibilita uma representação matemática e possui mecanismos de análise que permitem a verificação de propriedades e a verificação da corretude do sistema especificado [MLC96, Mur89, Pet81, Rei82].

A teoria inicial das RdPs foi apresentada pela primeira vez na tese de doutorado de *Carl Adam Petri*, em 1962, e intitulada *Kommunikation mit automaten* (Comunicação com autômatos). Trata-se de um modelo do tipo estado-evento, onde cada evento possui pré-condições que habilitam a sua ocorrência, e pós-condições resultantes deste, as quais são por sua vez pré-condições de outros eventos seqüentes.

Um passo importante no projeto de sistemas consiste na separação e identificação dos elementos passivos dos elementos ativos [GV03]. Nas RdPs, existem dois conjuntos disjuntos de elementos: *lugares* e *transições*. Entidades do mundo real interpretadas como elementos passivos são representadas como lugares (por exemplo, condições, recursos, etc). Entidades do mundo real interpretadas como elementos ativos são representadas como transições (por exemplo, eventos, ações, transmissão de mensagens,

etc.).

A realização de uma ação está associada a algumas pré-condições, ou seja, existe uma relação entre os lugares e as transições que possibilita a realização de uma ação. Da mesma forma, após a realização de uma ação, alguns lugares terão suas informações alteradas (pos-condições).

O comportamento de uma transição depende exclusivamente da sua *localidade*, que é definida como a totalidade dos objetos de entrada e saída (pré- e pós-condições, lugares de entrada e saída, etc.), juntamente com a própria transição.

As vantagens da utilização das RdPs na modelagem de sistemas são conhecidas [GV03].

- RdPs fornecem uma formalismo de modelagem que permite uma representação gráfica e é fundamentado matematicamente.
- Existe uma grande variedade de algoritmos para o projeto e análise de RdPs além de ferramentas de *software* desenvolvidas para auxiliar neste processo.
- RdPs provêem mecanismos para abstração e refinamento que são integrados ao modelo básico.
- Existe um grande número de ferramentas, tanto comerciais quanto acadêmicas, para o projeto, simulação e análise de sistemas baseados em RdPs.
- RdPs têm sido utilizadas em muitas áreas distintas e em conseqüência existe um grande número de especialistas no campo da modelagem.
- Existem várias extensões ao modelo básico de RdP.

Diferentes variantes do modelo de RdPs clássico têm sido desenvolvidas. Isto possibilita às RdPs suprir as necessidades existentes em diferentes áreas de aplicação, além de prover facilidades de comunicação e transferência de métodos e ferramentas de uma área para outra. Atualmente, além do modelo básico, existem extensões tais como redes temporizadas [Ram74], estocásticas [Mol81, AM89], alto-nível [Jen90] e orientadas a objetos [Jan98, Voj01].

## 3.2 Elementos Básicos

Uma RdP é composta dos seguintes elementos [DAJ95, MLC96, Fil00]:

**Lugares:** representam os componentes passivos do sistema. Sua função é armazenar marcas. Tais marcas podem ser acrescentadas e removidas pelo disparo das transições. O lugar cuja marcação influencia o disparo de uma transição é chamado pré-condição daquela transição. O lugar que tem sua marcação afetada pelo disparo de uma transição é chamado pós-condição daquela transição. Em sistemas de produção, lugares podem representar recursos tais como máquinas, ou partes em um *buffer*. A existência de uma ou mais marcas em um lugar pode representar a disponibilidade de um recurso em particular [DAJ95].

**Transições:** representam componentes ativos do sistema. Uma transição estará habilitada quando todas as suas pré-condições forem atendidas. Uma transição estará desabilitada caso ao menos uma de suas pré-condições não seja atendida. Uma vez habilitada, uma transição pode ser disparada. O disparo da transição remove um determinado número de marcas dos lugares que são pré-condições e gera novas marcas nos lugares que são pós-condições. Em sistemas de produção, as transições podem representar, por exemplo, atividades do sistema modelado.

**Arcos:** guiam o fluxo das marcas através da rede. Arcos conectam lugares a transições e transições a lugares (não são permitidos arcos entre elementos da mesma classe). É permitido a existência de mais de um arco entre os mesmos elementos (arcos multivalorados). O lugar de partida de um arco com destino a uma transição é uma pré-condição da transição. O lugar de chegada de um arco com origem em uma transição é uma pós-condição da transição.

**Marcas:** as marcas (também denominadas de fichas ou *tokens*) são os componentes dinâmicos do modelo. Seu fluxo através da rede determina o comportamento do sistema modelado. O número de marcas presentes em uma rede, bem como a sua localização, determinam o estado atual do sistema [Fil00].

Na Figura 3.1 temos a representação gráfica de uma RdP. Os lugares são representados por círculos e as transições por traços ou barras. Estes dois elementos são os vértices do grafo associado às RdPs. Os vértices são interligados por arcos dirigidos. Adicionalmente, pode haver inscrições tais como nomes, marcas, expressões, guardas, etc.

Para cada representação gráfica de uma RdP existe uma representação algébrica equivalente. Ela contém o conjunto de lugares, transições, arcos, e informações adicionais como inscrições.

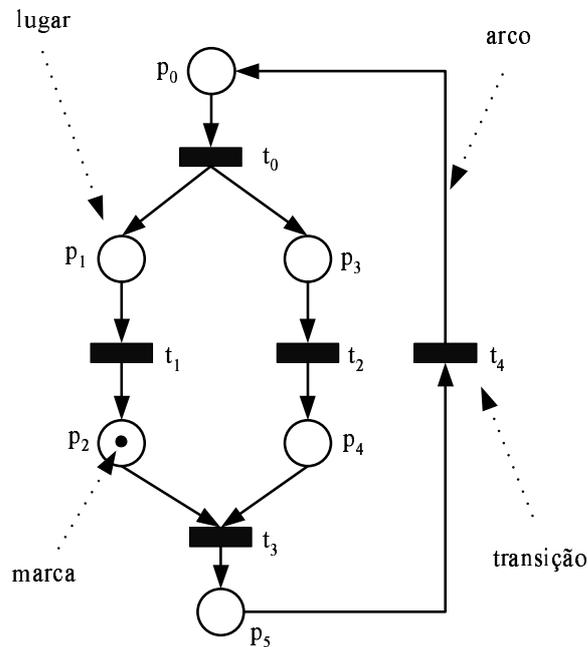


Figura 3.1: Exemplo de Rede

### 3.3 Estrutura de uma RdP

Nesta seção serão apresentadas três definições para RdPs. A primeira definição utiliza a teoria dos *bags* [Pet81] para representar a rede. A segunda definição utiliza notação matricial [Bra83]. A terceira define RdPs através de relações e pesos associados a estas relações [Rei82].

A estrutura de uma RdP em *bags* pode ser formalmente definida como segue:

**Definição 3.1** (Estrutura das Redes de Petri em Bags). *Uma RdP é uma quintupla  $N = \langle P, T, I, O, K \rangle$  onde*

- $P = \{p_0, p_1, \dots, p_n\}$  um conjunto finito não vazio de lugares.
- $T = \{t_0, t_1, \dots, t_m\}$  um conjunto finito não vazio de transições, onde  $P \cap T = \emptyset$ .
- $I : T \rightarrow P^\infty$  conjunto de bags que representa o mapeamento de transições em lugares de entrada.

- $O : T \rightarrow P^\infty$  conjunto de bags que representa o mapeamento de transições em lugares de saída.
- $K : P \rightarrow \mathbb{N} \cup \{\omega\}$  vetor de capacidades associadas aos lugares. O símbolo  $\omega$  é utilizado para representar que o lugar tem capacidade infinita.

Na representação gráfica, a existência de múltiplos arcos conectando lugares e transições pode ser indicada de forma compacta através de um único arco rotulado com seu peso ou multiplicidade  $w$  (veja a Figura 3.2).

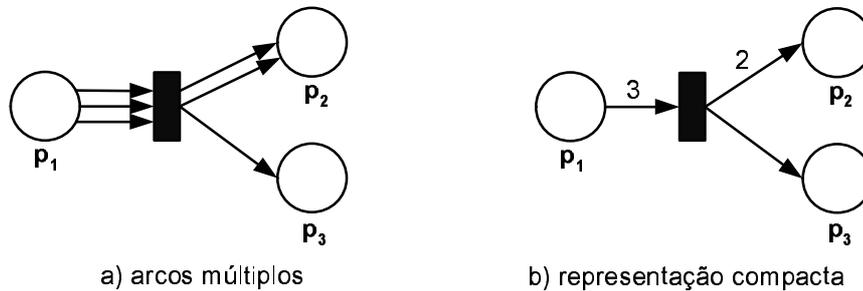


Figura 3.2: Arcos Multivalorados

Segundo a notação matricial, a estrutura das RdPs é representada por uma quintupla formada pelo conjunto de lugares, o conjunto de transições, a matriz de saída das transições e a capacidade de cada lugar, formalmente definida como segue:

**Definição 3.2** (Estrutura das Redes de Petri em Matrizes). *Uma RdP é uma quintupla  $N = \langle P, T, I, O, K \rangle$  onde*

- $P = \{p_0, p_1, \dots, p_n\}$  um conjunto finito de lugares.
- $T = \{t_0, t_1, \dots, t_m\}$  um conjunto finito de transições, onde  $P \cap T = \{\}$ .
- $I : P \times T \rightarrow \mathbb{N}$  é a matriz de pré-condições.
- $O : P \times T \rightarrow \mathbb{N}$  é a matriz de pós-condições.
- $K : P \rightarrow \mathbb{N} \cup \{\omega\}$  vetor de capacidades associadas aos lugares.

A estrutura das RdPs é representada através de relações por uma quintupla formada pelo conjunto de lugares, o conjunto de transições, o conjunto de arcos que interligam lugares às transições ou transições aos lugares e a valoração ou peso dos arcos, formalmente definida como segue:

**Definição 3.3** (Estrutura das Redes de Petri em Relações). *Uma RdP é uma quádrupla  $N = \langle P, T, R, W \rangle$  onde*

- $P = \{p_0, p_1, \dots, p_n\}$  um conjunto finito não vazio de lugares.
- $T = \{t_0, t_1, \dots, t_m\}$  um conjunto finito de transições, onde  $P \cap T = \emptyset$ .
- $R \subseteq (P \times T) \cup (T \times P)$  arcos que conectam lugares às transições e transições aos lugares.
- $W : R \rightarrow \mathbb{N}$  vetor de pesos dos arcos.

As definições básicas apresentadas nesta seção são referentes às RdPs Lugar/Transição, por ser o modelo mais difundido.

### 3.4 RdP Marcada

As marcas são informações atribuídas aos lugares. O número e distribuição das marcas nos lugares corresponde à marcação (estado) da rede em um dado momento. Nesta seção serão apresentados os conceitos de marcação, assim como a definição de uma RdP marcada.

**Definição 3.4** (Marcação). *Seja  $P$  o conjunto de lugares de uma RdP. Define-se formalmente marcação como uma função que mapeia o conjunto de lugares  $P$  a inteiros não negativos.  $M : P \rightarrow \mathbb{N}$ .*

A marcação de uma rede também pode ser definida na forma vetorial:

**Definição 3.5** (Vetor Marcação). *Seja  $P$  o conjunto de lugares de uma RdP. A marcação pode ser definida formalmente como um vetor  $M = (M(p_1), M(p_2), \dots, M(p_n))$ , onde  $n = \#(P)$ , para todo  $p_i \in P$ , tal que  $M(p_i) \in \mathbb{N}$ .*

Uma RdP com uma marcação associada é denominada uma RdP marcada. Uma RdP marcada pode ser definida por uma tupla formada pela estrutura da rede ( $N$ ) e uma marcação inicial ( $M_0$ ) associada à rede.

**Definição 3.6** (Rede Marcada). *Define-se uma RdP marcada pela tupla  $RM = \langle N, M_0 \rangle$ , onde  $N$  é a estrutura da rede e  $M_0$  é a marcação inicial.*

Normalmente, a presença de marcas em um lugar pode ser interpretada como a presença de um recurso de um determinado tipo. Graficamente estas marcas são normalmente representadas por inscrições no interior dos lugares.

O disparo de uma transição na RdP corresponde à ocorrência de um evento que modifica o estado do sistema. A evolução da marcação de acordo com a regra de disparo das transições simula o comportamento dinâmico do sistema.

### 3.5 Propriedades das RdPs

O estudo das propriedades de uma RdP permite uma análise detalhada do sistema modelado. As propriedades de uma RdP costumam ser divididas em *propriedades comportamentais*, que dependem do estado (ou marcação inicial) e da estrutura da RdP, e *propriedades estruturais*, que dependem apenas da estrutura da rede.

Dentre as principais propriedades comportamentais das RdPs, podem ser citadas a alcançabilidade, limitação, segurança, vivacidade e reversibilidade.

- A *alcançabilidade* de uma marcação em uma RdP refere-se a possibilidade de um sistema atingir um determinado estado. Uma marcação  $M_i$  é alcançável a partir de  $M_0$  se existir uma seqüência de disparos de transições que leve uma rede com marcação  $M_0$  a outra rede com marcação  $M_i$ .
- Uma RdP é *limitada* se e somente se o número de marcas de cada lugar da rede não supera  $k$ .  $M(p_i) \leq k$  para qualquer marcação alcançável. Um caso particular das redes limitadas ocorre quando o número de marcas em cada lugar é um, no qual a rede é dita *segura*. Em um ambiente de manufatura, a limitação de uma rede indica a ausência de *overflow* (caso o lugar represente áreas de armazenamento como em *buffers*) ou a limitação de recursos (caso os lugares modelem a disponibilidade de recursos).
- Uma RdP é *viva* se e somente se para cada marcação  $M_i$  existe uma seqüência de disparo de transições a partir de  $M_i$  que habilitam as transições  $t_i \in T$ . A vivacidade de uma RdP indica a ausência de bloqueios. Contudo, a ausência de bloqueio não implica em vivacidade.

- Um *bloqueio* (ou *deadlock*) em uma RdP ocorre quando se atinge uma marcação a partir da qual o disparo de uma ou mais transições não é mais possível.
- Uma rede é considerada *conservativa* se todas as suas transições adicionam nos lugares de saída exatamente a mesma quantidade de marcas que removem dos seus lugares de entrada. Em uma rede conservativa, o número de marcas permanece constante e não é alterado pelo disparo das transições. Portanto, a quantidade de marcas é um invariante. Uma rede é *sub-conservativa* se todas as transições adicionam nos lugares de saída no máximo o mesmo número de marcas que removem dos seus lugares de entrada. O número de marcas portanto, não pode aumentar.

Uma descrição mais detalhada destas e de outras propriedades das RdPs pode ser encontrada em [Mur89, MLC96].

### 3.6 Métodos de Análise e Validação de uma RdP

Uma grande vantagem na utilização de RdPs na modelagem de sistemas é a sua capacidade de formalização para a análise e verificação das propriedades do sistema modelado. A análise destas propriedades é implementada através de três métodos [ZZ94, MK80]: geração do grafo de alcançabilidade, análise da equação de estados e técnicas de redução. A validação das propriedades pode ser realizada através da simulação.

- Geração da árvore ou grafo de alcançabilidade – envolve a listagem de todas as marcações alcançáveis a partir da marcação inicial  $M_0$ . Cada nó da árvore representa uma marcação alcançável e cada arco representa o disparo de uma transição.
- Análise através da Equação de Estado – utilizada a matriz de incidência, que representa as conexões entre lugares e transições, para representar a estrutura da rede e caracterizar o comportamento dinâmico do sistema.
- Técnicas de redução – consistem em transformações que reduzem a dimensão do grafo de alcançabilidade, mas que asseguram parcialmente a conservação das propriedades a serem analisadas.
- Simulação - utilizada quando o sistema é relativamente complexo e sua análise através de métodos analíticos se mostra inviável. Neste

contexto, a simulação é utilizada para validação virtual do sistema, ou seja, para que se tenha uma maior confiança de que o sistema modelado apresenta as propriedades desejadas.

Na seqüência serão detalhados os métodos de análise baseados na árvore de cobertura, equação de estado e em técnicas de redução.

### 3.6.1 Árvore de Cobertura

*Árvore de cobertura* é um método de análise que se baseia na construção de uma árvore que possibilite a representação de todas as possíveis marcações de uma rede. A árvore de cobertura é utilizada para representar de forma finita um número infinito de marcações.

Para uma dada Rdp com uma marcação inicial, é possível obtermos diversas marcações para um grande número de transições potencialmente habilitadas [DCG91]. Essas marcações podem ser representadas por uma árvore, onde os nós são as marcações e os arcos as transições disparadas. Algumas propriedades, tais como limitação, segurança, transições mortas e alcançabilidade de marcações [Yen92], podem ser analisadas através da árvore de cobertura.

**Definição 3.7** (Árvore de Cobertura). *Seja  $N = \langle P, T, R, W, M_0 \rangle$  uma Rdp marcada. Define-se árvore de cobertura pelo par  $AC = (S, A)$ , onde  $S$  representa as marcações e  $A$  os arcos rotulados por  $t_j \in T$ .*

Para uma Rdp limitada, a árvore de cobertura é denominada *árvore de alcançabilidade*, dado que esta contém todas as possíveis marcações da rede. Uma outra possível representação finita das marcações acessíveis pode ser obtida do grafo de cobertura [Bra83].

**Definição 3.8** (Grafo de Cobertura). *Seja  $N = \langle P, T, R, W, M_0 \rangle$  uma Rdp marcada. Define-se grafo de cobertura pelo par  $GC = (S, A)$ , onde  $S$  representa o conjunto de todos os nós na árvore de cobertura e  $A$  os arcos rotulados por  $t_j \in T$  representando todos os possíveis disparos de transição tal que  $M_i[t_k > M_j$ , onde  $M_i, M_j \in S$ .*

Se uma rede é limitada o grafo de cobertura também é denominado de *grafo de alcançabilidade*. Este método de análise permite a análise das propriedades de limitação, segurança, conservação e problemas de cobertura de marcações. Todavia, não possibilita a resolução de vivacidade, alcançabilidade e reversibilidade em redes não-limitadas. Este problema está relacionado à introdução do símbolo  $\omega$  que é uma informação da qual não se sabe o seu real valor [MLC96].

### 3.6.2 Matriz de Incidência e Equação de Estado

O comportamento dinâmico de muitos sistemas pode ser descrito por equações diferenciais ou equações algébricas. A vantagem das técnicas algébricas sobre as técnicas baseadas na análise das árvores de cobertura é que a análise de propriedades pode ser efetuada pela resolução de equações lineares simples. As equações desenvolvidas governam o comportamento concorrente dos sistemas modelado por RdPs. Todavia, a solução destas equações é limitada, em parte devido à natureza não determinística dos modelos de RdP e por causa da restrição que soluções devem ser encontradas como inteiros não-negativos [Mur89, DAJ95].

**Definição 3.9** (Matriz de Incidência). *A Matriz de incidência  $A$  de uma RdP é uma matriz  $n \times m$  de inteiros, definida como:*

$$A = [a_{ij}]$$

e

$$a_{ij} = a_{ij}^+ - a_{ij}^-$$

onde  $a_{ij}^+ = w(i, j)$  é o peso do arco da transição  $i$  para seu lugar de saída  $j$  e  $a_{ij}^- = w(j, i)$  é o peso do arco do lugar de entrada  $j$  para a transição  $i$ .

**Definição 3.10** (Equação de Estado). *Seja  $m_k$  a marcação de uma RdP após sua  $k$ -ésima execução, com  $k \geq 0$ . A próxima marcação  $m_{k+1}$  é determinada pela Equação de Estado definida por*

$$m_{k+1} = m_k + Au_k$$

onde  $A$  é a matriz de incidência e  $u_k$  é um vetor de dimensão  $(m \times 1)$  de inteiros não negativos, chamado vetor de disparo, no qual cada entrada representa o número de vezes que a respectiva transição disparou durante a  $k$ -ésima execução da rede.

### 3.6.3 Redução

A análise de sistemas de grandes dimensões normalmente é uma operação custosa. Reduções são transformações aplicadas ao modelo de um sistema com o objetivo de simplificá-lo, e ainda preservando as propriedades do sistema a ser analisado. De modo oposto, existem técnicas para transformar um modelo abstrato em um modelo mais refinado que podem ser utilizadas para processos de *síntese*.

Existem várias técnicas de transformação para RdPs. Aqui serão citadas apenas as mais simples, que podem ser utilizadas para a análise de vivacidade, segurança e limitação por preservarem tais propriedades.

- Fusão serial de lugares, como mostrado na Figura 3.3(a).
- Fusão serial de transições, como mostrado na Figura 3.3(b).
- Fusão paralela de lugares, como mostrado na Figura 3.3(c).
- Fusão paralela de transições, como mostrado na Figura 3.3(d).
- Eliminação de lugares auto-laço, como mostrado na Figura 3.3(e).
- Eliminação de transições auto-laço, como mostrado na Figura 3.3(f).

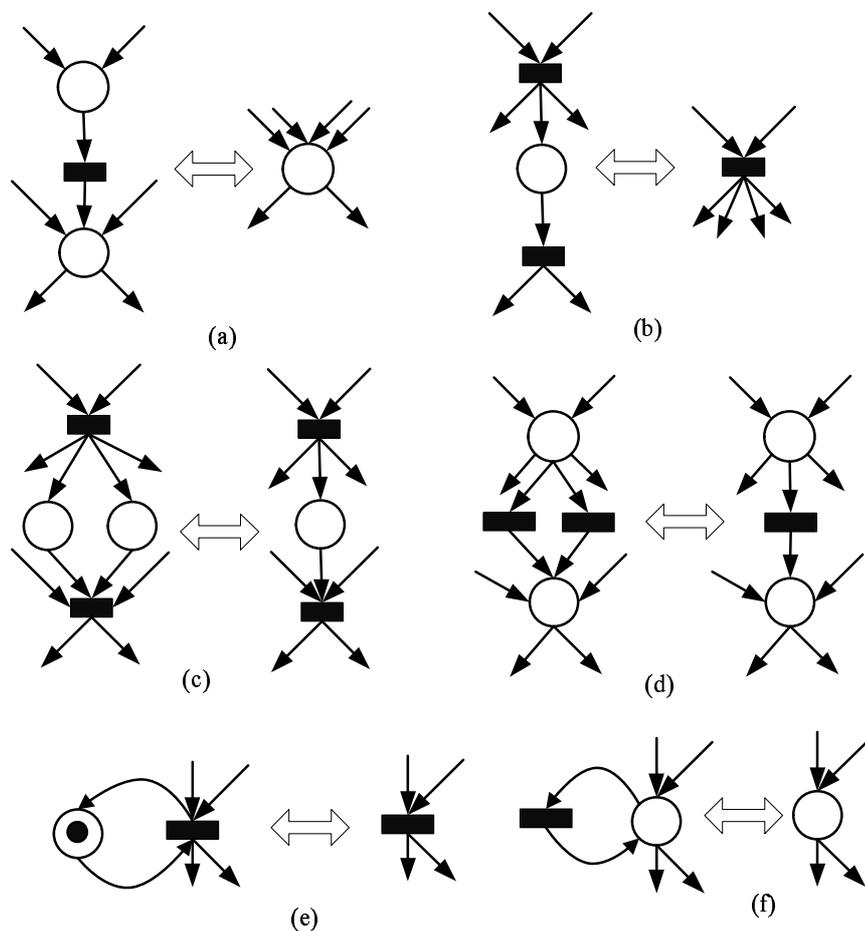


Figura 3.3: Seis Transformações Preservando Vivacidade, Segurança, e Limitação.

Segundo Murata [Mur89], a maior fraqueza das RDPs é o problema da complexidade, ou seja, modelos de RDP tendem a ser muito grandes para

a análise até mesmo de sistemas de tamanho reduzido. Portanto, é muito importante desenvolver métodos de transformação que permitam reduções hierárquicas ou graduais e preservem as propriedades do sistema a serem analisadas.

### 3.7 Classes de RdPs

Existem várias classificações para RdPs na literatura. A escolha da classe de RdP adequada para a modelagem de um sistema é crucial para obtenção de um modelo que seja conciso, simples, e que modele adequadamente o sistema apresentado. A seguinte classificação é baseada em [BC92, MLC96]:

- RdPs elementares.
  - RdPs lugar/transição.
    - RdPs ordinárias.
      - \* Máquinas de estado (*State Machine*).
      - \* Grafo marcado (*Marked Graph*).
      - \* Redes escolha-livre (*Free-Choice*).
- RdPs de alto nível.
  - RdPs Predicado-Evento.
  - RdPs Coloridas.

A primeira classe de RdPs são as *redes elementares*, onde lugares representam valores booleanos, ou seja, um lugar apresenta no máximo uma marca não estruturada.

A segunda classe são as *redes lugar/transição*, onde lugares representam valores inteiros, ou seja, um lugar apresenta um certo número de marcas não estruturadas. Um caso particular das RdPs lugar/transição são as *rede ordinárias*, nas quais o peso dos arcos é unitário.

A terceira classe são as *redes de alto nível*, onde lugares representam valores de alto nível, ou seja, um lugar apresenta um *bag* de marcas estruturadas.

Adicionalmente algumas extensões podem ser consideradas tais como redes temporizadas, redes com arcos inibidores e redes com prioridades. Tais extensões afetam tanto a concisão quanto o poder de modelagem dos modelos adotados.

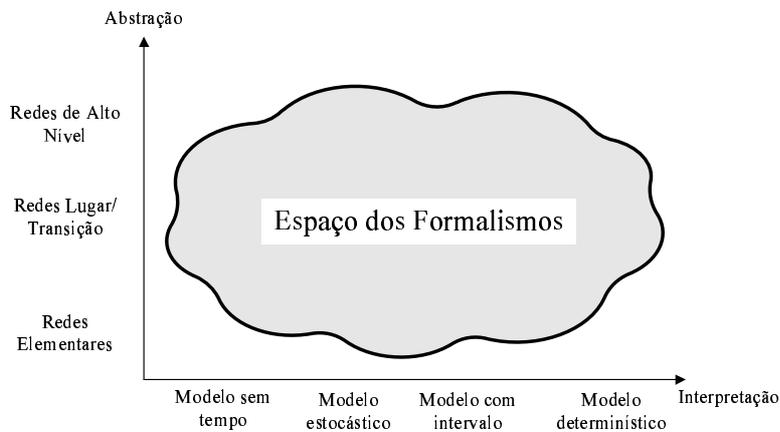


Figura 3.4: Espaço dos Formalismos.

Duas características de um modelo de RdP que as torna tão úteis são os níveis de abstração e as diferentes interpretações [Sil04]. Estas características especificam um espaço de formalismos apropriados para diferentes propósitos e são apresentados na Figura 3.4.

### 3.7.1 RdPs Ordinárias

Uma RdP ordinária é um grafo bipartido no qual lugares de capacidade ilimitada e transições alternam-se conectados por arcos orientados de peso unitário, formando a estrutura da rede.

RdPs ordinárias podem ser colocadas em várias classes. Cada classe tem um certo poder de modelagem e propriedades próprias [DAJ95]. Aqui serão descritas as seguintes sub-classes: grafo marcado, máquina de estado e as redes escolha-livre.

Um *grafo marcado* é uma RdP (ordinária) na qual cada lugar tem exatamente uma transição de entrada e uma transição de saída.

Uma *máquina de estado* é uma RdP (ordinária) na qual cada transição tem exatamente um lugar de entrada e um lugar de saída.

Uma *rede escolha-livre* é uma RdP (ordinária) tal que um arco com origem em um lugar é o único arco de saída (sem escolha) ou, é a única entrada para uma transição.

Uma RdP pura é uma rede que não contém auto-laços (termo encontrado na literatura como *self-loops*), isto é, nenhum lugar é ao mesmo tempo entrada e saída de uma transição.

Uma RdP marcada é uma rede que contém marcas em adição aos ele-

mentos descritos anteriormente. As marcações estão associadas aos lugares e seu fluxo através da rede é regulado pelas transições. Marcações são representadas graficamente por pontos.

A execução de uma RdP ordinária modifica a marcação da rede. A execução é realizada através do disparo de transições habilitadas. Uma transição está *habilitada* quando cada um dos seus lugares de entrada está marcado com pelo menos uma marca. O disparo da transição remove uma marca de cada lugar de entrada e coloca uma marca em cada lugar de saída.

No caso de transições que apresentam arcos de entrada com multiplicidade  $k > 1$ , a transição  $t_i$  irá disparar quando cada lugar de entrada tiver pelo menos um número de marcas igual a multiplicidade do arco que o conecta a  $t_i$ .

### 3.7.2 Extensões de RdPs

As extensões das RdPs fazem referência a modelos que incorporam regras adicionais de funcionalidade para aumentar seu poder de funcionalidade. São consideradas três subclasses [DA85].

- Modelos equivalentes às máquinas de *Turing* – correspondem às RdPs com arcos inibidores (que inibem o disparo de uma transição caso o lugar que é pre-condição da transição esteja marcado) e redes com prioridade.
- Modelos para sistemas contínuos e híbridos – correspondem às RdPs contínuas e RdPs híbridas.
- Modelos que evoluem em função do tempo ou de eventos externos – correspondem às RdPs sincronizadas, RdPs temporizadas e RdPs estocásticas.

#### RdPs com Arco Inibidor

Uma das principais limitações das RdPs é a impossibilidade da realização do teste *de zero* marcas em um lugar com capacidade ilimitada. A extensão mais simples que possibilita a realização do teste de zero marcas é a rede com arco inibidor. Esta extensão inclui um arco diferenciado dos demais, denominado arco inibidor, que inibe o disparo de uma transição caso o lugar que é pré-condição à transição esteja marcado [Pet81] [Bra83]. Graficamente, o arco inibidor é um arco de um lugar para uma transição (pré-condição), terminando com um pequeno círculo.

A regra de disparo das transições que tem arcos inibidores como entrada é alterada. Tais transições só estarão aptas a serem disparadas caso os lugares de entrada, relacionados às transições pelos arcos inibidores, não contenham marcas [MLC96].

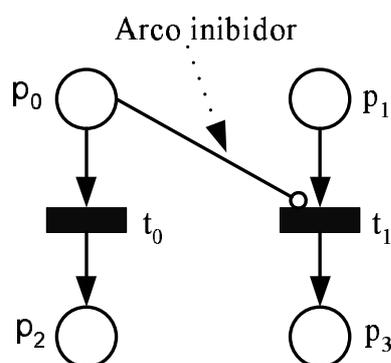


Figura 3.5: Rede com Arco Inibidor

A Figura 3.5 mostra uma rede com arco inibidor, possibilitando o teste de zero marcas no lugar  $p_0$ , ou seja, a transição  $t_1$  só estará habilitada quando não houver marcas em  $p_0$ .

### Redes de Petri Coloridas – CPN

A utilização de RdPs ordinárias para a modelagem de sistemas reais tem sido dificultada porque os modelos gerados são geralmente grandes, dificultando o seu entendimento. Nas redes ordinárias existe apenas um tipo de marca, o que não permite a diferenciação de recursos em um lugar, sendo necessários lugares distintos para expressar recursos semelhantes.

O principal objetivo das CPNs é a redução do tamanho do modelo, permitindo que marcas individualizadas (cores) representem processos distintos ou recursos em uma mesma sub-rede.

Inicialmente as marcas das CPNs eram identificadas por cores. Atualmente, as marcas podem ser estruturas de dados complexas e é possível efetuar operações sobre estes dados.

Jensen [Jen93] aponta três razões para construir modelos CPN:

- Um modelo CPN é uma *descrição* do sistema modelado e pode ser usado como uma especificação (do sistema que se quer produzir) ou como uma apresentação (do sistema que se quer explicar para outras pessoas).

- O comportamento de uma CPN pode ser *analisado* por simulação ou através de métodos de análise mais formais.
- O processo de modelagem utilizando CPN, que envolve a descrição do sistema e possibilita a análise do mesmo, dá ao modelador um maior *entendimento* do sistema modelado.

Jensen [Jen93] também lista alguma das propriedades que tornam as CPNs uma linguagem valiosa para o projeto, especificação e análise de muitos tipos de sistemas diferentes.

- Têm uma representação gráfica.
- Têm uma semântica bem definida que determina sem ambigüidades o comportamento de uma CPN.
- São gerais e podem ser usadas para descrever uma grande variedade de sistemas diferentes.
- Têm um conjunto reduzido, porém poderoso, de primitivas.
- Têm uma descrição explícita tanto de estados quanto de ações.
- Oferecem uma representação hierárquica.
- Integram a descrição do controle e sincronização com a descrição da manipulação de dados.
- Podem ser estendidas com o conceito de tempo.
- São estáveis diante de pequenas mudanças no sistema modelado.
- Oferecem simulação interativa na qual os resultados são apresentados diretamente no diagrama CPN.
- Têm um grande número de métodos de análise formal através dos quais as propriedades das CPN podem ser provadas.
- Têm ferramentas de *software* que suportam desenho, simulação e análise formal.

Uma introdução às CPNs é encontrada em [Jen93].

### Extensões Temporizadas

RdPs é um formalismo que possibilita a expressão do comportamento dinâmico de sistemas que tenham atividades concorrentes, assíncronas e não-determinísticas. O conceito de tempo não está explícito na definição original de RdPs. Isto motivou Ramchandani [Ram74] a desenvolver as RdPs temporizadas. A introdução da especificação de tempo é crucial quando são considerados problemas que envolvem a análise de desempenho, escalonamento, controle de tempo real, por exemplo. Para tanto, foram definidas várias extensões temporizadas para as RdPs.

Os tempos de um modelo podem estar associados aos lugares, às marcas (*Token-Timed Models*) e transições (*Transition-Timed Petri Models*) [Ram74]. Existe também uma abordagem que associa o tempo às transições pelo uso de intervalos (*Transition-Time Petri Nets*) [MF76].

Existem três maneiras de associar um valor ao tempo que o evento leva para ocorrer: determinística, estocástica e por intervalo. Se o tempo é determinístico, tem-se uma RdP determinística. Os primeiros trabalhos sobre TPN utilizavam atrasos de tempo determinísticos. Nas RdPs estocásticas [Mol81, Nat80, AM89], a duração de um evento é determinada por uma função de distribuição de probabilidade. A distribuição de probabilidade normalmente adotada é a distribuição exponencial, todavia, é possível utilizar outras funções de distribuição de probabilidade. Nas RdPs com intervalo, as durações definidas na rede estão entre um valor máximo e um valor mínimo, e o valor utilizado para disparar ou habilitar uma transição é tomado neste intervalo.

As redes estocásticas [Mol81, Nat80, AM89, Mol82], ou SPNs, permitem aplicar modelos de RdP temporizados em sistemas não-determinísticos (devido, por exemplo, a tempos randômicos de atraso e falhas) [DAJ95, p. 159]. Formalmente, pode-se dizer que uma SPN de tempo contínuo é temporizada nas transições e possui um conjunto de atrasos de tempo exponencialmente distribuídos, com taxa de disparo  $\lambda_i > 0$ .

As SPNs vivas e limitadas são isomorfas às cadeias de *Markov* homogêneas [Mol81, Mol82]. A principal vantagem da SPN é que ela simplifica a geração da Cadeia de *Markov* e provê uma representação mais compacta e lógica do sistema. Isto abre uma nova possibilidade de análise de desempenho de sistemas de produção: a modelagem do sistema é realizada com RdPs. Então, com base na marcação inicial, a árvore de cobertura pode ser gerada e uma cadeia de *Markov* equivalente pode ser obtida e analisada.

RdPs estocásticas generalizadas (GSPN) [AMBC<sup>+</sup>91, FSS<sup>+</sup>04a] apresentam transições imediatas (que disparam com tempo zero) e estocásticas.

As transições temporizadas têm tempo de disparo distribuído exponencialmente e disparam, em média, a  $1/\lambda$  unidades de tempo depois de habilitadas. É comum dizer que estas transições têm taxa de disparo  $\lambda$ . As transições imediatas disparam em tempo zero.

GSPNs permitem a utilização de arcos inibidores (utilizados para evitar que certas transições disparem quando alguma condição for verdadeira), funções de prioridade e distribuições de probabilidade discretas (do inglês, *random switches*). A função de prioridade é definida para as marcações que apresentam habilitadas ao mesmo tempo tanto transições estocásticas quanto transições imediatas (normalmente, as transições imediatas têm maior prioridade). As distribuições de probabilidade discretas são utilizadas para resolver conflitos entre duas ou mais transições imediatas. Uma descrição mais detalhada das GSPNs pode ser encontrada em [DAJ95, pp. 161-168].

RdPs temporizadas são a base para a análise de desempenho. Elas têm sido muito utilizadas para a modelagem e análise de desempenho de sistemas nas mais diversas áreas de aplicação [SMS03, FSS<sup>+</sup>04a, FSS<sup>+</sup>04b].

### Redes Hierárquicas

As redes hierárquicas proporcionam uma descrição estruturada dos sistemas, possibilitando sua especificação em níveis de abstração que pode ser refinada quando necessário. Nessa classe de rede, os lugares e as transições do níveis superiores podem ser refinados, ou seja, podem ser subredes da rede global. Estas subredes fornecem uma descrição mais detalhada do lugar ou da transição do nível superior. A interface entre um lugar ou transição de mais alto nível e a subrede correspondente é efetuada através de transições ou lugares denominados *port-transitions* e *port-places*, respectivamente [MLC96, JHS90].

O primeiro morfismo utilizado nesta abordagem para a obtenção de hierarquia em RdPs é a substituição de lugares. Cada lugar de alto nível designa uma página (denominada sub-página). Um lugar de alto nível é visto como uma caixa preta e a sub-página correspondente representa o detalhamento desse lugar. O lugar de alto nível e a página em que está inserido são, para a sub-página, denominados de super-página.

Do ponto de vista teórico, a hierarquia é apenas uma conveniência gráfica que não adiciona nenhum poder computacional. Todavia, o seu uso facilita a modelagem de sistemas de grandes dimensões. A Figura 3.6(a) apresenta uma RdP na qual temos um lugar de alto nível. A Figura 3.6(b) apresenta uma página que representa o refinamento do lugar  $p_0$ .

Os seguintes passos são necessários para a representação clara da

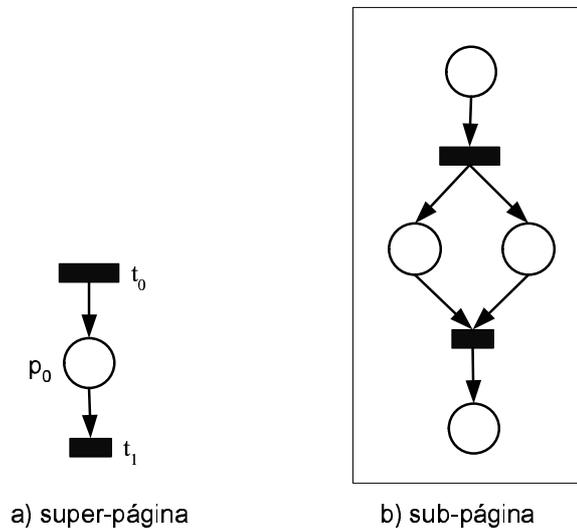


Figura 3.6: Exemplo de Hierarquia em Redes de Petri

semântica de substituição de lugares, que possibilitam a representação não-hierárquica equivalente da rede [JHS90]:

- Remove-se o lugar de alto nível.
- Insere-se a sub-página que representa o refinamento do lugar.
- Fundem-se os *port-transitions* da rede de mais alto nível e da sub-página.

As transições também podem ser substituídas por redes que representem o seu refinamento (páginas), de forma semelhante à substituição de lugares.

A Figura 3.7 mostra algumas das extensões de RdPs mais comuns. A partir do modelo elementar de RdP, pela adição de capacidade aos arcos e transições, chega-se às redes lugar/transição. A partir das redes lugar/transição, pela individualização das marcas (marcas correspondem a tipos de dados estruturados), chega-se às redes com marcas individualizadas tais como as redes PrT e as redes CPN. As RdPs acrescidas de tempo são obtidas através das redes lugar/transição pela introdução de informações explícitas de tempo. A informação de tempo pode ser determinística, levando às redes determinísticas (ex: TPN) ou estocástica, na qual a informação temporal associada ao elemento da rede apresenta é uma variável aleatória.

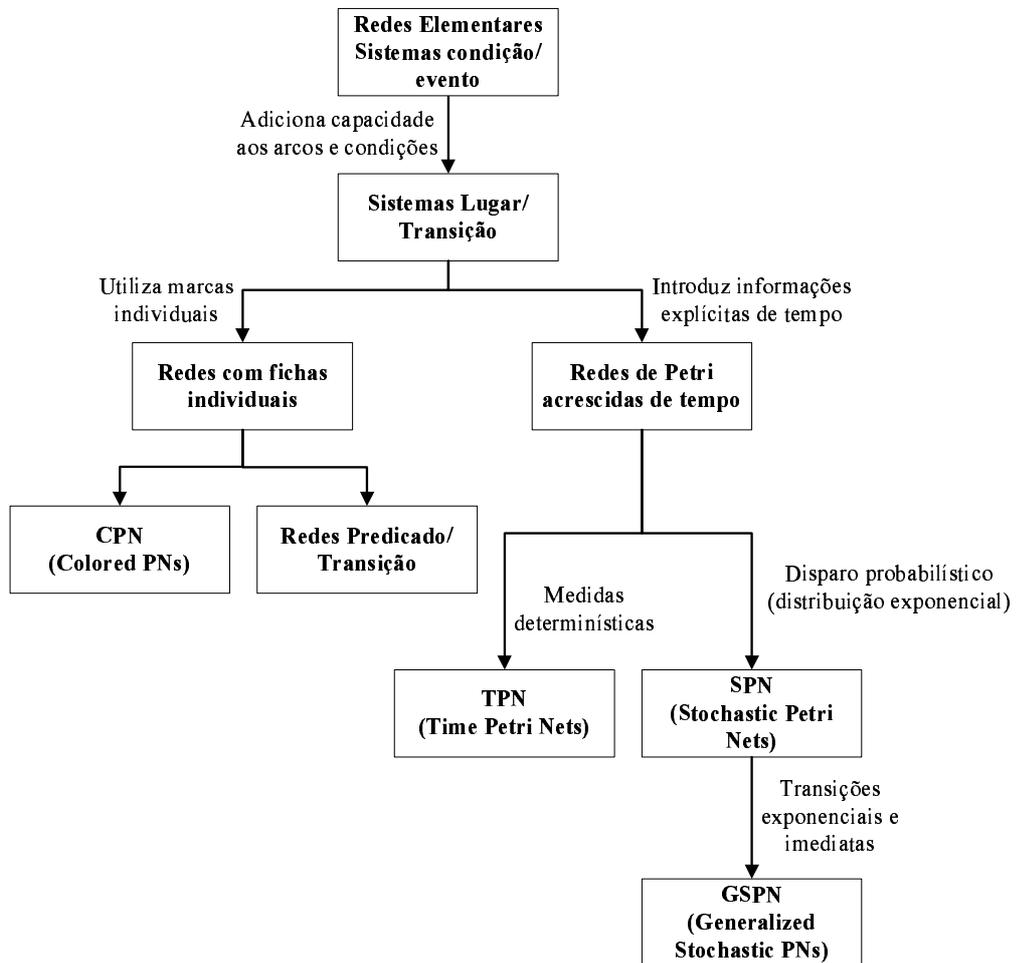


Figura 3.7: Extensões de RdPs

Conforme já salientado, existem muitas outras extensões de RdP tais como redes com arco inibidor, redes hierárquicas, redes orientadas a objetos, etc.

### 3.8 Considerações Finais

RdPs constituem-se em uma poderosa família de técnicas de modelagem de sistemas concorrentes, provendo um grande conjunto de ferramentas de análise. Várias das características das RdPs as tornam particularmente úteis para a modelagem e análise de sistemas de produção.

Os modelos iniciais de RdPs não incluem a noção de tempo e ob-

jetivam modelar apenas o comportamento lógico do sistema através da descrição das relações causais existentes entre os eventos. A introdução da especificação de tempo é crucial quando são considerados problemas que envolvem a análise de desempenho, escalonamento e controle de tempo real, por exemplo. Para tanto, foram definidas várias extensões temporizadas para as RdPs.

Para sistemas de produção reais, que são usualmente de grandes dimensões, os modelos RdPs gerados, utilizando-se RdPs ordinárias, apresentam um grande número de elementos, tornando a sua análise difícil ou mesmo impossível. Alguns tipos de RdP, tais como as RdPs hierárquicas, permitem uma representação mais compacta do sistema modelado. As RdPs hierárquicas são de particular importância por servirem de base para a formalização dos modelos apresentados neste trabalho.



# Capítulo 4

## Modelagem de Sistemas de Produção

*Neste capítulo são apresentados os modelos matemáticos utilizados na modelagem de sistemas de produção. São definidos os modelos para a estrutura do produto e para os recursos produtivos. A partir destes dois modelos é gerado o modelo do roteiro de produção. A junção de qualquer um dos modelos anteriores com os modelos de entrada, saída e fechamento, fornece o modelo do sistema de produção. O processo de modelagem inicia-se com um modelo de alto nível que é sucessivamente refinado até a geração do modelo de RdP final.*

### 4.1 Modelo de um Sistema

Um modelo é definido como a representação simplificada de um sistema real com o objetivo de estudo deste sistema. Um modelo pode ser classificado de várias formas [Col01]:

- Modelo matemático – usa notações simbólicas em termos de lógica e equações matemáticas para representar um sistema. Ex: RdPs podem ser utilizadas como um modelo matemático para a modelagem de sistemas de manufatura.
- Modelo físico – reproduz as características construtivas do sistema real em uma dada escala geométrica, observando as mesmas características físicas e estruturais do sistema real. Ex: protótipo de aviões, em menor escala, testados em túneis de vento.

- Modelo estático – representação de um sistema em um ponto particular do tempo. Ex: uma fotografia tirada em um dado instante.
- Modelo dinâmico – representação de sistemas que mudam com a influência do tempo. Ex: simulação da movimentação de produtos em uma linha de produção.
- Modelo determinístico – modelo no qual os componentes não apresentam uma probabilidade associada. Nos modelos determinísticos, uma vez determinada a entrada do sistema, a saída fica automaticamente determinada. Ex: modelo de uma linha de produção de automóveis automatizada.
- Modelo estocástico – modelo de simulação que possui probabilidade associada, ou seja, comportamento aleatório. Ex: modelo de uma máquina com probabilidade de falha.

Pode-se especificar, para um determinado sistema, vários modelos distintos, em função do aspecto que se deseja analisar. Uma das vantagens das RdPs é o fato de o mesmo formalismo poder ser utilizado para especificar, por exemplo, um modelo matemático, um modelo dinâmico, um modelo determinístico e ainda um modelo estocástico do sistema.

## 4.2 Modelagem Usando RdPs

RdPs são muito úteis na modelagem, análise, simulação e controle de sistemas de produção pelos seguintes motivos [DAJ95]:

- Capacidade de representar de forma natural a sincronização de processos, eventos concorrentes, causalidade, compartilhamento de recursos, conflitos, etc.
- Permite uma representação gráfica dos sistemas.
- Conflitos e *buffers* podem ser modelados de forma simples e eficiente.
- *Deadlocks* no sistema podem ser identificados.
- Como uma ferramenta formal, um modelo de RdP pode ser descrito matematicamente.

- Várias extensões das RdPs, tais como RdPs temporizadas, RdPs estocásticas, RdPs coloridas e RdPs lugar-transição possibilitam a análise qualitativa e quantitativa dos sistemas, permitindo a análise da utilização de recursos, efeito de falhas, *throughput*, etc.

Um processo comumente adotado para a modelagem de sistemas de produção usando RdPs consiste nos seguintes passos:

- Identificação dos recursos, operações e atividades do sistema.
- Estabelecimento da seqüência de atividades em cada processo.
- Representação das atividades geralmente através de *transições*.
- Representação de recursos geralmente através de *lugares*.
- Especificação da marcação inicial.

Na seqüência serão apresentados e formalmente definidos os elementos de modelagem utilizados neste trabalho.

### 4.3 Elementos de Modelagem

Para a modelagem dos sistemas de manufatura, os seguintes elementos necessitam ser definidos:

**Produto:** o elemento que se está manufaturando ou comprando. Um produto pode ser composto de outros produtos.

**Estrutura do Produto:** descreve as relações pai-filho entre os elementos que compõem o produto, além de atributos adicionais tais como o *lead time* para obtenção/manufatura de cada item que compõe o produto, e a quantidade de um elementos necessária para formar o produto final.

A partir das definições de Produto e Estrutura do Produto, o Modelo do Produto pode ser definido. Este modelo permite, por exemplo, o cálculo da necessidade de materiais para a manufatura de um dado produto, tempos mínimos de produção, além do escalonamento das atividades no tempo. Sua limitação resulta do fato de o mesmo não levar em conta as restrições de capacidade do sistema de manufatura. Portanto, para que se leve em consideração tais restrições, é necessário que se modele também os recursos produtivos e suas restrições produtivas:

**Recursos:** Todo e qualquer bloco funcional que possa ser utilizado no processo de manufatura, como por exemplo, máquinas e pessoas. Os recursos apresentam parâmetros cujos valores variam em função da atividade que realizam, tais como tempo de operação, tempo de recuperação em caso de falha, tempo médio entre falhas, etc.

**Atividades:** Operações realizadas pelos recursos.

O próximo passo consiste em determinar como cada produto será manufaturado, ou seja, definir para um dado produto os recursos utilizados e a as operações necessárias para a sua produção. Para tanto, é necessário definir:

**Planta de Produção:** representa os recursos alocados à realização de um dado processo de manufatura.

**Roteiro de Produção:** seqüência de operações efetuadas pelos recursos para a realização de um dado processo de manufatura.

Os passos básicos adotados no processo de modelagem são os seguintes (veja Figura 4.1):

- Definir o modelo de alto nível para o produto/recurso.
- Definir o modelo intermediário (também denominado de modelo hierárquico), usando RdPs, a partir do modelo de alto nível.
- Refinar os lugares do modelo intermediário.
- Gerar o modelo de RdP final.



Figura 4.1: Etapas do Processo de Modelagem

Para a representação da estrutura do produto e dos recursos produtivos são utilizados vértices e arcos de interconexão. Cada vértice tem propriedades associadas bem como um atributo *rede* que determina a sua representação de baixo nível, usando RdPs.

Na seqüência, os modelos da estrutura do produto, recursos e roteiro de produção serão formalmente definidos. Antes de formalizar os modelos propostos, é importante introduzir o conceito de alfabeto, que corresponde a um conjunto finito não vazio de símbolos, e é representado pela letra grega  $\Sigma$  (veja Definição A.16).

## 4.4 Estrutura do Produto (EP)

A representação gráfica de alto nível da estrutura do produto utiliza vértices e arcos de interconexão. Os vértices representam graficamente os elementos do produto. Os arcos, por sua vez, representam graficamente as relações pai-filho entre elementos do produto.

Por convenção, os itens de um produto obedecem a uma hierarquia vertical na qual elementos acima correspondem a itens pais e elementos imediatamente abaixo, representam itens filhos. Itens pais e itens filhos são ligados por arcos direcionados que partem dos itens filhos em direção aos itens pais. A representação da estrutura de um produto inicia-se com o vértice que o representa e parte verticalmente acrescentando os vértices dos seus itens constituintes. Um item filho pode ser um produto, o que torna a definição de estrutura do produto recursiva.

**Definição 4.1** (Modelo do Produto). *Define-se um produto como um grafo direcionado conectado e acíclico representado pela tupla  $Prod = \langle V, A \rangle$  onde  $V = \{v_0, v_1, \dots, v_n\}$  é um conjunto finito de vértices representando partes ou sub-produtos e  $A \subseteq V \times V$  corresponde às relações pai-filho, entre itens que são componentes de um mesmo produto. O par  $(v_i, v_j) \in A$  é um par ordenado onde  $v_i$  é um vértice que representa o item pai e  $v_j$  é um vértice que representa o item filho.*

Vários atributos tais como *rede*, *quantidade*, *lead-time*, e *estoque*, entre outros, podem ser associados aos vértices. Dentre estes atributos, merecem ser destacados a *quantidade* e a *rede*. A *quantidade* indica quantas unidades do subproduto representado pelo vértice são necessárias na manufatura do produto final. O atributo *rede* determina a RdP que será utilizada para refinar o vértice associado. Os demais atributos são utilizados para valorar as inscrições associadas aos lugares e transições da RdP usada no refinamento. Os atributos de um vértice serão formalmente definidos na seqüência.

**Definição 4.2** (Conjunto de Variáveis Tipadas). *Define-se  $Var$  como um conjunto de variáveis tipadas no qual cada variável tem um nome  $x$  e um tipo  $D$  expresso pela seguinte notação,  $x : D$ . Todos os nomes de variáveis de  $Var$  são distintos.  $Var = \{x_1 : D_1, x_2 : D_2, \dots, x_n : D_n\}$ .*

**Definição 4.3** (Substituição). *Uma substituição  $\{x_1/a_1, x_2/a_2, \dots, x_n/a_n\}$  é uma atribuição de valores  $a_i \in D_i$  para variáveis  $x_i : D_i$  do conjunto  $Var$  [DHP+93].*

**Definição 4.4** (Atributos de um Vértice). Cada vértice  $v \in V$  tem um conjunto de atributos, que é um subconjunto de  $Var$ . A função  $ATTRV : V \rightarrow Var$  retorna os atributos associados a um vértice.

**Definição 4.5** (Valor do Atributo de um Vértice). O valor de um atributo  $x_i$  é obtido através de uma função da forma  $attrv'_{x_i} : Var \rightarrow D_i$ , que aplica a substituição  $[x_i : D_i/a_i : D_i]$  em  $Var$ . Para simplificar a notação matemática, define-se a função  $attrv_{x_i} : V \rightarrow D_i$ , com domínio em  $V$ , tal que  $attrv_{x_i}(v) = attrv'_{x_i}(ATTRV(v))$ .

O grafo da EP deve satisfazer as seguintes propriedades:

1. Cada vértice  $v$  tem um conjunto de atributos, que é um subconjunto de  $Var$ . A função  $ATTRV : V \rightarrow Var$  retorna os atributos associados a um vértice.
2. Valores podem ser associados aos atributos de um vértice (um atributo é uma variável com um tipo associado). A função  $attrv_{x_i}(v)$  retorna o valor do atributo  $x_i$  que está associado ao vértice  $v$ .
3. Dois atributos obrigatórios são a *quantidade* e a *rede*, os quais podem ser obtidos como segue:

$attrv_{qt} : V \rightarrow \mathbb{N}^+$ , quantidade associada a um vértice. Quando não especificado, seu valor padrão é 1.

$attrv_{rede} : V \rightarrow \Sigma$ , identifica a RdP a ser utilizada para refinar um vértice.

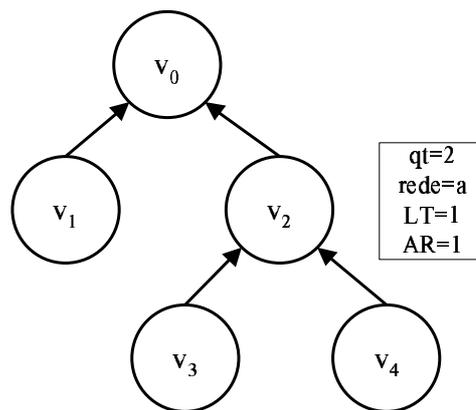


Figura 4.2: Modelo de Alto Nível da Estrutura do Produto

A Figura 4.2 mostra um exemplo de representação da estrutura de um produto. Os vértices representam partes do produto. O vértice  $v_0$  corresponde ao produto final, que é composto de dois itens representados pelos vértices  $v_1$  e  $v_2$ . Por sua vez, o item representado pelo vértice  $v_2$  é composto dos itens representados pelos vértices  $v_3$  e  $v_4$ .

Cada vértice tem um conjunto de atributos definido pela função  $ATTRV(v)$ . Na Figura 4.2 são mostrados os atributos associados ao vértice  $v_2$ . O atributo *quantidade* indica a quantidade de itens  $v_2$  necessários para a manufatura do produto final representado por  $v_0$ . O atributo *rede* indica a RdP que será utilizada para refinar este vértice. Além destes atributos obrigatórios, são mostrados mais outros dois atributos, *LT* e *AR*, cujo significado semântico é dado pelo modelador do sistema. Outros atributos podem ser facilmente associados aos vértices.

Este produto é representado matematicamente da seguinte forma:

$Prod = \langle V, A \rangle$  onde

$V = \{v_0, v_1, v_2, v_3, v_4\}$ ,

$A = \{(v_0, v_1), (v_0, v_2), (v_2, v_3), (v_2, v_4)\}$ .

Os atributos para o vértice  $v_2 \in V$  mostrado na figura são:

$ATTRV(v_2) = \{qt : \mathbb{N}, rede : \Sigma, LT : \mathbb{N}, AR : \mathbb{N}\}$ ,

$attrv_{qt}(v_2) = 2$ ,

$attrv_{LT}(v_2) = 1$ ,

$attrv_{AR}(v_2) = 1$ ,

$attrv_{rede}(v_2) = a$ , onde  $a \in \Sigma$  é um identificador da RdP que será utilizada para refinar  $v_2$ .

#### 4.4.1 Modelo Intermediário da EP

O processo de obtenção do modelo intermediário da estrutura do produto é simples. Os vértices são transformados em lugares. Entre cada lugar que representa um item pai e os lugares que representam itens filhos é colocada uma transição. Arcos direcionados com origem nos lugares correspondentes aos itens filhos são utilizados para conectá-los a esta transição. Um arco direcionado parte desta transição e chega ao lugar que representa o item pai. A quantidade associada a um item filho é representada pelo peso do(s) arco(s) de saída do lugar que o representa. O peso do arco de saída das transições é um.

Um “lugar” do modelo intermediário tem seu comportamento em função da rede que o refina, e portanto, pode não se comportar como um lugar convencional de RdPs. Por exemplo, em uma RdP comum, marcas são colocadas em um lugar e posteriormente removidas. No caso de um

lugar do modelo intermediário, este comportamento pode não se verificar caso a rede que o refina, por exemplo, acumule marcas.

A transição adicionada deve ser da mesma classe das transições utilizadas para refinar os lugares da rede para gerar um modelo de RdP coerente. Valores padrão são automaticamente associados aos atributos da transição (tipo, regra de disparo, tempo, duração, etc.).

O modelo intermediário, obtido através das transformações aplicadas ao modelo de alto nível, é definido matematicamente na seqüência.  $P_{prod}$ ,  $T_{prod}$ ,  $R_{prod}$  e  $W_{prod}$  correspondem, respectivamente, ao conjunto de lugares, ao conjunto de transições, ao conjunto de relações, e ao vetor de pesos dos arcos do modelo intermediário. A função  $dom$  retorna o domínio de uma relação binária (veja Definição A.10).

A função bijetora  $mapP : V \rightarrow P$  mapeia vértices do modelo de alto nível em lugares do modelo intermediário. A função  $mapP$  é bijetora e portanto admite uma função inversa. A função de mapeamento  $mapT : P \rightarrow T$  mapeia lugares do modelo intermediário em transições.

**Definição 4.6** (Modelo Intermediário do Produto). *Seja  $N = \langle P, T, R, W \rangle$  uma RdP e  $Prod = \langle V, A \rangle$  a estrutura de um produto. Define-se o modelo intermediário do produto através da tupla  $N_{prod} = \langle P_{prod}, T_{prod}, R_{prod}, W_{prod} \rangle$  onde*

$$\begin{aligned} P_{prod} &= \{mapP(v_i) \mid v_i \in V\} \\ T_{prod} &= \{t_i \in T \mid v_i \in dom(A)\} \\ R_{prod} &= \{(p_j, t), (t, p_i) \mid (v_i, v_j) \in A, \\ &\quad p_i = mapP(v_i), p_j = mapP(v_j), t = mapT(p_i)\} \\ W_{prod} &= \begin{cases} w_{prod}(p_i, t) = attrv_{qt}(v_i), \forall p_i \in P_{prod} \\ w_{prod}(t_i, p) = 1, \forall t_i \in T_{prod} \end{cases} \\ \text{Sendo} \\ mapP(v_i) &= p_i, mapT(p_i) = t_i. \end{aligned}$$

Cada vértice do modelo de alto nível tem um conjunto de atributos, e existe uma correspondência biunívoca entre os vértices do modelo de alto nível e os lugares do modelo intermediário, dada pela função  $mapP$ . Por definição, Os lugares do modelo intermediário herdam os atributos associados aos vértices correspondentes do modelo de alto nível. Os atributos de um lugar são formalmente definidos por:

**Definição 4.7** (Atributos de um Lugar). *Cada lugar  $p \in P$  tem um conjunto de atributos, que é um subconjunto de  $Var$ . A função  $ATTRP : P \rightarrow Var$  retorna os atributos associados a um lugar.  $ATTRP(p) = ATTRV(mapP^{-1}(p))$ .*

**Definição 4.8** (Valor do Atributo de um Lugar). *O valor de um atributo  $x_i$  é obtido através de uma função da forma  $attrp_{x_i} : P \rightarrow D_i$ , tal que  $attrp_{x_i}(p) = attrv_{x_i}(mapP^{-1}(p))$ .*

O modelo intermediário deve satisfazer às seguintes propriedades:

1. Cada lugar  $p$  tem um conjunto de atributos que é um subconjunto de  $Var$ . A função  $ATTRP : P \rightarrow Var$  retorna os atributos associados a um lugar.
2. Valores podem ser associados aos atributos de um lugar. A função  $attrp_{x_i}(p)$  retorna o valor do atributo  $x_i$  que está associado ao lugar  $p$ .
3. Dois atributos obrigatórios são *quantidade* e *rede*, os quais podem ser obtidos como segue:

$attrp_{qt} : P \rightarrow \mathbb{N}^+$ , quantidade associada a um lugar. Quando não especificado, seu valor padrão é 1.

$attrp_{rede} : P \rightarrow \Sigma$ , identifica a Rdp a ser utilizada para refinar um lugar.

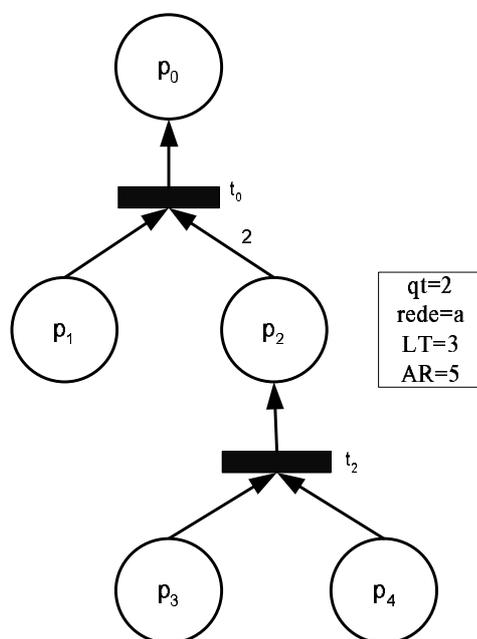


Figura 4.3: Modelo Intermediário do Produto

Um exemplo de representação hierárquica para a estrutura do produto é mostrado na Figura 4.3. Este modelo foi obtido através de transformações aplicadas ao modelo do exemplo mostrado na Figura 4.2.

Matematicamente, temos  $N_{prod} = \langle P_{prod}, T_{prod}, R_{prod}, W_{prod} \rangle$  onde:

$$\begin{aligned} P_{prod} &= \{p_0, p_1, p_2, p_3, p_4\}, \\ T_{prod} &= \{t_i \in T \mid v_i \in \{v_0, v_2\}\} = \{t_0, t_2\}, \\ R_{prod} &= \{(p_1, t_0), (t_0, p_0), (p_2, t_0), (t_0, p_0), (p_3, t_2), (t_2, p_2), (p_4, t_2), (t_2, p_2)\} = \\ &= \{(p_1, t_0), (t_0, p_0), (p_2, t_0), (p_3, t_2), (t_2, p_2), (p_4, t_2)\}, \\ W_{prod} &= \{w_{prod}(p_2, t_0) = 2, w_{prod}(t_0, p_0) = w_{prod}(t_2, p_2) = w_{prod}(p_1, t_0) = \\ &= w_{prod}(p_3, t_2) = w_{prod}(p_4, t_2) = 1\}. \end{aligned}$$

Os atributos para o lugar  $p_2 \in P$  são:

$$\begin{aligned} ATTRP(p_2) &= \{qt : \mathbb{N}, rede : \Sigma, LT : \mathbb{N}, AR : \mathbb{N}\}, \\ attrp_{qt}(p_2) &= 2, \\ attrp_{LT}(p_2) &= 1, \\ attrp_{AR}(p_2) &= 1, \\ attrp_{rede}(p_2) &= a, \text{ onde } a \in \Sigma \text{ é um identificador da RdP que será utilizada} \\ &\text{para refinar } p_2. \end{aligned}$$

#### 4.4.2 Substituição de Atributos na EP

Cada lugar do modelo intermediário para a estrutura do produto apresenta um conjunto de atributos. O atributo *rede* determina a RdP utilizada no seu refinamento.

A RdP utilizada no refinamento do lugar pode apresentar uma série de rótulos associadas aos elementos da rede. Tipicamente, um rótulo pode representar a marcação de um lugar, o peso de um arco, a guarda de uma transição, a inscrição de um arco, etc. [BCvH<sup>+</sup>]. Formalmente:

**Definição 4.9** (Rótulos de uma RdP). *A função ROTULO : P → Var retorna os rótulos associados aos elementos da RdP que refina um lugar do modelo intermediário. ROTULO(p<sub>i</sub>) = {rot<sub>i</sub> : D<sub>j</sub> | rot<sub>i</sub> ∈ attrp<sub>rede</sub>(p<sub>i</sub>)}. Rótulos podem corresponder a marcações em lugares, peso em arcos, tempo em transições, etc.*

Um dos objetivos do processo de modelagem apresentado é esconder a utilização de RdPs. Todavia, é interessante expor alguns destes rótulos para permitir a análise no alto nível do modelo gerado. Por exemplo, um rótulo pode representar a capacidade máxima de um buffer, ou o *throughput* de um sistema. A solução encontrada para esconder a estrutura da rede, mas expor os elementos da rede em cuja análise haja interesse consiste em estabelecer uma correspondência entre os atributos definidos no alto nível e

os rótulos presentes na rede utilizada no refinamento. Um atributo de um lugar e um rótulo da rede que o refina são considerados equivalentes se têm o mesmo nome. Neste caso, o valor do atributo é atribuído ao rótulo.

Esta atribuição deve preservar o domínio dos valores associados aos rótulos. Por exemplo, seja um atributo  $x_i : D_i$  associado ao lugar  $p_i$ ,  $rot_i : D_j$  uma inscrição da RdP que refina o lugar  $p_i$ , e  $[x_i/a_i]$  a atribuição do valor  $a_i$  ao atributo  $x_i$ . Para que a atribuição  $[rot_i/a_i]$  seja possível, é necessário que o atributo  $x_i$  tenha o mesmo nome associado ao rótulo  $rot_i$  e ainda que  $D_i \subseteq D_j$ .

A idéia por trás desta atribuição de valores de atributos em rótulos é simples: imagine um modelo de RdP que representa um *buffer* de capacidade limitada e um rótulo que indica a capacidade máxima deste *buffer*. O modelador do sistema poderia definir um atributo correspondente a esta capacidade máxima no alto nível e analisar, por exemplo, a influência da variação do mesmo no *throughput* do sistema. Deve-se observar que esta análise pode ser feita sem conhecimento prévio da estrutura da RdP que modela o *buffer*.

Os seguintes passos são adotados para mapear os atributos de um lugar do modelo intermediário nos rótulos da rede que o refina.

- Seja  $p_i \in P_{prod}$  um lugar do modelo intermediário a ser refinado.  $ATTRP(p_i)$  fornece o conjunto de atributos associados ao lugar  $p_i$ .  $ROTULO(p_i)$  fornece o conjunto de rótulos associados à rede que refina o lugar  $p_i$ .
- Seja  $x_i : D_i \in ATTRP(p_i)$ . Seja  $x_j : D_j \in ROTULO(p_i)$ . Se  $x_i = x_j$  e  $D_i \subseteq D_j$ , aplicar a substituição  $[x_j/attrp_{x_i}(p_i)]$  em  $ROTULO(p_i)$ .
- Repetir o procedimento para cada  $p_i \in P_{prod}$ .

Para exemplificar, a rede da Figura 4.4 (a) será utilizada para refinar o lugar  $p_2 \in P_{prod}$ . Esta RdP será denominada  $N_2$ . A notação  $M_2^0$  é utilizada para representar a marcação inicial da rede  $N_2$ .

Associado a  $p_2$  estão os seguintes atributos:

$$ATTRP(p_2) = \{qt : \mathbb{N}, rede : \Sigma, LT : \mathbb{N}, AR : \mathbb{N}\}.$$

$$attrp_{rede}(p_2) = N_2, attrp_{qt}(p_2) = 2, attrp_{LT}(p_2) = 3, attrp_{AR}(p_2) = 5.$$

Vamos supor ainda que o lugar  $p_{in,2} \in P_2$  tenha uma inscrição que corresponde à marcação do lugar, rotulada  $AR$  e que o arco  $(p_{in,2}, t)$  tenha uma inscrição que corresponde ao peso do arco, rotulada  $LT$ . Desta forma, temos:

$$ROTULO(p_2) = \{LT : \mathbb{N}, AR : \mathbb{N}\}.$$

$$N_2 = \langle P_2, T_2, R_2, W_2, M_2^0 \rangle \text{ onde:}$$

$$P_2 = \{p_{in,2}, p_{out,2}\},$$

$$\begin{aligned}
T_2 &= \{t\}, \\
R_2 &= \{(p_{in,2}, t), (t, p_{out,2})\}, \\
W_2 &= \{w(p_{in,2}, t) = 3, w(t, p_{out,2}) = 1\}, \\
M_2^0 &= \{m(p_{in,2}) = 5, m(p_{out,2}) = 0\}.
\end{aligned}$$

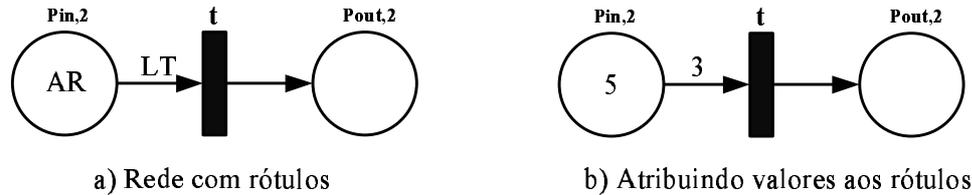


Figura 4.4: Atribuição de Valores aos Rótulos da Rede.

As redes que refinam os demais lugares são obtidas de forma similar. A Figura 4.4 (b) mostra a RdP  $N_2$ , onde as inscrições foram substituídas pelos seus respectivos valores.

### 4.4.3 Refinamento de Lugares da EP

A estrutura do produto permite capturar aspectos estáticos do mesmo, tais como os itens que o constituem e as respectivas quantidades necessárias para a sua manufatura. Para que aspectos comportamentais tais como *lead-time*, pedidos de compra/manufatura, quantidade em estoque, entre outras, possam ser representados, é necessário estender o modelo anterior, substituindo os lugares presentes pela RdP associada.

Conforme mencionado anteriormente, cada lugar (vértice no modelo original) tem associado um atributo *rede* que determina sua representação de baixo nível, usando RdPs. Esta rede pode ser definida arbitrariamente em função do comportamento que se deseja modelar, e deve seguir algumas regras básicas de criação.

Em primeiro lugar, a RdP utilizada deve apresentar um lugar de entrada e um lugar de saída bem definidos. O lugar de entrada da RdP que refina o lugar  $p_i$ , denominado  $p_{in,i}$ , será o local de destino dos arcos incidentes a  $p_i$ . O lugar  $p_{in,i}$  não pode ter arcos incidentes ao mesmo, ou seja, o conjunto *preset* (veja Definição A.12) de  $p_{in,i}$  deve ser vazio. Da mesma forma, o lugar de saída da RdP que refina  $p_i$ , denominado  $p_{out,i}$ , será utilizado como local de partida para os arcos que saem de  $p_i$ . O lugar  $p_{out,i}$  não pode ter arcos com origem no mesmo, ou seja, o conjunto *postset* (Veja Definição A.12) de  $p_{out,i}$  deve ser vazio. Isto pode ser visualizado na Figura 4.5. A caixa preta no interior do círculo representa uma RdP.

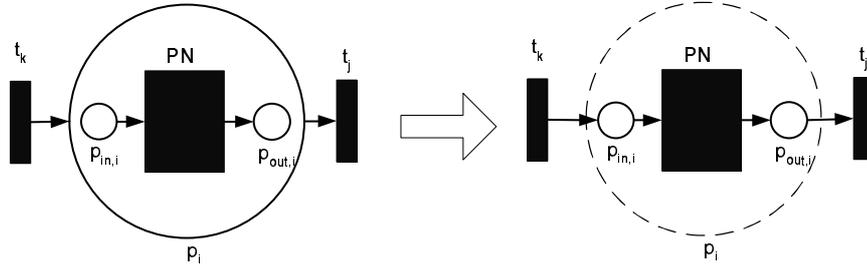


Figura 4.5: Exemplo de Refinamento de um Lugar

**Definição 4.10** (Modelo de Refinamento de um Lugar). Seja  $N_{prod} = \langle P_{prod}, T_{prod}, R_{prod}, W_{prod} \rangle$  a estrutura de um produto. A RdP utilizada para refinar um lugar  $p_i$  de  $P_{prod}$  é dada por  $N_i = \langle P_i, T_i, R_i, W_i, M_i^0 \rangle$ .  $p_{in,i}, p_{out,i} \in P_i$  correspondem respectivamente aos lugares de entrada e saída de  $N_i$ .  $\bullet p_{in,i} = \emptyset$  e  $p_{out,i}^\bullet = \emptyset$ .

Os tipos de RdPs utilizados para refinar os lugares e gerar o modelo final do produto devem ser da mesma classe ou de classes compatíveis, de forma a gerar um modelo final coerente.

Para a obtenção do modelo final do produto, os conjuntos de lugares das redes utilizadas para refinar os lugares de  $P_{prod}$  devem ser disjuntos. A mesma restrição é aplicada aos conjuntos de transições das redes utilizadas para refinar os lugares de  $P_{prod}$ .

O modelo final obtido pode não preservar as propriedades do modelo intermediário (sem *deadlocks*, limitada, etc.). Isto decorre da flexibilidade com que o modelador pode escolher os tipos de RdP utilizados no refinamento dos lugares. Tais propriedades podem ser preservadas, todavia, pela escolha criteriosa dos modelos de refinamento.

#### 4.4.4 Modelo de RdP Final para a EP

Na seqüência é apresentado formalmente o modelo de RdP final para a EP.  $P_{prod}, T_{prod}, R_{prod}, W_{prod}$  e  $M_{prod}^0$  correspondem, respectivamente, ao conjunto de lugares, o conjunto de transições, o conjunto de relações, o vetor de pesos dos arcos, e a marcação inicial do modelo final.  $fin : P_{prod} \rightarrow P_i$  é uma função que mapeia o lugar do modelo intermediário no lugar de entrada da rede que o refina.  $f_{out} : P_{prod} \rightarrow P_i$  é uma função que mapeia o lugar do modelo intermediário no lugar de saída da rede que o refina. Assume-se que todos os lugares  $p_i \in P_{prod}$  serão refinados.

**Definição 4.11** (Modelo Final do Produto). *Seja o modelo intermediário para a estrutura do produto definido por  $N_{prod} = \langle P_{prod}, T_{prod}, R_{prod}, W_{prod} \rangle$ . Seja  $N_i = \langle P_i, T_i, R_i, W_i, M_i^0 \rangle$  a rede utilizada para refinar o lugar  $p_i \in P_{prod}$ . Defina-se o modelo final de RdP para a estrutura do produto através da tupla  $N_{prodf} = \langle P_{prodf}, T_{prodf}, R_{prodf}, W_{prodf}, M_{prodf}^0 \rangle$  onde :*

$$\begin{aligned}
 P_{prodf} &= \bigcup P_i \\
 T_{prodf} &= \bigcup T_i \cup T_{prod} \\
 R_{prodf} &= \bigcup R_i \\
 &\quad \cup \{(fout(p_i), t_j) \mid (p_i, t_j) \in R_{prod}\} \\
 &\quad \cup \{(t_i, fin(p_j)) \mid (t_i, p_j) \in R_{prod}\} \\
 W_{prodf} &= \begin{cases} w_{prodf}(r) = w_i(r) \forall r \in R_i \\ w_{prodf}(fout(p_i), t_j) = w_{prod}(p_i, t_j), \forall (p_i, t_j) \in R_{prod} \\ w_{prodf}(t_k, fin(p_i)) = w_{prod}(t_k, p_i), \forall (t_k, p_i) \in R_{prod} \end{cases} \\
 M_{prodf}^0 &= M_0(p_i), \forall p_i \in P_i \\
 \text{Sendo,} \\
 \bigcap P_i &= \emptyset, \bigcap T_i = \emptyset \\
 fin(p_i) &= p_{in,i}, fout(p_i) = p_{out,i}
 \end{aligned}$$

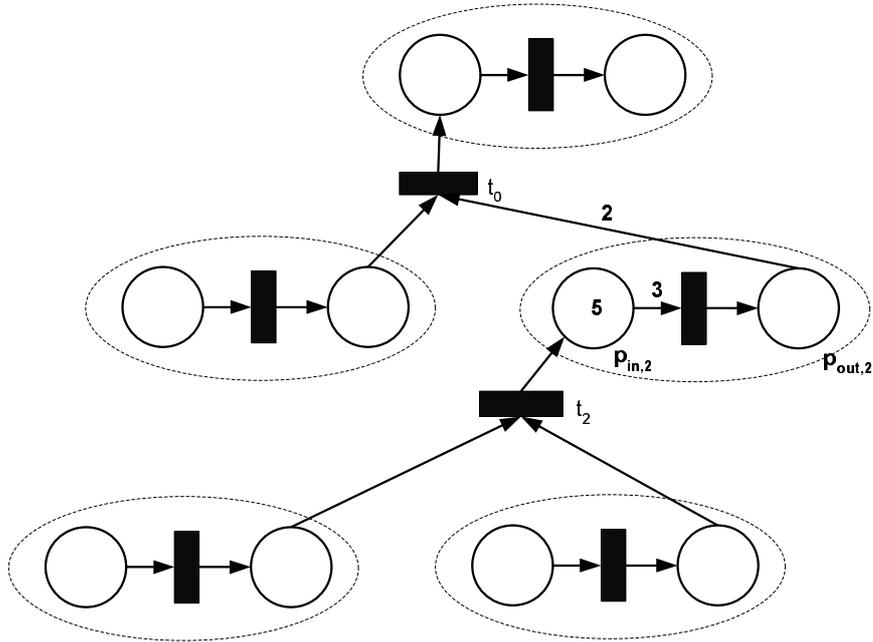


Figura 4.6: Modelo de RdP Final para a Estrutura do Produto

A Figura 4.6 mostra o modelo de RdP final para a estrutura do produto, resultante do refinamento do modelo intermediário mostrado na Fi-

gura 4.3. Uma elipse pontilhada é utilizada para destacar a RdP utilizada para refinar os lugares do modelo intermediário. A RdP utilizada no refinamento dos lugares é mostrada na Figura 4.4.

## 4.5 Recurso Produtivo (RP)

Um recurso produtivo (RP) é um bloco funcional capaz de realizar uma série de operações voltadas para o processo de manufatura. Enquanto a modelagem da estrutura do produto possibilita a análise de aspectos estáticos do sistema sendo modelado, a modelagem do recurso produtivo permite capturar aspectos dinâmicos do sistema, bem como possibilita a análise da capacidade produtiva do sistema modelado.

A exemplo da representação de alto nível da estrutura do produto, a representação de alto nível do recurso produtivo utiliza vértices e arcos de interconexão. Os vértices representam as operações que podem ser executadas pelo recurso produtivo. Os arcos, por sua vez, representam as relações de precedência que podem existir entre as operações.

**Definição 4.12** (Modelo do Recurso). *Define-se um recurso como um grafo direcionado representado pela tupla  $Rec = \langle O, E \rangle$  onde  $O = \{o_0, o_1, \dots, o_n\}$  é um conjunto finito de vértices representando operações e  $E \subseteq O \times O$  corresponde às relações de precedência entre as atividades<sup>1</sup> realizadas pelo recurso.*

Cada vértice tem um conjunto de atributos que serão utilizados em seu processo de refinamento. Dentre estes, destaca-se o atributo *rede* que determina a RdP a ser utilizada no seu refinamento. Os atributos de um vértice serão definidos na seqüência.

**Definição 4.13** (Atributos de uma Operação). *Cada vértice  $o \in O$  tem um conjunto de atributos, que é um subconjunto de  $Var$ . A função  $ATTRO : O \rightarrow Var$  retorna os atributos associados a um vértice.*

**Definição 4.14** (Valor do Atributo de uma Operação). *O valor de um atributo  $x_i$  é obtido através de uma função da forma  $attro'_{x_i} : Var \rightarrow D_i$ , que aplica a substituição  $[x_i : D_i/a_i : D_i]$  em  $Var$ . Para simplificar a notação matemática, define-se a função  $attro_{x_i} : O \rightarrow D_i$ , com domínio em  $O$ , tal que  $attro_{x_i}(o) = attr'o'_{x_i}(ATTRO(o))$ .*

O grafo do RP deve satisfazer as seguintes propriedades:

<sup>1</sup>Neste trabalho os termos operação e atividade são utilizados de forma equivalente.

1. Cada vértice  $o$  tem um conjunto de atributos, que é um subconjunto de  $Var$ . A função  $ATTRO : O \rightarrow Var$  retorna os atributos associados a um vértice.
2. Valores podem ser associados aos atributos de um vértice. A função  $attro_{x_i}(o)$  retorna o valor do atributo  $x_i$  que está associado ao vértice  $o$ .
3. Um atributo obrigatório é a *rede*, definido como segue:  
 $attro_{rede} : O \rightarrow \Sigma$ , identifica a RdP a ser utilizada para refinar um vértice.

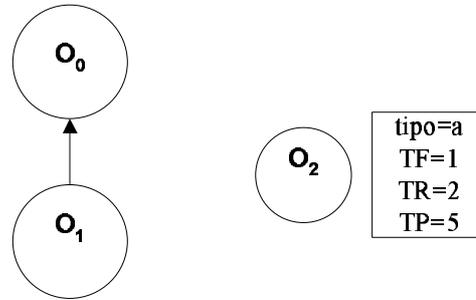


Figura 4.7: Modelo de Alto Nível de um RP

A Figura 4.7 mostra o exemplo de modelo de um recurso produtivo capaz de realizar três operações representadas pelos vértices  $o_0$ ,  $o_1$  e  $o_2$ . Entre as operações representadas pelos vértices  $o_0$  e  $o_1$  existe uma relação de precedência, o que significa que a operação representada por  $o_0$  não pode ser iniciada antes da operação representada por  $o_1$ .

Este recurso pode ser representado matematicamente como segue:

$Rec = \langle O, E \rangle$  onde

$O = \{o_0, o_1, o_2\}$

$E = \{(o_0, o_1)\}$

Os atributos para o vértice  $o_2 \in O$  do exemplo são:

$ATTRO(o_2) = \{rede : \Sigma, TF : \mathbb{N}, TR : \mathbb{N}, TP : \mathbb{N}\}$ ,

$attro_{TF}(o_2) = 1$ ,

$attro_{TR}(o_2) = 2$ ,

$attro_{TP}(o_2) = 5$ ,

$attro_{rede}(o_2) = a$ , onde  $a \in \Sigma$  é um identificador da RdP que será utilizada para refinar  $o_2$ .

### 4.5.1 Modelo Intermediário do RP

O processo de transformação do modelo inicial em uma RdP intermediária é similar ao processo adotado para a obtenção do modelo intermediário para a estrutura do produto. Os vértices são transformados em lugares. Uma transição é colocada entre um lugar que representa uma operação e os lugares que representam suas operações precedentes. Um arco direcionado com origem nos lugares que representam as operações precedentes é utilizado para conectá-los a esta transição. Um arco direcionado parte desta transição e chega ao lugar correspondente à operação que tem precedentes.

Existe, todavia, uma diferença básica: no modelo do recurso produtivo, uma relação entre dois vértices  $o_i$  e  $o_j$  significa que a operação representada por  $o_i$  não pode iniciar antes da operação representada por  $o_j$ . Para capturar este comportamento, é necessário adicionar ao modelo um lugar  $p'_i$  para cada lugar  $p_i$  que representa operações com precedentes. Este lugar será considerado o lugar de início da operação modelada. Um arco direcionado parte de  $p'_i$  e conecta-o à transição que liga  $p_i$  a seus precedentes. Isto leva ao modelo apresentado na Figura 4.8 :

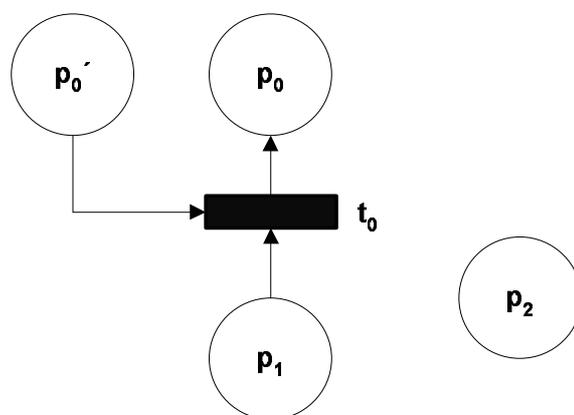


Figura 4.8: Recurso Produtivo Usando RdPs

Conforme salientado na Seção 4.4.1, um “lugar” do modelo intermediário tem seu comportamento em função da rede que o refina, e portanto, pode não se comportar como um lugar convencional de RdPs.

A transição adicionada deve ser da mesma classe das transições utilizadas para refinar os lugares da rede de forma a gerar um modelo de RdP coerente. Valores padrão são automaticamente associados aos atributos da transição (tipo, regra de disparo, tempo, duração, etc.). O mesmo se aplica

ao lugar adicionado.

Na seqüência é apresentado formalmente o modelo intermediário do RP.  $P_{rec}$ ,  $T_{rec}$ ,  $R_{rec}$  e  $W_{rec}$  correspondem, respectivamente, ao conjunto de lugares, o conjunto de transições, o conjunto de relações, e o vetor de pesos dos arcos do modelo intermediário.  $P_{NEW}$  corresponde ao conjunto de lugares  $p'$  que são acrescentados ao modelo para melhor representar a precedência entre as operações.  $R_{NEW}$  corresponde ao conjunto de relações que têm lugares de  $P_{NEW}$  como elemento de partida. A função injetora  $mapP : O \rightarrow P$  mapeia vértices em lugares. A função de mapeamento  $mapT : P \rightarrow T$  mapeia lugares em transições.

**Definição 4.15** (Modelo Intermediário do Recurso). *Seja  $Rec = \langle O, E \rangle$  o modelo de um recurso produtivo e  $N = \langle P, T, R, W \rangle$  uma RdP. Defina-se o modelo intermediário para o recurso produtivo através da tupla  $N_{rec} = \langle P_{rec}, T_{rec}, R_{rec}, W_{rec} \rangle$  onde*

$$\begin{aligned} P_{rec} &= \{mapP(o_i) \mid o_i \in O\} \cup P_{NEW} \\ T_{rec} &= \{t_i \in T \mid o_i \in dom(E)\} \\ R_{rec} &= \{(p_j, t), (t, p_i) \mid (o_i, o_j) \in E, \\ &\quad p_i = mapP(o_i), p_j = mapP(o_j), t = mapT(p_i)\} \cup R_{NEW} \\ W_{rec} &= \{w_{rec}(r) = 1, \forall r \in R_{rec}\} \\ &\text{onde} \\ P_{NEW} &= \{p'_i \mid p_i = mapP(o_i), o_i \in dom(E)\} \\ R_{NEW} &= \{(p'_i, mapT(p_i)) \mid p_i = mapP(o_i), o_i \in dom(E)\} \\ &\quad mapP(o_i) = p_i, mapT(p_i) = t_i \end{aligned}$$

Para o exemplo da Figura 4.8 temos:

$$\begin{aligned} N_{rec} &= \langle P_{rec}, T_{rec}, R_{rec}, W_{rec} \rangle \\ P_{rec} &= \{p_0, p_1, p_2, p'_0\}, \\ T_{rec} &= \{mapT(p_0)\} = \{t_0\}, \\ R_{rec} &= \{(p_1, t_0), (t_0, p_0), (p'_0, t_0)\}, \\ W_{rec} &= \{w_{rec}(p_1, t_0) = w_{rec}(t_0, p_0) = w_{rec}(p'_0, t_0) = 1\}. \end{aligned}$$

### 4.5.2 Substituição de Atributos no RP

Cada lugar do modelo intermediário para o recurso produtivo (que foi criado a partir de um vértice correspondente no modelo de alto nível) apresenta um conjunto de atributos. O atributo *rede* determina a RdP que será utilizada no seu refinamento, da mesma forma que ocorre na estrutura do produto.

A RdP utilizada no refinamento de um lugar do modelo intermediário pode apresentar uma série de rótulos associadas aos elementos da rede (ex: marcação nos lugares, peso nos arcos, tempo nas transições, etc.). Caso haja algum rótulo com o mesmo nome de algum dos atributos associados ao lugar do modelo intermediário, é necessário atribuir o valor do atributo ao valor do referido rótulo.

Esta atribuição deve preservar o domínio dos valores associados aos rótulos. Por exemplo, seja um atributo  $x_i : D_i$  associado ao lugar  $p_i$  do modelo intermediário. Seja  $rot_i : D_j$  um rótulo de algum elemento da RdP que refina o lugar  $p_i$ . Seja ainda  $[x_i/a_i]$  a atribuição do valor  $a_i$  ao atributo  $x_i$ . Para que a atribuição  $[rot_i/a_i]$  seja possível, é necessário que o atributo  $x_i$  tenha o mesmo nome associado ao rótulo  $rot_i$  e ainda que  $D_i \subseteq D_j$ .

É adotada a mesma abordagem apresentada na Seção 4.4.2 para mapear os atributos associados a um lugar do modelo intermediário nos rótulos associadas aos elementos da RdP que o refina.

É importante ressaltar que nem todos os lugares do modelo intermediário de RdP serão refinados. Os lugares  $p'_n$  criados para representar a precedência entre as operações de forma mais adequada não foram derivados diretamente dos vértices do modelo de alto nível. Em decorrência, eles não possuem um conjunto de atributos. Em particular, não possuem o atributo *rede* que poderia ser utilizado para definir a RdP a ser utilizada no seu refinamento.

### 4.5.3 Refinamento de Lugares do RP

O modelo intermediário do recurso produtivo obtido até o momento possui uma capacidade de modelagem limitada. Para que aspectos comportamentais tais como tempo de processamento, tempo de falha, tempo de recuperação, *setup* de máquina, compartilhamento de recursos, entre outros, possam ser representados, é necessário estender o modelo anterior, aplicando um refinamento nos lugares da rede.

O refinamento é realizado através da substituição dos lugares presentes na rede do modelo intermediário pela RdP associada ao lugar.

Cada lugar (vértice no modelo original) tem associado um atributo *rede* que determina sua representação de baixo nível, usando RdPs. Esta rede pode ser definida arbitrariamente em função do comportamento que se deseja modelar, e deve seguir algumas regras básicas de criação.

A RdP utilizada deve apresentar um lugar de entrada e um lugar de saída bem definidos. O lugar de entrada da RdP que refina o lugar  $p_i$  será o local de destino dos arcos incidentes a  $p_i$ . O lugar de saída da RdP utilizada

será utilizado como local de partida para os arcos que saem de  $p_i$ . Isto pode ser visualizado na Figura 4.9. A caixa preta no interior do círculo representa uma RdP.

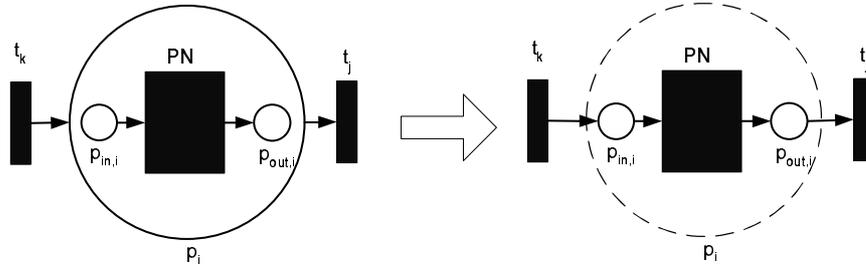


Figura 4.9: Exemplo de Refinamento de um Lugar

**Definição 4.16** (Modelo de Refinamento de um Lugar). Seja  $N_{rec} = \langle P_{rec}, T_{rec}, R_{rec}, W_{rec} \rangle$  um recurso produtivo. A RdP utilizada para refinar um lugar  $p_i \in P_{rec}$  é dada por  $N_i = \langle P_i, T_i, R_i, W_i, M_i^0 \rangle$ . Na rede,  $p_{in,i}, p_{out,i} \in P_i$  correspondem respectivamente aos lugares de entrada e saída de  $N_i$ .  $\bullet p_{in,i} = \emptyset$ .  $p_{out,i}^\bullet = \emptyset$ . Esta rede pode apresentar também um ou mais lugares que representam os recursos disponíveis para a realização da operação representada por  $p_i$ .

Os tipos de RdPs utilizados para refinar os lugares e gerar o modelo final do produto devem ser da mesma classe ou de classes compatíveis, de forma a gerar um modelo final coerente.

Para a obtenção do modelo final do recurso produtivo, os conjuntos de lugares das redes utilizadas para refinar os lugares de  $P_{rec}$  devem ser disjuntos. A mesma restrição é aplicada aos conjuntos de transições das redes utilizadas para refinar os lugares de  $P_{rec}$ .

### Compartilhamento do Recurso

Em um processo usual de produção, recursos podem ser compartilhados para a realização de tarefas distintas. O modelo de refinamento de uma operação pode ter um ou mais lugares que representam os recursos disponíveis para a realização da operação.

No processo de modelagem apresentado, cada recurso produtivo realiza uma ou mais operações. Cada operação está associada com um tipo de RdP que é utilizada no seu refinamento. Para operações que compartilham recursos, pode ser desejável efetuar a fusão dos lugares que representam

recursos compartilhados nas RdPs que refinam as operações. Para tanto, a seguinte função é definida.

**Definição 4.17** (Fusão de Lugares). *Seja  $N = \langle P, T, R, W, M_0 \rangle$  uma RdP e  $p_i, p_j \in P$ . A função  $fus(N, p_i, p_j) = N_{fus}$  efetua a fusão dos lugares  $p_i$  e  $p_j$  na rede  $N$ . O lugar resultante é denominado  $p_j$  (o lugar  $p_i$  é removido do modelo). A RdP resultante  $N_{fus}$  é dada pela tupla  $N_{fus} = \langle P_{fus}, T_{fus}, R_{fus}, W_{fus} \rangle$  definida da seguinte maneira:*

$$\begin{aligned}
 P_{fus} &= P \setminus \{p_i\} \\
 T_{fus} &= T \\
 R_{fus} &= (R \setminus \{(t, p_i) \mid t \in \bullet p_i\}) \setminus \{(p_i, t) \mid t \in p_i^\bullet\} \cup \\
 &\quad \{(t, p_j) \mid t \in \bullet p_i\} \cup \{(p_j, t) \mid t \in p_i^\bullet\} \\
 W_{fus} &= \begin{cases} w_{fus}(t, p_j) = w(t, p_i) \\ w_{fus}(p_j, t) = w(p_i, t) \\ w_{fus}(a, b) = w(a, b), (a \in P \wedge a \neq p_j) \vee (b \in P \wedge b \neq p_j) \end{cases} \\
 M_{fus}^0 &= \{m_{fus}(p) = m(p), \forall p \in P_{fus}\}
 \end{aligned}$$

Dados uma RdP  $N$  e dois lugares  $p_i, p_j \in P$ , a função  $fus$  efetua a fusão dos lugares  $p_i$  e  $p_j$ . Os arcos incidentes a  $p_i$  passam a ser incidentes a  $p_j$ . Os arcos que partem de  $p_i$  passam a ter origem em  $p_j$ . O lugar  $p_i$  é removido de  $P$ .

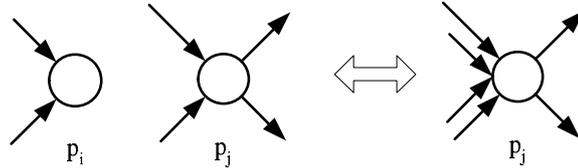


Figura 4.10: Fusão de Lugares

A Figura 4.10 mostra o resultado da aplicação da operação de fusão sobre os lugares  $p_i$  e  $p_j$ . A fim de tornar o modelo de refinamento de um lugar do modelo intermediário mais genérico, os lugares que representam os recursos disponíveis não foram explicitados na Definição 4.16.

#### 4.5.4 Modelo de RdP Final para o RP

Na seqüência é apresentado formalmente o modelo de RdP final para o RP.  $P_{ref}$ ,  $T_{ref}$ ,  $R_{ref}$ ,  $W_{ref}$  e  $M_{ref}^0$  correspondem, respectivamente, ao conjunto de lugares, o conjunto de transições, o conjunto de relações, o

vetor de pesos dos arcos, e a marcação inicial do modelo final.  $fin : P_{rec} \rightarrow P_i$  é uma função que mapeia o lugar do modelo intermediário no lugar de entrada da rede que o refina.  $fout : P_{rec} \rightarrow P_i$  é uma função que mapeia o lugar do modelo intermediário no lugar de saída da rede que o refina.

**Definição 4.18** (Modelo final do Recurso). *Seja o modelo de RdP para o recurso definido por  $N_{rec} = \langle P_{rec}, T_{rec}, R_{rec}, W_{rec} \rangle$ . Seja a RdP utilizada para refinar o lugar  $p_i \in P_{rec}$  dada por  $N_i = \langle P_i, T_i, R_i, W_i, M_i^0 \rangle$ . Defina-se o modelo final de RdP para o recurso produtivo através da tupla  $N_{ref} = \langle P_{ref}, T_{ref}, R_{ref}, W_{ref}, M_{ref}^0 \rangle$  onde:*

$$\begin{aligned}
P_{ref} &= \bigcup P_i \cup \{p'_k \mid p'_k \in P_{rec}\} \\
T_{ref} &= \bigcup T_i \cup T_{rec} \\
R_{ref} &= \bigcup R_i \\
&\quad \cup \{(fout(p_i), t_j) \mid (p_i, t_j) \in R_{rec}\} \\
&\quad \cup \{(t_i, fin(p_j)) \mid (t_i, p_j) \in R_{rec}\} \\
&\quad \cup \{(p'_k, t) \mid (p'_k, t) \in R_{rec}\} \\
W_{ref} &= \begin{cases} w_{ref}(r) = w_i(r) \forall r \in R_i \\ w_{ref}(fout(p_i), t_j) = w_{rec}(p_i, t_j), \forall (p_i, t_j) \in R_{rec} \\ w_{ref}(t_k, in(p_i)) = w_{rec}(t_k, p_i), \forall (t_k, p_i) \in R_{rec} \\ w_{ref}(p'_k, t) = w_{rec}(p'_k, t), \forall (p'_k, t) \in R_{rec} \end{cases} \\
M_{ref}^0 &= M_0(p_i), \forall p_i \in P_i \\
\text{Sendo} \\
\bigcap P_i &= \emptyset, \bigcap T_i = \emptyset \\
fin(p_i) &= p_{in,i}, fout(p_i) = p_{out,i}
\end{aligned}$$

Um passo **opcional** no processo de modelagem consiste em efetuar a fusão dos lugares que representam recursos compartilhados.

1. Seja  $N = \langle P, T, R, W, M_0 \rangle$ . Fazer  $N = N_{ref}$ .  $N$  representa a RdP que terá lugares fundidos.
2. Definir  $AV = \{p_r \mid p_r \in P, p_r \text{ representa um recurso compartilhado}\}$ .
3. Caso  $|(AV)| > 1$  fazer:
  - (a) Obter dois lugares distintos  $p_i, p_j \in AV$ .
  - (b) Fazer  $N_{ref} = fus(N, p_i, p_j)$ .
  - (c) Retornar ao passo inicial.

### Exemplo de Modelo Final para o RP

Existe uma grande flexibilidade quanto à RdP que pode ser utilizada para representar as operações. Sua definição é função basicamente dos aspectos comportamentais e estruturais que se deseja capturar.

Tabela 4.1: Atributos de uma Operação

Operação	Delay	Time to Fail	Time to Repair
$Op_n$	$D_n$	$TF_n$	$TR_n$

A Figura 4.11 mostra um exemplo de modelo em GSPN de uma operação.  $In_n$  e  $Out_n$  correspondem a *buffers* de entrada e saída, respectivamente.  $D_n$  corresponde ao tempo médio para a execução da operação.  $TF_n$  e  $TR_n$  correspondem, respectivamente, aos tempos médios de falha e recuperação do recurso, quando realizando esta atividade. O lugar  $p_{r,n}$  corresponde aos recursos disponíveis para a realização da operação. Convém observar a notação utilizada para representar transições em um modelo GSPN. As transições em negrito representam transições imediatas<sup>2</sup>, ou seja, com tempo zero de duração. As demais transições são estocásticas, ou seja, têm um tempo médio associado. Na Tabela 4.1 são mostrados os atributos associados a esta operação.

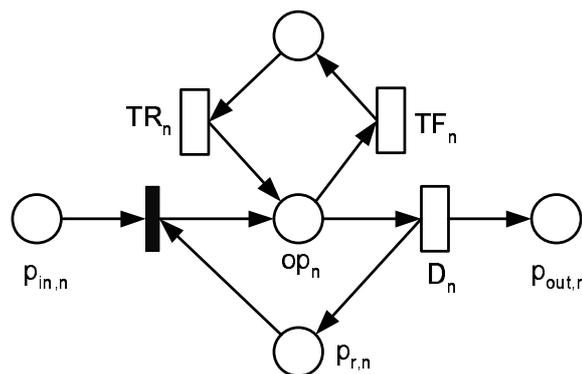


Figura 4.11: Modelo de uma Operação Usando RdPs

<sup>2</sup>Neste trabalho, transições (imediatas ou não) estão sendo genericamente representadas por retângulos em negrito. Quando necessário, será explicitado no texto que tal representação corresponde a uma transição imediata.

Um recurso pode realizar várias operações. Um dos objetivos da modelagem de sistemas de produção consiste em se obter, através de um modelo, uma configuração do sistema que possibilite a utilização ótima dos recursos. Aspectos fundamentais como a quantidade de recursos disponíveis, recursos compartilhados, etc., devem ser capturados pelo modelo, de forma a possibilitar análises de capacidade produtiva do sistema modelado.

A Figura 4.12 mostra um modelo de RdP para um recurso que realiza duas operações. O lugar  $p_{r,2}$  é compartilhado entre todas as operações e representa os recursos disponíveis.

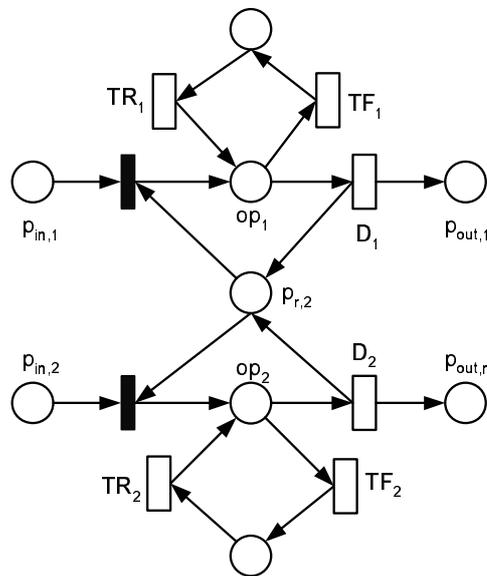


Figura 4.12: Modelo de Recurso com duas Operações Possíveis

Refinando-se as operações de  $N_{rec}$  com o modelo de RdP utilizado para representar as operações, chega-se ao modelo final para o recurso produtivo (veja Figura 4.13). Na figura, retângulos pontilhados são utilizados para destacar as RdPs utilizadas no refinamento dos lugares do modelo intermediário.

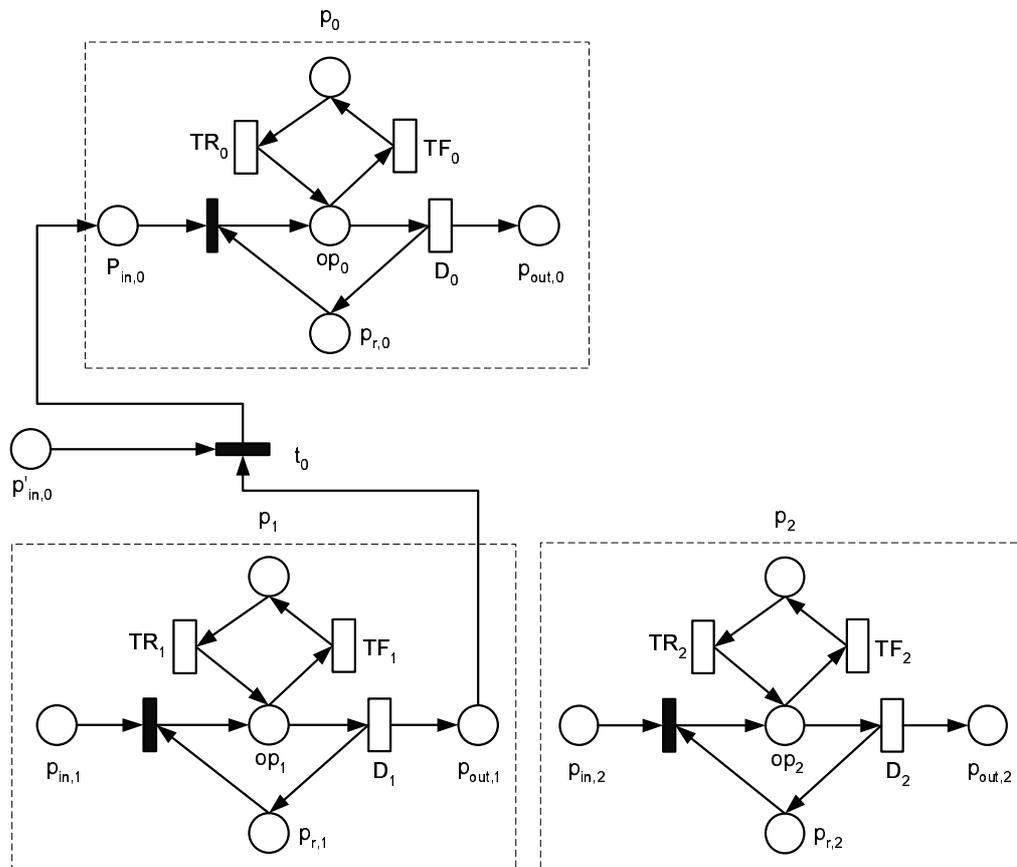


Figura 4.13: Modelo Final do Recurso

## 4.6 Roteiro de Produção (ROT)

O Roteiro de Produção determina a seqüência de operações necessárias para a manufatura de um dado produto. As operações são realizadas pelos recursos produtivos. Os seguintes passos são necessários para obter o roteiro de produção de um produto:

1. Escolher o produto a ser manufaturado.
2. Para cada sub-item definido na estrutura do produto, indicar o recurso/atividade que irá produzi-lo.
3. Gerar um escalonamento de atividades a partir da estrutura do produto e dos recursos produtivos.

A relação de precedência entre as operações é estabelecida a partir da estrutura do produto e dos recursos produtivos. Esta precedência de atividades não especifica quando cada atividade deverá ser iniciada, e sim que uma atividade não pode iniciar antes de outra. Para solucionar este problema, pode-se utilizar o escalonamento obtido através da EP, com base em alguma política de produção (ex: ALAP, ASAP, etc.). Para exemplificar abordagem, cita-se os sistemas MRP, que determinam o que produzir, quando e quanto, de forma a manter um estoque mínimo. O escalonamento obtido pelo MRP, utilizando as informações de *lead-time* e adotando-se uma política ALAP, poderiam ser utilizados para determinar, no ROT, quando cada operação deveria começar.

Este é apenas um dos procedimentos possíveis para obtenção do roteiro de produção a partir da estrutura do produto. Obviamente, outras formas de geração do roteiro podem ser definidas. Um exemplo típico é a geração do roteiro com base em uma política ALAP de produção. Outra possibilidade é a geração do roteiro atribuindo-se prioridades às operações concorrentes, com o objetivo de torná-las seqüenciais.

#### 4.6.1 Grafo do ROT

O primeiro passo na geração do ROT é a escolha do produto a ser manufaturado. Em seguida é preciso indicar, para cada sub-item do produto, qual recurso e qual atividade daquele recurso serão necessários para a sua produção. Isto pode ser visto na Figura 4.14.

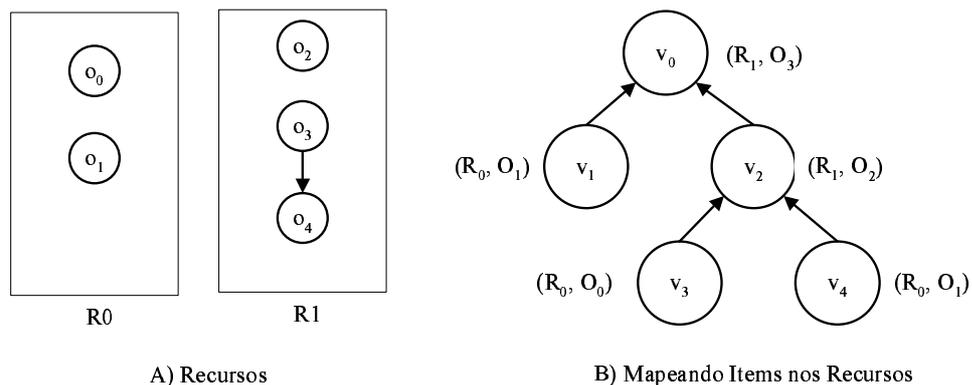


Figura 4.14: Associando Produto com Operações

A Figura 4.14 mostra dois recursos produtivos e a alocação dos mesmos para a manufatura de um dado produto. A inscrição ao lado do vértice  $v_0$ ,

por exemplo, indica que o item ou sub-produto representado por  $v_0$  será manufaturado através da operação representada por  $o_3$  do recurso representado por  $R_1$ .

Pelo mapeamento, as operações  $o_0$  e  $o_1$  do recurso  $R_0$  podem iniciar concorrentemente. A operação  $o_2$  do recurso  $R_1$  só tem início após a conclusão das operações  $o_0$  e  $o_1$ . Finalmente, a operação  $o_3$  do recurso  $R_1$  inicia-se após a conclusão das operações  $o_1$  e  $o_2$ .

Note que o mesmo recurso  $R_0$  é utilizado na produção/aquisição dos itens representados pelos vértices  $v_1$ ,  $v_3$  e  $v_4$ . Da mesma forma, o recurso  $R_1$  é utilizado na produção/aquisição dos itens  $v_0$  e  $v_2$ . Em particular, note que a mesma operação está sendo utilizada para manufaturar/adquirir os itens representados por  $v_1$  e  $v_4$ .

A Tabela 4.2 mostra o mapeamento dos recursos produtivos nos itens do produto.

Tabela 4.2: Mapeamento de Recursos na Estrutura do Produto

Vértice	Recurso	Atividade
$v_0$	$R_1$	$o_3$
$v_1$	$R_0$	$o_1$
$v_2$	$R_1$	$o_2$
$v_3$	$R_0$	$o_0$
$v_4$	$R_0$	$o_1$

Seja  $Prod = \langle V, A \rangle$  a estrutura de um produto. Seja  $Rec_i = \langle O_i, E_i \rangle$  o modelo do recurso produtivo associado ao vértice  $v_i \in V$ . Seja  $Recurso = \{Rec_i \mid 0 \leq i \leq n\}$  o conjunto de recursos disponíveis para a manufatura do produto.

A função  $aloc : V \rightarrow (Recurso \times O)$  determina, para um dado vértice, qual recurso/operação serão utilizados para a sua produção. Ela é utilizada para fornecer os valores mostrados na Tabela 4.2.

A função  $op : V \rightarrow O$  determina, para um dado vértice, qual operação será utilizada para a sua produção.  $op(v) = second(aloc(v))$ .

A função  $res : V \rightarrow Recurso$  determina, para um dado vértice, qual recurso será utilizado para a sua produção.  $res(v) = first(aloc(v))$ .

A função  $first$  retorna o primeiro elemento de um par ordenado (veja Definição A.18).

A função  $second$  retorna o segundo elemento de um par ordenado (veja Definição A.19).

O mapeamento da estrutura do produto em operações dos recursos resulta em um grafo direcionado, o qual é definido na seqüência.  $O_{rot}$  corresponde a um conjunto de vértices que representam operações.  $E_{rot}$  corresponde a um conjunto de arcos.  $A_{prod}$  representa os arcos que são obtidos a partir da estrutura do produto.  $E_{rec}$  representa os arcos que são obtidos a partir das relações de precedência existentes entre as operações dos recursos.

**Definição 4.19** (Modelo do Roteiro de Produção). *Seja  $Prod = \langle V, A \rangle$  a estrutura de um produto. Seja  $Rec_i = \langle O_i, E_i \rangle$  o modelo do recurso produtivo associado a um vértice  $v_i \in V$ . Defina-se o modelo do Roteiro de Produção através da tupla  $Rot = \langle O_{rot}, E_{rot} \rangle$ , onde*

$$\begin{aligned} O_{rot} &= \{op(v) \mid v \in V\} \\ E_{rot} &= A_{prod} \cup E_{rec} \\ \text{sendo} \\ A_{prod} &= \{(op(v_i), op(v_j)) \mid (v_i, v_j) \in A\} \\ E_{rec} &= \{e = (o_i, o_j) \mid e \in E_i, o_i \in O_{rot}, o_j \in O_{rot}, Rec_i \in \text{image}(\text{res})\} \end{aligned}$$

Esta definição é similar à definição do recurso produtivo apresentada na Seção 4.5. Deve-se observar, todavia, que apenas os arcos pertencentes a  $E_{rec}$  representam uma relação de precedência entre as operações (uma operação não pode iniciar antes da outra).

Como exemplo, será dada a definição matemática para o ROT mostrado na Figura 4.14. A estrutura do produto é representada matematicamente da seguinte forma:

$$\begin{aligned} Prod &= \langle V, A \rangle \text{ onde} \\ V &= \{v_0, v_1, v_2, v_3, v_4\} \\ A &= \{(v_0, v_1), (v_0, v_2), (v_2, v_3), (v_2, v_4)\} \end{aligned}$$

O recurso  $R_0$  é representado matematicamente como segue:

$$\begin{aligned} Rec &= \langle O, E \rangle \text{ onde} \\ O &= \{o_0, o_1\}, \\ E &= \emptyset. \end{aligned}$$

O recurso  $R_1$  é representado por:

$$\begin{aligned} Rec &= \langle O, E \rangle \text{ onde} \\ O &= \{o_2, o_3, o_4\}, \\ E &= \{(o_4, o_3)\}. \end{aligned}$$

Finalmente, o ROT é definido matematicamente da seguinte forma:

$$\begin{aligned} Rot &= \langle O_{rot}, E_{rot} \rangle \text{ onde} \\ O_{rot} &= \{op(v) \mid v \in V\} = \{o_3, o_1, o_2, o_0\}, \\ E_{rot} &= A_{prod} \cup E_{rec} = \{(o_3, o_1), (o_3, o_2), (o_2, o_0), (o_2, o_1)\}. \end{aligned}$$

### 4.6.2 Modelo Intermediário do ROT

Para a obtenção do modelo intermediário do ROT é aplicado um processo similar ao adotado para a representação do RP através de RdP. As operações correspondem a lugares de uma RdP. Os arcos representam relações pai-filho (arcos oriundos da EP) ou relações de precedência (arcos oriundos dos RPs). Para cada item pai é adicionada uma transição ao modelo. Um arco direcionado conecta o(s) item(s) filho(s) a esta transição. Um arco direcionado conecta a transição ao item pai.

De forma similar, para cada item que tem precedentes é adicionada uma transição ao modelo. Um arco direcionado conecta o item precedido à transição. Um arco direcionado conecta esta transição ao item precedente. Para cada item  $p_i$  que tem precedentes é adicionado um lugar  $p'_i$  ao modelo. Um arco direcionado conecta o lugar  $p'_i$  à transição.

Na seqüência é apresentado formalmente o modelo intermediário do ROT.  $P_{rot}$ ,  $T_{rot}$ ,  $R_{rot}$  e  $W_{rot}$  correspondem, respectivamente, ao conjunto de lugares, o conjunto de transições, o conjunto de relações, o vetor de pesos dos arcos do modelo intermediário.  $P_{NEW}$  corresponde aos lugares novos que são acrescidos em função da existência de relações de precedência entre operações dos recursos.  $R_{NEW}$  corresponde às relações novas que têm lugares de  $P_{NEW}$  como elementos de partida.  $E_{rec}$  corresponde ao conjunto de arcos obtidos a partir dos recursos produtivos (veja a Definição 4.19). A função bijetora  $mapP : O \rightarrow P$  mapeia vértices em lugares. A função  $mapT : P \rightarrow T$  mapeia lugares em transições.

**Definição 4.20** (Modelo Intermediário do Roteiro). *Seja  $Rot = \langle O_{rot}, E_{rot} \rangle$  o modelo de um roteiro de produção e  $N = \langle P, T, R, W \rangle$  uma RdP. Define-se o modelo intermediário para o roteiro de produção através da tupla  $N_{rot} = \langle P_{rot}, T_{rot}, R_{rot}, W_{rot} \rangle$  onde*

$$\begin{aligned}
 P_{rot} &= \{mapP(o_i) \mid o_i \in O_{rot}\} \cup P_{NEW} \\
 T_{rot} &= \{t_i \in T \mid o_i \in dom(E_{rot})\} \\
 R_{rot} &= \{(p_j, t), (t, p_i) \mid (o_i, o_j) \in E_{rot}, \\
 &\quad p_i = mapP(o_i), p_j = mapP(o_j), t = mapT(p_i)\} \cup R_{NEW} \\
 W_{rot} &= \{w_{rot}(r) = 1, \forall r \in R_{rot}\} \\
 \text{Sendo} \\
 P_{NEW} &= \{p'_i \mid p_i = mapP(o_i), o_i \in dom(E_{rec})\} \\
 R_{NEW} &= \{(p'_i, mapT(p_i)) \mid p_i = mapP(o_i), o_i \in dom(E_{rec})\} \\
 mapP(o_i) &= p_i, mapT(p_i) = t_i
 \end{aligned}$$

O modelo intermediário do ROT, para o exemplo mostrado na Figura 4.14,

é dado por

$N_{rot} = \langle P_{rot}, T_{rot}, R_{rot}, W_{rot} \rangle$  onde

$P_{rot} = \{p_3, p_1, p_2, p_0\} \cup \emptyset$ ,

$T_{rot} = \{t_3, t_2\}$ ,

$R_{rot} = \{(p_1, t_3), (t_3, p_3), (p_2, t_3), (t_3, p_3), (p_0, t_2), (t_2, p_2), (p_1, t_2), (t_2, p_2)\}$ ,

$W_{rot} = \{w_{rot}(r) = 1, \forall r \in R_{rot}\}$ .

A Figura 4.15 mostra o modelo intermediário para o ROT. Por questão de simplicidade, os atributos associados aos lugares do modelo intermediário não foram mostrados. Tais atributos correspondem aos mesmos atributos associados aos vértices do modelo de alto nível.

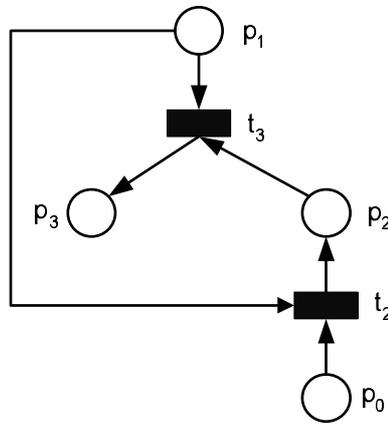


Figura 4.15: Modelo Intermediário para o ROT

Seguindo o processo de modelagem apresentado neste trabalho, os lugares do modelo intermediário obtido serão posteriormente refinados e será gerado um modelo final de RdP.

### 4.6.3 Refinamento de Lugares do ROT

As regras básicas para refinar as atividades do ROT são as mesmas utilizadas para refinar as operações de um recurso produtivo. Cada lugar que representa uma atividade do ROT tem um atributo *rede* que determina a sua representação de baixo nível, usando RdPs. Esta rede deve apresentar um lugar de entrada e um lugar de saída bem definido.

**Definição 4.21** (Modelo de Refinamento de um Lugar). *Seja  $N_{rot} = \langle P_{rot}, T_{rot}, R_{rot}, W_{rot} \rangle$  um roteiro de produção. A RdP utilizada para refinar*

um lugar  $p_i$  de  $P_{rot}$  é dada por  $N_i = \langle P_i, T_i, R_i, W_i, M_i^0 \rangle$ .  $p_{in,i}, p_{out,i} \in P_i$  correspondem respectivamente aos lugares de entrada e saída de  $N_i$ .  $\bullet p_{in,i} = \emptyset$ .  $p_{out,i}^\bullet = \emptyset$ .

A fim de gerar um modelo final coerente, os tipos de RdPs utilizados para refinar os lugares do modelo intermediário devem ser da mesma classe ou de classes compatíveis. Algumas propriedades verificadas no modelo de alto nível ou no modelo intermediário podem não estar presentes no modelo de RdP final. Isto é função da flexibilidade quanto à escolha da RdP utilizada no refinamento dos lugares. Para a preservação de propriedades é necessária uma escolha criteriosa dos modelos utilizados no refinamento.

#### 4.6.4 Modelo de RdP Final para o ROT

As regras básicas para obter o modelo de RdP para o ROT são as mesmas adotadas para obter o modelo de RdP do recurso produtivo.

**Definição 4.22** (Modelo Final de RdP para o Roteiro). *Seja o modelo intermediário para o roteiro de produção definido pela tupla  $N_{rot} = \langle P_{rot}, T_{rot}, R_{rot}, W_{rot} \rangle$ . Seja  $N_i = \langle P_i, T_i, R_i, W_i, M_i^0 \rangle$  a RdP utilizada para refinar o lugar  $p_i \in P_{rot}$ . Defina-se o modelo final de RdP para o roteiro de produção através da tupla  $N_{rotf} = \langle P_{rotf}, T_{rotf}, R_{rotf}, W_{rotf}, M_{rotf}^0 \rangle$  onde:*

$$\begin{aligned}
 P_{rotf} &= \bigcup P_i \cup \{p'_k \mid p'_k \in P_{NEW}\} \\
 T_{rotf} &= \bigcup T_i \cup T_{rot} \\
 R_{rotf} &= \bigcup R_i \\
 &\quad \cup \{(fout(p_i), t_j) \mid (p_i, t_j) \in R_{rot}\} \\
 &\quad \cup \{(t_i, fin(p_j)) \mid (t_i, p_j) \in R_{rot}\} \\
 &\quad \cup \{(p'_k, t) \mid (p'_k, t) \in R_{NEW}\} \\
 W_{rotf} &= \begin{cases} w_{rotf}(r) = w_i(r) \forall r \in R_i \\ w_{rotf}(fout(p_i), t_j) = w_{rot}(p_i, t_j), \forall (p_i, t_j) \in R_{rot} \\ w_{rotf}(t_k, in(p_i)) = w_{rot}(t_k, p_i), \forall (t_k, p_i) \in R_{rot} \\ w_{rotf}(p'_k, t) = w_{rot}(p'_k, t), \forall (p'_k, t) \in R_{NEW} \end{cases} \\
 M_{rotf}^0 &= M_0(p_i), \forall p_i \in P_i \\
 \text{Sendo,} \\
 \bigcap P_i &= \emptyset, \bigcap T_i = \emptyset \\
 fin(p_i) &= p_{in,i}, fout(p_i) = p_{out,i}
 \end{aligned}$$

$P_{rotf}, T_{rotf}, R_{rotf}, W_{rotf}$  e  $M_{rotf}^0$  correspondem, respectivamente, ao conjunto de lugares, o conjunto de transições, o conjunto de relações, o vetor de pesos dos arcos, e a marcação inicial do modelo final.  $fin : P_{rot} \rightarrow P_i$

é uma função que mapeia o lugar do modelo intermediário no lugar de entrada da rede que o refina.  $f_{out} : P_{rot} \rightarrow P_i$  é uma função que mapeia o lugar do modelo intermediário no lugar de saída da rede que o refina.

A Figura 4.16 mostra um modelo de RdP final para o ROT.

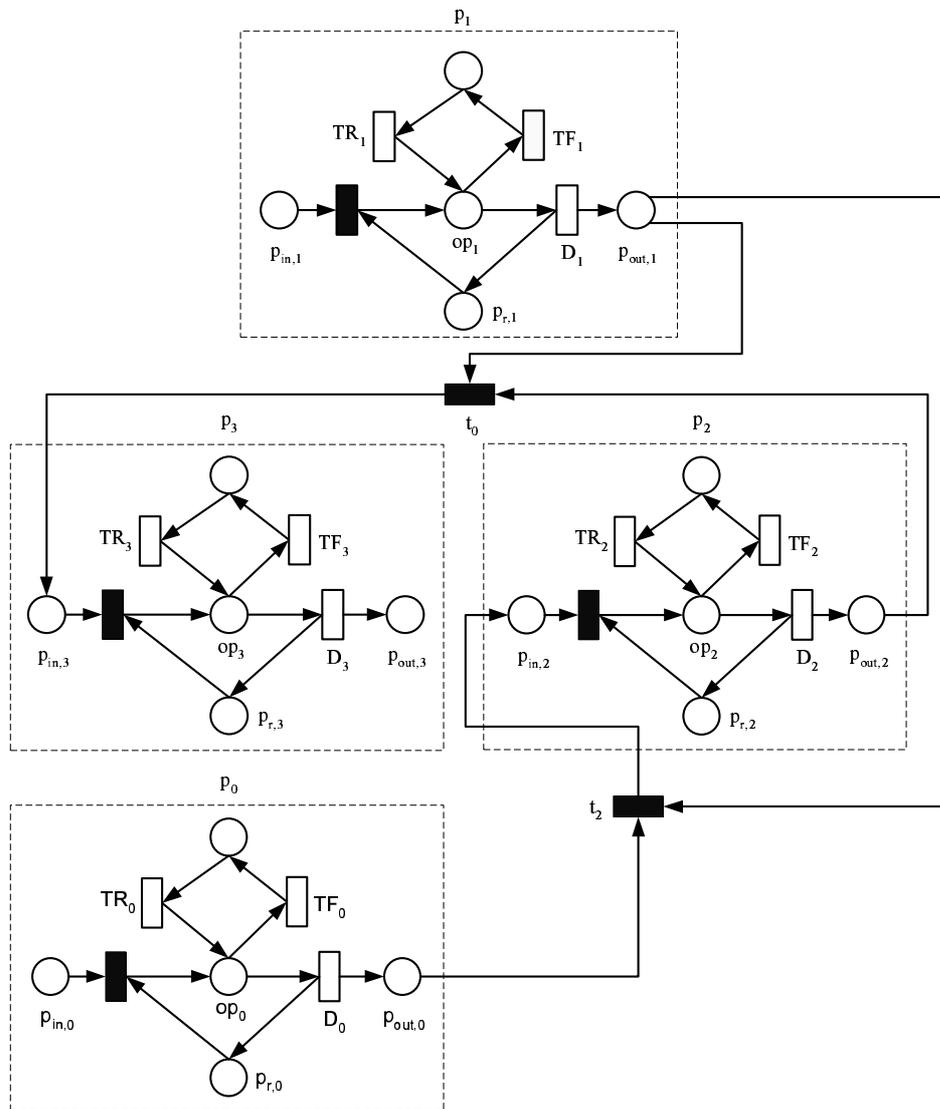


Figura 4.16: Modelo de RdP Final para o ROT

A RdP utilizada para refinar os lugares do modelo intermediário é mostrada na Figura 4.11. Retângulos pontilhados são utilizados para destacar as redes utilizadas para refinar os lugares do modelo intermediário.

## 4.7 Definição de Sistema Produtivo

A etapa final de modelagem consiste em definir o modelo para o sistema de produção como um todo. A modelagem aqui apresentada é baseada no trabalho de [BM02].

Um Sistema de Produção é um conjunto de atividades e operações inter-relacionadas envolvidas na produção de bens (caso da indústria) ou serviços [Mor02].

Um sistema de produção transforma matéria-prima em produtos (ou serviços). Produtos são compostos de *partes* em uma determinada ordem. Portanto, existe um fluxo dos insumos dentro do sistema. O plano de produção depende de fatores econômicos tais como demanda de mercado, disponibilidade de matéria-prima, fornecedores, do capital disponível, etc. [Tit95].

Baseado no planejamento, três espécies de processos podem ser considerados:

- Processos discretos – representados por variáveis discretas.
- Processos contínuos – representados por variáveis contínuas.
- Processos híbridos – representados tanto por variáveis discretas quanto por variáveis contínuas.

Em qualquer destes processos, para executar o planejamento, *ordens de serviços* são liberadas no sistema. Segundo [BM02], uma ordem de serviço é um conjunto não finito de *requisições de serviços* especificando como obter produtos. Portanto, um processo é uma ordem de serviço em execução.

Como mostrado na Figura 4.17, um *Sistema de Produção* tem duas estruturas que devem ser descritas separadamente: O *controle* e a *planta*. O controle gerencia a execução concorrente das ordens de serviço liberadas no sistema e informa o gerenciamento da produção sobre as ordens executadas [BPF97]. A planta é composta pelos recursos utilizados para executar diferentes operações.

A planta de um Sistema de Produção Discreto é basicamente composta de recursos dos seguintes tipos:

- Produção – transformam matéria-prima em partes ou produtos.
- Armazenagem – armazenam matéria-prima, partes ou produtos. São normalmente denominados *buffers*.
- Montagem – utilizados para conectar partes de um produto final.

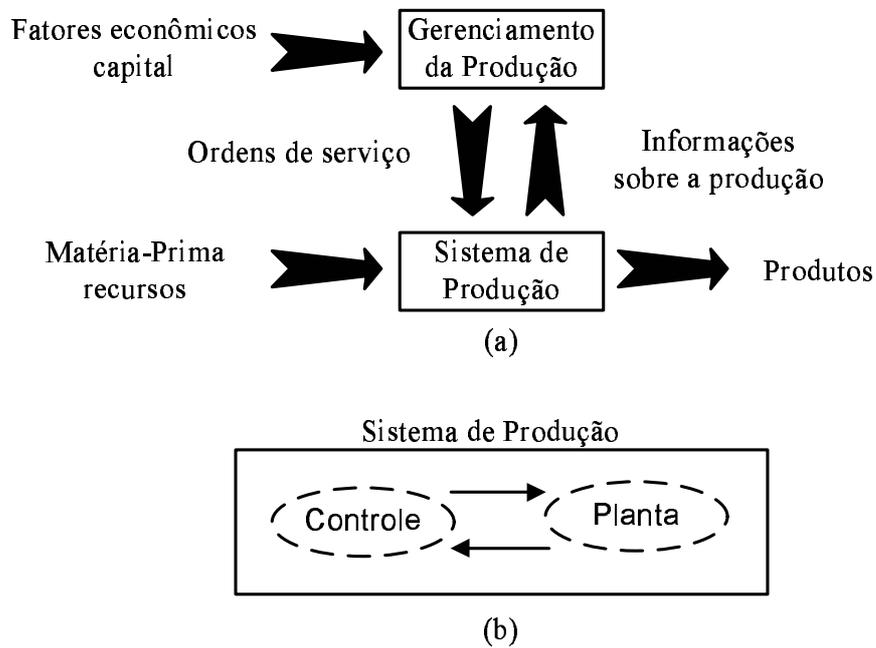


Figura 4.17: Visão Geral de Sistema de Produção

- Transporte – utilizados para transportar matéria-prima, partes ou produtos entre *buffers* do sistema.

Na seqüência, serão definidos os sub-modelos utilizados na definição do modelo do sistema de produção.

### Modelo do Sistema de Produção

Um Sistema de Produção pode ser caracterizado pelo inter-relacionamento complexo de suas atividades, condições, ciclos, repetições e conexões [Sil04].

A metodologia proposta para a modelagem do sistema de produção é composta dos seguintes sub-modelos:

1. Modelo de entrada (IFM) – Modela os pedidos emitidos para o sistema.
2. Modelo de saída (OFM) – Modela a conclusão dos pedidos.
3. Modelo de fechamento (CM) – Torna o modelo fortemente conectado.

4. Modelo do processo (PRM) – Modela o processo produtivo. Quaisquer dos modelos definidos para a Estrutura do Produto (EP), os Recursos Produtivos (RP) ou para o Roteiro de Produção (ROT) podem ser utilizados como modelos do Processo (PRM).

**Definição 4.23** (Modelo do Sistema de Produção Discreto). *Seja  $SM$  um sistema de produção discreto.  $SM = \langle IFM, OFM, CM, PRM \rangle$  define o modelo do sistema de produção discreto, onde  $IFM$  é o modelo de entrada das ordens de serviço,  $OFM$  é o modelo de saída das ordens de serviço,  $PRM$  é o modelo do processo, e  $CM$  é o modelo de fechamento.*

O IFM (veja a Figura 4.18) descreve a chegada de ordens de serviço. Ele representa o início da atividade de produção. Este modelo é formalmente definido como:

**Definição 4.24** (Modelo de Entrada). *Seja  $IFM = \langle P, T, R, W \rangle$  uma RdP definindo o modelo de entrada de ordens de serviço.  $T = \{t_0\}$ .  $p_0, p_{in,i} \in P$ , onde  $p_0 \neq p_{in,i}$ .  $p_0$  é chamado lugar de partida. Uma marca no lugar  $p_{in,i} \in P$  modela uma ordem de serviço.  $w(p_0, t_0) = 1$ .  $w(t_0, p_{in,i}) = n_i, \forall p_{in,i} \in P$  onde  $n_i$  é o número de ordens de serviço relacionadas com o produto  $i$ .*

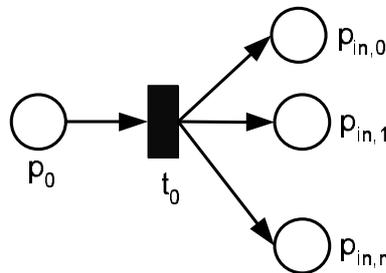


Figura 4.18: Modelo de Entrada

O OFM (veja a Figura 4.19) representa a conclusão de todos os processos de produção. A definição formal para este modelo é dada por:

**Definição 4.25** (Modelo de Saída). *Seja  $OFM = \langle P, T, R, W \rangle$  uma RdP definindo o modelo de saída de ordens de serviço.  $T = \{t_f\}$ .  $p_{out,i}, p_f \in P$ , onde  $p_{out,i} \neq p_f$ .  $p_f$  é chamado lugar de chegada. Uma marca no lugar  $p_f \in P$  modela a conclusão de todas as ordens de serviço.  $w(t_f, p_f) = 1$ .  $w(p_{out,i}, t_f) = n_i, \forall p_{out,i} \in P$  onde  $n_i$  é o número de ordens de serviço relacionadas com o produto  $i$ .*

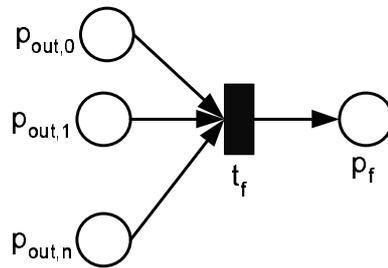


Figura 4.19: Modelo de Saída

A seguir é apresentado o modelo de fechamento. O CM (veja a Figura 4.20) é representado por uma transição conectando os lugares de partida e de chegada. O CM torna o modelo conectado e representa a repetitividade do processo produtivo.

**Definição 4.26** (Modelo de Fechamento). *Seja  $CM = \langle P, T, R, W \rangle$  uma RdP definindo o modelo de fechamento.  $T = \{t_c\}$ .  $P = \{p_f, p_o\}$ .  $p_f$  e  $p_o$  são os lugares de partida e chegada, respectivamente.*

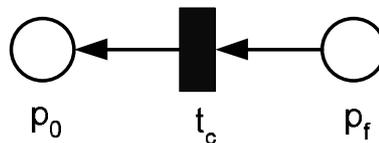


Figura 4.20: Modelo de Fechamento

Neste trabalho, qualquer um dos modelos de RdP definidos para a estrutura do produto, o recurso produtivo ou o roteiro pode ser utilizado para representar o modelo do processo.

A escolha de qual modelo utilizar é função basicamente do que se está querendo analisar no momento. Por exemplo, para uma análise mais agregada do sistema de produção, pode-se utilizar a estrutura do produto. Para uma análise de capacidade produtiva do sistema, é interessante modelar a planta produtiva. Neste caso, o modelo do recurso se mostra interessante. Finalmente, uma análise mais detalhada que leva em conta desde a posição dos produtos e recursos disponíveis, o modelo do roteiro de produção se mostra uma opção adequada.

**Definição 4.27** (Modelo do processo). *O modelo do processo, representado pela tupla  $PRM = \langle P, T, R, W, M_0 \rangle$ , é uma RdP que corresponde ao modelo de RdP definido para a estrutura do produto, ao modelo de RdP para o recurso produtivo, ou ao modelo de RdP para o roteiro de produção.*

Uma vez obtidos todos os modelos do processo, o próximo passo é construir o modelo do sistema. A abordagem para obter este modelo é a seguinte:

1. Fundir os lugares  $p_{in,u}$  do modelo de entrada e do modelo do processo, caso  $\bullet p_{in,u} = \emptyset$ . Denotar o lugar resultante  $p_{in,u}$ .
2. Fundir os lugares  $p_{out,u}$  do modelo de saída e do modelo do processo, caso  $\bullet p_{out,u} = \emptyset$ . Denotar o lugar resultante  $p_{out,u}$ .
3. Executar os passos 1 e 2 para  $u = 1, 2, \dots, n$ .

A Figura 4.21 mostra o resultado final da aplicação dos modelos de entrada, saída e fechamento, ao modelo final do roteiro de produção mostrado na Figura 4.16. Os lugares e transições  $p_0, t_0, p_{in,0}, p_{in,1}$  e  $p_{in,3}$  formam o modelo de entrada. Os lugares e transições  $p_{out,3}, t_f$  e  $p_f$  constituem o modelo de saída. Os lugares e transições  $p_f, t_c$  e  $p_0$  constituem o modelo de fechamento.

É importante observar que  $p_{in,2}$  não faz parte do modelo de entrada pois  $\bullet p_{in,2} \neq \emptyset$ . Da mesma forma,  $p_{out,0}, p_{out,1}$  e  $p_{out,2}$  não fazem parte do modelo de saída porque  $\bullet p_{out,0} \neq \emptyset$ ,  $\bullet p_{out,1} \neq \emptyset$  e  $\bullet p_{out,2} \neq \emptyset$ .

O fato de um mesmo lugar pertencer a mais de um modelo (por exemplo, modelo do processo e modelo de saída) é função da fusão de lugares ocorrida quando da geração do modelo final do sistema. Portanto, o sistema está completamente definido.

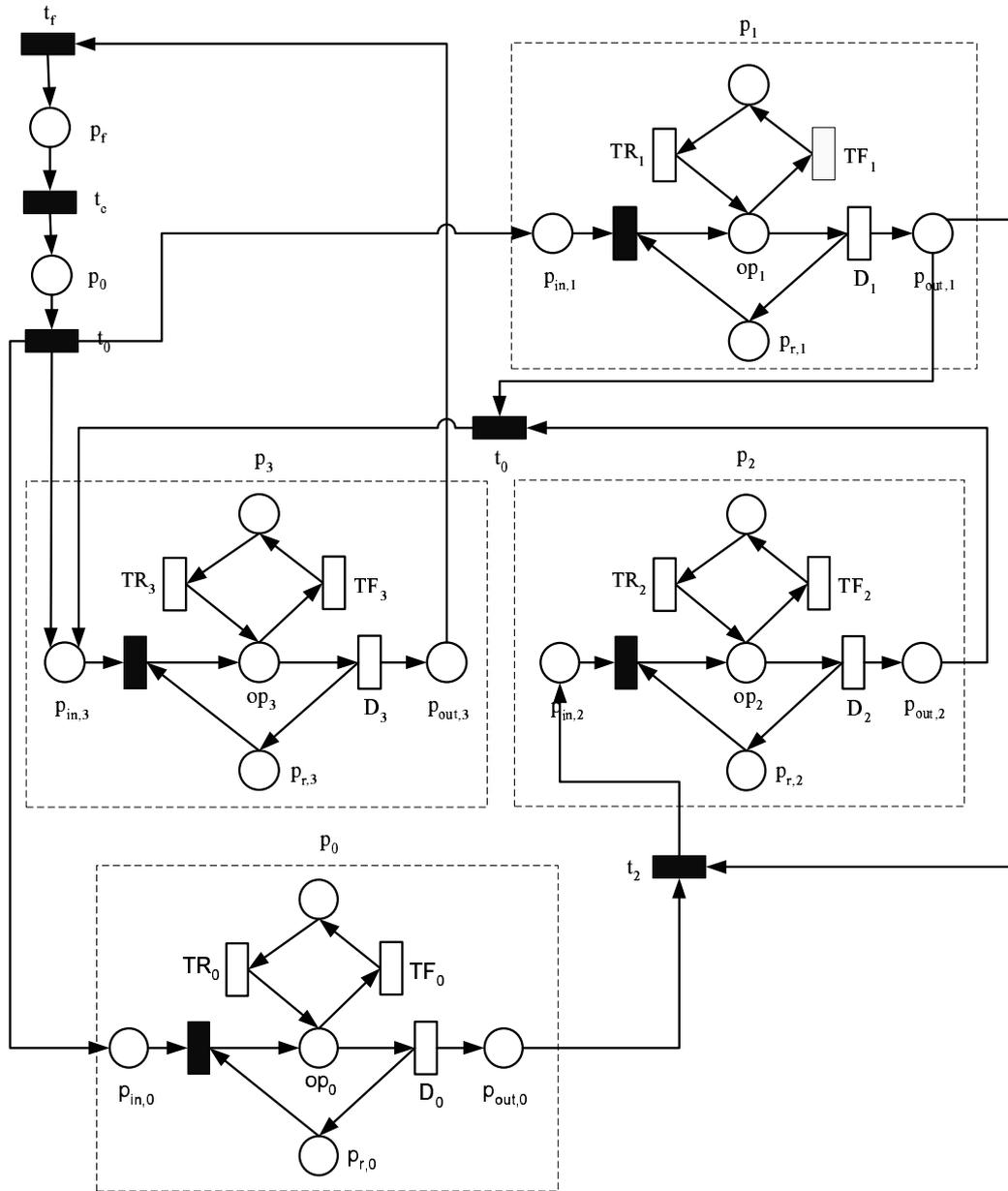


Figura 4.21: Modelo Final do Sistema

## 4.8 Considerações Finais

Neste capítulo foi apresentado o processo proposto para a modelagem de sistemas de produção. O processo de modelagem consiste em definir modelos de alto nível para a estrutura do produto e recursos produtivos. A partir destes modelos, é gerado um modelo para o roteiro de produção.

Neste trabalho, denomina-se modelo de alto nível ao modelo que não é baseado em RdPs. Os modelos obtidos são transformados em modelos intermediários de RdP e em seguida refinados até se obter um modelo final usando RdPs.

Há uma grande liberdade quanto à definição das RdPs utilizadas no refinamento dos lugares encontrados nos modelos intermediários. Todavia, tais redes devem ser da mesma classe da RdP final que será gerada. Além disso, devem apresentar lugares de entrada e saída bem definidos.

Para finalizar, foi definido o modelo do sistema de produção que é composto, além dos modelos anteriores, dos modelos de entrada, saída e fechamento. O modelo do sistema de produção assim obtido pode ser utilizado para análises qualitativas e quantitativas do sistema modelado.



# Capítulo 5

## Ambiente de Modelagem

*Neste capítulo é apresentada a ferramenta de modelagem PMSTool (Production Management System Tool) que suporta o processo de modelagem proposto neste trabalho. Inicialmente são apresentados o PNML (Petri Net Markup Language) e o PNK (Petri Net Kernel). Depois, são descritas as características básicas da PMSTool. Em seguida, é feita uma descrição funcional da mesma. Na seqüência, a interface gráfica da ferramenta é apresentada. Na seqüência, é descrito o processo de modelagem usando a PMSTool. Finalmente, é descrita a integração entre a PMSTool e o PNK.*

### 5.1 Introdução

Nesta seção será feita uma breve descrição do PNML e do PNK por terem uma utilização importante dentro da PMSTool. O PNML (*Petri Net Markup Language*) é o formato de arquivo para armazenamento dos modelos de RdP utilizado pela PMSTool. Trata-se de uma tentativa de padronizar a representação das RdPs o que simplificaria, por exemplo, a comunicação entre ferramentas de RdP distintas. O PNML possibilita a representação de qualquer tipo de RdP.

Por sua vez, o PNK (*Petri Net Kernel*) é um *framework* para o desenvolvimento de ferramentas baseadas em RdPs. Através da utilização do PNK, o desenvolvedor de ferramentas baseadas em RdPs necessita preocupar-se apenas com a definição de regras de negócio em vez de precisar re-implementar toda a funcionalidade que está normalmente presente em qualquer ferramenta baseada em RdP. Em síntese, a utilização do PNK simplificaria o processo de desenvolvimento de ferramentas baseadas em RdP.

A PMSTool teve como um dos seus requisitos básicos a integração com o PNK e a utilização da PNML para armazenar os modelos gerados.

### 5.1.1 Petri Net Markup Language (PNML)

Existem vários tipos de RdPs e várias ferramentas com suporte para estas redes, com seus respectivos formatos de arquivos. Dada esta diversidade de representação e tipos de rede, surge a dificuldade para se exportar ou importar RdPs entre ferramentas distintas. O PNML é um formato de intercâmbio ou troca de RdPs baseado em XML [RM01]. Trata-se de uma tentativa de padronizar a representação das RdPs. O objetivo é definir um formato de arquivo que suporte todos os tipos de RdPs. Os princípios que guiaram o projeto do PNML foram [JKW99, BCvH<sup>+</sup>]:

- O formato deveria ser legível para seres humanos e alterável através de um editor de texto convencional.
- Através do formato deveria ser possível representar qualquer tipo de RdP.
- O formato deveria fornecer tanta informação quanto possível de uma rede, mesmo que seu tipo fosse desconhecido.

Para poder representar qualquer tipo de RdP, a PNML considera uma RdP como um grafo rotulado, onde todas as informações extras podem ser adicionadas em rótulos que podem estar associados à própria rede, aos nós da rede ou aos seus arcos.

PNML suporta a definição de diferentes tipos de RdP. Um arquivo *Petri Net Type Definition (PNTD)* determina os rótulos que são permitidos para um dado tipo de rede. Através da atribuição de um tipo fixo a cada RdP, a sua descrição se torna não ambígua.

De forma a garantir que o máximo de informação possível possa ser trocada entre diferentes tipos de redes, PNML apresenta convenções em como definir rótulos com um significado específico. Em um *Documento de Convenções*, a sintaxe bem como o significado pretendido de todas as espécies de extensões são pré-definidos. Quando definindo um novo tipo de RdP, os rótulos podem ser escolhidos deste documento de convenções. Desta forma, o significado dos rótulos será bem conhecido e uma outra ferramenta baseada em RdP poderá interpretar a rede mesmo que não conheça o novo tipo de RdP.

PNML foi originalmente desenvolvido com um formato de arquivo para a versão *Java* do PNK.

### 5.1.2 Petri Net Kernel (PNK)

O PNK é uma infra-estrutura para a implementação de algoritmos relacionados com a simulação, análise e verificação de RdPs [EW01]. O PNK realiza operações básicas sobre RdPs tais como carregar, modificar características e salvar as redes. O desenvolvedor de uma aplicação baseada em RdPs não necessita desenvolver *parsers* para carregar uma rede a partir de um arquivo ou lidar com interfaces gráficas para representar uma rede [Hum99]. O programador necessita concentrar-se apenas na funcionalidade, tal como análise, verificação e algoritmos de simulação.

O PNK fornece várias interfaces:

**Application:** A interface *application* provê algumas funções sobre RdPs para o desenvolvimento de aplicações.

**Editor:** A interface *editor* descreve a interação entre o PNK e um editor. O PNK possui um editor de redes simples, o qual pode ser substituído por outro editor que satisfaça a interface *editor*. Ambas, a interface *application* e a interface *editor* podem ser utilizadas para o desenvolvimento de protótipos de ferramentas baseadas em RdPs.

**Net Type:** A interface *net type* define como um desenvolvedor de aplicação pode definir seu próprio tipo de RdP com extensões específicas. Uma característica importante do PNK é que ele não é restrito a um tipo específico de RdP. É possível definir novos tipos de rede.

Um tipo de rede é definido como uma implementação da interface *net type*, provendo informações como a representação de marcações, a descrição de modos de transição, a representação de inscrições em arcos, extensões para certos elementos da rede, tais como guardas nas transições e tempo nos lugares.

A Figura 5.1 mostra as diferentes camadas para especificação de um tipo de rede usando o PNK. A camada básica do PNK consiste das classes *Net*, *Place*, *Transition* e *Arc*, que definem modificadores e métodos de acesso para os elementos da rede. O restante é definido pela implementação do tipo de rede, que tem que ser feito para cada tipo de rede usado pela aplicação.

A implementação do tipo da rede define as características da mesma através da implementação dos atributos das respectivas classes: A classe *Place* requer uma implementação de *Marking*. A classe *Transition* requer a implementação da classe *Mode*, e a classe *Arc* requer a implementação de *Inscription*. Informações adicionais são implementadas em classes derivadas da classe *Extension*.

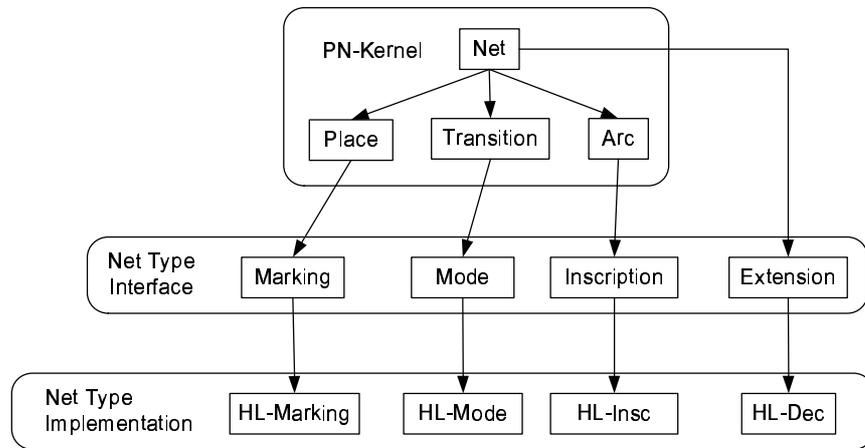


Figura 5.1: Camadas do Petri Net Kernel

As classes *Marking*, *Mode*, *Inscription* e *Extension* são implementadas como atributos adicionais das classes *Place*, *Transition*, *Arc* ou *Net*, respectivamente.

O PNK utiliza o PNML como formato de arquivo para o armazenamento de RdPs. Por ser um formato genérico, o PNML pode ser utilizado para a troca de dados da rede entre ferramentas distintas, desde que suportem o formato PNML. Desta forma, o PNK pode ser visto como uma ferramenta de integração fornecendo uma interface universal para ferramentas de RdPs.

## 5.2 PMSTool

Existem vários tipos de RdPs e, em consequência, várias ferramentas que suportem estes tipos de rede e seus respectivos formatos de arquivos. Cada ferramenta em geral adota seu próprio formato de arquivo, o que torna difícil a exportação/importação de RdPs entre ferramentas distintas. Um outro problema decorre do fato destas ferramentas normalmente serem voltadas para uma plataforma de *software* específica (ex: *Windows*, *Unix*, etc.). De maneira a evitar os problemas mencionados e possibilitar a comunicação com outras ferramentas, a PMSTool foi projetada para ser compatível com o formato de arquivo PNML, que é um formato de intercâmbio ou troca de RdPs baseado em XML. PNML é uma tentativa de padronizar a representação das RdPs. Por se tratar de uma tentativa de padronização da representação das RdPs, sua utilização implica em maior integração entre a ferramenta desenvolvida e outras ferramentas que su-

portem o formato PNML. Esforço foi realizado no sentido de tornar o processo de modelagem transparente em relação a utilização de RdPs.

A ferramenta pode ser utilizada como um módulo do PNK. O PNK fornece uma infra-estrutura para a construção de ferramentas baseadas em RdPs, além da análise, simulação e verificação das RdPs. O PNK provê uma API para ler uma rede a partir de um arquivo e para acessar os seus objetos, e portanto pode ser vista como um *Document Object Model (DOM)* para PNML.

A ferramenta foi implementada em *Java* para tornar o seu uso independente da plataforma. Sua arquitetura segue o padrão *Model-View-Controller (MVC)* [Mil03]. A idéia do padrão MVC é dividir uma aplicação em três camadas: *modelo*, *visualização* e *controle*. O *modelo* é responsável por representar os objetos de negócio, manter o estado da aplicação e fornecer ao controlador o acesso aos dados. A *visualização* representa a interface com o usuário, sendo responsável por definir a forma como os dados serão apresentados e encaminhar as ações dos usuários para o controlador. Já a camada de *controle* é responsável por fazer a ligação entre o modelo e a visualização, além de interpretar as ações do usuário e traduzi-las em uma operação sobre o modelo.

Um diferencial da ferramenta desenvolvida em relação às ferramentas apresentadas na Seção 1.6 é que esta é vista como uma ferramenta de planejamento e controle de sistemas de produção e não apenas como uma ferramenta de modelagem de sistemas de manufatura. Outro diferencial consiste no processo da modelagem, que inicia na estrutura do produto em vez de no chão de fábrica. Esta abordagem está mais próxima ao processo de planejamento e controle da produção real.

Em [SSM04] é mostrado a utilização da PMSTool na modelagem de sistemas de produção com ênfase no estoque.

### 5.2.1 Características

O projeto de sistemas modernos de produção requer o uso de técnicas e ferramentas de modelagem e avaliação de desempenho. Este trabalho apresenta a PMSTool, que é uma ferramenta baseada em RdPs que permite a modelagem de sistemas de produção.

Embora baseada em RdPs, a PMSTool torna transparente para o usuário a utilização de RdPs através da utilização de ícones. Ícones são utilizados como representações de fácil utilização e entendimento dos elementos do sistema produtivo. Estes ícones são automaticamente mapeados em um modelo de RdP final sem a interação do usuário. O usuário não necessita

conhecer a teoria das RdPs para poder utilizar a ferramenta. O usuário que domina a teoria das RdPs pode criar novos tipos e adicioná-los a uma biblioteca.

A PMSTool permite a modelagem da estrutura do produto, do recurso produtivo e do roteiro de produção. O roteiro de produção é definido a partir da estrutura do produto. Ícones e arcos são os elementos básicos de desenho na GUI.

Cada ícone tem associado um tipo e um conjunto de propriedades. As propriedades variam em função do tipo. Por sua vez, o tipo determina o mapeamento do ícone em uma RdP específica.

O usuário comum não precisa conhecer RdPs para poder utilizar a ferramenta. O usuário especialista, por sua vez, pode definir o formato das redes a serem mapeadas nos ícones apresentados ao usuário.

Em resumo, as principais características da ferramenta são:

- É específica para a modelagem de sistemas de produção.
- Abstrai o uso de RdPs através da utilização de ícones de alto nível.
- Cada ícone é mapeado de forma transparente em uma RdP.
- Usuário pode definir novos tipos de rede a serem mapeados aos ícones (biblioteca de ícones).
- Possibilita a modelagem da estrutura do produto e do recurso produtivo.
- Roteiro de produção é gerado automaticamente a partir da estrutura do produto e do recurso produtivo.
- O modelos resultantes são armazenados no formato PNML.

A PMSTool gera um modelo intermediário para a estrutura do produto, recurso e para o roteiro de produção. Este modelo intermediário é utilizado em conjunto com as redes associadas aos ícones para gerar um modelo de RdP final, que é armazenado em um arquivo PNML.

### 5.2.2 Visão Geral da Ferramenta

A Figura 5.2 mostra os diferentes módulos encontrados na PMSTool, bem como a interação entre os mesmos. Retângulos comuns representam as informações que devem ser fornecidas pelo usuário. Retângulos com extremos arredondados representam os módulos principais da ferramenta.

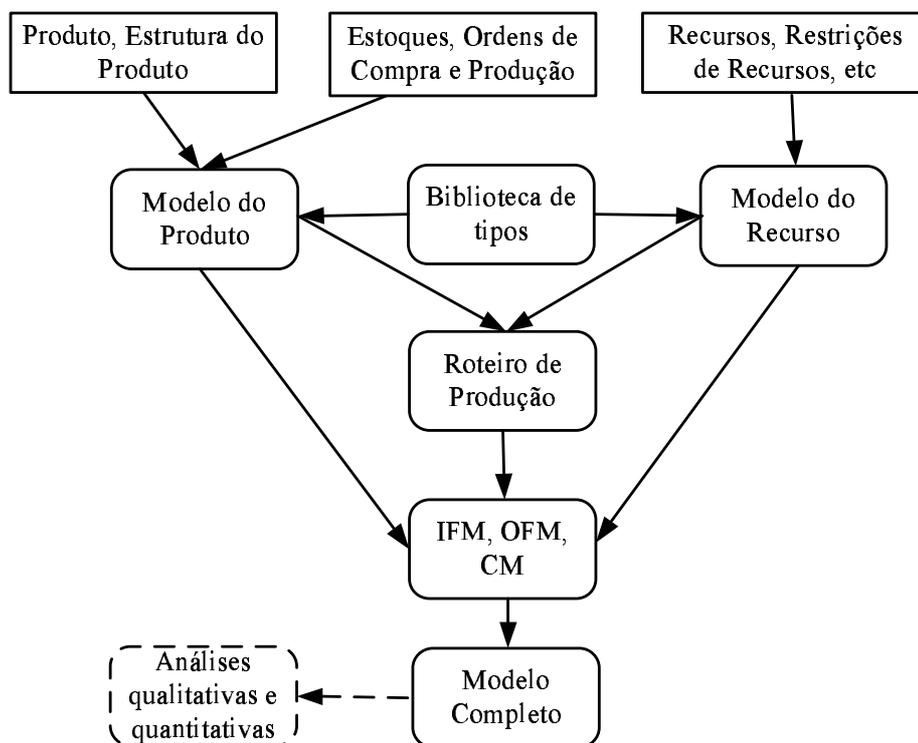


Figura 5.2: Arquitetura da Ferramenta

O ponto de partida consiste nas informações sobre o sistema a ser modelado, como por exemplo, a descrição dos produtos (lista de materiais que compõem a estrutura do produto, sua posição no estoque, etc.), e os recursos disponíveis (tipo, tempos de preparação, tempos médios de falha, etc.).

Em um primeiro passo, são gerados modelos para a estrutura do produto e os recursos produtivos. Estes modelos podem ser refinados posteriormente utilizando-se um ou mais sub-modelos de RdPs presentes na biblioteca de tipos (também chamada biblioteca de ícones). Os modelos da estrutura do produto e dos recursos produtivos são compostos de ícones de alto nível que apresentam um tipo associado, que determina, de forma, transparente, qual RdP será utilizada no seu refinamento.

A biblioteca de tipos corresponde a um conjunto de sub-modelos de RdPs que podem ser parametrizados e instanciados para facilitar a descrição de sistemas maiores. Os usuários com conhecimentos em RdPs poderão criar seus próprios ícones e acrescentá-los à biblioteca. A PMSTool

não disponibiliza a função de criação de *tipos*, o que pode ser feito através de um editor de RdP que suporte PNML. A criação de tipos é possível, por exemplo, quando se utiliza a PMSTool integrada com o PNK. A RdP utilizada para representar o tipo deve apresentar um lugar de entrada e um lugar de saída bem definidos, conforme explicitado no Capítulo 4.

A partir da estrutura do produto e dos recursos produtivos, é possível gerar o modelo de um roteiro de produção para o produto. Este modelo também é composto de ícones com um tipo associado, e pode ser refinado com a utilização dos tipos presentes na biblioteca de tipos.

O modelo completo do sistema, sob o qual é possível efetuar análises qualitativas e quantitativas, é composto de quaisquer dos modelos anteriores (Modelo da Estrutura do Produto, Modelo dos Recursos, Modelo de Roteiro) e dos modelos de entrada (IFM), saída (OFM) e fechamento (CM). Portanto, são possíveis três modelos finais distintos, um para o recurso, um para a estrutura do produto, e um para o roteiro de produção.

A escolha de qual modelo utilizar é função da necessidade do modelador. Por exemplo, o modelo da estrutura do produto serve para fornecer uma série de estimativas, tais como tempo mínimo para produção de um dado produto (com base nas informações de *lead-time*), escalonamento das atividades produtivas no tempo, além de possibilitar a simulação do envio das ordens de compra e produção. Este modelo é útil quando se deseja ter uma estimativa razoável sobre os tempos de produção e escalonamento das atividades. Todavia, tal modelo não leva em conta nenhuma restrição de recurso do sistema. O modelo do recurso pode ser utilizado quando se deseja obter um modelo mais completo e realista do sistema de produção que leve em conta não apenas os materiais necessários para a manufatura de um produto e o tempo, mas também a capacidade produtiva do sistema. Finalmente, o modelo do roteiro de produção pode ser utilizado para verificar se o que foi planejado a partir das informações obtidas através do modelo do produto e dos recursos produtivos é passível de ser realizado no sistema de manufatura modelado, considerando-se suas restrições de recursos, bem como escolher quais os melhores roteiros para a manufatura de um dado produto.

### 5.2.3 Estrutura do Código

O projeto da ferramenta e a sua implementação tiveram por base o padrão MVC. A divisão de código segue a seguinte estrutura:

**Camada de Dados** – código que manipula os dados da RdP, incluindo a leitura e o armazenamento dos dados. Nesta camada estão as classes

que representam objetos de uma RdP (lugar, transição, etc.).

**Camada de Controle** – módulos para controle de operação da rede, utilizando a funcionalidade da camada de dados.

**Camada de Visão** – classes responsáveis pela visualização gráfica das redes e o gerenciamento de eventos de edição através de classes auxiliares.

A camada de dados é responsável pela leitura e armazenamento dos dados. Ela também é responsável por expor os dados de uma forma que a camada de visão (interface gráfica) possa acessá-los e por validá-los. Esta camada responde às requisições de informação sobre o seu estado (geralmente feito pela camada de visão), e responde a instruções para modificar o seu estado (geralmente fornecidas pela camada de controle).

Dentre as várias funcionalidades encontradas nesta camada, destacam-se:

- Manipulação de arquivos (leitura, escrita, atualização).
- Conversão de/ para o formato PNML.
- Classes e métodos para transformar o modelo de alto nível em um modelo intermediário, e o modelo intermediário em um modelo de RdP final.
- Algoritmos para refinamento de lugares do modelo intermediário.
- Algoritmos para mapear os valores dos atributos dos lugares do modelo intermediário nos elementos correspondentes da rede que o refina.
- Algoritmos para aplicação do modelo de entrada, o modelo de saída, e o modelo de fechamento sobre os modelos do processo (estrutura do produto, recurso ou roteiro) e geração do modelo do sistema discreto.

Em linhas gerais, as transformações por que passa o modelo inicial envolvem a inserção automática de transições neste modelo, para gerar o modelo intermediário, e o refinamento dos lugares do modelo intermediário para a geração do modelo final (processo de síntese).

A camada de controle interpreta a entrada do usuário (ex: teclado ou *mouse* pressionados) e gerencia a camada de dados ou de visão para

modificar-se conforme apropriado.

A camada de visão gerencia a interface gráfica ou textual da aplicação. O usuário interage com a PMSTool através da sua interface gráfica.

### 5.2.4 Interface Gráfica

A GUI da PMSTool foi desenvolvida com o intuito de prover um ambiente de fácil utilização para a modelagem de sistemas de produção. A abstração quanto ao uso de RDPs constitui-se em um dos requisitos básicos no seu projeto. A ferramenta foi implementada em *Java* para tornar-se independente de plataforma.

A Figura 5.5 mostra uma tela da PMSTool. A tela é composta de três áreas principais: a barra de *menus*, a área de botões, e a área de desenho. A área superior contém alguns *menus* com comandos usuais para acesso a arquivo, modelagem de elementos do sistema de produção. A área de botões disponibiliza os mesmos comandos acessíveis através dos *menus*. Finalmente, a área de desenho possibilita o desenho do modelo utilizando ícones definidos para os arcos e vértices.

#### Menus

Na seqüência são descritos os comandos que estão disponíveis e podem ser utilizados através da área de *menu*. A Figura 5.3 mostra a área de *menu* da GUI da PMSTool. Existem três *menus* principais: *File*, *PMS* e *Draw*.



Figura 5.3: Área de Menu

O *menu File* agrupa as operações de manipulação de arquivos. Os seguintes comandos estão disponíveis:

**Create** – Cria uma área de desenho para a modelagem de um produto, recurso ou roteiro de produção.

**Open** – Abre um modelo já criado.

**Close** – Fecha o modelo que está aberto na ferramenta. Caso haja alterações no modelo, uma caixa de diálogo é mostrada perguntando se o usuário deseja salvar as alterações efetuadas.

**Save** – Salva o modelo correntemente aberto. Modelos finais para o produto, o recurso produtivo e o roteiro de produção são armazenados no formato PNML.

**SaveAs** – Possibilita que o modelo corrente seja salvo com um nome diferente.

**Exit** – Fecha a aplicação.

O *menu PMS* possibilita ao usuário a escolha do elemento de modelagem (estrutura do produto, recurso, roteiro de produção). Os seguintes comandos estão disponíveis:

**Product Structure (P)** – Com esta opção selecionada, a ferramenta estará configurada para a modelagem da estrutura do produto.

**Resource (R)** – Esta opção de *menu* selecionada possibilita a modelagem de recursos produtivos.

**Workplan (W)** – Esta opção de *menu* possibilita o mapeamento das operações nas partes de um produto, com o objetivo de gerar o roteiro de produção.

O *menu Draw* possibilita a escolha do elemento de desenho. A versão atual da PMSTool trabalha com dois elementos de desenho: ícones, representados por círculos, e arcos de interconexão. Existe também a opção de geração do modelo final no formato PNML.

**Ícones** – representados por círculos. Representam partes de um produto ou atividades de um recurso.

**Arcos de interconexão** – Representados por arcos direcionados. Representam relações pai-filho entre partes de um produto ou relações de precedência entre as atividades de um recurso.

### Área de Botões

A Figura 5.4 mostra a área de botões da PMSTool. Esta área contém um subconjunto dos comandos disponíveis através do *menu*. Da esquerda para a direita, é possível identificar quatro áreas distintas:

**File** – disponibiliza botões com ações para manipulação de arquivos tais como criação de uma rede, abertura de uma rede, gravação de uma rede, gravar a rede com um determinado nome, e fechamento da rede atual.



Figura 5.4: Área de Botões

**Draw** – disponibiliza comandos para desenho de ícones e arcos de interconexão, que são os elementos básicos de desenho.

**PMS** – Possibilita a escolha do elemento de modelagem (produto, recurso, roteiro de produção).

**Generate** – disponibiliza comando para gerar os modelos de RdP.

### Área de desenho

A área de desenho mostra o modelo atual. O *menu* PMS possibilita ao usuário a escolha do aspecto (estrutura do produto, recurso, roteiro de produção) a ser representado na área de desenho. Os elementos que podem ser desenhados na área de desenho são aqueles encontrados no *menu Draw* (ícones e arcos de interconexão).

Para desenhar um objeto na área de desenho, o usuário seleciona um item de desenho no *menu* ou na área de botões. Objetos podem ser adicionados e removidos da área de desenho.

É possível visualizar os atributos associados aos objetos desenhados e seus valores atuais através de uma tela que exhibe os atributos dos elementos desenhados. O atributo *rede* é utilizado para mapear o objeto e seus atributos na RdP correspondente. Esta rede será utilizada para gerar o modelo final de RdP.

PMSTool foi desenvolvida em *Java* [DD97] para garantir independência de plataforma. O modelo final é armazenado no formato *PNML* para permitir o seu uso por diferentes ferramentas de RdP.

### 5.2.5 Etapas da Modelagem Usando a PMSTool

O processo completo de modelagem de um sistema produtivo, utilizando a PMSTool, envolve as seguintes operações:

- Modelagem da Estrutura do Produto
- Modelagem do Recurso Produtivo

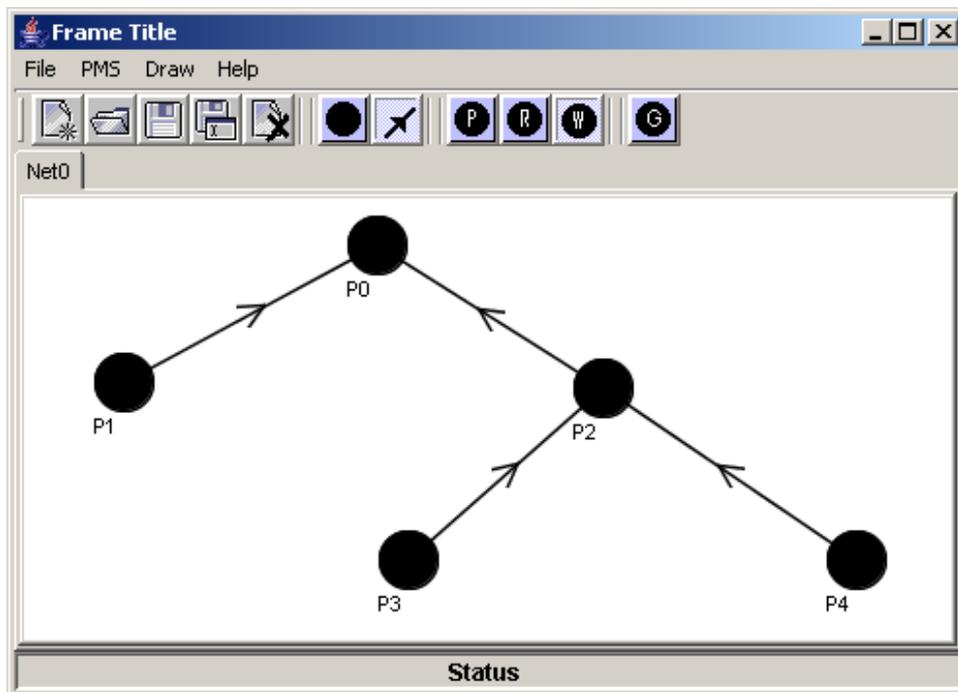


Figura 5.5: Interface Gráfica da PMSTool

- Mapeamento de cada parte do Recurso Produtivo na respectiva operação realizada por um dado recurso.
- Geração do modelo final no formato *PNML*.

O *menu PMS* permite a escolha do aspecto de modelagem (produto, recurso, roteiro). A mesma funcionalidade está presente na barra de ferramentas da PMSTool. Cada uma destas atividades será detalhada na seqüência.

### Modelagem da Estrutura do Produto

O usuário escolhe modelar a estrutura de um produto selecionando o *menu* correspondente na tela da PMSTool. Os elementos básicos de desenho são ícones e arcos direcionados. Cada ícone representa um item de um produto. Os arcos representam as relações pai-filho entre os itens.

Cada ícone tem um conjunto de atributos. A tela da Figura 5.6 mostra os atributos associados a um determinado ícone. O usuário pode atribuir valores aos atributos previamente exibidos ou pode definir novos atributos.

Name	Description	Value
LT	Lead Time	10
QT	Quantity	2

Figura 5.6: Atributos de um Item de Produto

O atributo *rede* (também denominado *type* na ferramenta) especifica qual RdP será utilizada no refinamento do ícone. Os atributos previamente exibidos para o usuário variam em função da *rede* selecionada.

A estrutura do produto assim modelada não pode conter ciclos, ou seja, não pode haver um caminho que permita partir de um ícone qualquer, visitar um ou mais ícones, e retornar ao ícone de partida.

### Modelagem do Recurso Produtivo

A modelagem do recurso produtivo é similar à modelagem da estrutura do produto. Na tela da PMSTool existe a opção de *menu* para a modelagem do recurso produtivo.

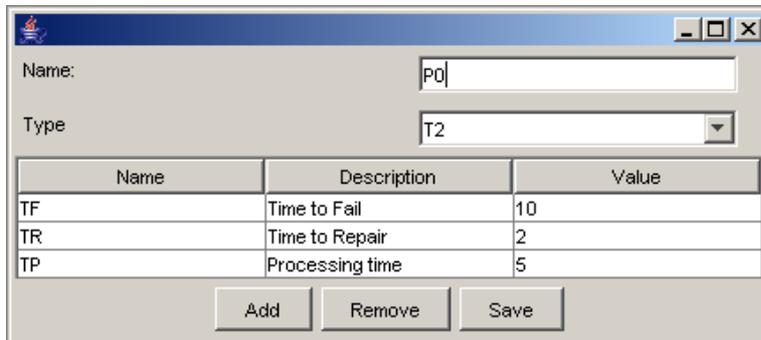
Os elementos básicos de desenho também são ícones e arcos de interconexão. Estes elementos, todavia, apresentam novo significado. Os ícones representam as operações que podem ser realizadas pelos recursos produtivos. Os arcos representam relações de precedência entre estas operações.

A tela da Figura 5.6 mostra os atributos associados a um determinado ícone. O atributo *rede* determina a RdP que será utilizada para refinar o ícone, bem como o conjunto de atributos exibidos na tela. O usuário pode adicionar/remover atributos.

### Mapeando Produtos nos Recursos/Atividades

A próxima etapa no processo de modelagem consiste em definir, para cada parte do produto, o recurso/atividade necessário para a sua produção.

Na tela da PMSTool, o usuário escolhe no *menu* correspondente a opção de modelar o roteiro de produção. Em seguida, o usuário necessita carregar o modelo do produto para o qual o roteiro será gerado.



Name	Description	Value
TF	Time to Fail	10
TR	Time to Repair	2
TP	Processing time	5

Figura 5.7: Atributos de uma Operação

A tela mostrada na Figura 5.8 permite ao usuário associar uma parte de um produto ao recurso/atividade que irá produzi-lo. Primeiramente, o usuário escolhe o recurso que será utilizado. As atividades relacionadas com o recurso selecionado são então exibidas para o usuário. O usuário, finalmente, escolhe a operação a ser associada com o produto.

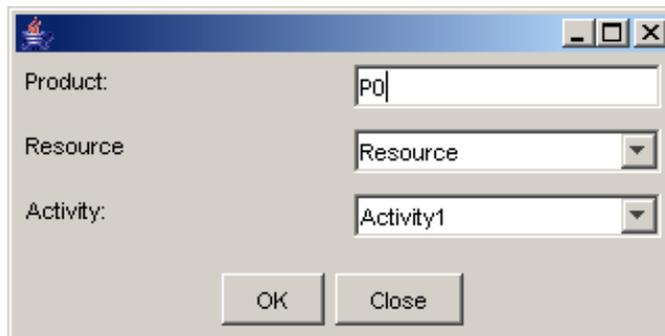


Figura 5.8: Mapeando Atividades nos Produtos

### Geração do Modelo Final

A geração do modelo final segue o processo de modelagem apresentado neste trabalho. Os elementos necessários para a obtenção do modelo final do roteiro de produção são:

- A estrutura do produto e seus atributos.
- Os recursos produtivos utilizados e seus atributos.

- O mapeamento do recurso/atividade nas partes do produto.

A descrição detalhada do modelo final foi feita na Seção 4.6.

### Definindo Novos Tipos de Rede

Um elemento importante no processo de modelagem é o atributo *rede* associado aos ícones que representam produtos e operações. Estes ícones determinam qual RdP será utilizada no refinamento do respectivo ícone. O atributo *rede*, portanto, abstrai a utilização de RdP na modelagem do sistema, uma vez que esse refinamento é feito de forma automática e transparente para o usuário.

O usuário que conhece a teoria das RdPs, todavia, pode definir novos tipos para serem utilizados na PMSTool. Para criar um novo tipo, o usuário especialista necessita criar a RdP que será representada pelo tipo, e armazená-la na biblioteca de tipos da PMSTool. Desta forma, outros usuários podem utilizar este novo tipo sem se preocupar com a RdP subjacente.

Esta rede deve seguir algumas regras básicas de criação. Em particular, ela deve apresentar um lugar de entrada e um lugar de saída bem definido. Informações importantes associadas à rede, tais como as inscrições a serem exibidas quando o usuário atribui valores aos atributos associados a um tipo na PMSTool, podem ser definidas. Os valores associados aos atributos de um dado *tipo* são mapeados nos valores das inscrições correspondentes.

Para a criação desta rede, qualquer editor de RdPs que suporte o formato PNML pode ser utilizado. A PMSTool não se propõe a ser um editor de RdPs. Uma possível configuração de uso é a utilização da PMSTool como um módulo do PNK. Desta forma, tem-se um editor de RdPs e uma ferramenta de modelagem de sistemas de produção integrados em um único ambiente.

### 5.2.6 Integração com PNK

A integração da PMSTool com o PNK foi motivada pelos seguintes fatores:

- A PNK fornece um conjunto de funcionalidades básicas que toda ferramenta baseada em RdP necessita.
- A PNK permite a criação de novos tipos de RdP.

A idéia do projeto do PNK foi tornar disponível para usuários e cientistas um *framework* para o desenvolvimento de ferramentas baseadas em RdP que disponibilizasse um conjunto de funcionalidades básicas que estão

presentes em qualquer baseada em RdP. Aplicações podem ser criadas e adicionadas ao PNK. Cada aplicação pode suportar um conjunto distinto de tipos de RdPs.

### Novos tipos de Rede

Para a integração com o PNK foram definidos dois novos tipos de rede:

- *pmsHighLevel* – utilizada para modelagem no alto nível.
- *pmsHierLevel* – utilizada para modelagem no nível intermediário.

A rede do tipo *pmsHighLevel* permite a modelagem, no alto nível, da estrutura do produto, dos recursos e do roteiro de produção. Para a modelagem são utilizados ícones e arcos de interconexão. Os modelos gerados são armazenados no formato PNML, dentro de uma pasta denominada *netTypeSpecifications*, onde são armazenadas as redes suportadas pelo PNK.

No sentido estrito da palavra, o tipo de rede *pmsHighLevel*, por não apresentar transições, não corresponde a uma RdP e sim a um grafo direcionado, e portanto não deveria ser armazenado no formato PNML. Deve-se salientar, todavia, que a especificação do PNML não impõe qualquer restrição quanto à existência de uma rede sem transições. Este tipo de rede pode ser selecionado pelo usuário da PMSTool que não tem conhecimento sobre RdP ou que não deseja trabalhar com modelos de RdP.

A rede do tipo *pmsHierLevel* é utilizada para representar o modelo intermediário do produto, recurso, ou roteiro obtido a partir do modelo de alto nível. Este tipo de rede corresponde a uma RdP hierárquica na qual os lugares serão refinados para se obter o modelo de RdP final.

Este tipo de rede pode ser selecionado pelo usuário da PMSTool que tem conhecimento sobre RdP e deseja trabalhar diretamente com os modelos de RdP em vez de trabalhar no alto nível. Os modelos de RdP intermediários e final são gerados automaticamente a partir do modelo de alto nível correspondente.

Quando integrada com o PNK, a PMSTool deve ser configurada como uma ferramenta baseada em RdP que suporta os tipos *pmsHighLevel* e *pmsHierLevel*, além dos tipos de redes correspondentes aos modelos finais (ex: rede lugar/transição, temporizada, estocástica, etc). Esta configuração é realizada no arquivo *toolSpecification.xml* que é um arquivo de configuração do PNK.

Tabela 5.1: Elementos da Tag Toolspecific

Elemento	Atributo	Descrição
toospecific	tool	ferramenta
	version	versão da ferramenta
Refnet	name	nome da rede
	type	tipo da rede
attribute	name	nome do atributo
	description	descrição do atributo
	value	valor do atributo
mapping	resource	recurso alocado
	operation	operação associada ao recurso
inplace	id	identificador do(s) lugar(es) de entrada
outplace	id	identificador do(s) lugar(es) de saída
value		representação textual

### Informações Específicas de Ferramenta

Cada ferramenta baseada em RdP normalmente apresenta um formato de arquivo próprio para armazenamento das redes geradas. O formato de arquivo PNML permite que informações específicas de cada ferramenta sejam armazenadas através de uma *tag* denominada *toolspecific*. As informações específicas da PMSTool, tais como rede utilizada no refinamento de um lugar, mapeamento partes de um produto nas operações correspondentes, lugares de entrada e saída de uma rede de refinamento, entre outras, foram armazenadas no formato PNML através da utilização de uma *tag toolspecific* cujos elementos constituintes são mostrados na Tabela 5.1.

A *tag toolspecific* apresenta os atributos *tool* e *version*, que indicam respectivamente a ferramenta que suporta aquela *tag* (no caso, a PMSTool), e a versão da ferramenta.

O elemento *refnet* identifica a rede que vai ser utilizada no refinamento de um lugar do modelo intermediário. Este elemento tem dois atributos, *name* e *type* que indicam respectivamente o nome da rede a ser utilizada no refinamento e o seu tipo.

Os elementos *inplace* e *outplace* identificam respectivamente os lugares de entrada e saída de uma rede utilizada no refinamento de lugares do modelo intermediário. Estas redes fazem parte da biblioteca de tipos utilizados

pela PMSTool e devem apresentar, por definição, um lugar de entrada e um lugar de saída bem definidos. Atributos podem ser associados aos modelos da biblioteca.

O elemento *attribute* identifica os atributos que podem estar associados a um vértice do modelo de alto nível ou a um lugar do modelo intermediário, por exemplo, ou às redes utilizadas no refinamento. Esta *tag* apresenta os atributos *name*, *description* e *type*, que correspondem respectivamente ao nome do atributo, à sua descrição, e ao valor do atributo.

O elemento *mapping* corresponde ao mapeamento de partes de produto nas operações que irão manufaturá-lo ou comprá-lo. Ele apresenta os atributos *resource* e *operation* que correspondem ao recurso e à operação a ser realizado na manufatura/aquisição de um dado produto.

Finalmente, o elemento *value* presente na *tag toolspecific* é utilizado para fornecer uma representação textual dos elementos da *tag*.

### Executando a PMSTool

Um possível cenário de execução da PMSTool consiste nos seguintes passos:

- Rodar o PNK
- Escolher o nível apropriado
  - Alto nível (*pmsHighLevel*)
  - Intermediário (*pmsHierLevel*)
- Rodar a PMSTool
- Escolher a opção de modelagem.
  - Produto
  - Recurso
  - Plano de Trabalho
- Desenhar o grafo ou a rede.
- Salvar o modelo final de RdP.

A Figura 5.9 mostra uma tela típica do PNK com alguns dos tipos de rede suportados. Para a escolha do nível de modelagem o usuário pode escolher, em função do seu perfil, o *pmsHighLevel* ou o *pmsHierLevel*.

A escolha do nível de modelagem indica que o usuário deseja efetuar uma modelagem no alto nível (ou no nível intermediário, no caso



Figura 5.9: Escolha do Tipo de RdP

de usuário com experiência em RdPs). A escolha do nível ocorre antes de executar a PMSTool por conta do funcionamento do PNK. Após a escolha do tipo de rede o PNK automaticamente habilita as ferramentas que suportam o tipo de rede escolhido. A Figura 5.10 mostra a seleção da PMSTool para execução.



Figura 5.10: Iniciando a PMSTool

A Figura 5.11 mostra as opções de modelagem da PMSTool, juntamente com a opção de geração automática do roteiro de produção. O significado de cada uma das opções de modelagem (produto, recurso, plano de trabalho) já foi explicado neste capítulo. A opção *Product* permite a modelagem da estrutura do produto. A opção *Resource* permite a modelagem do recurso produtivo. Finalmente, a opção *WorkPlan* permite o mapeamento de operações em partes do produto.



Figura 5.11: Geração do Modelo Refinado

Uma vez escolhido o elemento de modelagem, os modelos de alto nível, intermediário, final e do sistema são obtidos através da opção *Generate*. A partir de um modelo de alto nível para um produto ou recurso são gerados automaticamente o modelo intermediário, o modelo final, e o modelo do sistema para o produto ou recurso. A partir de um modelo intermediário para um produto ou recurso são gerados o modelo final e o modelo do sistema. Com a opção de geração do roteiro selecionada, são gerados o modelo de alto nível, o modelo intermediário, o modelo final e o modelo do sistema para o roteiro de produção.

A Figura 5.12 mostra o editor do PNK que foi estendido para representar de maneira adequada os elementos de modelagem utilizados pela PMSTool. No processo de modelagem são utilizados ícones e arcos de interconexão. A partir desta tela é possível acessar um *menu* através do qual pode-se definir qual o elemento de desenho a ser utilizado (ex: vértices, arcos, lugares, transições, etc.) bem como especificar o tipo de ação sobre os elementos de desenho (ex: criação, edição, remoção, etc.).

Uma aba denominada *toolspecific* é mostrada quando se está trabalhando com os tipos de RdP definidos para a PMSTool (*pmsHighLevel* e *pmsHierLevel*). O conteúdo da aba *toolspecific* varia em função do elemento de modelagem. Quando se está trabalhando com a estrutura do produto ou recursos produtivos, a aba mostra, para um dado vértice ou lugar, uma lista de tipos de rede que podem ser utilizados no processo de refinamento, além dos atributos associados a estas redes.

Ao se trabalhar com a opção de geração de produção, a aba *toolspecific* possibilita a seleção do recurso/operação a ser utilizado no processo de produção ou aquisição do produto representado pelo vértice (ou lugar). Cada uma das telas mencionadas será descrita na seqüência.

A Figura 5.13 mostra a tela da PMSTool que define os atributos para um dado vértice da rede. O atributo *Net* determina qual RdP será utili-

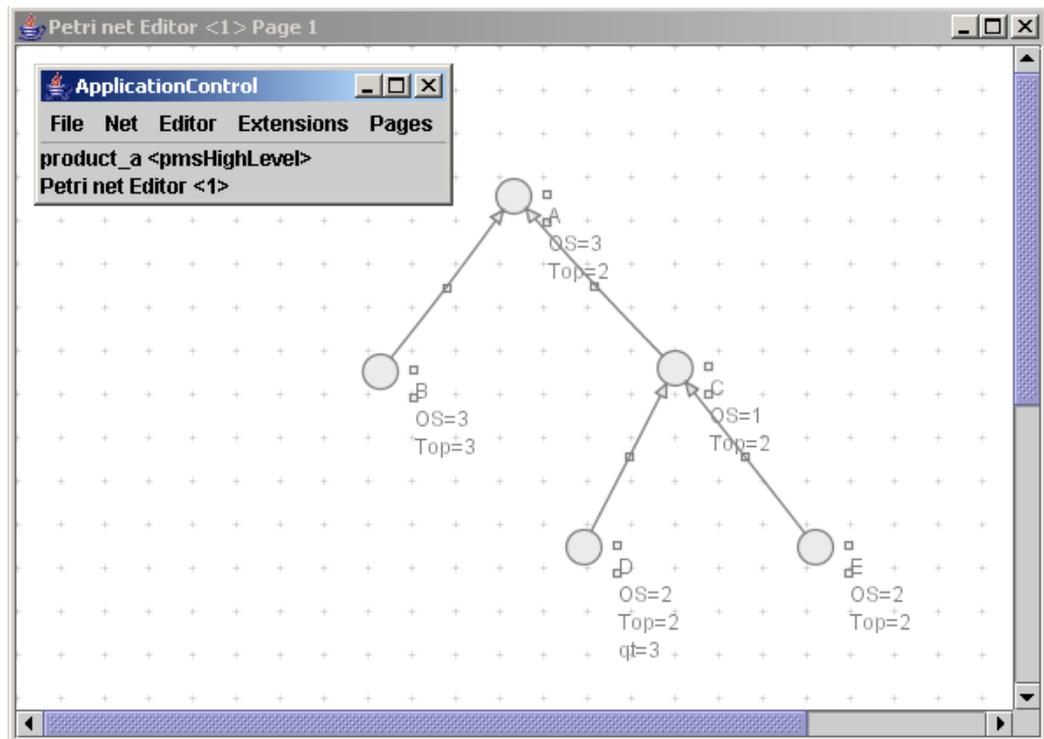


Figura 5.12: Editor do PNK

zada para refinar o vértice. Uma vez escolhida uma rede, seus atributos, caso existam, são automaticamente exibidos na tela. O usuário pode então atribuir valores aos atributos ou então definir seus próprios atributos, conforme definido no Capítulo 4.

A Figura 5.14 mostra a tela da PMSTool utilizada para mapear cada parte de um produto em uma operação correspondente. Nesta tela são mostrados os recursos disponíveis para a manufatura de (parte de) um produto. Estes recursos são obtidos da biblioteca de tipos existente. Quando o usuário seleciona um recurso, são mostradas as operações que aquele recurso pode realizar.

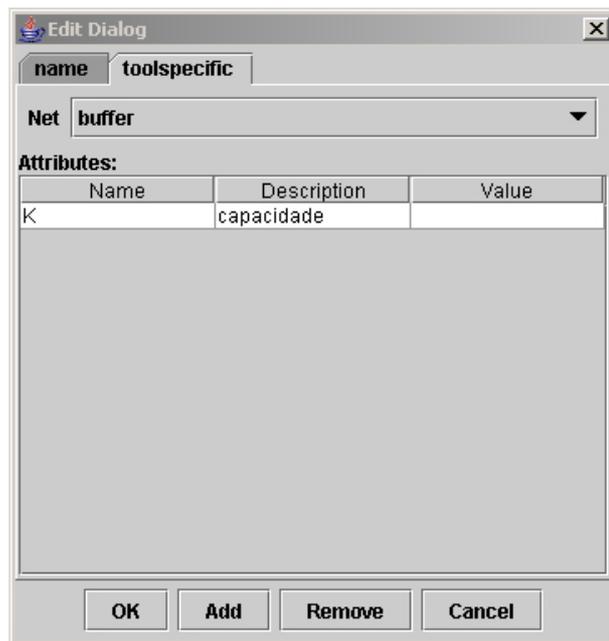


Figura 5.13: Definindo a RdP a ser Utilizada no Refinamento

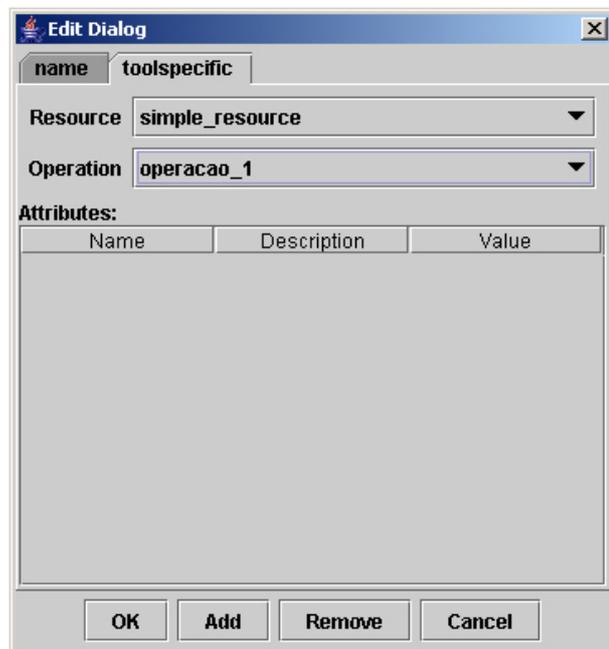


Figura 5.14: Mapeando Recurso e Operação

Ao selecionar uma operação, são mostrados os atributos associados à mesma, caso existam. O usuário pode então atribuir valores aos atributos ou criar e remover atributos através dos botões *Add* e *Remove*.

### Adicionando Redes à Biblioteca de Tipos

A Figura 5.15 mostra como o PNK pode ser utilizado para definir redes pertencentes à biblioteca de tipos. Primeiro, o usuário escolhe o tipo de rede a ser modelado. Em seguida, ele desenha a rede desejada utilizando o editor do PNK. Cada elemento da rede pode ter atributos associados. O modelador pode expor alguns destes atributos para o usuário que vai utilizar esta rede em um processo de refinamento. Para tanto, existe uma tela associada à rede sendo modelada na qual é possível definir os atributos a serem exibidos ao modelador no alto nível quando utilizar esta rede para refinar vértices do modelo de alto nível ou lugares do modelo intermediário (veja Capítulo 4).

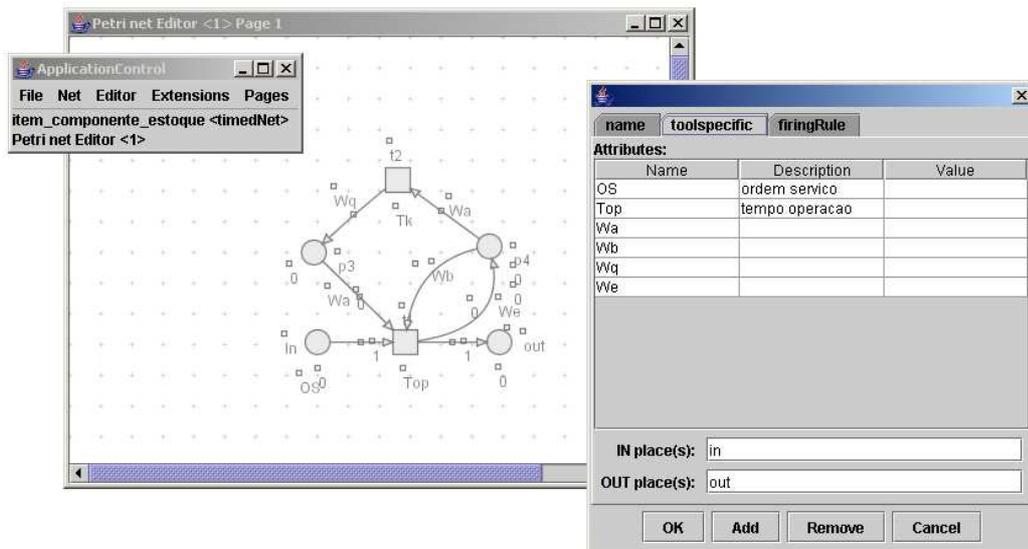


Figura 5.15: Rede Pertencente à Biblioteca

Os tipos de redes que pertencem à biblioteca de tipos são utilizados no processo de refinamento de lugares e, portanto, precisam ter um lugar de entrada e um lugar de saída bem definido. Na figura, são mostrados os campos correspondentes onde o usuário pode definir estes lugares de entrada e saída.

## 5.3 Considerações Finais

Neste capítulo foram apresentadas as principais características da PMSTool. Inicialmente falou-se sobre o formato de arquivo PNML, que é utilizado pela PMSTool para o armazenamento dos modelos gerados. Em seguida falou-se sobre o PNK que é um *framework* para o desenvolvimento de ferramentas baseadas em RdP.

Na seqüência falou-se sobre a PMSTool, suas características principais, bem como sua integração com o PNK. A PMSTool é uma ferramenta baseada em RdP que provê um ambiente de modelagem de alto nível para sistemas de produção. A ferramenta utiliza ícones que simplificam o processo de modelagem, abstraindo os detalhes de RdPs do usuário final.

A PMSTool dá suporte a metodologia de modelagem apresentada neste trabalho, ou seja, a modelagem da estrutura do produto, dos recursos produtivos e geração de um roteiro de produção a partir da estrutura do produto e dos recursos.

A utilização da ferramenta validou o processo de modelagem proposto no tocante a um de seus principais objetivos: a modelagem de sistemas de produção abstraindo o uso de RdPs.



# Capítulo 6

## Estudo de Caso

*Neste capítulo são apresentados os resultados da aplicação do processo de modelagem descrito neste trabalho a um problema real da indústria farmacêutica. Inicialmente será feita uma descrição do sistema de produção modelado. A seguir serão apresentados os modelos de RdP para a estrutura do produto, recursos produtivos, e o roteiro de produção.*

### 6.1 Introdução

O exemplo utilizado neste estudo de caso foi baseado no trabalho de Silva [Sil04] e objetiva aplicar o processo de modelagem apresentado neste trabalho na modelagem do processo de produção real do AAS (Ácido Acetil-Salicílico), realizado pelo Laboratório Farmacêutico de Pernambuco (LAFEPE), o segundo maior laboratório público do Brasil, que foi criado em 1966 para produzir medicamentos a baixo custo para as populações de menor poder aquisitivo. Dentre os diversos processos produtivos existentes no LAFEPE, foi selecionado o processo produtivo que contempla a fabricação do analgésico AAS 100 mg.

A primeira parte do estudo de caso consistiu em modelar a produção do AAS a partir da estrutura do produto. A segunda parte do estudo de caso consistiu na modelagem dos recursos produtivos e na geração de um roteiro de produção para o AAS. A vantagem da modelagem dos recursos produtivos é a possibilidade da captura de aspectos como falhas em máquinas, tempos de preparação de máquinas, compartilhamentos de recursos, etc.

## 6.2 LAFEPE

O LAFEPE tem unidades fabris de líquidos orais (não-antibióticos, antibióticos e gotas), sólidos orais (comprimidos, comprimidos revestidos, cápsulas e pós) e sólidos anti-retrovirais. O laboratório apresenta um rigoroso controle de qualidade. Os equipamentos são modernos e automatizados o que possibilita a produção anual de um bilhão de unidades.

O laboratório oficial é capaz de atender as áreas de injetáveis de pequeno e grande volume, pomadas e cremes, antibióticos penicilâmicos e instalar um novo almoxarifado para matérias-primas, o que lhe permite elevar a oferta de sua linha de produtos. Fazem parte do processo produtivo do LAFEPE cinco grandes áreas: o Planejamento e Controle de Produção (PCP), Custo, Produção (Chão-de-fábrica), Almoxarifado, Controle de Qualidade e Expedição.

**PCP** – responsável pelo planejamento e controle da produção.

**Custo** – representa a alta direção da empresa e suas decisões gerenciais.

**Produção** – responsável pela produção do medicamento propriamente dita.

**Almoxarifado** – responsável pelo armazenamento dos insumos utilizados na linha de produção.

**Controle de qualidade** – responsável por verificar e validar todo e qualquer elemento dentro do processo produtivo, a fim de atender aos requisitos de qualidade da empresa.

**Expedição** – responsável por receber o lote produzido e liberado pelo controle de qualidade para comercialização.

O inter-relacionamento entre estas áreas é mostrado na Figura 6.1.

## 6.3 AAS

A aspirina completou 100 anos em 1997 e é o medicamento mais conhecido e vendido em todo o mundo. Só nos EUA, são consumidos mais de 30 bilhões de comprimidos de aspirina por ano [Ago97]. A aspirina foi patenteada pela *Bayer* em 1899, e o seu nome deriva do nome da planta *Spiraea ulmaria* (“A” de acetilação, “spir” da planta **Spiraea ulmaria** e “in” sufixo empregado com freqüência em medicamentos naquela época).

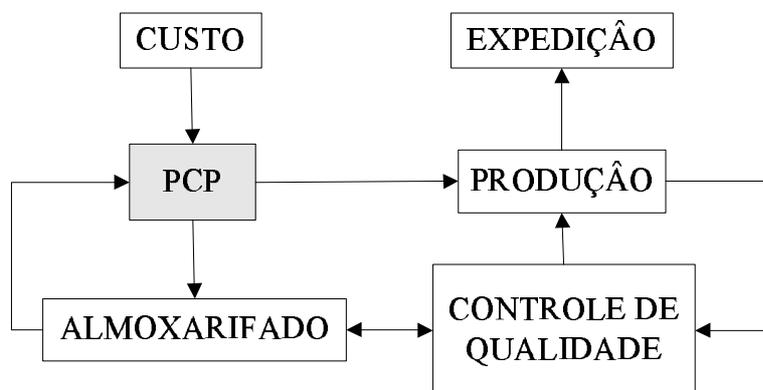


Figura 6.1: Fluxo de Produção do LAFEPE

O princípio ativo da Aspirina é o Ácido Acetil-Salicílico (AAS), que faz parte do grupo de medicamentos chamado *NSAID's - Non Steroidal Anti - Inflammatory Drugs* ou drogas não esteroidal anti-inflamatórias. O Ácido Salicílico possui alta e efetiva ação sobre redução de febres, com atuação analgésica e anti-inflamatória.

O processo de fabricação do AAS está dentro do fluxo do processo produtivo de sólidos do LAFEPE. O fluxograma mostrado na Figura 6.2 mostra as várias operações realizadas para a produção do AAS, tais como mistura, homogeneização, molhagem, granulação, secagem, calibração, compressão, revestimento, embalagem, etc.

O modelo de alto nível da estrutura do produto (veja Figura 6.3) foi definido a partir do fluxograma de produção do AAS e informações adicionais obtidas junto ao LAFEPE. Por exemplo, o *item 1* do fluxograma de produção do AAS corresponde ao *ícone 15* do modelo de alto nível.

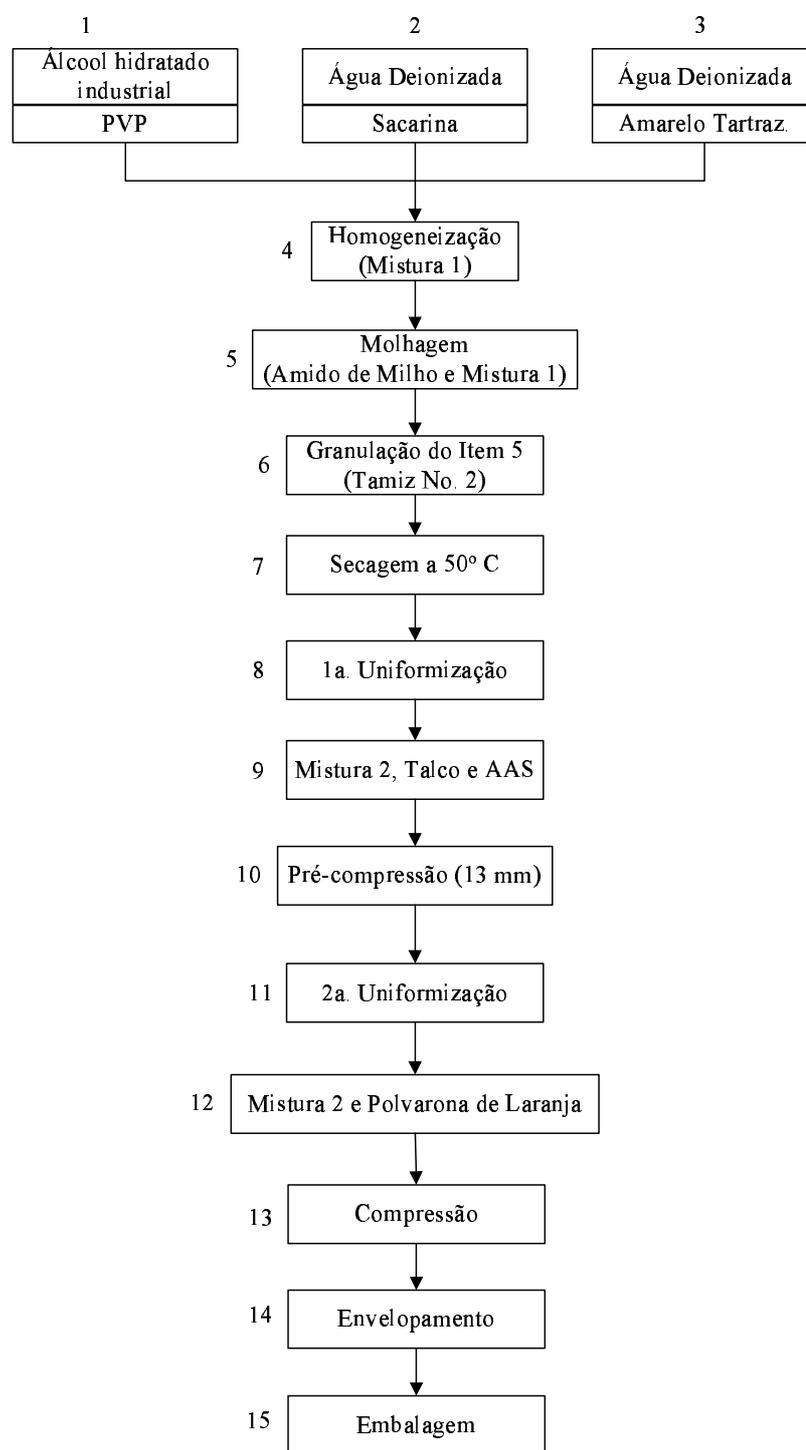


Figura 6.2: Fluxograma de Produção do AAS

## 6.4 Escopo do Estudo de Caso

O estudo de caso apresentado é baseado em um cenário de produção real do LAFEPE. Serão modelados a estrutura do produto, os recursos produtivos e gerado um roteiro de produção para o AAS.

O cenário real que serviu de exemplo para o estudo de caso foi montado a partir de um pedido de um determinado cliente do LAFEPE que desejava adquirir quatro milhões de unidades de comprimidos AAS. A linha de produção do AAS foi montada e disponibilizada segundo este pedido.

A análise quantitativa e qualitativa dos modelos gerados para a estrutura do produto, os recursos produtivos e o roteiro de produção não fazem parte do escopo deste estudo de caso.

## 6.5 Modelagem da EP para o AAS

O exemplo utilizado nesta seção foi retirado do trabalho de [Sil04] que buscava modelar e analisar as linhas de produção de comprimidos AAS feitos pelo laboratório LAFEPE. Todas as figuras e tabelas desta seção estão baseadas nesta referência.

Através da aplicação do processo de modelagem apresentado neste trabalho, e iniciando com um modelo de alto nível, foi possível gerar um modelo equivalente ao apresentado em [Sil04].

A Figura 6.3 mostra o modelo de alto nível da estrutura do produto AAS, que foi gerado a partir de dados fornecidos pelo laboratório LAFEPE. Deve-se observar que o modelo de alto nível está coerente com o fluxograma apresentado na Figura 6.2. Por exemplo, o vértice com rótulo 15, que representa a operação de embalagem, corresponde ao item quinze do fluxograma. Os vértices rotulados *AH* e *PVP* correspondem ao item 1 do fluxograma. E assim por diante.

**AH: Álcool Hidratado Industrial**  
**PVP: Polivinilpirrolidona**  
**AD: Água Deionizada**  
**SAC: Sacarina Sódica**  
**AT: Amarelo de Tartrazina**  
**TA: Talco Margesita**  
**AAS: Ácido Acetil Salicílico**  
**AM: Amido de Milho**  
**PL: Polvarona de Laranja**

**1: Homogeneizar**  
**2: Homogeneizar**  
**3: Homogeneizar**  
**4: Homogeneizar**  
**5: Amassadeira**  
**6: Granular**  
**7: Secar**  
**8: Calibrar**  
**9: Misturador**  
**10: Comprimir**  
**11: Calibrar**  
**12: Misturador**  
**13: Comprimir**  
**14: Envelopar**  
**15: Embalar**

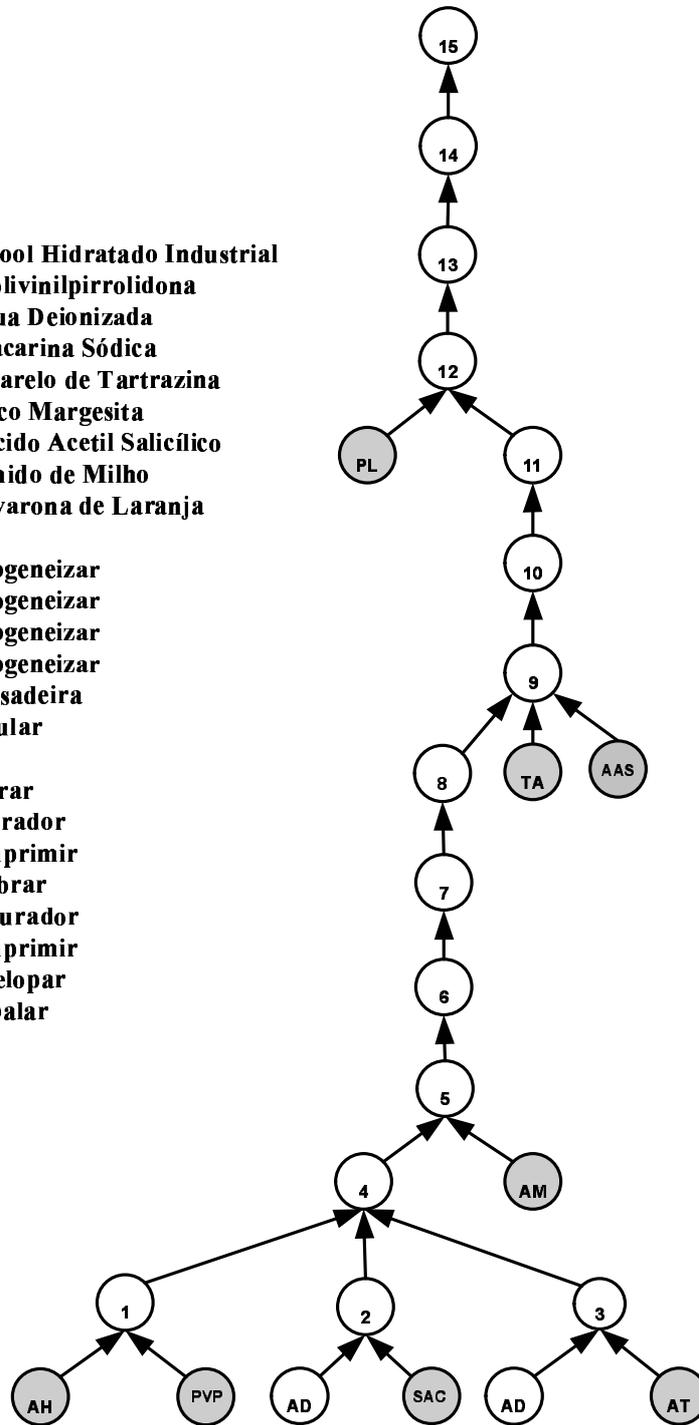


Figura 6.3: Modelo do Produto para AAS

A partir das informações mostradas na Figura 6.3 é possível identificar os insumos básicos (matéria-prima) utilizados na produção do AAS. Eles são mostrados na Tabela 6.1.

Tabela 6.1: Matéria-prima Utilizada na Produção do AAS

Código	Descrição
AAS	Ácido Acetil Salicílico
AH	Álcool Hidratado Industrial
PVP	Polivinilpirrolidona
AD	Água Deionizada
SAC	Sacarina Sódica
AT	Amarelo de Tartrazina
TA	Talco Margesita
AM	Amido de Milho
PL	Polvarona de Laranja

Associados aos vértices existem informações de estoque, que podem ser utilizadas para efetuar análises quantitativas no modelo. Alguns destes dados são mostrados na Tabela 6.2. Os dados exibidos são: o código do item, a capacidade máxima do estoque para o item, o estoque mínimo, o nível atual de estoque, o tempo de reposição, as ordens fixas (quantidade de reposição ou ressuprimento), e o consumo do item.

Tabela 6.2: Informações de Estoque do AAS

Código	$C_{max}$	$S_{min}$	Nível	Tempo Rep.	Ordens Fixas	Cons.
TA	3,3559 T	335,59 Kg	1,118 T	1 dia	1,118 T	50 Kg
PL	310,32 Kg	31,02 Kg	103,44 Kg	5 dias	103,44 Kg	4,5 Kg

A partir do modelo de alto nível, é gerado um modelo de RdP intermediário. O processo de obtenção deste modelo já foi descrito no Capítulo 4. Em linhas gerais, vértices são transformados em lugares, e entre cada vértice que representa um item-pai e os vértices que representam um item filho é acrescentada uma transição.

A Figura 6.4 mostra o modelo intermediário para o AAS. Um vértice numerado  $x$  do modelo de alto nível (veja Figura 6.3) é mapeado em um lugar correspondente rotulado “operação  $x$ ” no modelo intermediário. Por

exemplo, o vértice de rótulo 15 do modelo de alto nível corresponde ao lugar rotulado “operação 15” no modelo intermediário.

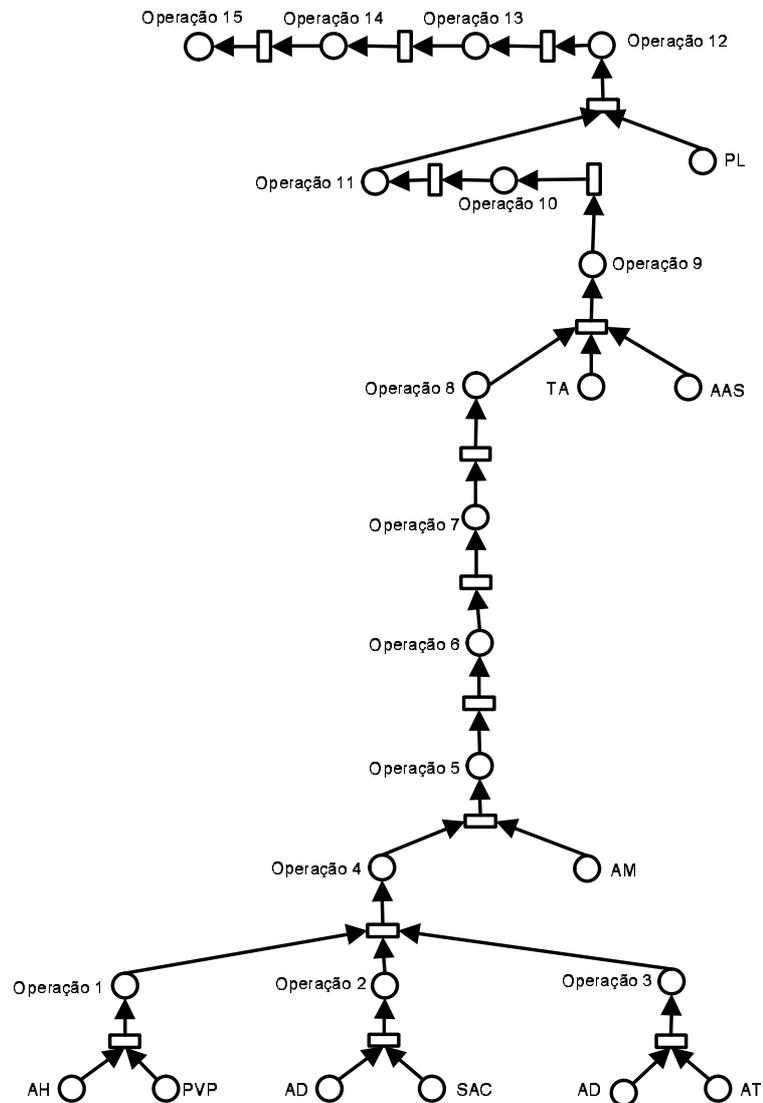


Figura 6.4: Modelo Intermediário para o AAS

Conforme descrito no Capítulo 4, cada lugar tem um conjunto de atributos. Um destes atributos é o tipo que identifica a RDP a ser utilizada para refinar este lugar. Os dois modelos básicos utilizados na modelagem são denominados Item Componente Simples (ICS) e Item Componente Estoque (ICE).

O ICS é caracterizado por não possuir restrição de quantidades disponíveis em estoque. Emitida uma ordem de serviço para um item componente simples, a operação a ser realizada no item pode ser iniciada independentemente de qualquer outro fator.

A produção de itens do tipo ICE, por sua vez, é caracterizada por depender da disponibilidade e quantidade de itens no estoque. Haverá um comprometimento da operação de produção do item correspondente quando não houver mais itens disponíveis no estoque. A operação a ser realizada não poderá ser completada até que o estoque seja devidamente repostado.

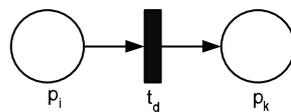


Figura 6.5: Modelo de Produto Simples

A Figura 6.5 mostra o tipo de RdP correspondente ao modelo ICS. Trata-se de uma RdP temporizada simples, na qual uma marca nos lugares  $p_i$  e  $p_k \in P$  modela uma ordem de serviço e a conclusão da ordem, respectivamente. O tempo de disparo da transição  $t_d$  corresponde à duração da operação.

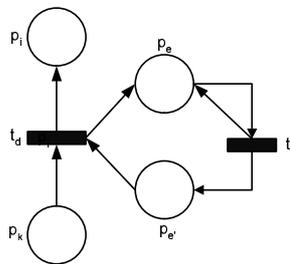


Figura 6.6: Modelo de Estoque

A Figura 6.6 mostra o tipo de RdP correspondente ao modelo ICE. Trata-se de um modelo mais elaborado que procura modelar a reposição do estoque. Neste modelo, uma marca nos lugares  $p_i, p_e, p_e'$  e  $p_k \in P$  modela uma ordem de serviço, nível de estoque, espaço disponível no estoque e a conclusão da ordem, respectivamente. O disparo das transições  $t_d$  e  $t_k$  modelam respectivamente a demanda e o *lead-time* (intervalo de tempo entre liberar uma ordem e receber as mercadorias no estoque).

A Figura 6.7 mostra o modelo de RdP final para a estrutura do produto AAS, resultado do refinamento da RdP intermediária utilizando os

modelos ICS e ICE. A maior parte dos vértices que representam partes da estrutura do produto AAS está associada com o tipo de RdP correspondente ao modelo ICS. Apenas os vértices que representam os itens *PL* e *TA* estão associados ao modelo ICE.

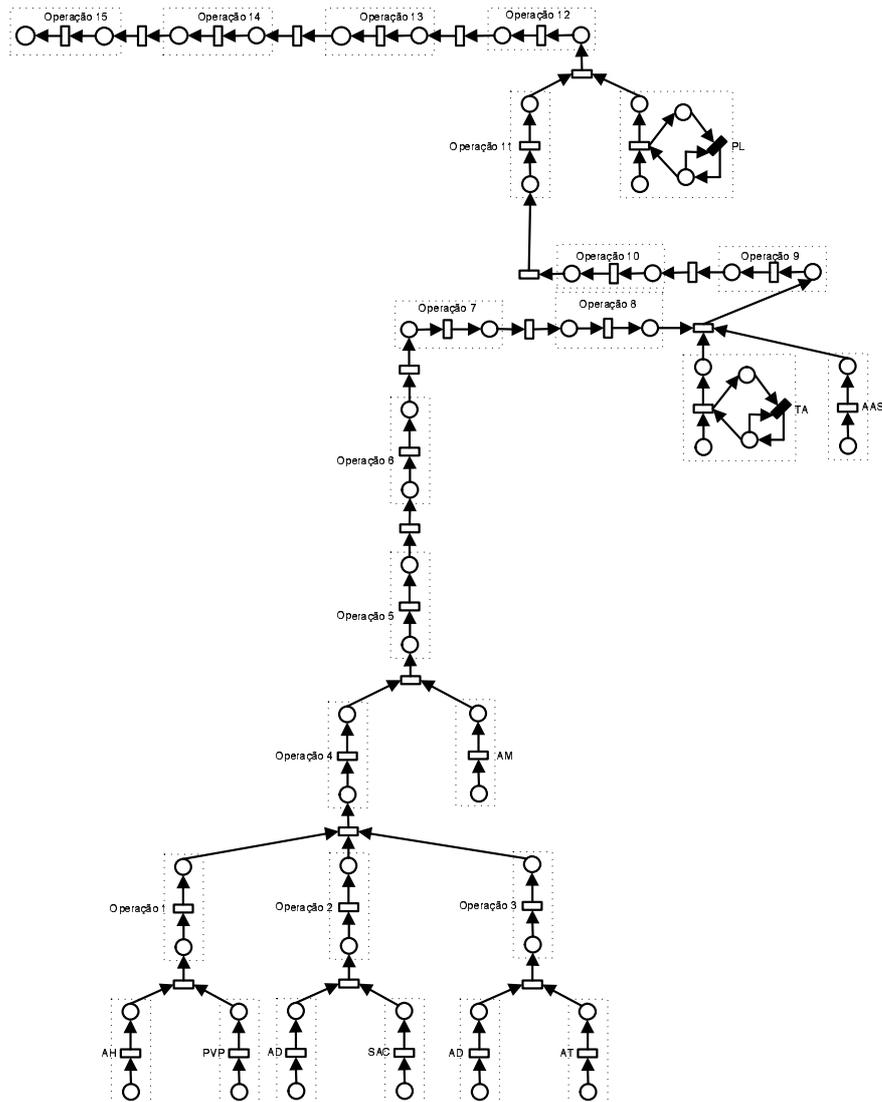


Figura 6.7: Modelo de RdP para AAS

O modelo final obtido é exatamente o mesmo modelo utilizado por [Sil04] para modelar a linha de produção do AAS e analisar a influência do estoque no processo produtivo do AAS.

O modelo completo do sistema de produção do AAS, que corresponde ao modelo de RdP final para o AAS acrescido dos modelos de entrada, saída e fechamento, é mostrado na Figura 6.8.

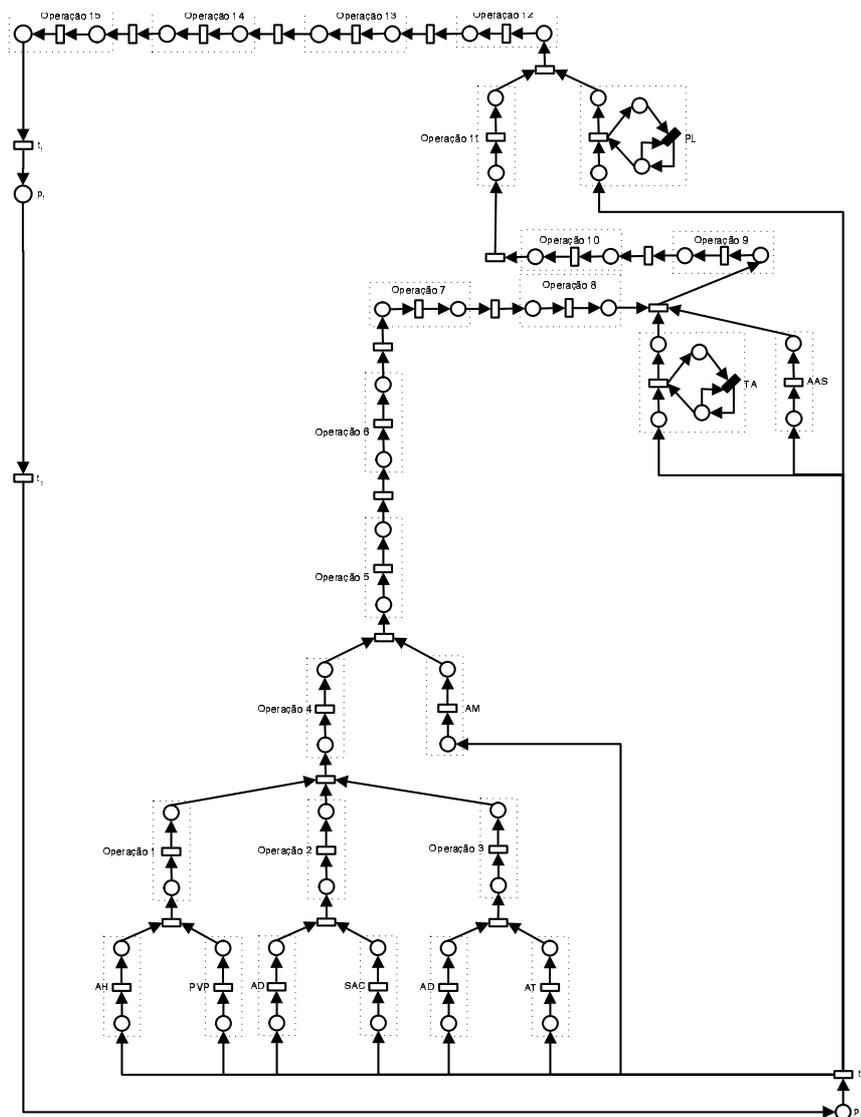


Figura 6.8: Modelo de Sistema de Produção AAS

Na figura,  $p_0$  e  $t_0$  fazem parte do modelo de entrada;  $p_f$  e  $t_f$  fazem parte do modelo de saída;  $p_f$ ,  $t_c$  e  $p_0$  fazem parte do modelo de fechamento. O modelo assim obtido pode ser utilizado para realização de análises tanto qualitativas quanto quantitativas sobre o sistema modelado.

## 6.6 Modelagem dos RPs para o AAS

A modelagem da estrutura do produto não leva em conta os recursos produtivos do sistema modelado. Também não captura, de forma clara, aspectos importantes tais como compartilhamento, indisponibilidade de recursos, entre outros. Portanto, também é necessário modelar os recursos produtivos do sistema.

Conforme definido na Seção 4.5, um recurso produtivo é um bloco funcional capaz de realizar uma série de operações voltadas para o processo de manufatura. Baseados num cenário de produção real da aspirina, os seguintes recursos foram identificados:

**Agitador mecânico** – realiza a operação de *homogeneização*.

**Misturador tipo V** – realiza as operações de *mistura* e *homogeneização*.

**Amassadeira** – realiza a operação de *molhagem*. O termo *molhagem* resulta do fato de os insumos serem colocados em uma amassadeira, molhados e homogeneizados até atingir o ponto de granulação.

**Granulador** – realiza a operação de *granulação*.

**Estufa de Secagem** – realiza a operação de *secagem* dos componentes.

**Compressor** – realiza a operação de *compressão*.

**Envelopadeira (emblastadeira)** - máquina para *envelopar* comprimidos.

**Calibrador** – efetua a operação de *calibração*.

**Operador** – realiza as operações de *movimentação*, que consiste em mover os insumos do almoxarifado para a linha de produção, e *embalagem* do produto acabado. Este recurso modela operações que não são automatizadas.

A Figura 6.9 mostra a representação de alto nível para os recursos produtivos do AAS.

O recurso *Misturador V* é representado como segue:

$Misturador V = \langle O, E \rangle$  onde

$O = \{mistura, homogeneizacao\}$

$E = \emptyset$

O recurso *Operador* é representado por:

*Operador* =  $\langle O, E \rangle$  onde

$O = \{movimentacao, embalagem\}$

$E = \emptyset$

O recurso *Estufa* é dado por:

*Estufa* =  $\langle O, E \rangle$  onde

$O = \{secagem\}$

$E = \emptyset$

O modelo de alto nível para os demais recursos pode ser obtido de forma semelhante.

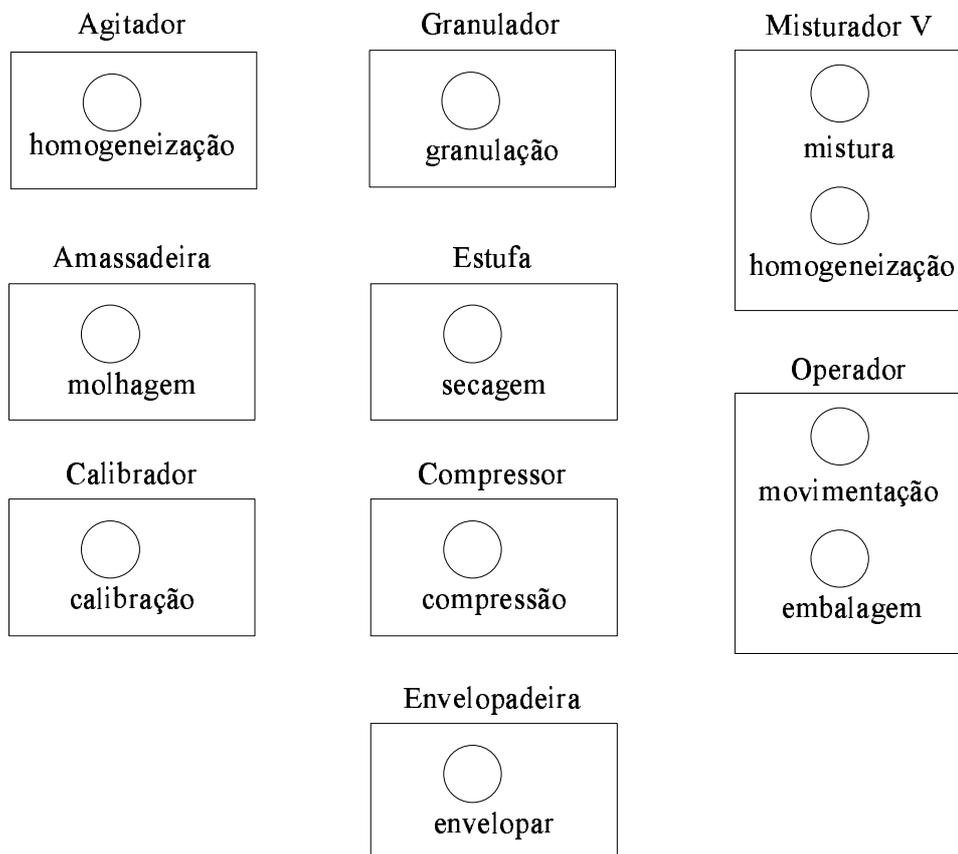


Figura 6.9: Modelo de Alto Nível para os RPs do AAS

## 6.7 Geração do ROT para o AAS

Conforme já foi salientado neste trabalho, os seguintes passos são necessários para a obtenção do roteiro de produção de um produto:

1. Escolher o produto a ser manufaturado.
2. Para cada sub-item definido na estrutura do produto, indicar o recurso/atividade que será utilizado para a sua manufatura/aquisição.
3. Gerar um escalonamento de atividades a partir da estrutura do produto e dos recursos produtivos.

Para este estudo de caso, o produto de interesse é o AAS, cuja estrutura é mostrada na Figura 6.3. Com base em um cenário de produção real do LAFEPE, foi possível identificar que:

- O recurso **Operador** efetua a *movimentação* dos insumos do estoque até a linha de produção.
- Um **Agitador Mecânico** realiza a *homogeneização* de AH e PVP.
- Um **Agitador Mecânico** realiza a *homogeneização* de AD e SAC.
- Um **Agitador Mecânico** realiza a *homogeneização* de AD e AT.
- A *homogeneização* da mistura anterior (Mistura 1) é realizada por um **Agitador Mecânico**.
- A *molhagem* do AM e Mistura 1 é realizada por uma **Amassadeira**.
- A *granulação* do item 5 é realizada por um **Granulador**.
- A *secagem* da mistura anterior é realizada por uma **Estufa de Secagem**.
- A *calibração* do granulado em tamiz de malha número 1,5 é realizada por uma **Calibrador**.
- A *mistura* do granulado (Mistura 2) com o AAS e TA é realizada por um **Misturador tipo V**.
- A *pré-compressão* é realizada por um **Compressor**.
- A *calibração* do granulado em tamiz de malha número 1,5 é realizada por uma **Calibrador**.

- A *mistura* do tamizado com PL é realizada por um **Misturador tipo V**.
- A *pré-compressão* é realizada por um **Compressor**.
- O *envelopamento* é realizado por uma **Envelopadeira**.
- A *embalagem* é um processo manual realizado pelo **Operador**.

Tabela 6.3: Mapeamento de Recursos para Produção do AAS

Item	Recurso	Atividade	Tempo (min)
AH	Operador	movimentação	1
PVP	Operador	movimentação	5
1	Agitador Mecânico	operação 1	15
AD	Operador	movimentação	5
SAC	Operador	movimentação	5
2	Agitador Mecânico	operação 2	15
AD	Operador	movimentação	5
AT	Operador	movimentação	5
3	Agitador Mecânico	operação 3	15
4	Agitador Mecânico	operação 4	6
AM	Operador	movimentação	5
5	Amassadeira	operação 5	30
6	Granulador	operação 6	390
7	Estufa de secagem	operação 7	2880
8	Calibrador	operação 8	12
TA	Operador	movimentação	5
AAS	Operador	movimentação	5
9	Misturador tipo V	operação 9	40
10	Compressor	operação 10	3
11	Calibrador	operação 11	12
PL	Operador	movimentação	5
12	Misturador tipo V	operação 12	30
13	Compressor	operação 13	3
14	Envelopadeira	operação 14	1
15	Operador	operação 15	29

Desta forma foi possível identificar quais recursos e quais operações eram necessárias para a aquisição ou manufatura de cada item constituinte da estrutura do produto AAS.

A Tabela 6.3 sumariza o mapeamento das partes do produto AAS (veja Figura 6.3) nos recursos produtivos. Também são mostrados os tempos médios para cada operação, considerando-se uma produção padrão de 2000 comprimidos. Por exemplo, a operação de secagem com estufa dura 48 horas ou 2880 minutos.

Deve-se destacar o compartilhamento de recursos nos seguintes casos:

- O recurso **Calibrador** é compartilhado pelos itens 8 e 11 da estrutura do produto.
- O recurso **Misturador V** é compartilhado pelos itens 9 e 12 da estrutura do produto.
- O recurso **Compressor** é compartilhado pelos itens 10 e 13 da estrutura do produto.

A Figura 6.10 mostra um roteiro de produção aplicado à manufatura do produto AAS cuja estrutura é mostrada na Figura 6.3. Para a geração do roteiro foram considerados os recursos produtivos mostrados na Seção 6.6. Na Figura 6.3 também se encontra a descrição das operações numeradas (de 1 a 15).

A descrição das operações no modelo de alto nível do roteiro de produção do AAS foi resumida por questão de simplicidade. As operações de movimentação foram numeradas de 1 a 9. As operações numeradas têm a descrição *operX* onde *X* corresponde ao número da operação. Desta forma, *oper1* corresponde à operação 1, que é realizada por um agitador mecânico, e corresponde a uma operação de homogeneização.

O processo de geração do roteiro de produção a partir da estrutura do produto e dos recursos produtivos está definido na Seção 4.6.

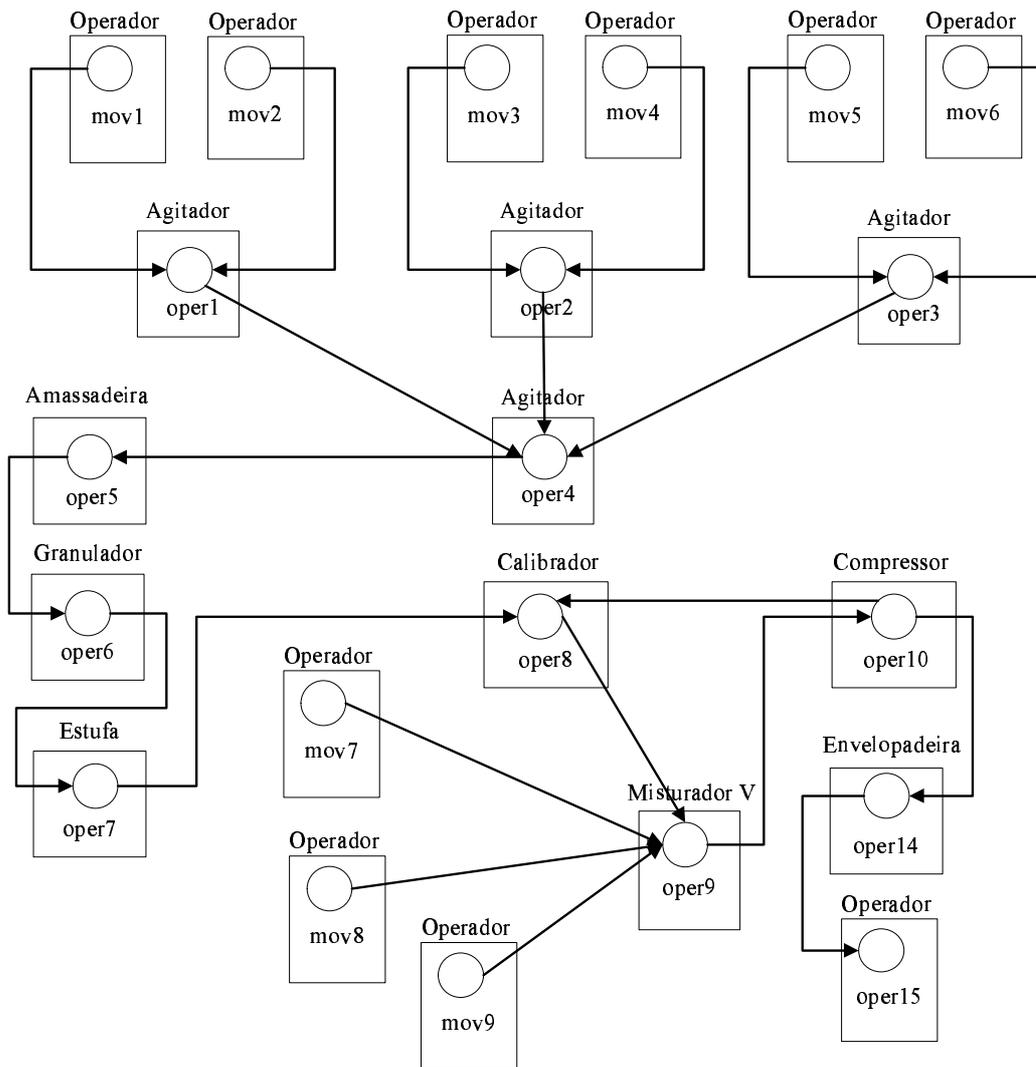


Figura 6.10: Modelo de Alto Nível para o ROT do AAS.

A partir do modelo de alto nível apresentado, o próximo passo consiste na geração de um modelo intermediário para o roteiro de produção, conforme descrito na Seção 4.6.2. A Figura 6.11 mostra o modelo resultante. Algumas operações tiveram seus nomes alterados por questão de simplicidade e para torná-los únicos.

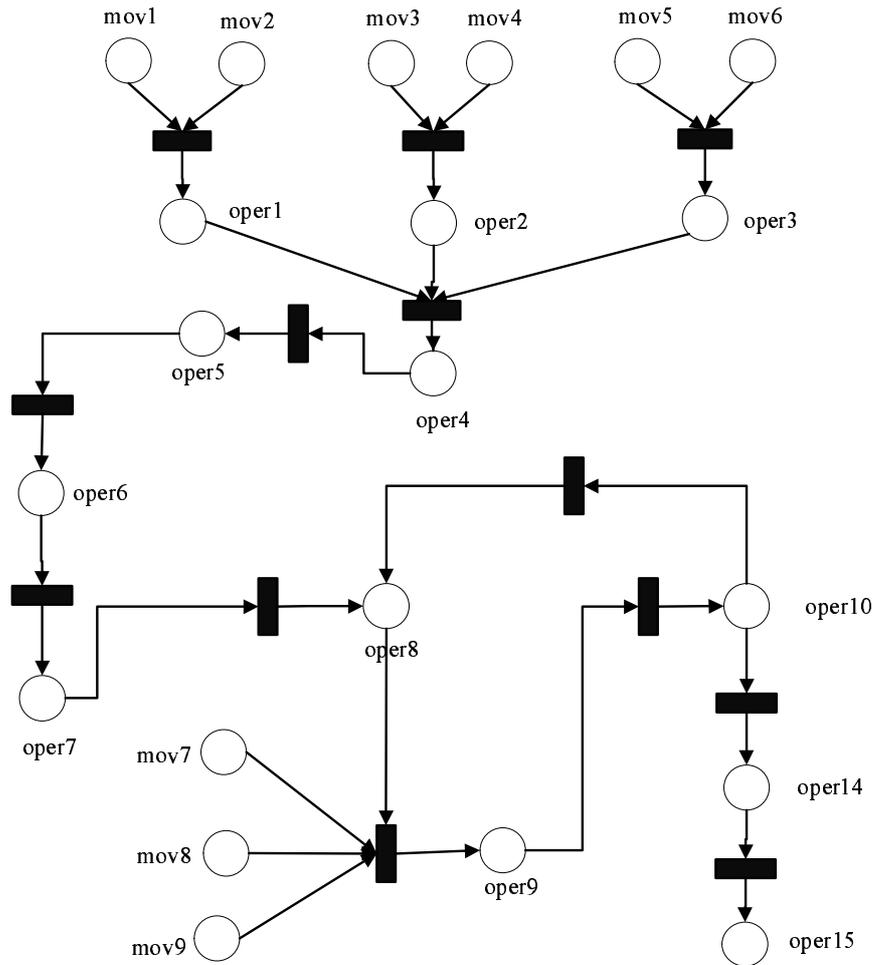


Figura 6.11: Modelo Intermediário para o ROT do AAS

Em seguida, a partir do modelo intermediário, é gerado um modelo de RdP final, conforme descrito na Seção 4.6.4. Este modelo é obtido através do refinamento dos lugares do modelo intermediário. Para o estudo de caso foram utilizados os modelos ICS (veja Figura 6.5) e ICE (veja Figura 6.6).

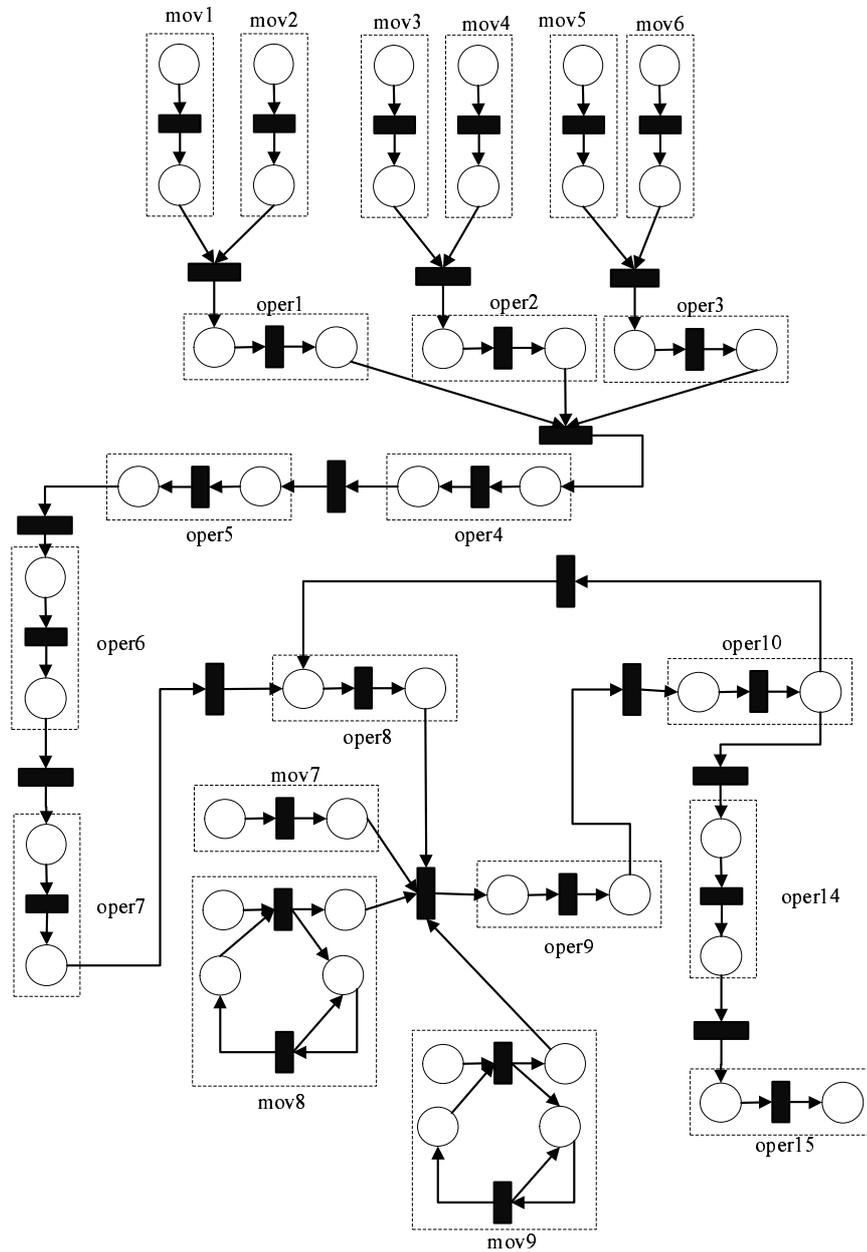


Figura 6.12: Modelo de Rdp Final para o ROT do AAS

A maior parte dos vértices que representam partes da estrutura do produto AAS são descritos com o tipo de Rdp correspondente ao modelo ICS. Apenas os vértices que representam os itens *PL* e *TA* são descritos pelo modelo ICE.

Estes modelos foram utilizados para refinar as operações dos recursos por permitirem uma análise de estoque sobre o modelo, que é um dos objetivos do trabalho original no qual este estudo de caso se baseia.

A escolha dos modelos a serem utilizados é função basicamente do que se está querendo modelar. Para uma análise de indisponibilidade de recursos, tempos de preparação, tempos de recuperação, compartilhamento de recursos, etc., modelos apropriados devem ser considerados. A Figura 6.12 mostra o modelo resultante.

Para finalizar, o modelo do sistema de produção do AAS pode ser gerado a partir do modelo final para o roteiro de produção obtido, e dos modelos de entrada, saída, e fechamento, conforme definido na Seção 4.7. A Figura 6.13 mostra o modelo resultante. Este modelo pode ser utilizado para efetuar análises qualitativas e quantitativas do sistema modelado.

O modelo do sistema apresentado na Figura 6.13 difere do modelo final do roteiro de produção do AAS pela presença dos elementos que compõem os modelos de entrada, saída e fechamento ( $t_f$ ,  $p_f$ ,  $t_c$ ,  $p_o$ ,  $t_o$  e arcos de conexão). O significado de cada um destes modelos, bem como o processo através do qual os mesmos são unificados para a obtenção de um modelo final, pode ser encontrado no Capítulo 4.

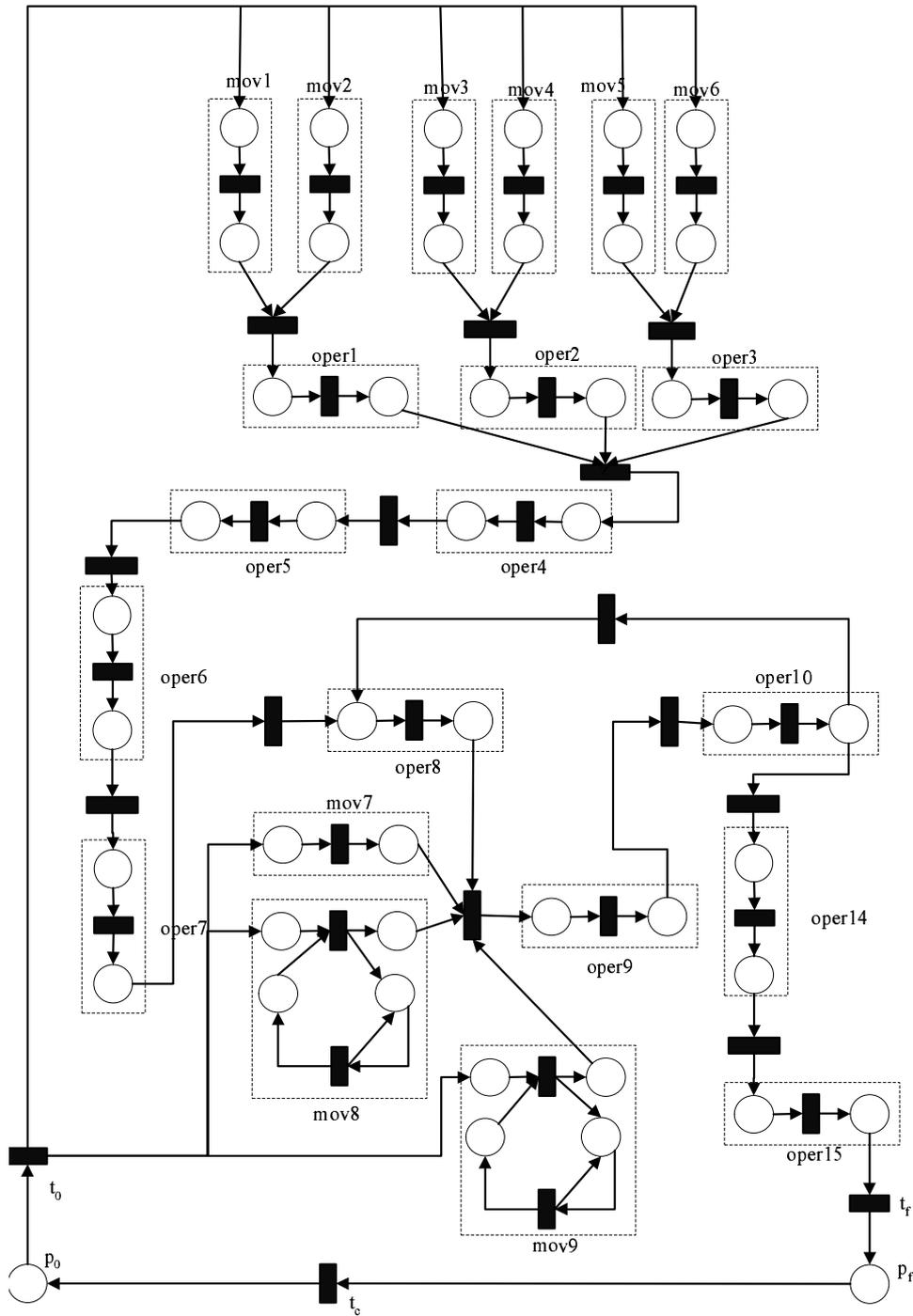


Figura 6.13: Modelo do Sistema de Produção do AAS.

## 6.8 Análise dos Modelos

Para complementar o estudo de caso, foi feita uma breve análise dos modelos gerados<sup>1</sup>. Uma análise mais detalhada dos modelos está fora do escopo deste trabalho.

Conforme já foi salientado, o processo de modelagem apresentado neste trabalho é composto basicamente de três etapas: geração do modelo de alto nível, geração de um modelo intermediário e geração do modelo de RdP final. Durante este processo de refinamento, algumas propriedades presentes no modelo intermediário, por exemplo, podem não estar presentes no modelo final. Em decorrência, o modelo final pode não capturar corretamente as características do sistema real que se deseja modelar e analisar. Isto é devido à flexibilidade com que o modelador pode definir novos modelos. A escolha criteriosa dos modelos de refinamento pode levar a um modelo resultante que preserve propriedades. Algumas propriedades, tais como a precedência entre operações que se verifica nos modelos do recurso e do roteiro de produção, são preservadas no modelo final.

Neste estudo de caso foram apresentados dois modelos que foram obtidos a partir da aplicação do processo de modelagem proposto: o modelo da estrutura do produto e o modelo do roteiro de produção para o AAS.

O modelo da estrutura do produto AAS é equivalente ao apresentado em [Sil04] onde foi feita uma análise de política de estoque sobre o modelo. Dentre os resultados obtidos naquele trabalho destacam-se a determinação do tempo para reposição do estoque sem que houvesse interrupção, construção do escalonamento global da produção, determinação dos níveis de utilização do estoque ao longo do tempo e determinação do caminho crítico de produção. Recomenda-se a leitura do referido trabalho para obter maiores detalhes da análise aplicada sobre o modelo da estrutura do produto AAS.

Neste trabalho foi efetuada a análise do modelo do roteiro de produção para o AAS (veja a Figura 6.13). Os tempos das operações são mostrados na Tabela 6.3. As informações de estoque utilizadas são mostradas na Tabela 6.2. Algumas reduções foram aplicadas ao modelo apresentado na Figura 6.13 para diminuir a quantidade de estados do grafo de alcançabilidade e simplificar a sua análise. Foram aplicadas reduções seriais nas transações<sup>2</sup> [Mur89] e redução das marcações pela aplicação de

---

<sup>1</sup>As análises foram realizadas com a ferramenta *INA* [RS99].

<sup>2</sup>Neste caso, o tempo da transição resultante corresponde ao somatório dos tempos das transições reduzidas.

proporcionalidade entre as marcações existentes<sup>3</sup>.

Após a simplificação do modelo, foi efetuada uma análise das propriedades básicas da rede bem como foi calculado o caminho crítico para o roteiro de produção apresentado.

Dentre as propriedades observadas, destaca-se que a RdP analisada é uma rede ordinária (a multiplicidade de cada arco é igual a um), homogênea (para qualquer lugar  $p$ , todos os arcos com início em  $p$  têm a mesma multiplicidade) e do tipo escolha-livre (um arco com origem em um lugar ou é o único arco de saída (sem escolha), ou é a única entrada para uma transição). Por ser do tipo escolha-livre pode-se afirmar também que a rede permite um conflito controlado, que pode ser resolvido pela escolha da transição que deve disparar através de um esquema de prioridades. A rede também apresenta invariantes de lugar. É interessante observar que as propriedades obtidas são propriedades desejadas em um sistema de produção.

Para o caminho crítico encontrado, calculou-se o menor tempo para a conclusão de um pedido e o resultado foi de 3.414 minutos. Este valor difere daquele que foi calculado em [Sil04], onde achou-se um caminho crítico com duração de 5.251 minutos. A diferença se deve ao fato de a análise apresentada naquele trabalho não levar em conta os recursos produtivos existentes, além de adotar uma política ALAP de produção. O modelo do roteiro de produção que foi analisado neste trabalho adota uma política ASAP e contempla não apenas a estrutura mais também os recursos disponíveis e a influência dos mesmos no processo de produção.

## 6.9 Considerações Finais

Neste capítulo, apresentou-se um estudo de caso que consistiu na aplicação do processo de modelagem proposto a um problema real da indústria farmacêutica.

Demonstrou-se que a estratégia de modelagem descrita neste trabalho possibilita a modelagem de alto nível da estrutura do produto, sem o conhecimento prévio da teoria das RdPs, e que o modelo final pode ser utilizado, por exemplo, para fazer análise qualitativa e quantitativa da influência do estoque no processo produtivo.

Também foi possível modelar os recursos necessários para a manufatura do produto e as operações por eles realizadas. Tais modelos são úteis, por exemplo, para a análise de capacidade produtiva de uma planta de

---

<sup>3</sup>Uma marca no lugar inicial corresponde a 20 mil unidades de comprimidos.

produção.

Finalmente, com base nos modelos da estrutura do produto e dos recursos produtivos, foi possível definir um roteiro de produção para o AAS. Este modelo possibilita a análise dos recursos utilizados no processo produtivo, influência do número de recursos disponíveis, compartilhamento de recursos, etc.

Para complementar o trabalho, foi efetuada a análise do modelo gerado para o roteiro de produção. Algumas propriedades desejadas em um sistema de produção foram identificadas no modelo gerado. Além disso, foi calculado o caminho crítico de produção para o AAS.

# Capítulo 7

## Conclusões e Trabalhos Futuros

*Este capítulo traz um resumo do trabalho apresentado, mostrando como os objetivos propostos foram atingidos, exibindo também quais os problemas encontrados durante o desenvolvimento do mesmo. Ao final são apresentadas algumas sugestões de trabalhos futuros.*

O objetivo do trabalho foi apresentar um processo e ferramenta para a modelagem de sistemas de produção, baseados em RdPs. O foco principal do trabalho consistiu em definir um modelo de alto nível que pudesse esconder do usuário, a utilização de RdPs. A estrutura do produto e os recursos produtivos foram modelados no alto nível. A partir destes modelos, foi gerado um roteiro de produção.

A PMSTool foi desenvolvida como uma ferramenta que dá suporte à modelagem apresentada neste trabalho. Ela provê um ambiente de modelagem que enfatiza não apenas a modelagem da manufatura, mas também o planejamento e controle da produção. Em síntese, o que se procurou foi desenvolver uma ferramenta de modelagem de sistemas de produção, de fácil utilização, e que auxiliasse na tomada de decisão. Conforme salientado, o conhecimento prévio de RdPs não é um requisito para utilização da ferramenta.

A maior dificuldade encontrada durante o desenvolvimento do trabalho consistiu na integração dos modelos (estrutura do produto, recursos, roteiro) e a sua formalização matemática.

Por ser uma ferramenta acadêmica, em sua versão inicial, a PMSTool carece de um melhor acabamento visual, estruturação de código e aprimoramento das funcionalidades. A ênfase foi dada na implementação da funcionalidade básica que validasse a metodologia de modelagem proposta.

Houve dificuldades na definição do estudo de caso a ser modelado. Foi

utilizado o estudo de caso apresentado em [Sil04] e incluído também a modelagem dos recursos e geração do roteiro de produção.

Outro grande problema consistiu em integrar o PMSTool com o ambiente PNK. Embora desenvolvido para prover o conjunto mínimo de funções que qualquer ferramenta baseada em RdPs necessita, o código fonte do PNK, particularmente o da versão *Java*, não seguiu os requisitos básicos da Engenharia de *Software* de forma a tornar simples o seu uso. Também merecem ser mencionados os *bugs* encontrados na ferramenta.

Em relação ao PNK, os seguintes problemas podem ser enumerados:

- O código fonte do PNK, versão 2.2 (versão mais atual do PNK disponível pela *Internet* à época deste trabalho), é datado do ano 2002, ou seja, não vem sofrendo atualizações e correções.
- O código fonte do PNK não estava devidamente comentado.
- A *Javadoc* (*Java API Documentation*) do PNK não foi corretamente gerada, em função de pouca ou nenhuma documentação no código fonte para as classes e métodos da aplicação.
- Não foi possível identificar utilização de boas práticas de Engenharia de *Software* que permitissem o reuso de código.
- Pouca documentação mostrando como estender o PNK e criar novos módulos.

Houve dificuldade para salvar os modelos gerados pela PMSTool no formato PNML. Um problema típico consistiu em posicionar graficamente os elementos da RdPs que eram gerados automaticamente durante a modelagem.

Uma das contribuições deste trabalho é o processo de modelagem que possibilita a modelagem de sistemas de produção abstraindo do usuário o formalismo matemático adotado e a geração automática de modelos de RdP. O estudo de caso confirmou a utilidade do processo para a modelagem de sistema de produção. O processo proposto neste trabalho apresenta as seguintes características que foram corroboradas no estudo de caso:

- Abstrai o uso de RdPs.
- Representa as características do sistema de forma clara e uniforme.
- Apresenta uma forma sistemática para a modelagem de sistemas de produção e partir da estrutura do produto e dos recursos produtivos.

- Representa o sistema do nível conceitual ao detalhado de acordo com a estrutura hierárquica das atividades do sistema, levando em conta a flexibilidade e capacidade de expansão.
- Gera automaticamente modelos de RdP para o sistema modelado.

Este resultado pôde ser implementado na ferramenta PMSTool, que possibilita, entre outras coisas:

- Modelagem de alto nível da estrutura dos produtos.
- Modelagem de alto nível dos recursos produtivos.
- Geração automática do roteiro de produção.
- Definição de uma biblioteca de tipos.
- Armazenamento dos modelos de RdPs no formato PNML.

A PMSTool pode ser utilizada como uma ferramenta não só para a modelagem de sistemas de produção, mas também como uma ferramenta de apoio a tomada de decisão.

Uma metodologia de modelagem de sistemas de produção foi desenvolvida e aplicada ao estudo de caso mostrado no Capítulo 6, alcançando os objetivos propostos para este trabalho. Como sugestão para trabalhos futuros, apresentamos:

- Dar suporte a modelos de refinamento (que constituem a biblioteca de tipos), com mais de um lugar de entrada e saída, para os lugares do modelo intermediário apresentado no capítulo de modelagem.
- Aprimorar a PMSTool, revisando a arquitetura, código fonte, correção de eventuais *bugs* e acrescentando novas funcionalidades.
- Desenvolver uma interface gráfica mais elaborada. Por exemplo, a GUI da PMSTool poderia ser aprimorada com a introdução de ícones diferenciados para representar os elementos modelados. Poderiam ser utilizados outros elementos de desenho além de arcos e ícones e cada qual poderia ter várias representações gráficas distintas, em função do aspecto a ser modelado.
- Desenvolver algoritmo de posicionamento dos elementos da rede na interface gráfica.

- Integração com o trabalho desenvolvido em [Sil04]. A PMSTool poderia funcionar como um *front-end* a partir do qual os modelos de RdPs utilizados em [Sil04] pudessem ser gerados.
- Novas abordagens para a geração automática do roteiro de produção.
- O processo poderia ser utilizado na modelagem de sistemas de produção que envolvam tanto a produção de bens quanto a de serviços.

# Apêndice A

## Teoria Básica

Neste apêndice apresentamos os conceitos básicos sobre teoria dos conjuntos, relações, funções e grafos. Tais conceitos são utilizados na formalização dos modelos apresentados no presente trabalho. Muitas das definições aqui apresentadas podem ser encontradas em [Ros99].

**Definição A.1** (Cardinalidade). *Seja  $S$  um conjunto. Se existem exatamente  $n$  elementos distintos em  $S$ , onde  $n$  é um inteiro não negativo, nós dizemos que  $S$  é um conjunto finito e que  $n$  é a cardinalidade de  $S$ . A cardinalidade de  $S$  é denotada por  $|S|$ .*

**Definição A.2** ( $n$ -tupla). *A  $n$ -tupla ordenada  $(a_1, a_2, \dots, a_n)$  é uma coleção ordenada que tem  $a_1$  como primeiro elemento,  $a_2$  como segundo, ..., e  $a_n$  como  $n$ -ésimo elemento. Em particular, 2-tuplas são chamadas de pares ordenados.*

**Definição A.3** (Produto Cartesiano). *Sejam  $A$  e  $B$  dois conjuntos. O Produto Cartesiano de  $A$  e  $B$ , denotado  $A \times B$ , é o conjunto de todos os pares ordenados  $(a, b)$  onde  $a \in A$  e  $b \in B$ . Ou seja,  $A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$ .*

**Definição A.4** (Bag). *Definimos um bag  $M$  sobre um conjunto não vazio  $C$ , por uma função  $M : C \rightarrow \mathbb{N}$ , onde  $m(c)$  representa o número de ocorrências do elemento em  $M$ . Bags são também denominados multiset.*

Utilizamos o símbolo  $[]$  para denotarmos os bags e  $\{\}$  para denotar os conjuntos. De forma alternativa, podemos representar o número de ocorrência de um elemento em um bag pela representação explícita desse elemento  $n$  vezes, ou seja,  $M_1 = [2a, 3b, c] = [a, a, b, b, b, c]$ .

**Definição A.5** (Relação). *Sejam  $A$  e  $B$  dois conjuntos. Uma relação  $r$  de  $A$  para  $B$  (ou de  $A$  em  $B$ ) é um subconjunto de  $A \times B$ . É costume denotar  $(a, b) \in r$  por  $A r B$ . O símbolo  $a r b$  é lido “ $a$  está  $r$ -relacionado a  $b$ ”.*

**Definição A.6** (Função). *Sejam  $A$  e  $B$  dois conjuntos. A função  $f$  de  $A$  para  $B$  é uma atribuição de exatamente um elemento de  $B$  para cada elemento de  $A$ . Nós escrevemos  $f(a) = b$  caso  $b$  seja o único elemento de  $B$  atribuído pela função  $f$  ao elemento  $a$  de  $A$ . Uma função  $f$  de  $A$  em  $B$  é denotada  $f : A \rightarrow B$ .*

Nós algumas vezes nos referimos a uma função como um *mapeamento* e dizemos que  $f$  mapeia  $A$  em  $B$ .

**Definição A.7** (Função injetora). *Uma função  $f$  é dita injetora, ou ainda um-para-um, se e só se  $f(x) = f(y)$  implica que  $x = y$  para todo  $x$  e  $y$  no domínio de  $f$ .*

**Definição A.8** (Função sobrejetora). *Uma função  $f$  de  $A$  em  $B$  é dita sobrejetora se e só se para cada elemento  $b \in B$  existem um elemento  $a \in A$  com  $f(a) = b$ . Em outras palavras, não sobram elementos em  $B$ .*

**Definição A.9** (Função bijetora). *Uma função  $f$  de  $A$  em  $B$  é dita bijetora caso seja ao mesmo tempo sobrejetora e injetora.*

**Definição A.10** (Domínio, co-domínio, imagem, pré-imagem, range). *Seja  $f$  é uma função de  $A$  em  $B$ . Nós dizemos que  $A$  e  $B$  são, respectivamente, o domínio e o co-domínio de  $f$ . Se  $f(a) = b$ , nós dizemos que  $b$  é a imagem de  $a$  e  $a$  é a pré-imagem de  $b$ . O conjunto de todas as imagens dos elementos de  $A$  é chamado imagem de  $f$ . Além disso, se  $f$  é uma função de  $A$  em  $B$ , nós dizemos que  $f$  mapeia  $A$  em  $B$ .*

Nós iremos evitar a terminologia “*range de uma função  $f$* ” porque muitos autores utilizam o termo *range* para designar o que foi aqui chamado de *imagem de  $f$*  e muitos outros autores o utilizam para designar o *co-domínio de  $f$*  [RW88, p. 108].

O domínio de uma função  $f$  é algumas vezes escrito  $dom(f)$ .

O conjunto *imagem* de  $f$  é escrito  $im(f)$  e pode ser definido matematicamente como  $im(f) = \{f(x) \mid x \in dom(f)\}$ .

As funções *dom* e *range* também podem ser definidas para relações binárias, o que pode ser encontrado em [Joh93].

**Definição A.11** (Grafo). *Um grafo simples  $G = (V, E)$  consiste de  $V$ , um conjunto não vazio de vértices e  $E$ , um conjunto de pares não ordenados de elementos distintos de  $V$ , chamados de arestas.*

**Definição A.12** (preset). Seja  $N = \langle P, T, R, W \rangle$  uma RdP. Define-se preset de um lugar como o conjunto das transições de entrada de  $p \in P$ . Da mesma forma, define-se preset de uma transição como o conjunto dos lugares de entrada de  $t \in T$ . Matematicamente, temos

- $p = \{t \mid (p, t) \in R\}$  = o conjunto de transições de entrada de  $p$ .
- $t = \{p \mid (t, p) \in R\}$  = o conjunto de lugares de entrada de  $t$ .

**Definição A.13** (postset). Seja  $N = \langle P, T, R, W \rangle$  uma RdP. Define-se postset de um lugar como o conjunto das transições de saída de  $p \in P$ . Da mesma forma, define-se postset de uma transição como o conjunto dos lugares de saída de  $t \in T$ . Matematicamente, temos

- $p^\bullet = \{t \mid (p, t) \in R\}$  = o conjunto de transições de saída de  $p$ .
- $t^\bullet = \{p \mid (t, p) \in R\}$  = o conjunto de lugares de saída de  $t$ .

**Definição A.14** (União finita). Define-se  $\bigcup A_i$  como a união finita dos conjuntos  $A_0, A_1, \dots, A_n$ . Matematicamente,  $\bigcup A_i = A_0 \cup A_1 \cup \dots \cup A_n$ .

**Definição A.15** (Interseção finita). Define-se  $\bigcap A_i$  como a interseção finita dos conjuntos  $A_0, A_1, \dots, A_n$ . Matematicamente,  $\bigcap A_i = A_0 \cap A_1 \cap \dots \cap A_n$ .

**Definição A.16** (Alfabeto). Um alfabeto é um conjunto finito não vazio  $\Sigma$  [letra grega sigma] cujos membros são símbolos, freqüentemente chamados letras de  $\Sigma$ . Dado um alfabeto  $\Sigma$ , uma palavra é qualquer seqüência de letras de  $\Sigma$ . Nós denotamos o conjunto de todas as palavras usando letras de  $\Sigma$  por  $\Sigma^*$  [sigma-estrela]. Qualquer subconjunto de  $\Sigma^*$  é denominado uma linguagem sobre  $\Sigma$ .

**Definição A.17** (Função de Nomeação). Seja  $\Sigma$  um alfabeto e  $X$  um conjunto qualquer. A função injetora  $\sigma : X \rightarrow \Sigma$  é definida como função de nomeação e atribui um nome único a cada elemento de  $X$ .

**Definição A.18** (Função first). Seja  $(x, y)$  um par-ordenado. A função *first* retorna o primeiro elemento do par, ou seja,  $first(x, y) = x$ .

**Definição A.19** (Função second). Seja  $(x, y)$  um par-ordenado. A função *second* retorna o segundo elemento do par, ou seja,  $second(x, y) = y$ .



# Apêndice B

## Glossário

**APICS** - *The Education Society for Resource Management* é uma organização educacional internacional sem fins lucrativos, fundada em 1957 como APICS - *The American Production and Inventory Control Society*.

Fonte: APICS. Online. <http://www.apics.org/About/>. 12/10/2004.

**Bag** – Um *bag* ou *multiset*, é uma generalização do conceito de conjunto que admite a repetição de elementos.

**Bug** – Erro de programação que causa um defeito no funcionamento de um programa.

**Java** – Java é marca registrada da *Sun Microsystems* e define uma plataforma de programação incluindo uma linguagem de programação, uma máquina virtual, um formato de arquivo (arquivo class) e um conjunto de classes (a API de Java).

**Lead-time** – Tempo de ressuprimento de um item.

**Máquinas de Controle Numérico (NC)** – São máquinas programadas através de fita magnética ou microcomputadores para executar um ciclo de operações repetidamente.

**Mouse** – Dispositivo móvel que controla a posição de um cursor na tela, e que conta com um ou mais botões.

**Recursos restritivos críticos (RRC)** – Recursos que limitam a capacidade produtiva do sistema.

**Sistema Flexível de Manufatura (FMS)** – Sistema no qual grupos de

máquinas de produção são ligados seqüencialmente por máquinas automatizadas de manuseio e transferência de materiais e integrados num sistema computadorizado [GF01].

**X-Windows** – Interface gráfica completa para os sistemas operacionais *Unix* e *Linux*, totalmente configurável e amplamente flexível para usuários e projetistas.

# Referências Bibliográficas

- [Ago97] William C. Agosta. Medicines and drugs from plants. *Journal of Chemical Education*, 74(7), Jul.1997.
- [AM89] M. Ajmone-Marsan. Stochastic petri nets: An elementary introduction. In *LNCS*, volume 424. Springer Verlag, 1989.
- [AMBC<sup>+</sup>91] M. Ajmone-Marsan, G. Balbo, G. Chiola, G. Conte, S. Donatelli, and G. Franceschinis. An introduction to generalized stochastic petri nets. In *Microelectronics and Reliability*, volume 31, 1991.
- [BC92] L. Bernardinello and F. De Cindio. A survey of basic net models and modular net classes. In Springer Verlag, editor, *LNCS*, volume 609, 1992.
- [BCM99] F. Basile, P. Chiacchio, and N. Mazzocca. International symposium on software engineering for parallel and distributed systems. In *Proceedings of IEEE*, page 3, 1999.
- [BCvH<sup>+</sup>] J. Billington, S. Christensen, K. van Hee, E. Kindler, O. Kummer, L. Petrucci, R. Post, C. Stehno, and M. Weber. *The Petri Net Markup Language: Concepts, Technology, and Tools*.
- [BM02] Tomaz Barros and Paulo Maciel. Resource estimation method based on timed petri nets for discrete production systems. In *Proceedings of the second IEEE International Conference on Systems, Man and Cybernetics (SMC'02), October 6-9, 2002, Hammamet, Tunisia, Volume 4*. IEEE Computer Society Press, Oct.2002.
- [BPF97] T. C. Barros, A. Perkusich, and J. C. A. Figueiredo. A fault tolerant coloured petri net resource allocation manager for manufacturing systems. In *Proceedings of IEEE International Conference on Systems Man, and Cybernetics. Orlando, Florida*,

- USA. IEEE International Conference On Systems Man And Cybernetics, 1997.
- [Bra83] G. W. Brams. *Réseaux de Petri: Théorie et Pratique*. Masson Editions, 1983.
- [CG96] H. Corrêa and I. Gianesi. *Just in Time, MRP2 e OPT*. Editora Atlas, 1996.
- [Cha99] Daniel Y. Chao. Petri net synthesis and synchronization using knitting technique. *J. Inf. Sci. Eng.*, 15(4):543–568, 1999.
- [Chi91] I. Chiavenato. *Recursos Humanos na Empresa*. Editora Atlas, São Paulo, 1991.
- [Col01] M. S. Colmanetti. Modelagem de sistemas de manufatura orientada pelo custeio das atividades e processos. Master's thesis, Escola de Engenharia de São Carlos, Universidade de São Paulo, São Paulo, SP, Brasil, 2001.
- [CW95] D. Y. Chao and D. T. Wang. Xpn-fms: A cad tool for fms modeling, analysis, animation, and simulation using petri nets and x window. *Int. J. Of Flexible Manufacturing*, pages 339–360, 1995.
- [DA85] R. David and H. Alla. *Petri Nets for Modeling of Dynamic Systems - A Survey*. Automatica, 1985.
- [DAJ95] A. A. Desroches and R. Y. Al-Jaar. *Applications of Petri Nets in Manufacturing Systems*. IEEE Press, 1995.
- [DCG91] G. Dicesare, S. J. Chang, and F. Goldbogen. Failure propagation trees for diagnosis in manufacturing systems. In *IEEE Trans. on Systems, Manuf. and Cybernetics*, 1991.
- [DD97] H. M. Deitel and P. J. Deitel. *Java, How to Program*. Prentice Hall, 1997.
- [DHP+93] F. Dicesare, G. Harhalakis, J. M. Proth, M. Silva, and F. B. Vernadat. *Practice of Petri Nets in Manufacturing*. Chapman & Hall, 1993.
- [EW01] C. Ermel and M. Weber. Implementation of parameterized net classes with the petri net kernel of the petri net baukasten. *LNCS 2128: Unifying Petri Nets — Advances in Petri Nets / H.*

- Ehrig, G. Juhás, J. Padberg, G. Rozenberg (Eds.)*, pages 79–pp, Dez.2001.
- [Fil99] C. Severiano Filho. *Produtividade & manufatura avançada*. Edições PPGEF, João Pessoa, PB - Brasil, 1999.
- [Fil00] F. M. C. Filho. Particionamento em hardware/software usando redes de petri. Master's thesis, Centro de Informática, Universidade Federal de Pernambuco, Recife, PE, Brasil, 2000.
- [FSS<sup>+</sup>04a] S. F. L. Fernandes, W. J. Silva, M. J. C. Silva, N. S. Rosa, P. R. M. Maciel, and D. F. H. Sadok. On the generalized stochastic petri net modeling of message-oriented middleware systems. In *23rd IEEE International Performance, Computing, and Communications Conference (IPCCC 2004)*, Embassy Suites Phoenix North, Phoenix, Arizona, 2004.
- [FSS<sup>+</sup>04b] S. F. L. Fernandes, W. J. Silva, M. J. C. Silva, N. S. Rosa, P. R. M. Maciel, and D. F. H. Sadok. Performance analysis of message-oriented middleware using stochastic petri nets. In *22nd Brazilian Symposium on Computer Networks (SBRC 2004)*, Gramado, RS - Brazil., 2004.
- [GF01] N. Gaither and G. Frazier. *Administração da produção e Operações*. Pioneira Thomson Learning, 8 edition, 2001.
- [GKZ95] R. German, C. Kelling, and A. Zimmermann. Timenet - a toolkit for evaluating non-markovian stochastic petri nets. *Performance Evaluation*, pages 69–87, 1995.
- [GL81] H. J. Genrich and K. Lautenbach. System modelling with high-level petri nets. *Theoretical Computer Science 13*, pages 109–136, 1981.
- [GV03] Claude Girault and Rüdiger Valk. *Petri Nets for System Engineering*. Springer, 2003.
- [Hum99] Humboldt-Universität zu Berlin – Institut für Informatik, Germany. *PNK User's Manual*, 1999.
- [Jan98] V. Janoušek. *Modelling Objects by Petri Nets*. PhD thesis, Department of Computer Science and Engineering, Faculty of Electrical Engineering and Computer Science, Technical University of Brno, Czech Republic, 1998. In Czech.

- [Jen90] K. Jensen. Coloured petri nets: A high-level language for system design and analysis. In *Lecture Notes in Computer Science*, volume 483. Springer Verlag, 1990.
- [Jen93] K. Jensen. An introduction to the theoretical aspects of coloured petri nets. *Lecture Notes in Computer Science; A Decade of Concurrency*, 803:230–272, Jun.1993.
- [JHS90] K. Jensen, P. Huber, and R. M. Shapiro. *Hierarchies in Coloured Petri Nets*, volume 483. Springer-Verlag, 1990.
- [JKW99] M. Jungel, E. Kindler, and M. Weber. *The Petri Net Markup Language*. Humboldt-Universität zu Berlin – Institut für Informatik, Germany, 1999.
- [Joh93] R. Johnsonbaugh. *Discrete Mathematics*. Prentice Hall, 3 edition, 1993.
- [MF76] P. M. Merlin and D. J. Farber. Recoverability of communication protocols: Implications of a theoretical study. In *IEEE Transaction Communication*, volume COM-24, Cambridge, MA: MIT, Set.1976.
- [Mil03] D. Mills. Understanding mvc. *Oracle Magazine*, 2003. Consultado em 10 dez. 2004. Disponível em <http://otn.oracle.com/oramag/webcolumns/2003/techarticles/mills-mvc.html>.
- [MK80] Tadao Murata and J. Y. Koh. Reduction and expansion of live and safe marked graphs. *IEEE Trans. on Circuits and Systems*, Cas-27(1):68–70, Jan.1980. NewsletterInfo: 6.
- [MLC96] P. Maciel, R. Lins, and P. Cunha. *Introdução às Redes de Petri e Aplicações*. X Escola de Computação Campinas-SP, julho de 1996.
- [Mol81] M. K. Molloy. *On the Integration of Delay and Throughput Measures in Distributed Processing Models*. PhD thesis, UCLA, Los Angeles, CA, 1981. PhD thesis.
- [Mol82] M. K. Molloy. Performance analysis using stochastic petri nets. In *IEEE Trans. Computers.*, volume C-31, pages 913–917, Set.1982.

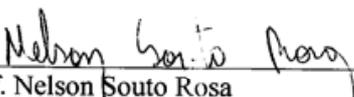
- [Mon83] Y. Monden. *Toyota Production System: Practical Approach to Management*. Industrial Engineering and Management Press, 1983.
- [Mor02] Daniel A. Moreira. *Administração da Produção e Operações*. Pioneira Thomson Learning, 2002.
- [Mur89] Tadao Murata. Petri Nets: Properties, analysis and applications. In *Proceedings of IEEE*, 1989.
- [Nah02] Fiona Fui-Hoon Nah. *Enterprise Resource Planning Solutions and Management*. IRM Press, University of Nebraska, USA, Abr.2002.
- [Nat80] S. Natkin. *Les Reseaux de Petri Stochastique et leur Application a l'Evaluation des Systèmes Informatique*. PhD thesis, CNAM, Paris, France, 1980. Thèse de Docteur Ingegnieur.
- [Pet81] J. L. Peterson. *Petri Nets an Introduction*. Prentice-Hall, 1981.
- [Ram74] C. Ramchandani. Analysis of asynchronous concurrent systems by petri nets. In *Project MAC, TR-120*, Cambridge, MA: MIT, Fev.1974.
- [Rei82] W. Reisig. *Petri Nets: An Introduction*. Springer-Verlag, 1982.
- [RI00] R. S. Russell and Bernard W. Taylor III. *Operations Management*. Prentice Hall, 3 edition, 2000.
- [RM01] Erik T. Ray and Christopher R. Maden. *Learning XML*. O'Reilly & Associates, Inc., 2001.
- [Ros99] K. H. Rosen. *Discrete Mathematics and its Applications*. Prentice Hall, 2 edition, 1999.
- [RS99] S. Roch and P. Starke. *INA – Integrated Net Analyzer, Version 2.2*. Humboldt-Universität zu Berlin – Institut für Informatik, Germany, 1999.
- [RW88] K. A. Ross and C. R.b. Wright. *Discrete Mathematics*. Prentice Hall, 2 edition, 1988.
- [SCJ97] N. Slack, S. Chambers, and R. Johnston. *Administração da Produção*. Atlas, 1 edition, 1997.

- [Shi96] S. Shingo. *Sistema Toyota de Produção – Do Ponto de Vista da Engenharia de Produção*. Bookman, Porto Alegre, RS – Brasil, 1996.
- [Sil04] M. J. C. Silva. Modelando políticas de estoques: Uma abordagem baseada em redes de petri. Master's thesis, Centro de Informática, Universidade Federal de Pernambuco, Recife, PE, Brasil, 2004.
- [SMS03] M. J. C. Silva, P. R. M. Maciel, and W. J. Silva. Modeling of manufacturing system for performance analysis: An approach based on gspn. In *European Simulation and Modelling Conference (ESMC's 2003)*, University of Naples II, Naples, Italy, 2003.
- [SSJS00] C. Simões, J. Smereka, R. Justino, and S. Silva. Erp – enterprise resource planning. Universidade Nova de Lisboa – Faculdade de Ciências e Tecnologia – Departamento de Informática. Consultado em 03 jul. 2004. Disponível em <http://students.fct.unl.pt/users/smss/erp/index.html>, 2000.
- [SSM04] M. J. C. Silva, W. J. Silva, and P. R. M. Maciel. Modelling and analysis in production system: an approach based on petri net. In *IEEE - SMC 2004 International Conference on Systems, Man and Cybernetics*, The Hague, The Netherlands, 2004.
- [Tec99] Technische Universität Berlin, Germany. *User Manual for TimeNET 3.0*, 1999.
- [Tit95] M. Tittus. *Control Synthesis for Batch Processes*. PhD thesis, Chalmers University of Technology, Goteborg, Sweden, Out.1995. PhD thesis.
- [Tub00] D. F. Tubino. *Manual de Planejamento e Controle da Produção*. Editora Atlas, 2000.
- [VBW92] T. E. Vollmann, W. Berry, and D. C. Whybark. *Manufacturing Planning and Control Systems*. Irwing, 3 edition, 1992.
- [VMS89] J. L. Villaroel, J. Martínez, and M. Silva. Graman: A graphic system for manufacturing system design. *IMACS Symposium on System Modelling and Simulation*, pages 311–316, 1989.

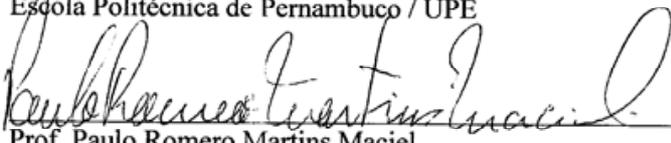
- [Voj01] T. Vojnar. *Towards Formals Analysis and Verification over State Spaces of Object Oriented Petri Nets*. PhD thesis, Department of Computer Science and Engineering, Faculty of Electrical Engineering and Computer Science, Brno Univeristy of Technology, Czech Republic, 2001. In Czech.
- [Yen92] H. C. Yen. Unified approach for deciding the existence of certain petri net path. *Information and Computation*, 1992.
- [ZF98] A. Zimmermann and J. Freiheit. TimeNET<sub>MN</sub> – an integrated modeling and performance evaluation tool for manufacturing systems. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, 1998.
- [ZFH01] A. Zimmermann, J. Freiheit, and A. Huck. A petri net based design engine for manufacturing systems. *International Journal of Production Research*, 39(2):225–253, 2001.
- [ZH98] A. Zimmermann and G. Hommel. Modelling and evaluation of manufacturing systems using dedicated petri nets. *Int. Journal of Advanced Manufacturing Technology*, 15:132–138, 1998.
- [ZZ94] R. Zurawski and M. Zhou. Petri nets and industrial applications: a tutorial. In *IEEE Transactions on Industrial Electronics*, 1994.



Dissertação de Mestrado apresentada por **Wellington João da Silva** a Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título, “**Um Ambiente de Suporte à Modelagem Hierárquica por Redes de Petri para Sistemas de Produção**”, orientada pelo **Prof. Paulo Romero Martins Maciel** e aprovada pela Banca Examinadora formada pelos professores:

  
\_\_\_\_\_  
Prof. Nelson Souto Rosa  
Centro de Informática / UFPE

  
\_\_\_\_\_  
Prof. Ricardo Massa Ferreira Lima  
Escola Politécnica de Pernambuco / UPE

  
\_\_\_\_\_  
Prof. Paulo Romero Martins Maciel  
Centro de Informática / UFPE

Visto e permitida a impressão.  
Recife, 4 de fevereiro de 2005.

  
\_\_\_\_\_  
**Prof. JAELSON FREIRE BRELAZ DE CASTRO**  
Coordenador da Pós-Graduação em Ciência da Computação do  
Centro de Informática da Universidade Federal de Pernambuco.